

Intelligent Networks

in

Jan Thörner

Intelligent Networks

Jan Thörner

Artech House
Boston • London

Library of Congress Cataloging-in-Publication Data

Thörner, Jan

Intelligent networks/Jan Thörner

Includes bibliographical references and index.

ISBN 0-89006-706-6

1. Telecommunication systems. 2. Telephone systems. I. Title.

TK5102.5.T536 1994

621.382-dc20

94-15140

CIP

British Library Cataloguing in Publications Data

Thörner, Jan

Intelligent Networks

I. Title

621.38

ISBN 0-89006-706-6

© 1994 ARTECH HOUSE, INC.

685 Canton Street

Norwood, MA 02062

All rights reserved. Printed and bound in the United States of America. No part of this book may be reproduced or utilized in any form or by any means, electronic or mechanical, including photocopying, recording, or by any information storage and retrieval system, without permission in writing from the publisher.

International Standard Book Number: 0-89006-706-6

Library of Congress Catalog Card Number: 94-15140

10 9 8 7 6 5 4 3

Contents

Preface		xi
Acknowledgments		xiii
Chapter 1	Building, Operating, and Using Large, Complicated Systems	1
1.1	Large System Evolution Occurs in Waves	1
1.2	Large System Dilemma	4
1.3	User Willingness	7
1.4	The World's Largest System	10
Chapter 2	Intelligent Networks	11
2.1	New Trends and Network Economy are the Driving Forces behind Intelligent Networks	11
2.1.1	Trends in Society	11
2.1.2	Trends in Technology	12
2.1.3	Trends in Communication	12
2.1.4	The New Natural Resources	13
2.1.5	New Trends and Economic Reasons Require a New Telecommunication Concept	13
2.2	The Intelligent Network Concept	15
2.2.1	The Intelligent Network Conceptual Model	16
2.2.2	The Introduction of Platforms	23
2.2.3	What the Intelligent Network Really Is	24
2.2.4	Applicability to All Access Networks	28
2.3	Network Service Implementation	29
2.3.1	Conventional Service Implementation	30
2.3.2	Service Implementation on Intelligent Networks	30
2.4	The Evolution of Intelligent Networks	34

2.4.1	How It All Began	34
2.4.2	Bellcore's IN/1, IN/2	35
2.4.3	Advanced IN	35
2.4.4	Capability Sets from ITU-T and ETSI	35
2.5	Intelligent Network Users: Requirements and Obligations	38
2.5.1	Service Subscribers and Service Users	38
2.5.2	Service Providers	39
2.5.3	Network Operators	39
2.6	The Term "Intelligent Network" Will Not Be Used beyond the 1990s	40
	References	40
Chapter 3	Platforms for Intelligent Networks Today	43
3.1	General Trends	43
3.2	A Basic Telephone Network is a Valuable Platform	43
3.3	Valuable Functions in an Existing Network	44
3.4	Intelligent Network Implementations with or without CCSS No. 7	46
3.4.1	Signaling Types	46
3.4.2	Common Channel Signaling System No. 7	47
3.4.3	Implementations that Use Service Control Points	49
3.4.4	Implementations that Use Service Switching and Control Points	53
3.4.5	Mated Pairs of Service Control Points and Service Switching Control Points	54
3.5	Telephone Traffic Theory Applied to New Services	55
3.6	Intelligent Network Management Functions	58
3.6.1	The Service Management Point	58
3.6.2	Charging, Billing, and Administration	58
	References	60
Chapter 4	Exploitation of Intelligent Networks in Services Today	61
4.1	Services Available before Intelligent Networks	62
4.1.1	General Aspects	62
4.1.2	Conventional Non-IN-Based Services	63
4.2	Service Implementation on Intelligent Networks	68
4.2.1	Service Creation on an Intelligent Network	69
4.2.2	Service Provisioning and Customization	74
4.2.3	Network Introduction of Services	75
4.3	The First IN-Based Services	76
4.3.1	Basic Characteristics of Some IN-Based Services	77
4.3.2	Value-Added Features	81
4.3.3	Creating Advanced IN-Based Services	83
4.4	Families of IN-Based Services	87

4.5	Complementary Services to Intelligent Networks	89
	References	89
Chapter 5	Risks and Threats Facing Intelligent Networks	91
5.1	Identifying the Weak Points	91
5.2	Solutions to the Problems	93
5.2.1	Redundant Paths and Backup Capabilities	93
5.2.2	Local and Temporary Service Implementations	94
5.2.3	Fast Withdrawal or Temporary Closing of a Service	94
5.2.4	Built-In Control and Supervisory Functions	94
5.2.5	Service Interworking Management Tools	95
Chapter 6	New Service Demands in the Future	97
6.1	Trends, Tendencies, and Demands in the 90s	99
6.2	Towards Full Mobility in the Fixed Networks	106
6.2.1	A New User Structure	106
6.2.2	The Impact of Full Mobility on Services	111
6.3	Service Administration and Management Evolution	116
6.3.1	Service Prototyping, Testing, and Supervision	117
6.3.2	The Evolution of Customer Control Capabilities	120
6.3.3	Charging and Billing	123
6.3.4	Security and Reliability Demands	126
6.4	Service Interworking	127
6.4.1	Understanding the Complexity of Service Interworking	127
6.4.2	Structuring Service Interworking	128
6.4.3	Service Interaction	129
6.4.4	Solutions to the Service Interaction Problem	132
	References	133
Chapter 7	Future Network Evolution in Harmony with Intelligent Networks	135
7.1	Different User Needs	135
7.1.1	Service Subscribers and Service Users	136
7.1.2	Service Providers	137
7.1.3	Network Operators	137
7.2	Traffic Demands on the Network	138
7.3	Network Traffic Solutions	140
7.3.1	Priority	141
7.3.2	Redundancy on the Network	142
7.3.3	Handshaking Functions and Acknowledgments	142
7.3.4	A Flexible and Adaptive Routing System	143
7.3.5	Encryption	143
7.3.6	Fault-Free Connections	144
7.3.7	Overload Functions (Call Gapping, Windowing)	144

7.4	Impact on Future Network Structure	145
7.4.1	New Network Topologies	146
7.4.2	Mated Pairs of Service Control Points	148
7.5	Allocation and Optimization of Resources on Networks	150
7.5.1	Resource Allocation and Optimization for Announcement Machines	150
7.5.2	Resource Allocation and Optimization for Service Logic	152
7.5.3	The Evolution of Resource Allocation and Optimization	153
7.6	Interworking between Intelligent Network Platforms	157
Chapter 8	Building, Operating, and Using Large, Complicated Telecommunication Networks	159
8.1	Large Intelligent Network System Evolution	160
8.2	Intelligent Networks Facilitate the Building of Large Systems	160
8.3	Solutions to the User Willingness Problem	168
8.4	Conclusions	171
Appendix		173
A.1	Service-Independent Building Blocks Defined in Capability Set 1	173
A.2	Capability Set 1 Target Set of Service Features	175
A.3	Capability Set 1 Target Set of Services	179
About the Author		183
Index		185

Preface

The worldwide telecommunication network is the greatest and most complex system that exists. Telecommunication is one of the most important, most expansive, and most powerful tools in the 90s, in almost any business area, and it is going to play a much more important role in the private domain. Many services that, today, are provided by other media, in the 90s, will be taken over by telecommunication networks. To fulfill all the requirements this implies, a quite new approach to building, maintaining, changing, and providing services is needed. A much quicker reaction to market demands as well as opportunities to customize services will be other natural consequences. The solution to fulfilling all those requirements is intelligent networks (IN), a concept that was introduced in the 80s and will be widely used in the 90s in all networks.

The first phase of intelligent network implementation is more a new way of thinking or of structuring than a new network. The basic functionality in intelligent networks, the *service control function* (SCF), which hosts the control (or the intelligence) for handling services, will be implemented in *service control points* (SCPs) or in *service switching and control points* (SSCPs). Before intelligent networks, this functionality was spread to several nodes in the network, for example, for a freephone number (i.e., 800 number). So, normally, when IN is introduced into a network, it brings about not a sudden increase in total network intelligence, but rather a new structuring of the already existing one. An increase in intelligence occurs first in the subsequent phases of intelligent network implementation, when such features as mobility, flexibility in charging and billing, advanced customer control facilities, and advanced routing are introduced. These enhancements would, however, not be possible without the new structuring introduced in the first phase.

It is not a coincidence that the intelligent network concept was born in the 80s and not before. It was in the 80s that networks reached, on a widespread scale, the level of technology required for intelligent networks to be introduced, namely, a broad usage of stored program control (SPC) exchanges, digital transmissions, and modern signaling systems like the Common Channel Signaling System (CCSS) No. 7.

The goal of this book is to provide a broad knowledge of intelligent networks, by exploring both the theoretical models of the International Telecommunication Union (ITU-T) (autumn 1993) and the practical experiences of implementing an intelligent network on a real network. Theoretical discussions accompany examinations of the impact of network implementation. User-related aspects of the intelligent network, from the subscribers', the service providers', and the network operators' views, are covered. The book examines the advantages of using intelligent networks as well as the risks to be faced. It also attempts to describe what intelligent networks represent in the long term—a flexible intelligence for any access form and any network.

Chapter 1 defines two major problems facing all networks in the 80s and 90s: (1) updating software in large, complex systems, like a telephone network, and (2) increasing the use of existing services. Chapter 2 takes a more general look at the intelligent network concept, including both practical issues and standardization in ITU-T and ETSI. The chapter also discusses the likely future of intelligent networks. Chapter 3 describes various network implementations of intelligent networks as well as the charging and billing functions. Chapter 4 describes the services available in the first years of intelligent networks, including services outside the intelligent network platform. Chapter 5 explores the risks and threats to intelligent networks that implementers face if they are not careful. Chapter 6 takes a step into the future and considers some new services and functions that will develop as a result of user demands: enhanced mobility, service interworking, and charging and billing. Chapter 7 examines the impact of new services on future IN platforms. Resource allocation and mobility are covered as well. The chapter also discusses the various ways that cooperation between intelligent network platforms can be accomplished. Finally, Chapter 8 speculates on how intelligent networks can help solve the problems raised in Chapter 1.

Acknowledgments

This book would have been much more difficult to write without the support I received from my employer, Telia, and from all my colleagues there. I want to thank Telia for its financial support and thank all my colleagues, who have been involved in one way or another, for their encouragement and support. However, the ideas and opinions expressed in the book are entirely my own and are not meant to reflect the policy, position, or opinion of Telia.

I also want to thank the International Telecommunication Union-Telecommunication Standardization Bureau (ITU-T) for allowing me to use two figures and a table for which they are the copyright holder (Figures 2.1 and 2.2 and Table A.1). The choice of this material is my own and therefore not the responsibility of ITU-T in any way.

Chapter 1

Building, Operating, and Using Large, Complicated Systems

1.1 LARGE SYSTEM EVOLUTION OCCURS IN WAVES

Defining a large system is always difficult. Nevertheless, a large system *can* be described as a system that possesses at least one of these characteristics:

- Requires considerable software to operate it,
- Is extremely complex,
- Covers a large geographic area,
- Contains a large number of users.

Additional criteria for describing large systems may exist, but these are generally the first characteristics that come to mind.

Some large systems are standalone systems, complete with built-in, independent intelligence. These systems may and often do communicate with other systems, but their main characteristic is that they are able to work autonomously, without intervention from the outside world. Excellent examples include airplanes, ships, and space shuttles, all of which are very complex systems that have nearly all control logic or intelligence on board.

Other systems feature distributed intelligence, whereby a single piece of equipment is only a small part of a larger, complex system. The entire system works by regulated co-operation among all the equipment inside, and, more importantly, it often runs in an operation mode that does not allow the whole system to be taken down for a change of hardware, testing, or another reason. The best example, of course, is the worldwide telecommunication network. A telecommunication network must function 24 hours a day, 365 days a year, which means that maintenance and improvements to the system must be carried out during operation.

Looking back at the evolution of large systems, an interesting phenomenon can be observed: large systems tend to evolve in waves (or follow a distinct wave form).

2 Intelligent Networks

- *A time to sow (spring)*. Phase one is the investment in and development of a basic system (a platform). In this phase, no results are visible but considerable amounts of money are normally spent.
- *A time to reap (autumn)*. In phase two, investments must be exploited to earn back the money spent during phase one. This is also the phase in which visible results are produced, and the money required to maintain the system is less than the money spent in phase one.
- *A need to sow (spring) again*. There will come a time, however, when we can no longer continue to run the existing system (existing platform) because it is becoming old-fashioned. Consequently, we must open our wallets again to invest either in improving the existing system or in developing a completely new one.

Consequently, the cycle starts all over again. (See Figure 1.1.)

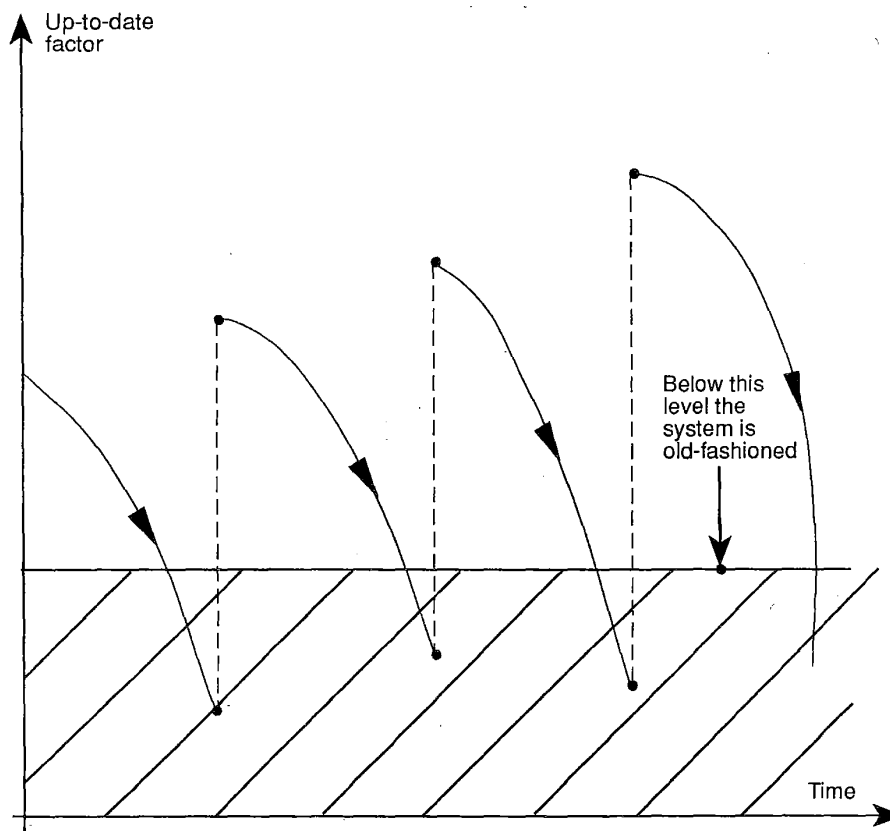


Figure 1.1 Large systems evolve in waves.

The system becomes old-fashioned because general demands increase as the technical evolution continues, and new systems are constructed with greater capabilities. This means that the original system becomes less modern; in other words, the *up-to-date factor* of the system continuously decreases. Once it falls below a certain low level, the system becomes a liability rather than an asset to its owner. Ideally, an organization should upgrade a system or replace it with a new one *before* the system becomes a liability. However, if the system is large and complex, this is a very expensive and time-consuming task. In reality, most systems experience a period, before they are upgraded or replaced, when they are more of a liability than an asset.

Obviously, this discussion is applicable to telecommunication networks. To understand that it can be considered a common trend, we need only look at the U.S. space program:

- During the 50s and 60s, large sums of money were spent building up a *basic system* that would enable human beings to leave the confines of earth.
- The *first visible triumphs*, for the general public, were the first successful launch of an unmanned rocket, which occurred around 1960; followed some years later by the first manned space flight; and the landing of a man on the moon in 1969. In between those events, a large amount of effort was expended, but it was not visible to the general public.
- In the 70s and part of the 80s, the *basic system* was improved.
- The next *visible triumphs* were the ability to reach other planets and the space shuttle, which provided a system that could be used more than once.

Telecommunication networks also evolve in waves, *from the technological point of view*, beginning with *stored program control* (SPC) technology in the 70s, followed by fiber optics and the *integrated service digital network* (ISDN) in the 80s, and continuing with *intelligent networks* and broadband in the 90s. However, this technological progress, where a large amount of money is spent, is not perceived directly by the user.

From the 60s to the 70s, the most visible advancement in telecommunication, *from a subscriber point of view*, was a more reliable network. During the 70s, value-added services, for example, call forwarding and call transfer services, became available in private automatic branch exchanges (PABXs); during the 80s, these services were also available in public networks. Services like freephone (800 numbers) and 900 numbers became available in the public networks in the 80s and 90s (often built first with node-based solutions and not with intelligent network platforms). In the near future, services offered via broadband access will be generally spread among subscribers.

Features visible to the general public are one thing; results that constitute the evolution of a large system are another. Just because results do not become visible at once does not mean that work is not proceeding. In fact, it is often quite the opposite; by the time results become visible, the work is almost completed.

data, and facsimile, as they do today, but also broadband—in the future, images will be used extensively.

Ultimately, possibilities will spring from the networks. And as they grow, these possibilities will increase demands. The demands will encourage further technical evolution, which will, in turn, increase the possibilities, and so on.

2.1.4 The New Natural Resources

All natural resources have one characteristic in common: they can be exploited to obtain a better living. Often, natural resources cannot be used directly, but must be refined for use. Natural resources are also limited and must be handled with care so we may share them now and in the future. Extravagant use of resources could have disastrous effects on future generations. In the past, natural resources have included iron, wood, water, and air, to mention a few.

In the same way that we must be careful not to overexploit water and air, we must also take care in the way we use telecommunication resources when communicating over distances. These resources are also limited and may not last unless they are handled carefully. Examples of these resources include:

- Radio frequencies
- Telephone numbers (number series)

Two serious problems facing many networks today are that radio frequencies are rapidly filling up and telephone numbers are running out. When a number series in one area is exhausted, adjusting it causes a ripple effect. This means that numbers must be changed for many people, not only in that area but in surrounding areas as well, which is extremely time-consuming and expensive and often causes irritation.

