INTERNATIONAL TELECOMMUNICATION UNION

ITU-T

TELECOMMUNICATION
STANDARDIZATION SECTOR
OF ITU

Q.1219

(04/94)

INTELLIGENT NETWORK

INTELLIGENT NETWORK USER'S GUIDE FOR CAPABILITY SET 1

ITU-T Recommendation Q.1219

(Previously "CCITT Recommendation")

FOREWORD

The ITU-T (Telecommunication Standardization Sector) is a permanent organ of the International Telecommunication Union (ITU). The ITU-T is responsible for studying technical, operating and tariff questions and issuing Recommendations on them with a view to standardizing telecommunications on a worldwide basis..

The World Telecommunication Standardization Conference (WTSC), which meets every four years, etablishes the topics for study by the ITU-T Study Groups which, in their turn, produce Recommendations on these topics.

The approval of Recommendations by the Members of the ITU-T is covered by the procedure laid down in WTSC Resolution No. 1 (Helsinki, March 1-12, 1993).

ITU-T Recommendation Q.1219 was prepared by ITU-T Study Group 11 (1993-1996) and was approved under the WTSC Resolution No. 1 procedure on the 7th of April 1994.

NOTE

In this Recommendation, the expression “Administration” is used for conciseness to indicate both a telecommunication administration and a recognized operating agency.

CONTENTS

SUMMARY

This Recommendation is intended to provide a detailed guide for the IN capabilities provided by Capability Set 1. This guide includes example service scenarios, as well as details to ensure an understanding and implementation of IN CS-1. This guide is targeted towards a wide audience that includes Users that only require a general “talking” knowledge on IN and how it will be utilized, as well as Users that need a detailed “working” knowledge of IN in order to complete their work function within an IN structured environment. This Recommendation is accompanied by two annexes which provide detailed example service scenarios and basic call state machine models.

INTELLIGENT NETWORK USER'S GUIDE FOR CAPABILITY SET 1

(Geneva, 1994)

1 Scope

1.1 Target audience

The Target Audience includes a broad spectrum of Intelligent Network (IN) Users of this guideline. At one end of the spectrum is a set of Users that only require a general "talking" knowledge of IN and how it will be utilized. At the other end of the spectrum is a set of Users that need a detailed "working" knowledge of IN in order to complete their work function within an IN structured network. Specifically, this guideline is written for use by Service Providers and Equipment Vendors (as used in Recommendation Q.1211), and Manufacturers and Network Operators (as used in Recommendation I.312/Q.1201).

These needs within the Target Audience can be satisfied with this Users Guide as discussed in 1.2, Intended Use.

1.2 Intended use

The intent of this User's Guide is to provide a detailed Implementors' Guide for the IN capabilities provided in Capability Set 1 (CS-1). This guide includes example service scenarios, as well as details to ensure an understanding and implementation of IN CS-1.

This User's Guide includes a broad understanding of General IN Concepts, as well as the relationship of CS-1 to these concepts as follows:

1) Overall IN Objectives;
2) CS-1 Objectives;
3) Service Drivers for CS-1;
4) CS-1 Architecture;
5) Infrastructure for CS-1;
6) Service Examples;
7) Physical Deployment Scenarios;
8) Future IN Capability Sets.

This User's Guide will also serve as a reference document and direct the User that requires more detailed information to the specific clauses in the detailed IN Recommendations, i.e. the Q.1210-Series.

1.3 Framework outline of Q.1200-Series

Table 1, taken from Recommendation Q.1200, clause 1, provides the IN Recommendation Structure[1]:

[1] Structural distribution is accomplished through the tens digits and the ones digits. For example, Q.1211 is the Principles and Introduction Recommendation for CS-1, and Recommendation Q.1224 is the description of the Distributed Functional Plane for CS-2, etc.