Good *planning* in the use of these resources is essential. One of the most important considerations for network economy in future networks will be *resource allocation and optimization*. (This is further discussed in Section 7.5.) To facilitate current planning of resource use and future resource allocation and optimization, we need an instrument that will provide a good network overview. For both tasks, although they are not its primary aim, the intelligent network can be of assistance.

2.1.5 New Trends and Economic Reasons Require a New Telecommunication Concept

Trends in society, as well as possibilities presented by technological evolution, combined with economic considerations, have fueled development in telecommunication networks, from SPC exchanges in the 70s to the evolution towards the intelligent network in the 80s and 90s.

When we make a telephone call today, we want to either reach a particular *person* or perform a particular *function*, involving, for example, a bank, an insurance company, or a

pizzeria. When we make the call, we need to know the exact physical location we are calling so we reach the right telephone. In future, we want to be able to call a person or perform a function by using a *unique number*, no matter where that person or that function is physically located. We will just call the unique number and the network, not we, will know where to route the call.

For some functions, who we reach or which location we call is of no importance, as long as we get the right response and assistance. For example, calls to a bank can be of two types. Either we want to talk to our own personal bank contact (a person), or we simply want to call the bank to discuss general matters (a function). In the latter case, it does not really matter which branch we reach or who answers, as long as we can receive the same assistance from all branches.

On the other hand, if we want to have a pizza delivered, the exact location we call is of extreme importance—unless, of course, we like cold pizza. We need to be sure that the pizzeria that takes our order is located in our general vicinity. Here, the value of intelligent networks becomes apparent. With intelligent networks, we do not need to know where the nearest pizzeria is located. We simply dial the same unique “pizza number,” no matter where we are in the country, and the network finds the nearest pizzeria. The result is a hot pizza.

These scenarios, in which a network provides the exact level of service a user desires, illustrate one of the main goals of the introduction of the intelligent network concept. Given that goal, identifying the needs of a subscriber becomes the first task. A *preliminary structure* that probably covers most user needs can be easily established from the scenarios described.

Customers want to be able to call

- A particular person, regardless of his or her location;
- A particular person at a particular location or within a particular area;
- An alternative person if the first cannot be reached;
- A particular location, rather than a particular person or a particular function;
- A particular function, rather than a particular person or a particular location;
- Any function situated within x miles of the caller.

These customer requirements of the future, which will be solved by intelligent networks, together with the ongoing development and future possibilities of intelligent network technology, suggest that in the 1990s the telephone could play a much greater role in our lives than ever imagined. The term “telephone,” as it is used here, also refers to future terminals with greater built-in intelligence and with a more intelligent user interface than the poor interface available in today’s ordinary DTMF pushbutton phone. An increasing number of daily functions currently handled by other media will be provided by telecommunication network services in the 90s. This will occur first through the implementation of intelligent networks in *public switched telephone networks* (PSTNs) and later in narrowband and broadband in the *integrated services digital network* (ISDN).

Today’s users can get a taste of the future when they use a telephone to

- Obtain bank services,
- Obtain the daily news, stock market reports, and weather forecasts,
- Turn on the heat in their weekend cottage before arriving.

These examples point to the fact that the telephone has already begun to compete with other media, such as newspapers, letters, radio, television, and text-television. Such services are normally offered in telecommunication networks through a premium rate number, with the customer paying per service or per minute. In future, services based on broadband ISDN—for example, movies—will be offered to subscribers through the telephone network via a premium rate number controlled by an SCP in an intelligent network. Subscribers will pay for the movies via their telephone bills.

Other major areas for combined intelligent network/broadband services include interactive video services, which will allow you to sit at home and search through a large database, obtain information, and view pictures on your screen while searching. For example, from the comfort of your sofa, you could review various destinations and obtain a view of the beach, hotel restaurants, hotel rooms, and so on, before deciding where to spend your holiday. Real estate agents could use this service to permit buyers to examine different properties without leaving home. Last but not least, adventure games will probably be available through interactive video services. Intelligent network technology and plans around the world support this evolution.

What then is so special about intelligent networks? Briefly, the driving force behind introducing the intelligent network concept in networks all over the world is the goal of reducing the time required to implement a new service in the network. Conventional techniques and conventional software programming require somewhere between two to five years from the date a decision until a service becomes available to users, that is, the subscribers. The goal of intelligent networks is to reduce this time to a maximum of six months! Section 2.2 describes how that can be achieved.

2.2 THE INTELLIGENT NETWORK CONCEPT

When the intelligent network concept was first introduced in the 1980s, it was soon followed by such concepts as intelligent network services, intelligent network technology, and the like. The natural question is, did any intelligence exist in networks prior to the advent of intelligent networks?

Obviously, there has been a continuous growth of intelligence in networks since long before the term “intelligent network” was coined. Accordingly, the term “intelligent network technology” is appropriate because it describes a new type of technology, the core of which is centralized service control located in SCPs. SCPs are defined in Section 2.2.1 and are further described in Chapter 3 and subsequent chapters. The principal task of SCPs in early intelligent networks is to control the setting up of *number translating services* in the network; the most important part of the task is determining the location to which a service, for example, a freephone call, should be routed.

Services controlled by SCPs, for instance, freephone calls and credit calls, were available on the network before intelligent network technology evolved, in other words, before service control was moved from the local node (the exchange) to a centralized node (the SCP). Subscribers have been using these services for many years, and when network operators switch to intelligent network technology, by moving service control from a local (or transit) exchange to an SCP, subscribers will hopefully see no difference in use.

The obvious question then is, has the service suddenly gone from being unintelligent to intelligent? The obvious answer is, of course not. The concept of "intelligent network service" is therefore misleading. Consequently, this book uses "services" instead of "intelligent network services" when discussing *services in general*. Services can thus be implemented by using conventional, node-based technology directly in the nodes *or* by using intelligent network technology in the network *or* directly in the terminals.

This range of choices may create a situation in which different services working simultaneously are implemented in different ways. In this book, if the implementation method is of importance, services implemented using intelligent network technology are referred to as *intelligent network-based (IN-based) services*, services implemented in terminals are *terminal-based services*, and services implemented in conventional node-based ways are *node-based services*. Actually, it is quite possible that, during an updating period, for example, a network could have both a node-based and an IN-based implementation of a particular service running simultaneously for some time.

The next section defines and discusses the first standardized model for the intelligent network, the intelligent network conceptual model of the International Telecommunication Union (ITU-T) and the European Telecommunication Standardization Institute (ETSI).

2.2.1 The Intelligent Network Conceptual Model

It is essential to distinguish between *functions* and *implementations* in the intelligent network concept. It is also essential to distinguish between intelligent network architectures and the intelligent network conceptual model. While architectures will evolve with increasing service requirements and emerging technologies, the conceptual model is intended to remain consistent. It was created by ETSI and ITU-T and is described in the Q.120x series from ITU-T.¹ The goal of the conceptual model is to encourage better understanding of the intelligent network concept. The model, illustrated in Figure 2.1, is a four-plane structure, consisting of

- A service plane
- A global functional plane (GFP)
- A distributed functional plane (DFP)
- A physical plane

1. There is a logical structure in the numbering of the Q documents to facilitate use. The Q.120x series describes the conceptual model. Q.121x describes the corresponding standards for *Capability Set 1* (CS1), Q.122x describes *Capability Set 2* (CS2), and so on.

The following list illustrates the relationship between implementation and the theoretical model (conceptual model).

<i>Implementation</i>	<i>Conceptual Model</i>
a) Services (The plane that users see)	Service plane
b) SIBs (Service-independent building blocks)	Global functional plane
c) FE (Functional entities, e.g., CCF, SCF, SSF)	Distributed functional plane
d) Physical products (E.g., SCP, SSP, SDP)	Physical plane

The Service Plane

The service plane, which is described in ITU-T rec. Q.1202 [1], provides a view that is exclusively service-oriented. In other words, it shows services without indicating how they have been implemented. A service might be implemented on an intelligent network platform (IN-based) or a conventional technique might be used (node-based or terminal-based).

Service features (SF) (see Figure 2.1) are the smallest functions on this plane. A service is composed of one or more SFs. An SF is larger than a *service-independent building block* (SIB) but smaller than a service. The service plane also contains customization of services and management services.

An important matter to consider with regard to the service plane is service and service feature interaction. In other words, do undesired effects occur when two or more services or service features are used together? Service interaction, which is discussed in more detail in Section 6.4, can be divided into two parts:

1. Interactions that result from the specifications of the services (and therefore are independent of the implementation).
2. Interactions that result from how the services were implemented. (The same service, from the user's perspective, may be implemented in different ways, causing different types of interactions with other services. See Section 6.4.)

Only the first type of interaction can be solved on the service plane. The second type must include the other planes.

The SFs on the service plane are mapped to the *global functional plane* (GFP) by combining SIBs on the GFP using *global service logic* (GSL).

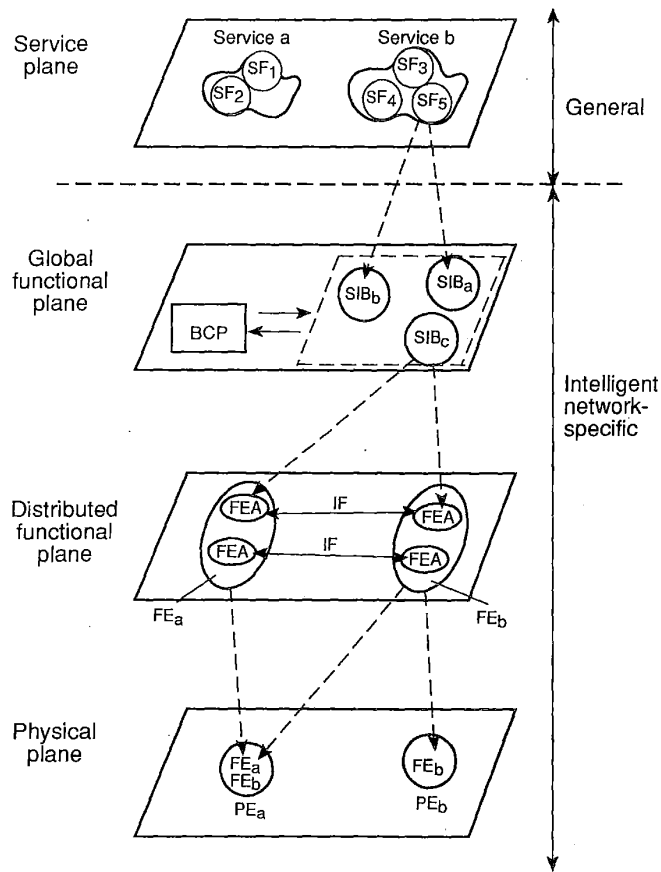


Figure 2.1 The intelligent network conceptual model. (ITU-T Rec. Q.1201, Figure 3.1/a.²)

The Global Functional Plane

The GFP, which is described in ITU-T rec. Q.1203 [2] on architecture and in Q.1213 [3] on CS1, is the plane on which SIBs are found. GFP is also the plane on which network operators create basic new services.

SIBs are the smallest building block that can be found in an intelligent network. A SIB is defined as “a standard reusable networkwide capability residing in the global functional plane used to create service features” [2]. The *basic call process* (BCP) is a special

2. Full text of Q.1201 may be obtained from the ITU Sales Section, Place des Nations, CH-1211 Geneva 20, Switzerland, Tel: +41 22 730 51 11, Fax: +41 22 730 51 94.

SIB that handles all activities necessary for a normal call. Service features are composed of one or more SIBs; the SIBs are chained together by global service logic (GSL). The SIBs themselves, including BCP, are service-independent and have no knowledge of the next SIB in the chain. Consequently, GSL is the only element on the GFP that is service-dependent. The description of how the SIBs are linked together is often called a service script.

A first set of SIBs that will be commonly used, Capability Set 1 (CS1), is recommended by ITU-T and ETSI. CS1, at its current status (autumn 1993), is described in Section 2.4.4 and it is also described, together with SIBs, service features, and services, in the Appendix. Examples of service creation in CS1 are described in Section 4.2.1. There are, however, differences between the CS1 from ITU-T and the CS1 from ETSI. ETSI has, for instance, added seven extra SIBs. (Note that this discussion concerns the CS1 of autumn 1993).

CS1 will not be a standard, only a recommendation. The successors, however, CS2, CS3, and so on, are (at the time of publication) intended to be standards. The focus in subsequent *capability sets* will be on such aspects as mobility and interworking between intelligent networks and other networks.

A SIB on the global functional plane must be found in at least one *functional entity* (FE) on the *distributed functional plane* (DFP), but it can also be found in more than one FE.

The Distributed Functional Plane

The DFP, which is described in ITU-T rec. Q.1204 [4] on architecture and in Q.1214 [5] on CS1, contains FEs, which are a unique group of special functions that form a part of the total intelligent network concept. The FEs provide a detailed description and a functional specification to be applied in physical realizations. The cooperation between FEs is achieved by *information flows* (IFs). The DFP architecture is vendor- and implementation-independent.

Figure 2.2 illustrates a typical intelligent network configuration today, including the interworking between FEs. Bear in mind that there is only one *service management function* (SMF), because a network can only contain a single *master*. It controls a number of *service control functions* (SCFs) and a larger number of *service switching functions* (SSFs) in the network. But, as Section 7.6 discusses, we may have to consider cooperation between SMFs in future, for instance, when mixing traffic from different network operators.

The FEs on the DFP, shown in Figure 2.2, are

- *Call control agent function* (CCAF). This function handles user access to the services. The CCAF can be found in exchanges with subscribers, that is, local exchanges.
- *Call control function* (CCF). This function handles all normal calls and the switching of calls and services, both IN-based and non-IN-based. The CCF can also recognize a service that is going to be handled by the intelligent network.

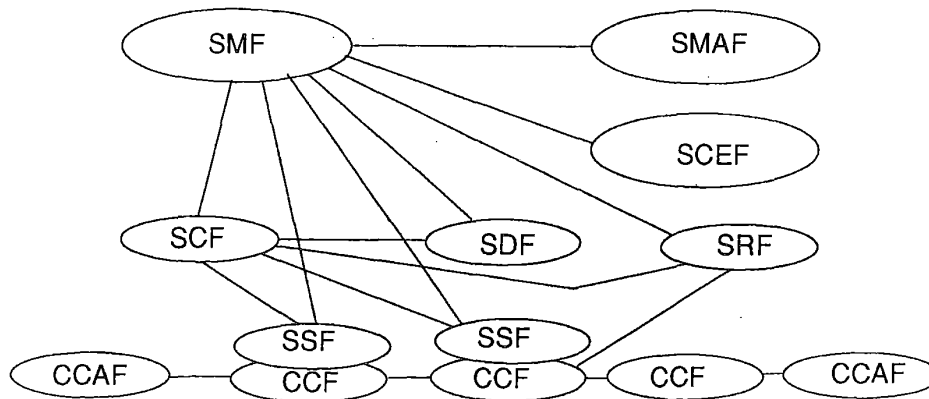


Figure 2.2 The distributed functional plane. (ITU-T Rec. Q.1211, Figure 2/Q.1211.³)

- *Service switching function (SSF)*. This function handles communication between the CCF and the SCF. The SSF is also the function for switching a call or service to a particular location (particular number) on order from the SCF.
- *Service control function (SCF)*. This is the core of the intelligent network. This function controls the whole process of handling calls or services by giving instructions to the SSF/CCF, the SDF, and the SRF. Knowledge of the specific data and logic of a service itself is only found in the SCF.
- *Service data function (SDF)*. This function assists the SCF, providing data about customers and the network.
- *Specialized resource function (SRF)*. This function is used to give and receive data to and from users, much like control facilities for code receivers and speech machines.
- *Service creation environment function (SCEF)*. This function handles defining, developing, and testing of IN-based services.
- *Service management agent function (SMAF)*. This function handles the human interface with the SMF for, for example, operation and maintenance staff.
- *Service management function (SMF)*. This function operates like a spider at the center of web, managing the distribution, control, and maintenance of IN-based services.

The FEs on the DFP are allocated to *physical entities (PEs)* on the physical plane.

3. The full text of Q.1211 may be obtained from the ITU Sales Section, Place des Nations, CH-1211 Geneva 20, Switzerland, Tel: +41 22 730 51 11, Fax: +41 22 730 51 94.

The Physical Plane

The physical plane, which is described in ITU-T rec. Q.1205 [6] on architecture and in Q.1215 [7] on CS1, shows how the aforementioned functions on the DFP can be implemented in physical products on the network, that is, how an FE on the DFP is allocated to a physical entity (PE), or physical node.

An FE must be found in at least one PE, but a PE can consist of more than one FE. (As an example, the PE service switching and control point (SSCP) consists of two FEs, the SCF and the SSF.) However, there cannot be more than one FE of the same type in one PE. In other words, according to the theoretical model, there can not be two SCFs in one SCP. If for some reason you still want to locate two SCFs in the same place, they must, from a theoretical point of view, be regarded as two different PEs, that is, two SCPs.

Moreover, it is not possible to divide an FE between two physical nodes. (Implementation aspects of PEs are discussed below and in Chapters 3 and 7.) Examples of physical nodes on this plane include:

- *Service switching point (SSP)*. This node contains the SSF and call control function (CCF), recognizes an IN-based service, and communicates with the node containing the SCF (often the SCP). If the SSP is a local exchange to which the users are connected, the CCAF is also performed. The SRF may be integrated in the SSP or directly in an IP.
- *Service control point (SCP)*. This node contains the SCF, namely, the service logic that controls the IN-based services.
- *Service switching and control point (SSCP)*. This is a physical node that contains both the SSF and the SCF.
- *Service management point (SMP)*. This node is used for operation and maintenance tasks, such as provision of services, collection of statistics, and so on, which together form the service management function (SMF).
- *Service data point (SDP)*. This node contains the SDF, that is, data used by the service script. An SDP may be accessed from either the SCP or the SMP.
- *Intelligent peripheral (IP)*. This node contains the *specialized resource function* (SRF) and is used to communicate with the users of a service. It can be used for voice recognition, code receiving, protocol conversion, customized messages, and so on, and for communicating with one or more SSPs. The SRF might as well be integrated directly in the SSP.
- *Adjunct (AD)*. This node is similar to an SCP, but has a high-speed interface to the SSP.
- *Service node (SN)*. This node is similar to an SCP and an AD, but has a direct point-to-point connection with the SSP for speech and signaling.
- *Service creation environment point (SCEP)*. This node contains the SCEF and is used for defining, developing, and testing IN-based services.

- *Service management agent point (SMAP)*. This node contains the SMAF and is used to access the SMP. Access can be given to the network operation staff, as well as to external, specially selected customers.

Implementation Aspects

The most common way to implement the SCF and SSF is in the SCP and SSP, respectively. The SCF, for instance, is the control intelligence of services such as freephone. In the early stage, before the first SCP was installed, many networks implemented the SCF for number translation services, such as freephone in a transit exchange. (This is a very common way to start offering these services in a network.) When intelligent network technology was introduced and the first SCP was implemented, the most natural evolution was to move the functionality represented by the SCF to the SCP, along with control of the freephone service.

The SSF is the function that switches the call upon receipt of orders from the SCF. The normal way to implement an intelligent network is to separate geographically the SCF and the SSF and to have one or more SCPs (with the SCF) control an entire network of SSPs for a service. But, to do this, a signaling possibility between the SSPs and the SCP is needed, that is, a signaling system with both high capacity and a flexibility that also allows communication of information other than normal call setup data. In other words, what is needed is a *data communication channel* separated from the speech channels. *Common channel signaling* (CCS) provides that facility, and consequently, CCS is a fundamental base for building the required signaling within an intelligent network. In the *Open System Interconnection* (OSI) application layer (layer 7) in *Common Channel Signaling System No. 7* (CCSS No. 7), which is the most common CCS system today, a protocol, *Transaction Capability Application Part* (TCAP) is introduced to provide the intelligent network with the signaling required. Within TCAP there is a component layer, *Intelligent Network Application Protocol* (INAP) that is used for intelligent network communication, for example, between SSPs and SCPs, between SCP and SDP, and so on. See Section 3.4 and Figure 3.2 for more information about CCSS No. 7 and the TCAP/INAP.

Of course, this provides incentive for the INAP, based on the CCSS No. 7 signaling system, to be available in both the SCP and the SSP. It will, however, be some time before INAP is implemented in many networks; in the meantime, the following approaches can and have been used. Telecommunication networks that contain a separate signaling network, that is, a CCSS No. 7 network but where INAP is not implemented yet—as in the first years of intelligent networks in Sweden—can still physically separate the SSF and the SCF prior to implementing the INAP. This can be done by using the facilities in the CCSS No. 7 to make speech and signaling (quasiassociated or nonassociated signaling) take different paths in the network. (The technique is described further in Chapter 3, Section 3.4.) For networks in which CCS is not implemented, the best way to implement an intelligent network is to allow the SCF and the SSF to be physically present in the same node, that is, in an SSCP. This way, a signaling protocol is not required; the communication between

the SCF and the SSF is performed via an internal interface within the software in the SSCP instead.

The advantage of these last two approaches is that IN-based services can be made available earlier, instead of having to wait for the INAP to be implemented, which is frequently one of the network components for which we must wait the longest. This is because it also influences the nodes (the SSPs) and therefore must be implemented both in one or two SCPs and in a large number of SSPs.

The disadvantage of the SSCP method of implementing an intelligent network is that we must also switch the speech channel to the location of the SCF, that is, to the SSCP. This slows down the capacity of many network services, since the possibility of having a free speech channel between a node (local or transit) and the SSCP determines whether you can reach the SCF or not. This could be especially devastating for mass calling services, such as televoting, as well as for premium rate services. (Refer to Section 3.4 for a description of SSCP and to Section 3.5 for a discussion of mass calling services).

One advantage of the intelligent network is, as described above, that it allows a service to cover an entire network and be controlled from one central point (an SCP) in the network instead of from many (local) exchanges. This facilitates such administrative and management procedures as introduction, change, and withdrawal of services. Centralized control and centralized administration of a service also provide the operator with a better overview of that service in the network. As it becomes easier to introduce new services by using the intelligent network concept, a much wider range of services will undoubtedly become available.

2.2.2 The Introduction of Platforms

The goal of the intelligent network concept is, as described in Section 2.1, to reduce the time required to implement a new service from between two to five years, using conventional technology, to less than six months, using intelligent network technology. How is this possible? It becomes possible by taking a completely new approach to the whole service implementation complex—by building the intelligent network using a *platform* concept (see Figure 2.3).

An advantage of this method is that all the software commonly required for a range of services is already included on a stable platform that seldom changes. Consequently, when a new service must be designed and programmed, only the parts that are unique to the service need to be developed and placed on the platform to make the service ready. Furthermore, the unique service parts are based on supersophisticated programming by means of service-independent building blocks (SIBs). The “programming” of a new service will therefore differ considerably from conventional programming. A suitable range of SIBs will be put together to form a service, and, once appropriate parameters have been set, the service will be ready. In future, using the intelligent network concept, a new service will be introduced in months or perhaps even weeks, provided all the SIBs needed actually exist. If a SIB is missing and must be designed, it will necessarily take longer to introduce a service. But the probability that all the SIBs required for a new service on a

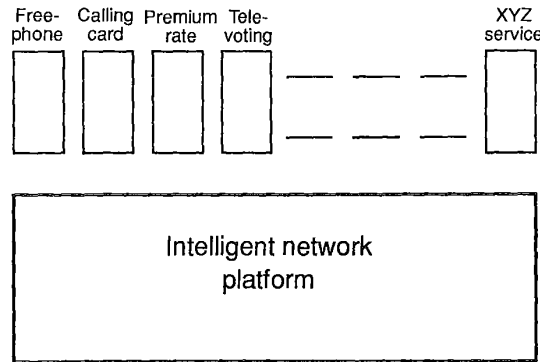


Figure 2.3 The intelligent network employs a platform concept.

major intelligent network will actually exist should in future be closer to 100% than to 90% because the growing number of services on the intelligent network will result in a growing library of SIBs.

2.2.3 What the Intelligent Network Really Is

Phase One: Separation of Switching and Intelligence

The introduction of intelligent networks represents something of a revolution in that we are starting to separate the intelligence—that is, the logic to set up a call or to control a service in the network—from the switching. Intelligence and switching functions have historically been very closely integrated in the telephone exchange, with no possibility of separating them. A call is set up according to a step-by-step procedure, with each node deciding which node to proceed to next, making it impossible to obtain a total overview of the call setup from one location. The situation is the same for setting up a service, like freephone, through the network. See Figure 2.4.

When the control (the SCF) of a service, such as a number translation service, like freephone or premium rate, is moved from the nodes to an SCP, control of the whole service is centralized at a single point on the network. This point, which is the SCP, has a total overview of the execution of the service, including logic, data, and management. If the service is changed or withdrawn from the network, or if a new service is introduced, it is mainly the SCP that is affected. Of course, we have to “teach” the SSPs to call the SCP upon recognizing an indicator that a service must be activated and then to wait for orders from the SCP. This can be described as a *liberation* of intelligence from switching in the networks. See Figure 2.5.

Along with this liberation, it also becomes possible to put intelligence in more than one place on the network. Provided there is good and secure network signaling, we can distribute intelligence, that is, the control facility of a service, anywhere on the network.

can be achieved. Both the intelligent network method and conventional methods of implementation are examined.

2.3.1 Conventional Service Implementation

The conventional method of introducing services is, as everyone knows, a rather time-consuming and complicated task. The introduction of a new network service using conventional techniques (conventional programming, testing, and integration in every network node) requires between two and five years in an ordinary network today. The requirement to reduce this time drastically is the principal driving force behind the intelligent network concept, as discussed in Section 2.1. A lack of flexibility characterizes the service packages available in a conventional network. The packages are often highly inflexible, stable, and limited in number of services. Moreover, these services have usually been available to users for quite some time.

This last observation is especially true, as many of the services introduced in the network by conventional techniques were available in PABXs long before they were introduced into public networks. Examples include call forwarding unconditional (CFU), call transfer (CT), and call completion to busy subscriber (CCBS), all of which were implemented in PABXs in the 1970s before becoming a reality in some public networks in the 1980s. Consequently, subscribers who have been using both PABXs and the public network have been using these services for at least a decade.

The number of services implemented by conventional means in public networks is normally somewhere around 20 to 30 and will probably never exceed 50 in a particular network. The way in which a specific service is implemented, which, from the subscriber's point of view, could affect the way the service behaves, often differs from network to network. Nevertheless, a certain network usually never uses more than one method of service implementation for a single service if conventional techniques are used.

Furthermore, in a particular network, it is extremely difficult to find different features or variants of a particular service adapted to different customer (subscriber) needs. This means that each subscriber, or user, in a network has exactly the same type of service. However, when comparing different networks, the same service may have been implemented using a different method and may be functioning differently. Often, these minor differences in service implementation are disastrous when a service is used by users in different networks. Call completion to busy subscriber (CCBS) is an excellent example of a service that, at the time of publication, is very difficult to get to work outside your own system.

2.3.2 Service Implementation on Intelligent Networks

The main goals of introducing services using the intelligent network concept are to solve the problems and eliminate the inflexibility described in Section 2.3.1 and to reduce the time required to implement a new service to less than six months—from the time the decision is made until the service is available to subscribers. This can be accomplished by us-

ing a stable platform that contains common functionality for many services. The unique parts of a particular service are placed on top of the platform, creating a complete service for the users. Only a minor part of the service has to be developed. Furthermore, the unique parts are built using the SIB technique. (See Section 2.2.1.) From a library of SIBs, which are large software building blocks, a suitable number of SIBs are chosen to form a service. (See Figure 2.10.) Once the necessary parameters are established, the service is placed on the platform, tested, and, finally, offered to the subscribers. (See Figures 2.11 and 2.12.) Consequently, the testing procedure is rather short. But there is another type of testing that will become absolutely essential in future, namely, testing of the interaction between a new service and existing services. (See Section 6.4.)

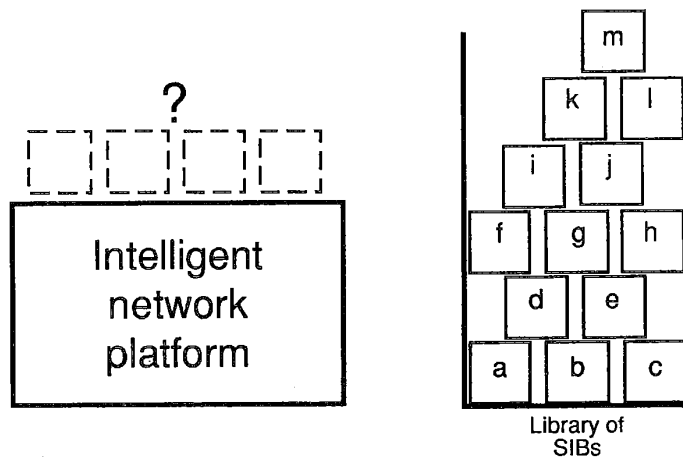


Figure 2.10 The method for building services in intelligent networks.

Service-Independent Building Blocks

Section 2.4.4 and the Appendix describe the SIBs in CS1, from ITU-T. In future, CS1, or at least the same functionality as CS1, will probably be implemented in all intelligent networks; however, as CS1 was not ready nor stable at the time intelligent networks became available, the SIBs in CS1 are not exactly the same as the SIBs used in networks today. Today, SIBs are, of course, highly influenced by the products offered by different intelligent network vendors. Although the basic principles for creating services, using platforms, and offering SIBs are the same for all vendors, no two vendors offer identically functioning SIBs. Other differences between CS1 SIBs and the SIBs offered by vendors are the size and, consequently, the number of SIBs needed to create a particular service. In other words, different intelligent network implementations use different levels of SIBs; some use a few large SIBs, while others use a large number of smaller SIBs to achieve the same

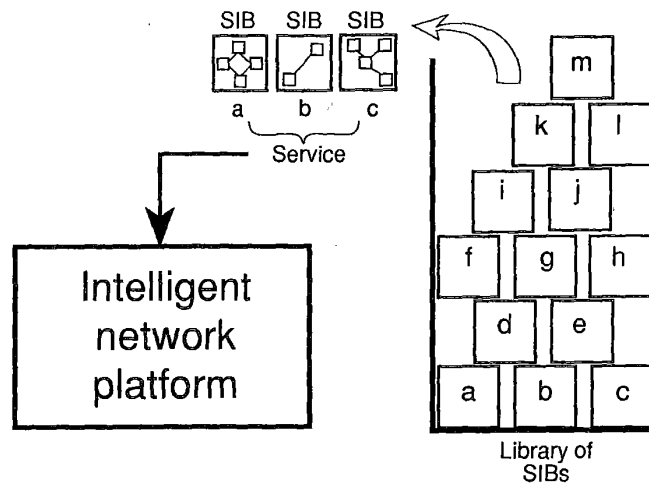


Figure 2.11 SIBs are combined to form a service.

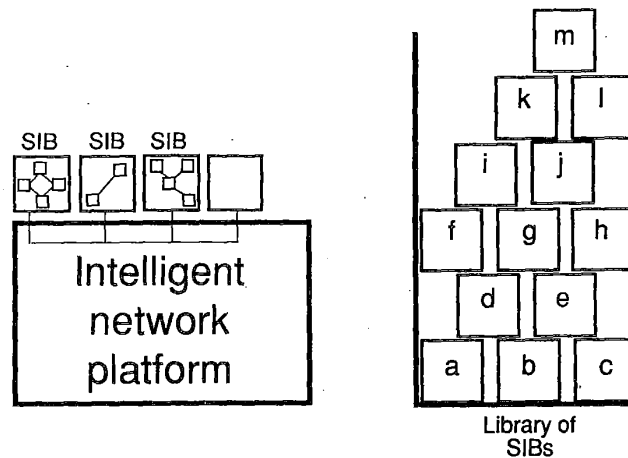


Figure 2.12 The service is placed on the platform.

service functionality. The general method of using basic software blocks is, however, the same for all intelligent network implementations.

Typical functionality in intelligent network implementations (driven by real needs) includes SIBs that handle

- Number analysis

- Time-dependent translation
- Call origin-dependent translation
- Percentage distribution of calls to different destinations
- Queuing possibility

Combining SIBs into Services

SIBs are combined in a particular way to form a complete service. (See Figures 2.10, 2.11, and 2.12.) When the appropriate SIBs have been combined, only a few parameters need to be established to make a service ready. If the appropriate SIBs exist and the person creating the service is experienced, the procedure can be accomplished in about one month. (Section 4.2.1 describes how to build services for intelligent networks in more detail and provides examples.)

Service Provisioning and Customization

Once a service is created, it is considered a *basic version* of the desired service. But different users of the service have different requirements; therefore, the service has to be adapted to meet the special needs of each customer. This is called service provisioning and customization. Consider, for example, how a pizzeria delivery service that has multiple locations and thus needs to locate the origin of calls uses the freephone service. Here, the network operator and the pizzeria management will have to work together to divide delivery service into geographic areas. The divisions can also be time-dependent, that is, customer calls can be routed to different locations depending on the time the calls are placed. (Section 4.2.2 describes provisioning and customization of services and provides more detailed examples.)

Service Implementation Demands an Overall Approach

The aforementioned IN-based service is implemented on the intelligent network platform on the network (in an SCP) and can then be reached from many SSPs. This is a new approach to the technical aspects of installing services whereby, primarily, the time needed to design a service is drastically reduced. But there are numerous other matters that must be taken care of before a service can be made available to subscribers. Therefore, to really reduce the introduction time for services, it is necessary to take an *overall approach to introducing an intelligent network*. The technical platform described above is only one factor and will not be worth anything unless the other systems in the network—management systems, charging and billing systems, and so on—can be updated in the same short time. Accordingly, an overall approach to the whole chain of systems must be considered in the rapid introduction of services on a network. (See Figure 2.13.) (This is further discussed in Section 6.3.)

The software design period, which, prior to development of the intelligent network concept, was the major component of introducing new services, will become the shortest

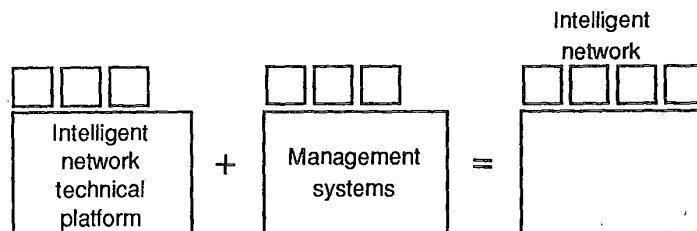


Figure 2.13 Introducing intelligent networks requires an overall approach.

part. Instead, introduction time of a service will be determined in future by the ability of the surrounding systems to adapt to the IN-based way of building services. Consider the following:

- An organization decides to implement a new, conventional service as soon as possible. Software design will take 18 months, testing and integration, six months; in total, it will take 24 months from start until the service can be used on the network.
- The start time for software design is also the start signal for preparation of the management systems that must be ready when the service is put into operation 24 months later. Management system preparation takes 12 months, so there is no hurry to complete that task compared to the 24 months needed for technical development.
- When the intelligent network is introduced, technical development is reduced from 24 to six months, a reduction of 18 months.
- But development of the management system still takes 12 months! To avoid a bottleneck, management system development must also adopt an IN-based way of working that will reduce development time to six months or less.

Withdrawal of an Undesired Service

Organizations must also be able to close or withdraw an undesired service quickly from a network. This is an often-overlooked but highly important matter, usually necessitated by a severe situation in the network, for example, misuse of a service, problems with service interaction, or charging problems. The activities around a withdrawal must also include closure of the service in the management systems.

2.4 THE EVOLUTION OF INTELLIGENT NETWORKS

2.4.1 How It All Began

Intelligent networks emanated from the first introduction of the 800 service (freephone) in the United States in the 1980s. The term “intelligent network” was introduced by BellCore

around 1984 and signaled the start of a new era in telecommunication. At this time it became clear that centralized intelligence for controlling services nationwide could provide such extensive benefits that it was worthwhile to build a separate node *service control point* (SCP) exclusively for this purpose.

2.4.2 Bellcore's IN/1, IN/2

So, when intelligent network was conceived, planning for its future began. BellCore was first on the scene again, defining two concepts, IN/1 and IN/2, which, if implemented in order, would gradually increase the intelligence in a network.