TABLE 1/Q.1219

00 – General	
10 – CS-1 20 – CS-2 30 – CS-3 40 – CS-4 50 – CS-5 60 – CS-6 70 – CS-7 80 – CS-8 90 – Vocabulary	1 – Principles Introduction 2 – Service Plane (not included for CS-1) 3 – Global Functional Plane 4 – Distributed Functional Plane 5 – Physical Plane 6 – For future use 7 – For future use 8 – Interface Recommendations 9 – User's Guide

Specifically, the IN Recommendations provided in CS-1 are listed below:

Q.1211: Introduction to Intelligent Network Capability Set 1

This Recommendation provides an introduction to IN CS-1 by providing an overview and definition of CS-1 and by describing its main characteristics and overall capabilities. It further defines the service aspects, network aspects, and functional relationships that form the basis of the CS-1 capabilities.

Q.1213: Global Functional Plane for Intelligent Network CS-1

This Recommendation provides the functional characteristics of the Global Functional Plane associated specifically with IN CS-1. The following functional characteristics are specific to IN CS-1 and are addressed in this Recommendation:

– CS-1 Service Independent Building Blocks (SIBs).

– The Basic Call Process SIB.

– The relationships between the Service Plane and the GFP.

– Stage 1 SDL diagrams are provided for the CS-1 SIBs where needed to clarify SIB operation.

Q.1214: Distributed Functional Plane for Intelligent Network CS-1

This Recommendation provides the following:

– The IN Distributed Functional Plane architecture for IN CS-1, in terms of a subset of the general IN DFP.

– Static and Dynamic models of the functional entities related to IN service execution.

– SIB Stage 2 descriptions to identify information flows and functional entity actions for IN CS-1.

– Detailed information flow descriptions, including information elements and functional descriptions, as the basis for specifying IN protocols.

– A starting point for the study of call party handling capabilities beyond the two-party call setup and clearing.

Within the Q.1210-Series Recommendations, Q.1214 describes the distribution of GFP functionality defined in Recommendation Q.1213 in a service and vendor/implementation independent manner, as constrained by the capabilities of the embedded base of evolvable network technology. This provides the flexibility to allocate distributed functionality into multiple physical network configurations, as described in Recommendation Q.1215, and to evolve IN from CS-1 to some future CS-N. It also provides a framework from which IN protocols are specified for CS-1, as described in Recommendation Q.1218.

Q.1215: Physical Plane for Intelligent Network CS-1

This Recommendation describes the physical plane of the IN architecture for CS-1. The physical plane of the IN CS-1 identifies the different physical entities and the interfaces between these entities.

Q.1218: Intelligent Network Interface Recommendations

This Recommendation defines the Intelligent Network Application Protocol (INAP) for the support of the capabilities required by the CS-1 target services over the CS-1 interfaces (SSF-SCF, SCF-SDF, and SCF-SRF) as defined in Recommendation Q.1211. It defines some of the possible protocol stack scenarios, the operations which flow between the entities and the procedures to be followed at each functional entity.

Q.1219: Intelligent Network User's Guide for Capability Set 1

This Recommendation is intended to provide a detailed implementor's guide for the IN capabilities provided in IN CS-1. It includes example service scenarios, as well as details to ensure an understanding and implementation of IN CS-1.

Q.1290: Vocabulary of Terms Used in the Definition of Intelligent Networks

This Recommendation provides a vocabulary of terms and definitions which have been studied for application in the documentation of Intelligent Networks.

1.4 Initial set of capabilities

(Reference, Q.1211, Introduction to Intelligent Network Capability Set 1.)

CS-1 is the definition and selection of an initial subset of IN capabilities that meet the following general criteria:

a) CS-1 is a subset of the target IN architecture, i.e. the Target Architecture is the Long Term View of the evolvable architecture that will provide the vendor, service, and network independence specifies in Q.1201.

b) CS-1 is a set of definitions of capabilities that are of direct use to both manufacturers and network operators (as used in Recommendation Q.1201) and Service Providers and Equipment Vendors (as used in Recommendation Q.1211), i.e. interface recommendations are provided for the manufacturers, equipment vendors, and network providers, and Service definitions are provided for the Serve Providers.