IN/1, the easiest to implement, was a straightforward concept for introducing number-translating services like freephone, that is, services for which the SCP was a simple "translator" from one telephone number to another. Concepts similar to IN/1 have been implemented in most countries today and represent a natural first step toward implementing intelligent networks.

With the IN/1 concept, when a new service was introduced, *both* the SSP and the SCP had to be updated, a fact that was seen as a drawback for the future. Consequently, it was decided that in the IN/2 concept *only* the SCP would be updated for a new service. The idea was to simply download the software for a service from the SCP to the SSP via the network. Very soon, however, it became evident that the IN/2 concept was too difficult to implement in one step, so an interim concept was discussed (IN/1+). This was also later abandoned and instead focus became fixed on a new concept, *advanced IN* (AIN).

2.4.3 Advanced IN

The main idea of AIN, compared to its predecessors, is to be *service-, switch-, and equipment-independent*.

In 1991, BellCore presented a technical advisory on AIN Release 1. It was primarily meant to be a target document; a direct step to AIN Release 1 would be impossible. The result, however, has been that CS1, from ITU-T/ETSI, has become more or less a subset of BellCore's AIN Release 1.

2.4.4 Capability Sets from ITU-T and ETSI

In the standardization of the intelligent network concept in ITU-T and ETSI, work has been divided into a number of concepts, the first ones being Capability Set 1 (CS1), Capability Set 2 (CS2), Capability Set 3 (CS3), and so on.⁴ The term "capability set" refers to the set of services and service features that can be constructed by using the SIBs contained in each concept.

4. Note that this discussion relates to the status of intelligent network technology in autumn 1993.

There are, however, distinctions between the capability sets from ITU-T and those from ETSI. For CS1, ITU-T has defined 13 SIBs plus a SIB for the *basic call process* (BCP). ETSI has defined the same SIBs plus seven more to create an ETSI CS1. The CS1 from ITU-T is described briefly here and in more detail in the Appendix. Examples of service creation are described in Section 4.2.1.

CS1 was formulated as a standard, but will only be a recommendation. CS2 and CS3, on the other hand, are aimed at being standards, and their focus is on such items as mobility and interworking between intelligent networks and other networks. The main idea is to standardize interfaces and SIBs, but not the services themselves. However, to facilitate understanding of how to use the SIBs, CS1 also describes a set of service features and services that can be built using its SIBs. These services represent a *minimum set of services* that can be built with the help of the SIBs in CS1 and may be particularly useful when an intelligent network implementation is being tested for conformance to a capability set. (Conformance testing determines whether a system meets the requirements of a certain standard.) In future, it will, of course, be possible to build more than these services; this is also a main goal of the intelligent network concept.

In CS1, the following limitations are imposed:

- Services should be *single-ended*, that is, the service should concern only one user.
- Services should have a *single point of control*, that is, only one service control function may be involved, and an SSP may have a relationship with only one SCP. Direct communication between two SCFs is not permitted in CS1.
- Services defined by CS1 will support the networks PSTN, ISDN, and PLMN.
- The only interfaces defined in CS1 are between:
 - The service control function (SCF) and the service switching function (SSF);
 - The service control function (SCF) and the specialized resource function (SRF);
 - The service control function (SCF) and the service data function (SDF).

The SIBs defined in CS1 from ITU-T (Rec. Q.1213) [3] have the following functions. (The SIBs in CS1 and the minimum set of services they support are described in detail in the Appendix; examples of service creation are given in Section 4.2.1.)

<i>CS1 SIB Functions</i>	<i>Description</i>
Algorithm	For mathematical algorithms
Charge	To influence charging
Compare	To compare two values
Distribution	Call distribution to different destinations
Limit	To limit the number of calls for a service
Log call information	To log detailed information about a call
Queue	Sequencing of calls to a called party
Screen	To check if a number is in a specified list
Service data management	Management of end user data

<i>CSI SIB Functions</i>	<i>Description</i>
Status notification	To check the status of a service
Translate	To translate input information into output information
User interaction	To exchange information between the network and involved parties
Verify	To compare collected information with expected format
Basic call process	A specialized SIB, providing the basic call capabilities

The following list contains the minimum set of services that can be built using the SIBs in CS1. (A functional description of each SIB and the services it supports is provided in the Appendix.) The services are, consequently, *fully* supported by the SIBs in CS1 from ITU-T (Rec. Q.1211) [8]:

<i>CSI Minimum Services</i>	<i>Description</i>
Abbreviated dialing (ABD)	Use of short numbers
Account card calling (ACC)*	Call from any terminal, charging an account
Automatic alternative billing (AAB)*	Billing via a separate account
Call distribution (CD)	Routing incoming calls to different destination(s)
Call forwarding (CF)	Forwarding of incoming calls to another number
Call rerouting distribution (CRD)	Rerouting of incoming calls due to a condition of busy, after a certain number of rings
Credit card calling (CCC)*	Calls from any access points charged to a specified account
Destination call routing (DCR)	Routing of incoming calls depending on time, date, originating area, calling line, etc.
Follow-me diversion (FMD)	Remote control redirection of incoming calls
Freephone (FPH)	Reverse charging
Malicious call identification (MCI)	Logging of incoming calls
Mass calling (MAS)	Preparation for a mass calling service
Originating call screening (OCS)	Restriction of outgoing calls
Premium rate (PRM)	Partial pay back for a call to the called party
Security screening (SEC)	Security screening of users seeking access to network, system, or applications
Selective call forwarding on busy/no answer (SCF)	If called party is busy or does not answer, only particular, preselected calls may be forwarded
Split charging (SPL)	Split charges between calling and called parties
Televoting (VOT)	Preparation for a televoting session
Terminating call screening (TCS)	Restriction of incoming calls
Universal access number (UAN)	Using one number for several incoming lines
Universal personal telecommunications (UPT)	Using one unique number across multiple networks at any access

(continued)

User-defined routing (UDR)	The user may specify routing of outgoing calls
Virtual private network (VPN)	Building a private network using public network resources

*These services, initially different from each other, are very similar today. CCC, for instance, can also be used without a card. All three, ACC, AAB, and CCC have the common characteristic that a call may be charged to an account.

The following two services are only *partly* supported by the SIBs in CS1:

<i>Partly SIB-Supported Functions</i>	<i>Description</i>
Call completion to busy subscriber (CCBS)	Calling party is automatically informed when called party becomes free
Conference calling (CON)	Multiple parties participate in one single conversation

Standardization efforts are under way for CS2, but the SIBs and services have not been defined yet (autumn 1993).

2.5 INTELLIGENT NETWORK USERS: REQUIREMENTS AND OBLIGATIONS

A general requirement of services in an intelligent network is that they should offer at least the same degree of availability to the user as ordinary telephone traffic, which is quite high in most countries today, at least compared to many other systems used in daily life. (Regarding future user needs, please refer to Chapter 7.)

2.5.1 Service Subscribers and Service Users

Service subscribers and service users are the first groups who come to mind when intelligent network users are considered. The distinction between the two is that service users use the services and service subscribers own the subscription (pay the bills). Often they are the same people, but not always. Within a family or a company, for instance, there is often one person who owns the subscription and several others who use the services.

Service users reach services via routines managed and offered by the network, that is, via their terminals (telephones). The services themselves are provided by service providers, who can be network operators or external providers offering service via the network. Users want services well-suited to their private and/or professional lives. They want good, easy-to-use, and stable services. These same requirements apply to the user interface. A "stable" service is one that functions as close to 100% of the time as possible and does not constantly change due to the operating sequence.

changing) a basic telephone network is enormously difficult, very expensive, and time-consuming. So, you will have to live with the basic telephone network you already have, at least in the near future, which, in fact, corresponds to the timeframe for building the first intelligent network platform. Of course, you can make minor improvements to your telephone network, but major changes, such as introducing a networkwide Common Channel Signaling System (CCSS) No. 7, take a very long time. Consequently, you have to find an intelligent network solution that makes the best use of your existing network and “excuses” the parts that are not quite up to date.

The following sections discuss different approaches to implementing an intelligent network, based on the basic telephone networks from which you can start. The discussion does not cover every aspect and detail of implementing an intelligent network, but the main components and the main system functions in a network are considered from the perspective of an intelligent network implementation.

3.3 VALUABLE FUNCTIONS IN AN EXISTING NETWORK

The tasks of building, changing, or expanding an intelligent network platform are facilitated if the existing network possesses any of the following characteristics:

- A CCSS No. 7;
- An automatic message accounting (AMA) record technique (toll ticketing) charging system;
- Standardized interfaces;
- A single vendor situation (i.e., SSPs and SCPs from the same vendor);
- Stored program control (SPC) exchanges, both local and transit;
- Digital transmission and Pulse code modulation (PCM);
- Dual tone multifrequency (DTMF) interfaces widely available to the subscribers (i.e., pushbutton terminals instead of older dial pulse (DP) telephones);
- Calling line identification capabilities;
- A well-established organization for marketing and managing value-added services (VASs).

These are probably the most important characteristics. You can manage without all of them, but the more you have, the more rapidly and inexpensively you can implement a good intelligent network platform and start to provide IN-based services.

A CCSS No. 7 is a valuable component to have in your network because it provides a 64-Kbps “data communication channel” for almost free use, which is necessary for, for example, handling communication between SSPs and SCPs. The AMA record technique provides the capability to move all charging activities outside the network (outside the technical system), which affords a greater flexibility in the tasks of setting and changing tariffs, providing specific billing, and so on. The use of CCSS No. 7 and the AMA record technique are discussed in more detail in Sections 3.4 and 3.6, respectively.

Standardized interfaces facilitate the use (purchase) of products already available on the market and are also a means of obtaining a good overall structure in the network. Sometimes, however, standardized protocols tend to be performance-consuming, regarding CPU capacity.

Being able to rely on a single vendor may make building an intelligent network platform and managing the network easier. Everything works more smoothly because you do not have too many different products or too many different vendors to cope with. It may not be good for your wallet in the short run, however, to be stuck with only one vendor.

The benefits of SPC exchanges, digital transmission, and PCM have become so obvious today that we often forget to mention them. It is true, however, that there are still many older non-SPC exchanges operating on networks, mainly as local exchanges, and their subscribers are often excluded from the use of many IN-based services due to older signaling protocols. These protocols make it impossible for a user to, for example, transfer the identity of the calling party out on the network, which is a requirement of many new services, both IN-based and node-based. Also, non-SPC exchanges often present users with restrictions to services that need high charging flexibility for different tariffs, like, for example, premium rate services.

Another problem with older exchanges is that they often have no capability for receiving DTMF pulses. A DTMF interface at the subscriber's phone or terminal is mandatory for using most value-added services (VASs), including those implemented on the intelligent network platform. While it is true that subscribers connected to nonSPC exchanges may use DTMF phones, signaling between the local and the transit exchange nonetheless presents a huge obstacle to using many services.

Calling line identification (CLI) is a very useful function in the implementation of VASs. This function makes it possible during call setup to have the calling party's identity available for use in different places or to have it transferred to the called party. This can be used in a variety of ways in combination with VASs and is further discussed in Section 6.2.2.

As was previously mentioned, it is not only the technical platform that is important in the implementation and marketing of IN-based services; a well-established organization that is already accustomed to conventional (not IN-based) services offers another excellent "platform" to start from.

Customers who have public switched telephone network (PSTN) access are undoubtedly the most interesting target group for IN-based services today and in the foreseeable future. This is because PSTN access (a normal telephone) can be found in practically every home and in every business today, while integrated services digital network (ISDN) access and ISDN terminals are not yet widespread and are not expected to become readily available for many years. To provide all customers with an ISDN access is expensive and will take a long time. However, IN-based services combined with access forms like ISDN, both narrowband and broadband, will probably become a greater market in the long run.

3.4 INTELLIGENT NETWORK IMPLEMENTATIONS WITH OR WITHOUT CCSS NO. 7

The most important items for standardization in intelligent networks are the interfaces, with the SSP/SCP interface being the most important and the most urgently needed in the short term. The reason for this is that it is the *basic* interface. If it is implemented, a first generation of intelligent network services, for example, translating services, like freephone and premium rate, can be opened. It is not necessary to have interfaces to, for example, service data points (SDP) and intelligent peripherals at once; these can be implemented in a second step.

3.4.1 Signaling Types

There are two basic parts of a communication link between two points on a network:

1. The speech channel, on which calls are conveyed (a "speech" channel is equivalent to other channels that transmit information between users, i.e., for data, text, pictures, and so on);
2. The signaling channel, which handles control information for the call, setting up a speech channel, releasing it, transferring numbers, and so on.

Signaling between exchanges can be carried out in two main ways, by using either

1. *Channel-associated signaling* (CAS), or
2. *Common channel signaling* (CCS).

CAS was formerly the only method for signaling offered on networks. It is called "channel-associated" because the signaling for the speech channel is conveyed in close relation to the speech channel itself. (In the old analog systems, signaling was carried out on the same physical wire the speech was transmitted on). CAS is also carried out in a fixed and static manner that does not allow, for example, new signals for new functions to be added. The number of possible messages that can be transferred is limited, and the capacity is low due to this rigid connection to only one speech channel. This is because CAS was established in the former analog networks in which there was no need to do more than transfer simple call setup information.

When digitization came into practice and the SPC technique was introduced, exchanges were interconnected via *pulse code modulation* (PCM) transmission techniques. Suddenly, in a 2-Mbps system, there was one 64-Kbps signaling channel for handling signaling for 30 speech channels. The first PCM systems were created according to the old CAS principle, but they eventually gave way to the CCS principle, which was more advantageous. CCS offers a more efficient use of signaling channels and the freedom to transfer call setup data and any other information via these channels. Moreover, as the SPC exchanges are controlled by CPUs, it is very beneficial to establish a direct data communication

channel between the CPUs in both exchanges that need to communicate, i.e., to introduce CCS links.

A CCS system between two exchanges works with two autonomous signaling channels, one in each direction. These two channels are synchronized by the two nodes so that signals forwarded on one channel are acknowledged on the returning channel and vice versa. What is interesting to consider here is that if we only use the signaling channel to handle call setup data for the 30 speech channels, we normally only use 10–20% of the total capacity (which is 64 Kbps). The other 80–90% is free and can be used for other purposes, such as, for example, to transfer messages between SSPs and SCPs on an intelligent network.

The most widespread common channel signaling system in use today is CCSS No. 7, standardized by ITU-T. It is widespread in many networks, but, nevertheless, there are a lot of networks that still use older signaling systems. The advantages of using CCSS No. 7 are obvious, and they are not only related to the building of intelligent networks. All networks are more or less heading towards CCSS No. 7.

3.4.2 Common Channel Signaling System No. 7

This section does not attempt to provide an in depth description of CCSS No. 7, which would be a very complex task. However, as CCSS No. 7 is very important in intelligent network implementations, we examine the parts that are essential to implementation. Readers who are interested in acquiring a more comprehensive knowledge of CCSS No. 7, should refer to [1–3].

In addition to aiding intelligent network implementation, CCSS No. 7 also offers the freedom of transferring signaling between the two exchanges in three different modes (see Figure 3.1):

1. An associated mode (path “a” in Figure 3.1);
2. A quasiassociated mode (a predetermined path “b” in Figure 3.1);
3. A nonassociated mode (path “b” chosen in some more dynamic way, maybe randomly).

Do not confuse the signaling modes with the CAS, which, as we saw in Section 3.4.1, is something entirely different.

In the associated mode, signaling *always* takes the same path (not the same channel) as the speech channel on the network. In the quasiassociated and nonassociated modes, signaling is routed independently of the speech channel, that is, speech can go one way and signaling another. In the quasiassociated mode, there are predetermined paths for signaling to follow. The nonassociated mode points to a more general situation in the future in which the goal is to let the signaling setup be completely independent and work autonomously. Quasiassociated could be described as a special case of, or a first step towards, the nonassociated mode.

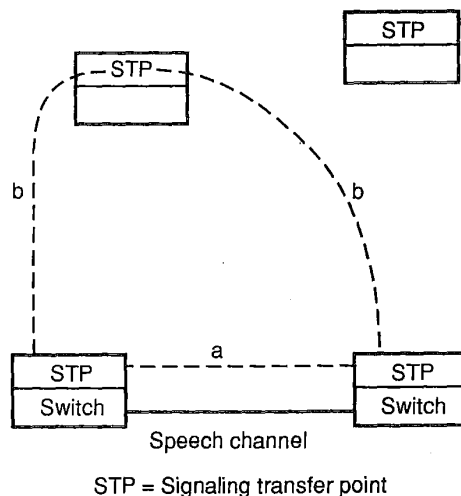


Figure 3.1 Signaling modes in CCSS No. 7.

The quasiassociated and later nonassociated modes separate signaling paths from the speech paths. Therefore, it is very convenient to consider two different (functional) networks, a transmission network for transferring speech and data and a signaling network. The nodes in the network that handle (originate and terminate) signaling in a CCSS No. 7 network are called SPs (signaling points). To transfer the signaling between the SPs, STPs (signaling transfer points) may be used, but not necessarily. The STPs then form, together with the SPs, a separate network for signaling, for our purposes, called a signaling network. An SP, or an STP, is often a normal exchange as a transit exchange, but it can also be a node of its own.

For the purposes of this text, we will regard the signaling network as a network of its own—a 64-Kbps data communication network capable of handling, for example, the communication between the SSP and the SCP.

Figure 3.2 illustrates the structure of CCSS No. 7. The part that is of primary interest in relation to intelligent networks is the *IN application protocol* (INAP) within the component sublayer of the *transaction capabilities application part* (TCAP) on the application layer, that is, layer 7 of the OSI model. The figure does not show every detail of CCSS No. 7 structure; rather, it indicates where the parts that are relevant to intelligent networks are located with respect to the OSI 7-layer model and the signaling 4-layer model. The acronyms in the figure stand for:

<i>Acronym</i>	<i>Term</i>
ASE	Application service element
TCAP	Transaction capability application part
ISP	Intermediate service part

ISUP	ISDN service user part
SCCP	Signaling connection and control part
TUP	Telephone user part
DUP	Data user part
MTP	Message transfer part

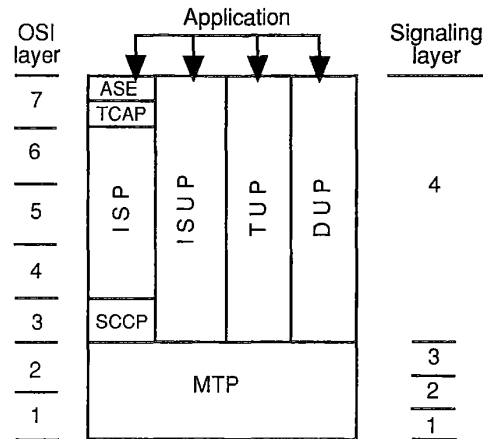


Figure 3.2 The structure of CCSS No. 7.