c) CS-1 provides network capabilities to support services either defined, or in the process of being defined, by CCITT, or by other organizations, (e.g. Universal Personal Telecommunications Service, Freephone, and Virtual Private Network services like Private Numbering Plan), i.e. the CS-1 network capabilities were not strictly driven by CCITT Stage 1 service descriptions such as UPT, Freephone, and PNP, but were provided to incorporate the capabilities needed to support services being defined on an ad hoc basis, thus meeting the guidelines set out in Recommendation I.312/Q.1201 for providing Service Independence.

d) CS-1 is the first standardized stage of evolution based upon the existing technology base and on evolvability requirements to ensure that the existing technology base will be re-usable in an IN structured environment, i.e. the Basic Call State Model description reflects the current view of switching systems existing in physical switching systems, the Service Control Function and Specialized Resource Function are modelled after existing installed systems, and the protocols specified include those already available for implementation.

The CS-1 architecture may be supported by, but is not limited to the Public Switched Telephone Network (PSTN), the Integrated Switched Digital Network (ISDN), and mobile networks.

For IN CS-1, the initial subset of IN capabilities focuses primarily on normal call processing scenarios. The Recommendations for IN CS-1 do not completely model or describe failure scenarios or error handling procedures. However, general information on the error handling semantics can be found in 4.2.2.5/Q.1214, which identifies error cases and SSF/CCF application processing procedures associated with DP processing and response processing from the SCF; Annex B/Q.1214, which identifies SSF/CCF – SCF information flow scenarios for error cases; 3/Q.1218, which defines general INAP procedures for error handling such as timer expiry for the SSF, SCF, SRF, and SDF protocol state machines (3.1.1.5, 3.1.2.5, 3.1.3.4, and 3.1.4.4, respectively); and clause 2/Q.1218, which specifies several error types for the INAP for IN CS-1. Also, A.2.7 (Time Handling Aspects) presents service examples with failures. Refer to 12.4.3/Q.1400 and in 2.4/Q.775 and 3.2.1.4/Q.775 for general guidelines on error handling procedures.

1.5 State of Maturity of the CS-1 Recommendations

CS-1 is the first Recommendation set providing a framework which will help the various Administrations to develop or customize services independently of any specific manufacturer's Intelligent Network elements. Service constructs are also designed to be independent of the physical implementation. It is important to recognize that Intelligent Network studies address several complex areas, and where unanimous agreements could not be reached at this time, due to lack of complete understanding of the involved complexities, material was moved to appendices and annexes.

CS-1 must be recognized as not being specified in 100% detail, but is conceptually complete. In some cases there may be insufficient detail to allow different manufacturers to truly build 100% compliant IN capabilities. CS-1 does not address the Operational aspects of an IN network, Service Creation, or the administrative tools (i.e. SMS), needed to run it.

However, CS-1 does provide a framework that will help Administrations to develop or customize services independently of the manufacturer, within reasonable limits. As such, CS-1 can now be released for achieving implementation experience. It is recognized, as with any project of this size, that interworking the various multi-manufacturer implementations may not be fully realizable and some future fine tuning of the Recommendations may be required in light of that implementation experience. CS-1 is viewed as the base line from which future IN Recommendations will evolve.

1.6 Service Decomposition for CS-1

Beginning with an overview of the INCM, in terms of the network view provided by each plane of the INCM model, the service modelling can be placed in the perspective of the work completed on IN CS-1. The focus on the determination and meaning of CS-1 targeted services is quite different from the concept of "standard" services, and warrants an effective explanation.

In the Service Plane of the IN Conceptual Model (INCM), services are decomposed into their Service Features (SF), which are the features that comprise the service. Full service descriptions must be available for the new service being analysed prior to identifying the Service Independent Building Blocks (SIBs) required to support the service

Given that a catalogue of Services, SFs, and SIBs exist, the following description explains how analysis of a new service may lead to the extension of existing SIBs or identification of new SIBs (refer to Figures 1/I.329/Q.1203 and 4/I.329/Q.1203).