The *telephone user part* (TUP) is employed by existing networks that use CCSS No. 7 for normal speech traffic (normal calls). Networks can also use a modified TUP for communication between the SSP and the SCP. This is described in more detail later in the text.

3.4.3 Implementations that Use Service Control Points

When SCPs are used to control the intelligence for a service, only signaling is used between the SSP and the SCP. This section describes three approaches to implementing intelligent networks with SCPs; the differences between the three approaches depend on the signaling capabilities available. To facilitate understanding, the examples describe the implementation of a service that uses number-translating, the freephone service. The same approaches are also valid for other number-translating services, such as premium rate and universal number.

A Freephone Implementation that Uses CCSS No. 7 with the TCAP and the INAP

Using TCAP/INAP offers numerous benefits. First, it provides a standardized protocol, which allows an open approach to different systems and different vendors of intelligent network equipment. Second, and more important, it offers the capability of transferring other types of information (not circuit-related information) besides information related to

call setup in the interface. Third, it provides more security in the communication from the protocols. Fourth, services that use devices to broadcast recorded announcements to subscribers during call setup (examples are provided in Chapter 4), can use the signaling network to control the announcement device in the subscriber's own transit or local exchange by a remote SCP.

Figure 3.3 shows a typical freephone service setup that uses the TCAP/INAP on a CCSS No. 7 network. This solution allows signaling and speech to be separated; the TCAP/INAP works as a protocol conveyed by signaling channels, which, in turn, are regarded as a *separate network* (from the speech network), or a signaling network. The freephone service steps, illustrated in Figure 3.3, are:

1. The local exchange hosting the calling party recognizes a freephone call from subscriber A by analyzing the called number.
2. When it finds a freephone number, the local exchange routes the call (both speech and signaling) to an SSP, which is often the transit exchange.
3. At the SSP, call setup is suspended while a setup to the SCP is made via the CCSS No. 7 signaling network, using the messages conveyed by the TCAP/INAP.
4. In the SCP, the freephone number is translated to the C number (i.e., the destination number for the freephone call).
5. The C number is returned on the signaling network to the SSP that required the translation.
6. When the SSP receives the C number back, normal call setup (signaling and speech) is resumed to connect to the destination local exchange and, finally, the C subscriber.

The TCAP/INAP is the recommended way of signaling between SSPs and SCPs and, in the future, it will undoubtedly replace all other methods. There have not been a lot of TCAP/INAP implementations to learn from to date (December 1993), however, and it is to be expected that they will use more loads in the CPUs in the nodes than the modified TUP (see below) and similar implementations. This is most important, of course, in implementations of mass calling services, like televoting, which can entail sudden bursts of traffic. Perhaps the best solution is to have two protocols on the intelligent network, one that guarantees high security and performance of features (TCAP/INAP) and one that provides less security but offers fast response when that is the essential requirement. This is also discussed in Section 7.4.1.

A Freephone Implementation that Uses CCSS No. 7 and a Modified TUP

The normal procedure when the CCSS No. 7 has a TUP but not a TCAP/INAP is to add some extra signals to the TUP exclusively for handling the communication between the SSP and the SCP. A freephone number setup with a TUP modified in this way follows exactly the sequence illustrated in Figure 3.3. The only difference is the method of signaling

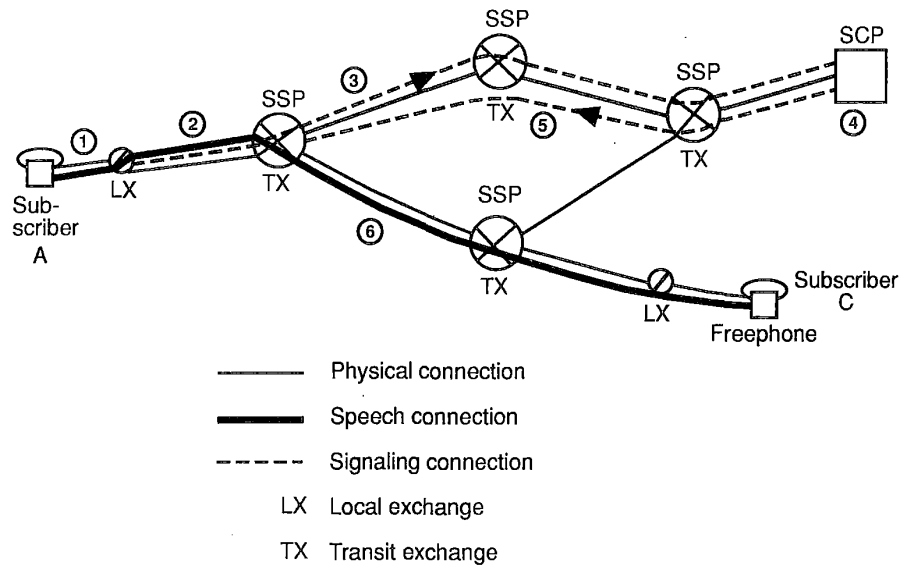


Figure 3.3 A freephone setup with a CCSS No. 7 network and an SCP solution.

between the SSP and the SCP. A modified TUP solution uses the facility within CCSS No. 7 that is offered by the *quasiassociated or nonassociated modes* (see Section 3.4.1 and Figure 3.1) to create different paths for speech and signaling. Only the signaling channel is required, however, to transfer a number to the SCP for translation into another number, which must be returned to indicate where the freephone call must be routed.

Since the speech channel is not needed and the distance between the calling SSP and the called SCP could be very large, it is best simply to have no speech channel at all and to use only signaling. But this is not easily done in networks in which CCSS No. 7 was mainly implemented to set up normal calls. Signaling requires a physical connection. There is a method, however, that can solve this. (It was used in Sweden and some other networks in the first years of intelligent networks.)

The software in SSPs and in the SCP defines direct routes between all SSPs and the SCP. In practice, however, there are no physical connections, only hardware devices on both sides that are physically loop-connected back to the SSP/SCP. In other words, the SSPs and the SCP “believe” there are direct speech channels between them, but there are only loop-connected devices at both ends (see Figure 3.3). When a number translation is needed, the SSP at the calling party’s end calls the SCP, just as if it were a normal telephone call, and a “speech channel” is reserved at both ends of the SSP/SCP connection (but no transmission link exists). As the speech channel (which does not physically exist) is reserved at both ends, signaling can be exchanged between SSP and SCP (dotted lines labeled 3, 4, and 5 in Figure 3.3). CCSS No 7 uses the quasiassociated or nonassociated

mode. When the number is returned to the SSP, the freephone call is set up, just like any normal call setup to C (the destination number).

The advantages of this solution are that it is easy to implement and takes less time to start up with (a simple) intelligent network, compared to the implementation of TCAP/INAP (of course, you must have a CCSS No. 7 network). It is usually not very performance-consuming in the CPUs and it provides short answering times from the SCP.

Drawbacks to the modified TUP solution, compared to the TCAP/INAP solution, are that the type of information that may be exchanged between the SSP and the SCP is very limited and that, after delivering a translated number, the SCP loses control over the rest of the service. This means, for example, that the SCP has no knowledge of the remaining part of the setup of the freephone number. It does not know if there was any answer, if the call failed because of congestion or another reason, and so on.

A Freephone Implementation when CCSS No. 7 Is Not Available

If a freephone implementation employs the SCP, but CCSS No. 7 is not available, you must create a way of communicating between the SSP and the SCP without a speech channel, that is, by using a communication link outside the network. (See Figure 3.4.) Usually a data communication X.25 network can be used (X.25 is described in many books, for instance [2]). The disadvantage of this method is somewhat longer response times compared to the CCSS No. 7 solutions. However, if there are no alternatives and if the response times are acceptable—for example, if no real-time requirements exist and a delay of 3–5 seconds is acceptable, or if the communication is a facsimile or a file transfer—this is a possible interim solution. Many networks today have an X.25 network that interconnects the switches and the SCPs. The X.25 network is usually used for network management and exchange operations. It should be easy to use as an interim solution for communication between SSPs and SCPs before CCSS No. 7 is available. Remember too that there are new players in the intelligent network market. Data communication companies are starting to develop SCPs, and they have experience and a long tradition of protocols like X.25.

With this method, it is quite simple to set up a freephone call:

1. The local exchange hosting the calling party recognizes a freephone call from subscriber A by analyzing the called number.
2. When it finds a freephone number, the local exchange routes the call (both speech and signaling) to an SSP, which is often the transit exchange.
3. At the SSP, call setup is suspended, while a setup through a separated data communication network (preferably an X.25) is made to the SCP.
4. At the SCP, the freephone number is translated to C (the destination number).
5. The C number is returned on the data communication network to the SSP that requested the translation.
6. When the SSP gets the C number back, normal call setup (signaling and speech) is resumed to connect to the destination local exchange and, finally, the C number subscriber.

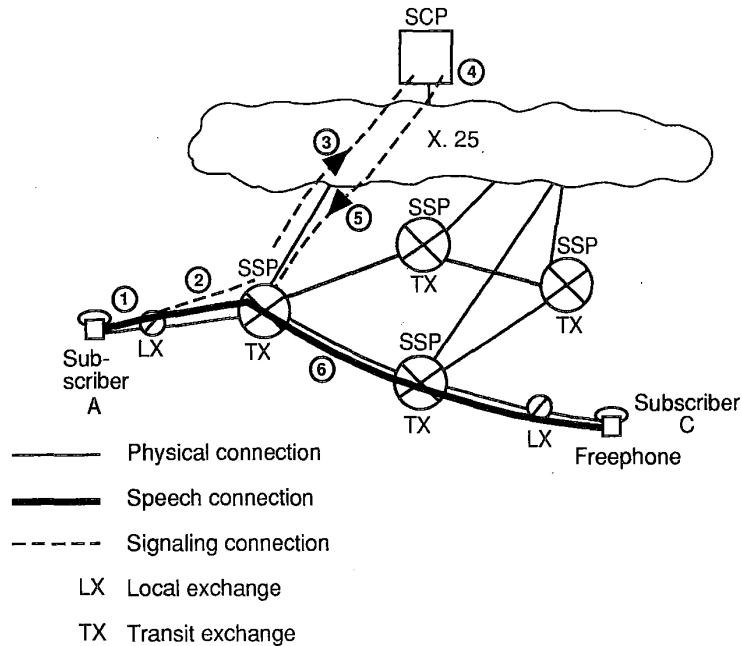


Figure 3.4 A freephone setup using the SCP concept, when CCSS No. 7 is not available.

3.4.4 Implementations that Use Service Switching and Control Points

A service switching and control point (SSCP) is a physical node that contains both the service switching function (SSF) and the service control function (SCF). The interface between the two is an internal software interface within the node itself.

SSCPs are often implemented when no CCSS No. 7 exists on the network, because it is convenient to route the speech channel to the node where SCF is located. SSCP is also the oldest way of implementing intelligent networks; even before the intelligent network concept came into widespread use, the SSCP was a dedicated service node in a central location on the network.

In addition, if a service requires an announcement device, a single announcement device situated at the SSCP can serve a large area (see also Section 7.5.1). Traffic for the service, including signal flow on the speech channel, would be routed to the host SSCP of the device.

In implementations that employ the SSCP solution (see Figure 3.5), the freephone call service works as follows:

1. The local exchange hosting the calling party recognizes a freephone call from subscriber A by analyzing the called number.
2. When it finds a freephone number, the local exchange routes the call (both speech and signaling) to the transit exchange.

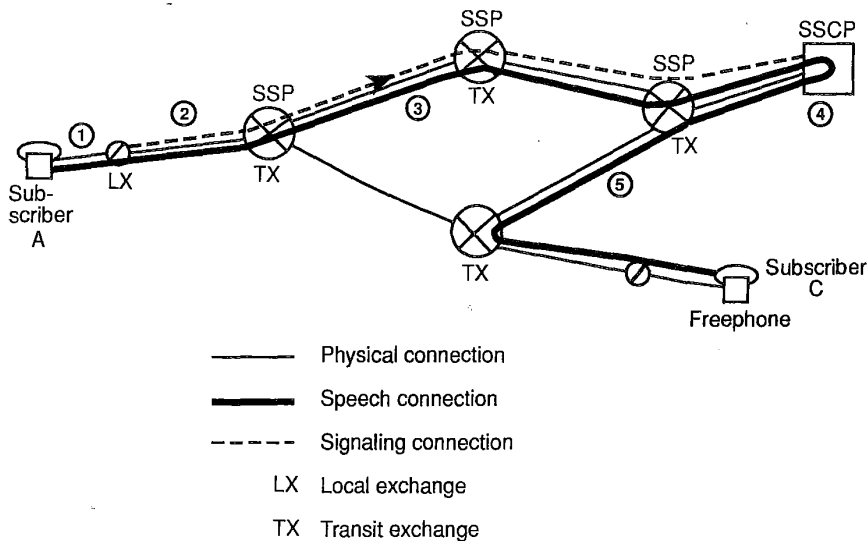


Figure 3.5 A freephone setup using the SSCP solution.

- At the transit exchange, a normal call through the network is set up to the SSF in the SSCP.
- In the SSCP, the SSF calls the SCF to get the freephone number translated to C (the destination number). After that, the C number is returned to the SSF (all of these are examples of internal signaling within the SSCP).
- When the SSF gets the C number back, a normal call (signaling and speech) is set up from the SSCP to the C number subscriber.

It can be seen that the speech path must follow signaling exactly, that is, speech and signaling cannot be separated from each other with the SSCP method. This means that the speech path has to be routed together with the signaling path to the SSCP (where the SSF and the SCF are). In other words, the speech path has to be set up through the SSCP before it connects to the C number subscriber, which often is a waste of speech paths, because there is the risk that the total way in the network for the speech will be very long, compared to the SCP solution. This can cause problems, as speech channels in the network may run out much easier in this way. Consequently, this can cause congestion for other call setups.

3.4.5 Mated Pairs of Service Control Points and Service Switching and Control Points

When an SCP or an SSCP is introduced for a certain service, it is often given the responsibility of all handling and control of the service, for the entire network. This is a classic

case of “putting all of our eggs in one basket.” If the SCP/SSCP fails, availability of the service is lost throughout the *entire* network. To avoid this, some kind of redundancy is necessary. A very handy way of accomplishing this is to work with *mated pairs of SCPs/SSCPs*. Two SCPs/SSCPs working as a mated pair should:

- Be physically separated on the network, and
- At every moment, possess identical software and subscriber data.

Distribution of service traffic differs, though, in the SCP and the SSCP solutions. For mated SCPs, the solution to calling the SCP is often very simple because the normal routing schemes on the network can be used. In the SSPs, the two SCPs in a mated pair can be defined as *first* and *second* choice due to routing algorithms. When the SSP is calling an SCP, the first attempt tries to reach the first choice. If for some reason it does not answer—due, perhaps, to failure or congestion—the network automatically tries the second choice.

With mated SSCP, however, it is not always as straightforward. In this case, the SSF is physically located with the SCF. The solution greatly depends on the signaling used. What is important, in the case of a failure in the SSCP, is whether we notice it in the calling node and can reroute the call to the other SSCP, as in the SCP case, or notice the failure first when reaching the SSCP itself. The latter case is more complicated, as several solutions exist.

Customer control, management functions, service setup using mated pairs, and possible implementation solutions of the future are further discussed in Chapter 7, Sections 7.3.2 and 7.4.2.

3.5 TELEPHONE TRAFFIC THEORY APPLIED TO NEW SERVICES

Until recently, telephone traffic theory, for example, calculating the number of channels in routes between telephone exchanges, dimensioning exchanges themselves, or calculating congestion, was relatively straightforward. We were able to predict traffic relatively well by using general telephone traffic theory. Experience showed that traffic peaks during certain hours of the day, which has led to the “busy hour” concept—the hour of the day with the highest traffic. The duration of calls has always been assumed to average 2–3 minutes, which is true for a *basic call* [3].

The busy hour traffic concept has been used to calculate the load on a network, to allow dimensioning of all transmission and switching equipment. Basic telephone calls have been the dominating traffic on the networks. When value-added services were introduced on networks, such as call forwarding, call transfer, call completion to busy subscribers, and other node-based services for which control is located in the local exchange, traffic theory was still valid. This was because these services were closely *related to basic calls*, that is, the services were initiated by basic calls and they followed the *same load pattern*. Furthermore, the network load from value-added services has been rather limited compared to the load from basic calls.

Chapter 4

Exploitation of Intelligent Networks in Services Today

This chapter describes the current situation (autumn 1993) regarding the introduction of IN-based services, including some of the most common services.

The *first common* set of services to be generally implemented in many networks will most likely be Capability Set 1 (CS1), created by ITU-T and ETSI. CS1 will not become a standard, it will instead be a set of services and SIBs for *benchmark testing*. If the services in CS1 can be built on a network, the network conforms to CS1. Instead, what will be standardized are the physical interfaces, for example, the Intelligent Network Application Protocol (INAP) within the Transaction Capability Application Part (TCAP) in OSI layer 7. Because CS1 was not stable until 1993, it will take a few years to implement it on the networks. (CS1 services, features, and SIBs are described in Section 2.4.4 and in the Appendix.)

The *reality* in networks today (1993) is quite the opposite; as CS1 is stable first now, services, features, and service-independent building blocks (SIBs) are all vendor-dependent, that is, they are dependent upon the way vendors implemented the IN-based services. There are different sets of services on networks, and the same service can be implemented by different methods—IN-based, node-based, or terminal-based—on different networks. Moreover, in the case of IN-based services, the different implementations share no commonality of SIBs. Not even the level (the size) or the functionality of the SIBs are the same. Some SIBs are small (only a few lines of software code); some are larger and can almost be regarded as features.

This chapter focuses on a pragmatic situation involving the most common services implemented today or that will be implemented in the near future. Both services in general and services based on CS1 are described. Because the IN-based services will exist in an environment that includes other (older) services, a total service approach is taken. Consequently, services outside the IN platform are also handled.

4.1 SERVICES AVAILABLE BEFORE INTELLIGENT NETWORKS

4.1.1 General Aspects

Before describing how to best exploit intelligent network platforms to build services, we should focus briefly on the conventional services that existed prior to the introduction of intelligent networks.

To review, Section 2.2 described the three different methods for implementing services: (1) terminal-based, (2) node-based, and (3) IN-based (also called network-based).

Conventional, non-IN-based services were introduced in the 70s and 80s, usually, first as node-based services. Later, some of these services, such as abbreviated dialing and repetition of last dialed number, were moved to terminals, and they became terminal-based services. These conventional services will not be removed from the network simply because intelligent networks are introduced. In future, of course, some of them may *disappear*, because they no longer fulfill modern requirements, or because their functions are extended, or because they are being replaced by other new services. The new services could, in turn, be IN-based, terminal-based, or node-based.

So, instead of being discarded, some services implemented on the network before the introduction of IN, mainly node-based services, will probably be replaced by services implemented either on the terminals or on the intelligent network platform. The advantage of moving services to the terminals is that it minimizes signaling needs between the terminal and the local exchange, for example, when setting up or changing a list of abbreviated numbers. Porting of services to the intelligent network platform will allow a network operator to obtain a better overview and provide improved service management. There will also be a greater demand for interoperability of new services and the older, node-based services.

To sum up, the general trend today is to abandon the old *one and only* way of implementing services on nodes and to consider, when justified, terminal-based or IN-based implementation of conventional services (see Figure 4.1). This trend is further discussed in Chapter 6.

The service repertoire built *before intelligent network technology* may vary considerably from network to network. Some services, however, are or will be present in almost all networks. Without claiming to be complete, this section describes a majority of these services. The function of a basic service may also have small variations from network to network *from the user's point of view*.

In IN-based implementation there is a *planned way* of allowing variations in a particular service, to fulfill different customers' needs. This is one of the *benefits* of building services with the intelligent network platform, namely, the concept of *service provisioning and customization*. Basic services are tailored to suit the special requirements of individual customers. This concept is discussed in Section 4.2, and examples are provided in Section 4.3.

But even two services that appear identical to users might, because of different implementation methods, behave differently, for example, in the way they interact with other services. These differences, which are *implementation-dependent*, are referred to in this

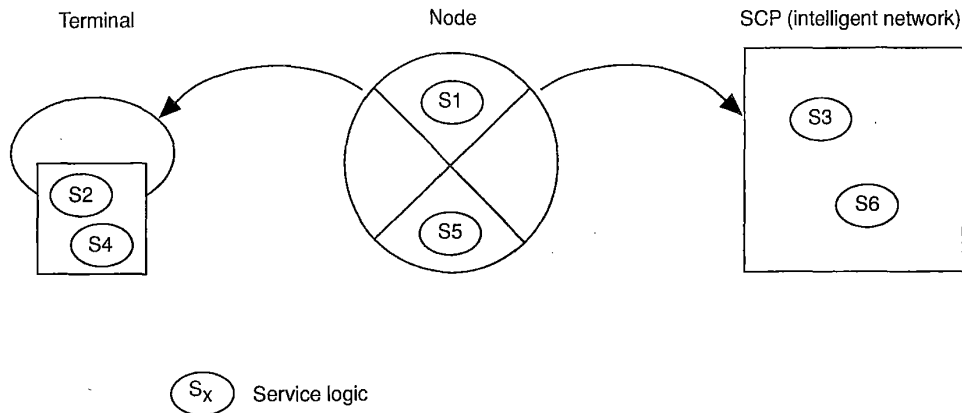


Figure 4.1 Traditional, node-based implementation is no longer the only way today.

text as *service dialects*. Service dialects existed long before intelligent networks were introduced. And, with intelligent networks, service dialects will continue to exist, because many different intelligent network platforms will exist. (Customized services and service dialects are also discussed in Section 6.4, which deals with service interworking.)

4.1.2 Conventional Non-IN-Based Services

While conventional services can be either node-based or terminal-based, the majority today is node-based. Figure 4.2 shows from above the user interface (from the service user's point of view) and from below how deeply the service is rooted (integrated) in the network. "Deeply rooted" is an apt description because it refers to the fact that once a service is embedded (rooted) in the network, it is very difficult to change it. Originally, as we discussed above, the trend was to implement all services on the nodes. However, extensive signaling between nodes and terminals will result if the services become popular. In this case, it is more convenient to move whatever can be moved from the node to the terminal to decrease the load on the nodes and signaling systems. That is why some services today are only implemented in terminals. Good examples of services that are commonly shifted from node-based to terminal-based implementation today include repetition of last dialed number and abbreviated dialing. Figure 4.3 illustrates this situation. Once terminal-based services have been implemented, they are also not easily changed due to, among other things, the large number of terminals spread among subscribers.

The following list describes conventional services that were common in most networks prior to intelligent networks. When possible, the most common method of implementation is mentioned as well. To simplify matters, "A" represents the calling party, "B" represents the called party, and "C" represents the next party involved, where applicable.

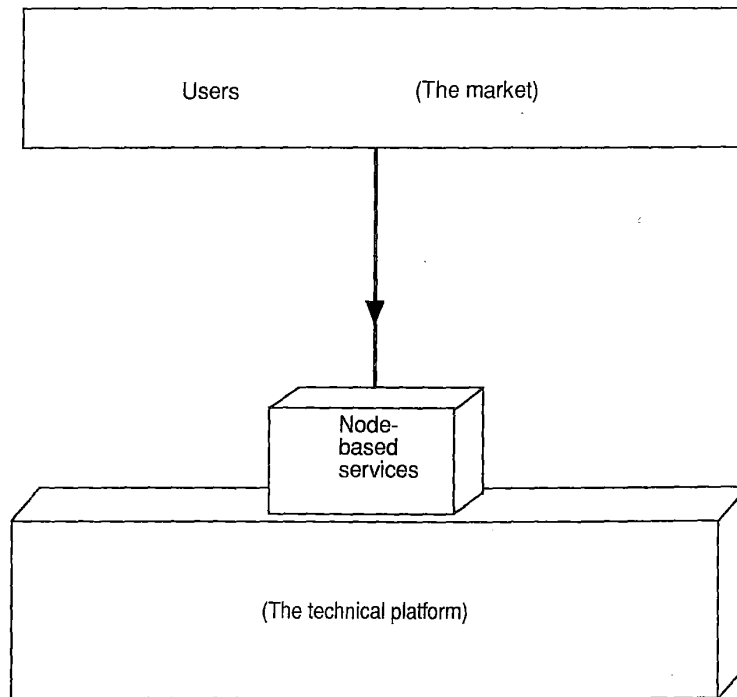


Figure 4.2 Node-based implementation of services.

- *Abbreviated dialing (short number)*. Telephone numbers that A dials frequently may be keyed in as short codes (only 3–4 digits instead of the 10–20 that normally need to be dialed). The short code is then translated to the actual number. Although this service was originally implemented in the local exchange, it is an excellent example of a service that ought to be (and today often is) implemented in the terminal (the telephone) instead.
- *Automatic alarm call (wake-up service)*. This service was first known as the wake-up service. Today, however, it is more generally used to activate a terminal at a pre-determined time. At the indicated time, the service (network) initiates a call to the terminal.

It is important to restrict this service from being combined with services like *call forwarding*. If those two services can be used together, a practical joker can order a wake-up call for another person. To safeguard against inappropriate use of the service, users can only order the service from the phone that is to be called (alarmed). (See also Section 6.4, which deals with service interaction problems.)

Today, this service is usually implemented in the local exchange, but new terminals could easily include it in future. Implementing the alarm function in terminals is technically very easy and can decrease the need for signaling in the networks.

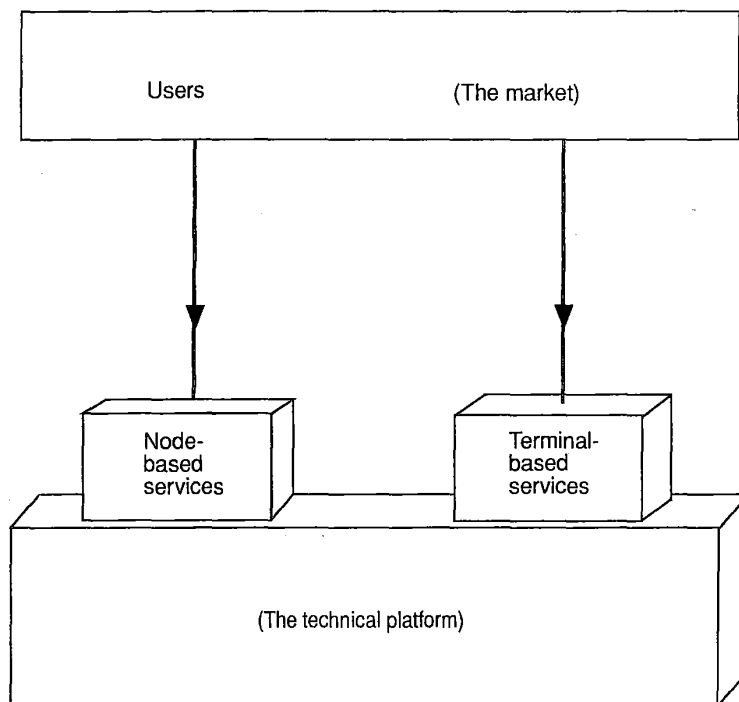


Figure 4.3 Node-based and terminal-based implementation of services.

- *Call completion to busy subscriber (CCBS)*. If A calls B and B is busy, A may initiate *call completion to busy subscriber* from her terminal before putting down her handset. When B becomes free, A's telephone rings in a special manner (a distinctive tone). If A still wants to reach B, she has to lift her handset within a certain period of time to be connected to B. Once B lifts his handset a call is established.

Today, this service is easy to implement if A and B are located at the same local exchange. It is more problematic, however, if two separate local exchanges are involved, as the means of transfer becomes more complicated. From the implementation point of view, the service is divided between the local exchanges of A and B.

- *Call completion when no reply (CCNR)*. If A calls B and B does not answer, A may initiate *call completion when no reply* from her terminal before putting down her handset. Now the network must watch to see when B becomes available. It does this by noticing when B lifts off and releases his handset. Once this happens, A's telephone rings in a special manner (a distinctive tone). If A still wants to reach B, she has to lift her handset within a certain period of time to be connected to B. Once B lifts his handset a call is established.

This service is easy to implement if A and B are at the same local exchange. It is more problematic, however, if two separate local exchanges are involved because the signals between exchanges are more complicated. From the implementation point of view, this service is divided between the local exchanges of A and B.

- *Call forwarding unconditional (CFU)*. If *call forwarding unconditional* is initiated, all incoming calls to B will be forwarded unconditionally to C. The C number must be initiated (and removed) from B's telephone. Today, this service is often implemented at B's local exchange.

A service with the same functionality can easily be implemented on the intelligent network platform by using number translation features. Extra features, like remote control via the customer control interface may then be easily added, if desired.

- *Call forwarding (with condition) (CF)*. Incoming calls to B will, if the necessary conditions are fulfilled, be forwarded to C. The C number must be initiated (and removed) from B's telephone. The most common conditions that require call forwarding today are if B (1) is busy or (2) does not answer within a certain period of time or after a certain number of signals (ringing cycles). This service is often implemented at B's local exchange today.
- *Call hold (CH)*. A party uses the call hold service to temporarily interrupt one call in order to initiate a second call or service. CH is often used in combination with the *multiparty* or *call transfer* services described below. It often has a node-based implementation today.
- *Call pickup*. This service makes it possible for B to receive an incoming call on any phone. Of course, this assumes that B is aware that he is being called. Call pickup can be used, for instance, in a large office where B can hear his phone ring but is too far away to reach it. He can receive the call by picking up the phone nearest him and keying in a code (password). This service has often a node-based implementation today.

This service can also be combined with a *paging system*. When B hears the tone on his pager, he can pick up any phone, dial a code (password), and the incoming call will be connected to that phone.

- *Call screening services (call barring)*. This term refers to a number of services with the same goals: to eliminate A's ability to make dedicated outgoing calls or to prevent B from receiving dedicated incoming calls. For example, A might be forbidden to make international calls because they are very expensive. *Outgoing call screening* services are often used when users have access to phones but do not have to pay for calls, for example, employees. *Incoming call screening* is often used to protect someone from receiving calls from certain A numbers.

Outgoing call screening can be:

- Geographic, where A is only allowed to call within a certain area.
- Limited in number, where A is only allowed to call a predetermined group of subscribers. This service is also called *closed user group* (CUG).
- Dependent on duration, where A is only allowed to speak for a certain period of time before the call is interrupted.

- Dependent on time, where A is only allowed to make calls within certain time periods.
- Restricted, where A is not allowed to call expensive services, such as premium rate services.

Incoming call screening can be:

- Geographic, where B only wants to receive calls from a certain area or areas.
- Personal, where B wants to or does not want to receive calls from certain A subscribers.

These services often have a node-based implementation today, that is, in the local nodes of A or B.

- *Call transfer (CT)*. Call transfer, in contrast to call forwarding, is always carried out during an ongoing call. It is commonly used with call hold (see description). A calls B, and during the call B initiates a call hold for a moment (temporarily leaving contact with A) to make a call to C, perhaps to ask C a question. B now can *pendulate* between A and C. If B decides to connect A directly to C, B can initiate a call transfer to C, that is, B leaves the call and A and C are connected.

This service is often implemented at the local exchange level today.

- *Call waiting (CW)*. A makes a call to B, who is already involved in a telephone call with C. If B subscribes to the *call waiting* service, she is notified (very discretely) that someone is trying to call her during her conversation with C. At the same time, A will not meet a busy tone, just a normal ringing tone. In this situation, B can either ignore the new call or ask C to hold for a moment while she switches to the new call. After talking to A, B can return to her original call with C.

This service often has a node-based implementation today.

- *Fixed destination call (hot line)*. This service makes it possible for A to initiate a call to a predetermined B subscriber without dialing a number. It can be done at the instant that A lifts her handset or after a few seconds, to permit A to make a normal call. This service often has a node-based implementation in the local node of A.
- *Immediate service (ringback price)*. This service allows a caller to obtain information about the price of a call as soon as it is finished, for example, when the caller is using someone else's phone. The service often has a node-based implementation.
- *Malicious call identification*. This service makes it possible for the B subscriber to request a trace of a malicious call. It is usually used when a subscriber receives obscene calls or the like. After receiving an obscene call (A probably puts down the handset after a short period), B has a certain period of time to initiate the service from the terminal. Once this has been done, the A number will be registered by the network and made available to B later on. This service has often a node-based implementation today.
- *Multiparty service*. This service involves calls in which more than two parties are involved, usually *threeparty calls*. Actually, the service performs more than one function. During a normal two-party call from A to B, it allows one party to initiate a call hold (temporarily interrupt the call, see description) and make an inquiry to a third party (C). She can then alternate between the two calls, establish a three-party call,

or leave the call and allow the other two to continue (call transfer). This service often has a node-based implementation today.

- *Repetition of last dialed number (redial facility).* When A calls B and does not reach B for some reason (B is busy or not at home) or if A wants to make a second call to B, this service comes in very handy. It simply remembers the last number A dialed. By dialing a special code at her terminal, A can initiate a new call to B automatically.

This service was initially implemented in the local exchange; however, it is an excellent example of a service that should be (and today usually is) implemented in the terminal (the telephone) instead.

- *Remote control facility.* This mechanism allows a remote capability to be established for many of the services described in this list, even without implementation of an intelligent network and service control points with customer control facilities. Normally, to initiate any of these services, the user must be physically present at the right phone. With a remote control facility, however, the subscriber may initiate a service from any phone. This capability can be implemented in many ways. It has been available in Sweden since 1992 as *remote control of call forwarding unconditional*.

The technical solution, which is applicable to other services as well, is accomplished in the following manner. If, for example, a call forwarding unconditional should be done on someone's home phone while he is at another location, he may reach the remote control facility by making a freephone call. This freephone call is set up to separate control equipment on the network. Such a call may be made from any DTMF phone. After providing a password, for identification and validation, the subscriber may initiate call forwarding to any other telephone number in the world. The service is initiated by the control equipment, which creates the same sequence of star (*), pound sign (#), and the digits that the subscriber would have inserted if at home, and sends the sequence to the subscriber's local exchange.

Of course, the remote control facility is very easy to implement when customer control on an intelligent network is used. Services that are similarly complemented by remote control, for example, call forwarding unconditional, are then easily implemented on the intelligent network platform.

4.2 SERVICE IMPLEMENTATION ON INTELLIGENT NETWORKS

Services implemented on intelligent networks are not built in the traditional way. On the contrary, the intelligent network platform provides greater flexibility to service creation in general and also to the tailoring of services to suit the exact requirements of a particular customer.

This section examines each step of service implementation, like service creation, provisioning, and customization to the needs of the customer, and, finally, network introduction.

enhanced by other service features" [2]. Consequently, we can extend the model in Figure 4.4 by adding the feature level. See Figure 4.5.

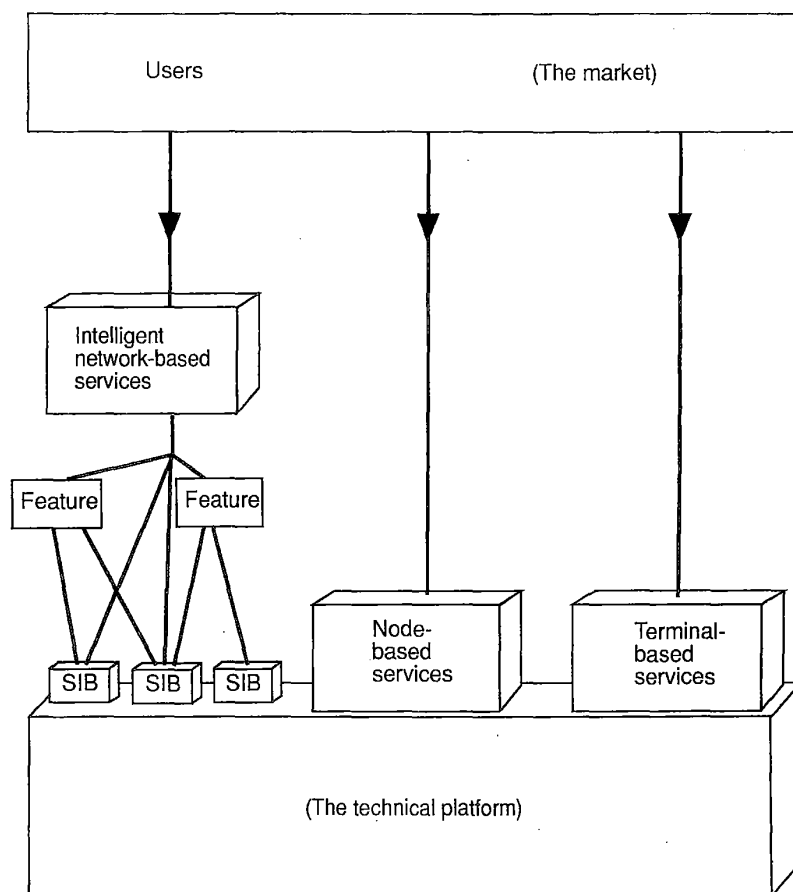


Figure 4.5 The *feature level* is added to service implementation.

Examples of services include freephone and televoting.

Service Creation Steps

The procedure for creating a new service can be easily divided into three main steps:

- Step 1 is the creation of a description of the service based on the user's perception of the way it should function. Normally, this step is best carried out by the marketing department of a network operator. Examples are descriptions of the general

requirements for time- and origin-dependent routing possibilities and customer control functions.

- Step 2 is the creation of a description of the technical aspects of the service, including all error situations and how to handle them. This step is carried out by the technical department and is often done in *software description language* (SDL) according to ITU-T. SDL is often used because it is a high-level language (HLL), but the main reasons are that it is widespread, well known, and standardized.
- Step 3 is the implementation step, which entails loading the service logic into the SCP and updating the routing information in the SSPs and local nodes.

Once you become accustomed to them, steps 1, 2, and 3 can be carried out very quickly (in a few months).

In the very long run, service creation may be carried out by third parties on behalf of customers; that is, the service creation tools may be made available by network operators to customers, who may do the work themselves or hire consultants (third-party service creation). Whether this will become reality or not depends on the possibility of having, first, a *stable, well-defined intelligent network platform*, preferably with standard interfaces and application programming interfaces (API) and second, *extremely good tools* for testing services, for example, regarding service interaction. A new service, wrongly defined and not sufficiently tested, could harm other services and their users.

Service Creation Examples

This section describes some service creation examples based on ITU-T Q 1211 [2]. ITU-T rec. Q.1211 describes the mapping between services and service features in Capability Set 1 (CS1). A table presents the mapping, that is, it indicates which service feature is used in which service. (See Appendix A for a description.) Mapping is divided into *core* and *optional* features. Core features are features that are fundamental to the service, without which the service would not make sense as a commercial offering. Optional features, as the name implies, are not required by the service, and the service can be a commercial offering without them.

1. Malicious Call Identification Service

The malicious call identification (MCI) service is built by combining the following core features:

- *Call logging (LOG)*. Allows a record to be prepared each time a call is received by a specific telephone number.
- *Originating call screening (OCS)*. Allows the served user to bar calls (call screening) from certain areas based on the district code of the area from which the call originates.

No optional features are of interest here.

2. Call Forwarding Service

The call forwarding service is built on one core feature:

- *Call forwarding* (CF). Allows the user to unconditionally address incoming calls to another number. (It should be noted that a service and a service feature can have the same name.)

Call forwarding can incorporate two optional features:

- *LOG*. (See example 1.)
- *Customer profile management* (CPM). Allows the subscriber to manage the service profile in real time, that is, change answering places, control the announcements to be played, perform call distribution, change time-dependent routing, and so on.

Combining the optional feature CPM and the core feature CF creates the remote control of call forwarding unconditional service, the node-based implementation of which is described in Section 4.1.

3. Automatic Alternative Billing Service

The automatic alternative billing (AAB) service allows a user to call from any phone and have the call billed to a separate account instead of charging the calling or the called line. It is built by combining the following core features:

- *Authorization code* (AUTZ). Allows a user to get special calling privileges according to the authorization profile. Different sets of calling privileges can be assigned to different authorization codes, and a code can be shared by multiple users. The feature was first used in virtual private networks (VPN), to allow users within the VPN to override certain calling restrictions of a VPN station. Its use is expected to be broader in future.
- *Originating user prompter* (OUP). Allows the service to provide an announcement that asks the caller to enter a digit or series of digits via a DTMF phone or to enter a voice instruction via a more advanced service in future.

Two optional features may be added:

- *Abbreviated dialing* (ABD). This feature allows abbreviated dialing digit sequences to represent an actual dialing digit sequence; for example, a two-digit code may represent a complete directory number.
- *LOG*. (See example 1.)

Building a Simple Personal Communication System

To better understand how two services can be combined to provide more benefits than each service can separately, consider the combination of two of the services just described.

number translating services. Depending on the origin of the call, different C numbers can be given back by the SCF to the SSF.

Time- and Origin-Dependent Routing

This feature, which is a combination of time-dependent and origin-dependent routing, is a very powerful tool that is mainly used with number translation services. The feature allows networks to, for example, route calls from area 1 to answering place 1 and from area 2 to answering place 2 during normal working hours. At all other times, the calls from both areas might be routed to answering place 3. All this is handled by the SCF in the intelligent network.

Customer Control

This feature allows service subscribers and service providers to gain access to, read, and change *their own data*. Usually, only the network operator is allowed to do this. In its simplest form, customer control may be carried out from a normal DTMF phone. However, as the capabilities for making changes become more sophisticated, more advanced terminals will be required. On intelligent networks most customer control accesses will involve the service management point (SMP), which in turn updates the SCP, but simple changes can be made directly in the SCP.

Call forwarding or the handling of old, node-based services from a normal phone can be seen as the first step towards customer control, but the possibility of influencing service by those methods is very limited compared to what is possible on an intelligent network.

The most common uses of customer control in the early phases of intelligent networks will probably be in three main areas. First, the feature could be employed to control both time- and origin-dependent routing. For example, the feature could be used to change answering places, to change time-dependent routing for freephone or premium rate services, and so on. (See the example in Section 4.3.3.)

A major company with numerous credit call cards spread among its employees suggests a second area. Customer control access into the service management function (SMF) would allow the company to manage all of its facilities. The users (employees) themselves would also be able to make certain changes. (See the example in Section 4.3.3.)

A third area would be managing traffic flow, that is, routing management. Consider a major company spread out over a large geographical area, where the customer control function may be used to change the routing in the network between two subsidiaries located far apart. The telecommunications manager may use time-dependent routing and customer control to change the routing each time tariffs change, thereby ensuring that the most inexpensive routes in the global network are always utilized. Many networks have different charges at different times, for example, lower charges when traffic is low (evenings, nights, holidays). When these lower tariffs are in effect at different times (because

of different time zones), a telecommunications manager can change the routing to save money for the company.

Security is a basic issue for customer control. Security is important to the owner of a service, especially where the risk of mistakes, abuse, and fraud is concerned. It is also important to the network operator, who is giving a third party access to the system in a way that was not possible before. Mistakes or deliberate misuse of the customer control function may affect the rest of the network, other services and service providers, and other subscribers.

Calling Line Transfer Capability

This general feature can be used in many different services. The most commonly used or discussed are the calling line identification presentation (CLIP) and the calling line identification restriction (CLIR) services. With CLIP, the calling line's identity, the telephone number (the A number) is forwarded to the called party (the B subscriber), where it can be used either for presentation on B's phone or for other purposes. CLIR is a service available to A subscribers who do not want their numbers to be forwarded to B subscribers.

Some B subscribers, however, do not want to receive any calls from individuals who have CLIR service, that is, individuals who do not wish to identify themselves. This disagreement has led to at least two schools of thought. One group feels that it is a violation of personal integrity to have numbers presented to the B subscriber; the other group thinks it is everyone's right to know the identity of a caller. There are also legal aspects in different countries that must be considered regarding restrictions for calling line transfer capabilities.

There is a positive spinoff effect, however. This feature makes it possible for the calling line's identity to be available in other parts of the network during a call setup. In future, it will be possible to use this very interesting feature in combination with numerous services.

Section 6.2.2 offers a closer look at future uses of calling identities.

4.3.3 Creating Advanced IN-Based Services

If we add the features discussed in Section 4.3.2 to the basic services described in Section 4.3.1, more advanced IN-based services may be created. Obviously, there are many combinations, and we cannot possibly cover all of them here. Therefore, I will focus on two examples that show how value-added features can be added to IN-based services, creating, in this instance, an advanced freephone service and an advanced credit (card) call service.

Advanced Freephone Service

Let us look at how the advanced freephone service, with sophisticated features, can be used in a real business by returning to the pizza company example described in Section

- *The terminal used.* If the terminal owns the subscription, it might be a company that has mobile terminals for its employees. Employees might take the terminals with them while working, but the bill goes to the company.
- *The NAP used.* Once full mobility is introduced, there will still in some cases be a need to charge according to physical location, that is, according to where the terminal is connected to the network. In this case, the conventional method of charging the access point is used. An example might be a company's connection to a network to which several terminals can be connected.
- *A third party.* This includes all other ways to charge a call. Examples of a third party include an account, another person, or another subscription.

In the first three cases, it should also be possible to charge the called side, that is, the person, terminal, or NAP called. Furthermore, it should be possible to split the charge between the parties involved.

6.2.2 The Impact of Full Mobility on Services

Moving towards full mobility means we must continuously change the services we have implemented, for each step we take. Mobility will probably be one of the greatest revolutions for value-added services since their creation, as it deals with almost every service that existed prior to mobility. In this section, we will examine some of the most important services available in networks today, services planned for the future, and services soon to be implemented to see how they fit into a network with partial or full mobility.

Calling and Called Identification Services

Of course the calling party identification service comes to mind first, as it is similar to the calling line identification (CLI) service that exists today. See Section 4.3.2. This service, which identifies the calling party prior to call setup, offers, among other things, more security to the called party.

Another identification service that will probably be of interest in the future is the called party identification service. But, wait a minute, don't we know who we are calling? Let us consider an example. If I call someone to discuss business, I want to be 100% sure that the person on the other end is the person I wish to speak with. If I call a bank, I want to be sure that I reach the right department, regardless of who answers. Well, you might ask, Do you know what number you dialed? Obviously, I know what number I dialed, but I do not know where I have been connected. The bank could have two, three, or more numbers for the office (a normal number, a freephone number, and so on), and these numbers could be used together with: time- and origin-dependent routing; customer control for routing the number to different places; call forwarding (unconditional, on busy, or on no answer); and so on. These variables make me highly uncertain of which physical location I have been connected to. Introducing a called identification service could minimize the risk of abuse and fraud, among other things.

Identification Cards

One way to identify yourself as a calling or a called party is to use a personal card in combination with a password, the same way you do with an automatic teller machine at your bank. The credit card service was probably the first type of service that used identification by card in telecommunications.

The Calling Party

Section 4.3.2 describes the calling line identification (CLI) services CLIP, for presentation of calling line on the called subscriber's terminal, and CLIR, for restriction to it, prior to the introduction of mobility. The ability to identify the calling party will be one of the most important features also when full mobility is introduced. However, we cannot talk simply about "calling line identification" anymore, as this is only one of the cases. Instead from now on we must talk about the *calling identity* (CI) to cover all cases.

CI could be used for presentation at the called party, as in the CLIP service, but it could also have other, more sophisticated uses. In future, it could be available on different parts of the network during set up of a call or during a call. A good example is when both you and the person you wish to call move around on the network. During the call setup and the call, it is extremely important that the two terminals can handle your call and the services you intend to use during the call. It is also important to check the two NAPs against one another, because you could be connected to exchanges with different capabilities. A third example could be that you have two different personal service profiles. Perhaps one person has a low-budget set of minimal services.

So, the CLI service Section 4.3.2 describes is only valid when we have a one-to-one connection between the subscriber and the terminal (the phone) and a one-to-one connection between the terminal and the NAP. But, as we see in Section 6.2.1, when we enter the world of full mobility, this will no longer be the case. A person will be able to use any terminal for calls. Moreover, it is possible to move a terminal and use it in different places.

Which number, then, should be transferred for use on the network or for presentation at the called side—the caller's personal number or the number of the terminal? Or should it be an entirely different number? The answer is that in future the CI will be divided into at least five different identities, depending upon the actual use. So, from now on, we consider as relevant:

1. The calling person,
2. The personal function,
3. The terminal identity,
4. The terminal function,
5. The physical location, that is, the line or NAP (network access point).

The customer control function will enable added-value service capabilities to move between these five identities between calls or during a call.

Let us use a few examples to examine this new structure of the calling identity, or perhaps, we should say, the *calling party*:

1. *The calling person.*

When I call my spouse, children, parents, and so on, I could, in future, receive special privileges (such as priority) compared with other callers. When I call my insurance company, they could use my identity, which is transferred with the call, to immediately find my personal file on their computer and give me a quick reply to my questions. This personal identity will be independent of the terminal and the NAP used.

2. *The personal function.*

For example, the general manager of a company might call a bank or local or government authorities and receive special rights and access capabilities into the company's data that are associated with the particular called party identity *general manager*, regardless of who the person actually is. Again, the personal function will be independent of the terminal or the NAP the person uses.

People can have more than one personal function. For example, the individual in our example could be both general manager of company ABC and chairman of the local golf club.

3. *The terminal identity.*

In future, many different types of terminals, with different technical capabilities, will be available on networks. Consequently, it is extremely important that users be able to have their terminal identities transferred when they call another terminal or a service. Perhaps, for example, a user's terminal can only be used locally in a particular area. Or, perhaps the terminal has technical limitations that prevent it from connecting to a called party. The terminal might, for example, be a normal phone, and the user is trying to reach a service only created for broadband access.

This terminal identity will be independent of who uses the terminal or from which NAP the calls come.

4. *The terminal function.*

It might be that a terminal can only be used by a certain group, for example, only the members of the local golf club and no one else. A terminal can also be dedicated for emergency calls only, with geographical restrictions, and so on. Again, this terminal function will be independent of who uses the terminal or from which NAP the calls come.

5. *The physical location.*

The calling identity might be defined by the physical location on the network, for example, the NAP. If someone wants to order a pizza or call for assistance when

his or her car breaks down on the highway, physical (geographical) location is what is important. And physical location will be independent of the person who calls or the terminal used.

Furthermore, when parties make or receive calls, they will have to distinguish between their different roles in some way. This can be done in several ways, depending on the type of mobility implemented. Identities can be distinguished:

- Through the use of identification cards (see Section 6.1, item 8). By inserting different cards into a terminal, users can distinguish between their possible roles as private person (calling person) or personal function. These cards can also identify the different functions of a certain terminal and can be used in combination with a PIN code for security reasons.
- Through the use of different access codes, that is, different numbers for private persons and individual personal functions.

Combinations of the five identities described in the preceding list represent the next step. For example:

- The general manager (from item 2) may only be able to make the type of call he wants (for example, international) if he is using a certain type of terminal (as in item 4) in the company (without geographical restrictions).
- A person (item 1) maybe only get full access to special information if she is located at a special location, for example in her office (as in 5).
- A person holding a function (item 2) and working partly at home may have all his calls paid by his company if he is calling from his home (as in 5).

A next step is to combine these five types of calling identifications with time dependence:

- The person working partly at home might only get his calls paid by the company if he calls during working hours.
- A person might be identified by her function when she calls during working hours and as a private person during her leisure time without having to change these roles manually.

The Called Party

Like the calling party, the called party can be structured according to personal identity, personal function, terminal identity, terminal function, or NAP.

When you call someone today, you usually need to know the exact location (the NAP) of the person and the number to that location. Furthermore, you must remember or look up the number you dial. Now, in about 50% of the cases, there is no answer.

Value-added services like call forwarding and others can increase this figure somewhat but, based on experience, not much. This is because people forget or do not bother to use these services, mainly because of bad user interfaces or poor knowledge about service benefits.

The ultimate goals for calls in future should be to have much better terminals than are generally available today and to use letters instead of numbers when dialing. We would also like to have a more *personal* way to call, for example, when I dial "dentist" I want to reach *my dentist* (a personal function) and no one else. In the same way, my neighbor wants to reach his dentist (not mine) when he dials "dentist." But, on the other hand, when we dial "plumber," we might want to reach the same person. So, if we could find an easier way to dial or, perhaps, just speak to the terminal instead of dialing, these would be tremendous improvements over communication today.

Call Forwarding Services

The first service in this category that comes to mind is call forwarding unconditional (CFU). We must, however, also consider call forwarding in combination with a condition, like no reply or when busy. Consider the following example.

Mr. Smith is the general manager of company ABC. He is also chairman of the local golf club. Mr. Smith is married and lives in a house near his work and near his golf club. For his different roles, he has different numbers:

- As a private person, his number is 111 (a personal identity);
- As the general manager of company ABC, his number is 222 (a personal function);
- As chairman of the local golf club, his number is 333 (a personal function).

At a certain time, perhaps calls to all three numbers are routed to his car terminal. For example, let us assume that he is on his way to the airport in half an hour and he wants to be sure that he does not miss any calls while he is abroad, as he cannot answer any of them outside the country.

He cannot use call forwarding to forward all three to only one person. As a private person, he probably wants his calls to go to his wife, that is, he initiates call forwarding unconditional (CFU) from a personal identity to a personal identity. As the general manager, he wants his calls to go to his secretary, initiating CFU from one personal function to another personal function. As chairman of the golf club, he has four choices: (1) the secretary of the golf club (CFU from a personal function to a personal function); (2) the clubhouse (CFU from a personal function to a physical location); (3) the mobile terminal owned by the golf club (CFU from a personal function to a terminal function); (4) Mr. Haley, a friend and a fellow member of the golf club, but with no official function there (CFU from a personal function to a personal identity). Obviously, Mr. Smith will have a lot to do before his plane leaves in half an hour.

Let us assume that Mr. Smith succeeded in initiating all the CFUs before he left. When he returns one week later, he will have to transfer (collect) all his numbers again. He

will have to remember where all his CFUs were made and, after talking to everyone and canceling all the CFUs, he will have a clear picture of what happened at his home, at his company, and at the golf club while he was away.

In this example Mr. Smith only had three roles. In future, people could have perhaps five or more roles. It might then be very easy for them to forget where they have assigned calls for each of their roles while away. All CFUs could probably be canceled, and the roles (all of them) could be taken back immediately, so this is not a problem. However, if they cannot remember who received their calls in the separate roles, these people will have a problem. How will they find out what happened during their stay abroad? Obviously, we will need some type of function that provides a complete list of all our numbers (for all our roles) and where they go at a particular moment in time. We will also probably need to combine this function with a history function, so that we can find out what happened yesterday, for example.

Concepts Such as Virtual Private Networks

Full mobility is very well suited to virtual private networks (VPNs), especially when full mobility is being created inside a VPN. In general, the preceding discussions about calling line identification and call forwarding for networks is also valid for mobility inside a VPN.

Services Independent of Full Mobility

Some services will not be affected by the introduction of full mobility and can be used in fully mobile networks. Such services are often of the type *call initiating* and include:

- Abbreviated dialing, which translates a short number into a real one for initiating a call;
- Repetition of last dialed number, which repeats a number for initiating a call;
- Wakeup service, which initiates a wakeup to a predetermined number;
- Hot line, which initiates a call to a predetermined number.

These services also make good candidates for implementation on terminals, as we have seen in previous chapters.

6.3 SERVICE ADMINISTRATION AND MANAGEMENT EVOLUTION

The area of service administration and management is a key area that will impact the future evolution of service implementation. This area will be important because it will influence to what extent we can

- Expand the number of services and customized services on networks;
- Allow end users (service providers and subscribers) access to data and program areas in SCPs and SDPs (either directly or indirectly via SMP);

SCP and so works dynamically. When calls are being sent from SSPs to the SCP, at the same time, we are waiting for return calls from the SCP to the SSPs. Windowing determines how often calls can be sent to the SCP based on the number of outstanding calls, that is, the number not answered yet. If too many outstanding calls accrue, sending is temporarily stopped, until outstanding calls are answered. When a certain number of answers have been returned, sending is continued. When windowing is used from all SSPs to the SCP, the actual load on the SCP decides the traffic flow from each SSP, thus risk of overloading the SCP is less than when call gapping is used.

7.4 IMPACT ON FUTURE NETWORK STRUCTURE

Almost all traffic carried on the telephone networks during the last decades has been real-time voice communication, that is, normal telephone calls between two people. The greatest demand from users, especially the calling party, has been to obtain a quick response from the network after dialing a number (a few seconds) to find out if there is a possibility to reach the called party or not. If congestion occurs on the network or if the called party is busy, we want to know that immediately after dialing the last digit. Even a ten-second wait is intolerable. Once a call has been established, however, we do not notice or care about single bit errors, as when we are speaking we can understand each other anyway.

If we look ahead 5–10 years, many experts forecast a major change in the user pattern of subscribers in PSTN. Traditional real-time voice communication will still be used a lot, of course, but it will probably only account for about 50% of all traffic. The rest will come primarily from facsimile and mail (computer and voice) traffic. The narrowband integrated services digital network (ISDN) will also have grown, but to far from the same size as the PSTN, of course. Broadband communication, which is primarily used for business, education, and leisure, will also grow.

Many experts believe that ISDN access will replace a great deal of existing PSTN accesses by this time, and that ISDN will be heavily used in the United States and Europe. Personally, I do not believe this will be the case. There are some very concrete facts that support my view:

- ISDN access forces an extension of a new physical access (termination) to every new customer. This is a very expensive task that I do not think is possible, at least not during the economic recession we are experiencing in the beginning of the 90s.
- The possibilities offered by facsimile and computer mail communication on the PSTN provide many of the benefits that were supposed to be offered by ISDN.

However, I do believe that broadband ISDN communication will expand, as there are only a few alternatives offered in PSTN. When there are alternatives in PSTN, you often have to multiplex numerous 64 Kbps channels to a higher bandwidth, which is a rather complicated and expensive alternative.

If what we discussed above is true, in 5–10 years we will end up with a PSTN that carries two major types of traffic, with about a 50–50 division between them:

- Conventional voice communication, like we have today, with a demand for a 1-2 second connection to the called party.
- Facsimile and computer or voice mail traffic, with demands for response times of 5-10 seconds or more and for such features as multicast capabilities and called party receipts.

ISDN accesses, both narrowband and broadband, will increase in parallel, however, and the ability to cooperate with the major PSTN network will be mandatory for the minor ISDN. I believe that ISDN cannot survive without that cooperation from PSTN.

The SCPs, first built and used in the PSTN, may in the early phases of ISDN also be used to handle traffic (especially number translation services like freephone and premium rate) between

- PSTN and narrowband ISDN,
- PSTN and broadband ISDN,
- The ISDN networks themselves,
- Narrowband and broadband ISDN.

Of course, communication capabilities are determined by the weakest party, and this determination could be managed by the SCPs on intelligent networks.

7.4.1 New Network Topologies

The strict hierarchical switching networks we have had traditionally often have from four to six levels of exchanges. These can range from a very small analog local exchange at the bottom to the digital transit and/or international exchange at the top. Introduction of SPC exchanges and digital transmission, however, has led to

- Increasingly larger exchanges than before,
- A reduced number of switching levels in many networks to two or three.

Consequently, when an intelligent network is introduced, we often have two or three levels on the switching network complemented by some SCPs that control IN-based services. In this first phase, the SCPs are not working hierarchically, which means they only form one level of intelligence or one level of service logic on the network. The trend in the switching networks is to further reduce the number of levels to one or two, usually by building combined exchanges for many purposes.

Where intelligent networks are concerned, we can spot another trend. The number of SCPs is starting to grow as the number of IN-based services increases. Not only is the number of IN-based services growing, the number of service providers and service subscribers that use a single service is also growing. This growth results in an increase both in the signaling traffic going to the SCPs and in the load on the SCPs themselves. In the area of service, we will see a distinction between services with relatively moderate demands on

Appendix

This appendix describes briefly service-independent building blocks (SIBs), service features (SFs), and supported services in capability set 1 (CS1) from ITU-T. (Autumn 1993). For detailed descriptions, please refer to the standards themselves.¹

A.1 SERVICE-INDEPENDENT BUILDING BLOCKS DEFINED IN CAPABILITY SET 1

CS1 includes 13 SIBs (ITU-T recommendation Q.1213), which represent a minimum set required to define the CS1 targeted services in ITU-T recommendation Q.1211, described in Section A.3. The basic call process (BCP) has been defined as a specialized (14th) SIB. The CS1 from ETSI also includes, in addition to these 14 SIBs, another seven SIBs (for full descriptions, please refer to the standards).

<i>SIB</i>	<i>Description</i>
Algorithm	Applies a mathematical algorithm to data to produce a data result. May be used for reducing or increasing counters, for example, for mass calling services (televoting, and so on).
Basic call process (BCP)	A specialized SIB that provides basic call capabilities.
Charge	Allows customizing of charging rates for situations that require different rates than the basic call process. Examples are free of charge and fixed rate.
Compare	Compares two values, if greater than, equal to, or less than. Used for checking a current time against time and date routing, comparing a current value of number of calls with a predetermined value, and so on.

1. Full text of the source material may be obtained from the ITU Sales Section, Place de Nations, CH-1211 Geneva 20, Switzerland, Tel. +41 22 730 51 11, Fax +41 22 730 5194.

<i>SIB</i>	<i>Description</i>
Distribution	Distributes calls to different destinations according to a percentage distribution or time and date routing.
Limit	Limits the number of calls for a service.
Log call information	Logs detailed information about a call in a separate file. Information can be the time of the call, the time the call ended, the dialed number, the destination number, the calling line identification, the queuing time for the call, and so on.
Queue	Provides sequencing of calls to be completed to a called party. Provides the user with the capability to pass a call if resources are available, queue a call if resources are unavailable, play a recorded message to the caller during the queuing time, release the call from the queue when resources become available. The maximum number of calls that can be queued or the maximum time calls can be queued may be specified.
Screen	Checks to ensure that a number is in a specified list, for example, checks PIN code, account number, destination number. Used for functions like call screening and account calling (with or without card).
Service data management	Fetches, replaces, increases, or decreases end user-specific data.
Status notification	Checks the status of a resource, to wait for the right status, to supervise and save changes in status, or to end the supervision of a resource.
Translate	Translates input information into output information, for example, translates a number into one or more numbers. Used for number translation services like freephone and for abbreviated dialing, call forwarding, and call transfer services.
User interaction	Allows information to be exchanged between the network and involved parties (calling or called parties). Used for playing an announcement machine recording and for receiving digits from the parties involved. Can also be used to decide the number of times a recorded message should be played. The message may be interrupted when digits are dialed.

Verify

Compare collected information with expected format. Often used after the SIB: user interaction.

A.2 CAPABILITY SET 1 TARGET SET OF SERVICE FEATURES

Table A.1 (first and second part) shows the mapping between services and service features in CS1 (ITU-T recommendation Q.1211).

Table A.1
Q.1211 Mapping Between Services and Service Features

Services	Service Features (first part)																		
	ABD	ATTC	AUTZ	AUT	ACB	CD	CF	CFC	GAP	CHA	LIM	LOG	QUE	TRA	CW	CUG	COC	CPM	CRA
ABD	C											O						O	
ACC	C		C									O							
AAB	O		C									O							
CD						C						O						O	
CF							C					O						O	
CRD								O			O	O	O					O	O
CCBS					C							O			O				
CON												O					O	O	
CCC	O		C									O							
DCR						C						O						O	
FMD												O						O	
FPH				O		O		O	O		O	O	O					O	O
MCI												C							
MAS						O			O		O	O	O					O	O
OCS												O						O	
PRM						O		O	O		O	O	O					O	O
SEC				C								O						O	
SCF								C				O						O	
SPL						O		O	O		O	O	O					O	O
VOT						O			O		O	O	O					O	O
TCS												O						O	
UAN						O		O	O		O	O	O					O	O
UPT			C									O						O	O
UDR												O						O	
VPN	O	O	O	O		O				O		O	O	O		O	O	O	O

Table A.1 continued

Services	Service Features (second part)																		
	CRG	DUP	FMD	MAS	MMC	MWC	OFA	ONC	ONE	ODR	OCS	OUP	PN	PRMC	PNP	REVC	SPLC	TCS	TDR
ABD		O																	
ACC												C							
AAB												C							
CD									C	O									O
CF																			
CRD									C										
CCBS																			
CON					O	C													
CCC												C							
DCR										O									O
FMD			C																
FPH	O	O		O					C	O	O	O				C			O
MCI											C								
MAS				C						O	O	O							O
OCS											C								
PRM	O	O							C	O	O	O		C					O
SEC																			
SCF																			
SPL	O	O							C	O	O						C		
VOT				C						O	O	O							O
TCS																		C	
UAN	O								C	O	O	O							O
UPT		O	C									O	C				C		O
UDR										O									O
VPN	O		O				O	O				O			C				O

C = Core The particular service feature is fundamental to the service, i.e., in the absence of the service feature the name of the service does not make sense as a commercial offering to the service subscriber.

O = Optional The service feature is not core, i.e., without this service feature the name of the service would still make sense as a commercial offering to the service subscriber. Therefore, the service feature can be regarded as an optional enhancement to the service.

Service Features Fully Supported in CS1

The following is a list of the service features that are fully supported in CS1 (for full descriptions, please refer to the standards).

<i>Service Feature</i>	<i>Description</i>
Abbreviated dialing (ABD)	Allows the definition of abbreviated dialing.
Attendant (ATTC)	Allows virtual private network (VPN) users to access an attendant position within the VPN.
Authentication (AUT)	Verifies that a user is allowed to exercise certain options.
Authorization code (AUTZ)	Allows a VPN user to override calling restrictions.
Call distribution (CD)	Specifies the percentage of calls to be distributed among two or more destinations.
Call forwarding (CF)	Forwards incoming calls to another number.
Call forwarding on busy/don't answer (CFC)	Forwards particular incoming calls if the called user is busy or does not answer within a specified number of rings.
Call gapping (GAP)	Restricts the number of calls that can be routed to a particular destination.
Call limiter (LIM)	Specifies a maximum number of calls that can be simultaneously routed to a particular destination.
Call logging (LOG)	Creates a log record for each call received at a specified number.
Call queuing (QUE)	Queues calls meeting busy or no answer within a predetermined time. An announcement is given to the calling party.
Closed user group (CUG)	Allows a user to be a member of a set of VPN users who are normally authorized to make and receive calls only within the group.
Customer profile management (CPM)	Allows users to manage their customer profiles in real time.
Customized recorded announcement (CRA)	Allows a call to be completed to a (customized) terminating announcement instead of a subscriber line.
Customized ringing (CRG)	Allocates a distinctive ringing to a list of calling parties.
Destinating user prompting (DUP)	Prompts called parties with a specific announcement.
Follow-me diversion (FMD)	Routes incoming calls to a new location.

<i>Service Feature</i>	<i>Description</i>
Mass calling (MAS)	Allows the processing of huge numbers of incoming calls for example, calls generated by broadcasted advertising o games.
Off net access (OFA)	Allows VPN users to access the VPN from any non-VPN exchange.
Off net calling (ONC)	Allows VPN users to call outside the VPN.
One number (ONE)	Permits two or more terminating lines, also at different locations, to have a single telephone number.
Origin-dependent routing (ODR)	Enables users to accept or reject calls and, if they acceptance a call, route the call according to the calling party's geographical location.
Originating call screening (OCS)	Allows users to bar calls from certain areas.
Originating user prompter (OUP)	Allows users to provide an announcement requesting callin parties to enter a digit or series of digits.
Personal numbering (PN)	Supports a universal personal telephony (UPT) number that uniquely identifies each UPT user and is used by calling parties to reach the UPT user.
Premium charging (PRMC)	Allows for the payback of part of the cost of a call to a value-added service provider.
Private numbering plan (PNP)	Allows subscribers to maintain a numbering plan within the private network that is separate from the public network.
Reverse charging (REVC)	Allows subscribers to be charged for the entire cost of calls they receive.
Split charging (SPLC)	Allows calling and called parties to share the cost of a call.
Terminating call screening (TCS)	Screen calls based on the terminating number called.
Time-dependent routing (TDR)	Enables users to accept or reject calls and, if they accept a call, route the call according to time and date. (Other treatments than routing can be done, according to time and dates.)

Service Features Partially Supported in CS1

The following is a list of the service features that are partially supported in CS1 (for full descriptions, please refer to the standards).

<i>Service Feature</i>	<i>Description</i>
Automatic call back (ACB)	Allows the called party to automatically call back the calling party of the last call.
Call hold with announcement (CHA)	Allows subscribers to place a call on hold with options to play music or a customized message.
Call transfer (TRA)	Allows a user to place a call on hold and transfer the call to another location.
Call waiting (CW)	Notifies the called party, when on a call, that another party is trying to reach him or her.
Consultation calling (COC)	Allows the user to place a call on hold, in order to initiate a new call for consultation.
Meet-me conference (MMC)	Reserves a conference resource for making a multiparty call, indicating the date, time, and duration of the conference.
Multiway calling (MWC)	Establishes multiple, simultaneous calls among parties.

A.3 CAPABILITY SET 1 TARGET SET OF SERVICES

These services should be considered as a minimum set of services for CS1 (ITU-T recommendation Q.1211). Table A.1 (first and second part) shows the mapping between services and service features in CS1. (For full descriptions, please refer to the standards.)

Services Fully Supported in CS1

The following services are fully supported in CS1.

<i>Service</i>	<i>Description</i>
Abbreviated dialing (ABD)	Allows abbreviated versions of telephone numbers. Translation to the real number is done by the network or by the terminal.
Account card calling (ACC)	Allows users to make calls from any terminal (with or without card reading facilities) and have the call charged to an account rather than the terminal used. Used with a secret PIN code.
Automatic alternative billing (AAB)	Charges calls made from any terminal to an account rather than to the calling or called party.
Call distribution (CD)	Routes incoming calls to different destinations based on specified criteria.
Call forwarding (CF)	Forwards calls to another number.

<i>Service</i>	<i>Description</i>
Call rerouting distribution (CRD)	Allows users to have incoming calls encounter a triggering condition.
Credit card calling (CCC)	Charges calls made from any access point to a specified account number.
Destination call routing (DCR)	Routes calls to destinations according to specified criteria such as time, date, originating area, or calling line identity
Follow-me diversion (FMD)	Allows users to remotely control redirection of incoming calls to other locations.
Freephone (FPH)	Allows reverse charging.
Malicious call identification (MCI)	Logs incoming calls.
Mass calling (MAS)	Allows the network operator to temporarily allocate a single number to a user. Includes instantaneous high-volume traffic routed to one or multiple locations (for example, televoting)
Originating call screening (OCS)	Specifies that outgoing calls may be restricted or allowed.
Premium rate (PRM)	Allows for the payback of part of the cost of a call to a value-added service provider.
Security screening (SEC)	Screens network users attempting to access the subscriber's network, systems, or applications.
Selective call forwarding on busy/no answer (SCF)	Allows called parties to forward preselected calls if busy or don't answer.
Split charging (SPL)	Allows calling and called parties to split call charges.
Televoting (VOT)	Allows users to vote by phone. Calling parties call one of two or more numbers to cast their vote.
Terminating call screening (TCS)	Specifies that incoming calls be either restricted or allowed.
Universal access number (UAN)	Allows a user with several terminating lines in any number of locations to be reached by one number.
Universal personal telecommunications (UPT)	Allows users to access telecommunication services via unique personal telecommunication number (PTN) across multiple networks at any network access.
User-defined routing (UDR)	Allows users to specify how outgoing calls shall be routed.
Virtual private network (VPN)	Allows users to build a private network with public network resources.

Services Partially Supported in CS1

The following two services are only *partly supported* in CS1.

<i>Service</i>	<i>Description</i>
Call completion to busy subscriber (CCBS)	Allows calling parties who encounter a busy destination to be informed when the busy destination becomes free, without having to make a new call attempt.
Conference calling (CON)	Connects multiple parties in a single conversation.