1) *List Service Features*

   Decompose the new service into its SFs.

2) *Service Feature Definition*

   Define each SF by describing the service provided from the end users (Subscriber) perspective. This definition is referred to as the Service Prose. Information should be available from the Stage 1 Service Description.

3) *Describe Usage Procedure*

   Describe the chain of events seen by the user for this SF. This includes service subscription, activation, modification, and call scenarios for the SF.

4) *Describe Service Feature in terms of SIB*

   Describe the SF in terms of the modular network functions represented by SIBs.

5) *Map to existing SIBs*

   Compare the above (Steps 3 and 4), with the characteristic lists for established SIBs.

6) *Test against 2 and 3*

   Verify the robustness of the SF by analysing the SIB representation with the SF Definition and Usage Procedure (from Steps 2 and 3). Failure to pass this verification indicates that the analysis in Steps 4 and 5 was incorrect or incomplete.

7) *Describe Additional Network Functionality Required*

Describe what functions must be provided by the network, in addition to those of existing SIBs, to fully support the SF.

8) *Extend existing SIB, or define new SIB*

If possible, extend the capabilities of an existing SIB (e.g. additional “type”) to provide the additional functionality required to support the SF. If such extension is not possible, then define a new SIB. Complete the definition of the extended or new SIB by providing the detailed information in 3.4/Q.1203 (Method to describe SIBs).

2 Intelligent Network objectives

(Reference, I.312/Q.1201, Principles of Intelligent Network Architecture.)

The Intelligent Network (IN) is an architectural concept for the creation and provisioning of telecommunications services which is characterized by:

a) extensive use of information processing techniques;

b) efficient use of network resources;

c) modularization of network functions;

d) integrated service creation and implementation by means of re-usable standard network functions;

e) flexible allocation of network functions among physical entities;

f) portability of network functions to physical entities;

g) standardized communications between network functions via service independent interfaces;

h) service provider access to the process of composition of services through the combination of network functions;

i) service subscriber control of subscriber specific service attributes;

j) standardized management of service logic.

IN CS-1 primarily addresses IN characteristics b) through e) and g). These characteristics are reflected in the Q.1210-Series Recommendations as follows:

b) *Efficient use of network resources*

The specific bundling of functionality in each Functional Entity provides a means for efficient use of network resources. In particular, the SRF FE definition supports the efficient use of specialized telecommunications resources, and the SDF FE definition supports the efficient use of service data resources.

c) *Modularization of network functions*

There are three types of modularization of network functions in IN CS-1, namely Service Independent Building Blocks (SIBs), Functional Entities, and Application Service Elements (ASEs).

SIBs define a modularization of functionality from a Global Functional Plane perspective. Thirteen SIBs and one specialized SIB are defined in Recommendation Q.1213.

Functional Entities provide a modularization of functionality in the Distributed Functional Plane for physical entity deployment. Functional Entities are defined in Recommendation Q.1211. Models for the Call Control Function (CCF), Service Switching Function (SSF), Service Control Function (SCF), Specialized Resource Function (SRF), and Service Data Function (SDF) FEs are described in Recommendations Q.1214 and Q.1218. The IN CS-1 Recommendations do not model the Call Control Agent Function (CCAF), Service Creation Environment Function (SCEF), Service Management Access Function (SMAF), or Service Management Function (SMF) FEs described in Recommendation Q.1211.

ASEs provide a modularization of functionality in the Physical Plane for protocol deployment. Twenty-five ASEs are defined in Recommendation Q.1218. See 6.5 for additional information of the ASEs for IN CS-1.

d) *Integrated service creation and implementation by means of re-usable standard network functions*

SIBs may provide a modularization of functionality for service creation in future capability sets. Also, for IN CS-1 there is a strong coupling of functionality among SIBs, Information Flows, and the protocol operations, which facilitates integrated service creation and implementation by means of re-usable standard network functions.

e) *Flexible allocation of network functions among physical entities*

The definition of FEs and the FE models facilitate a flexible allocation of network functions among physical entities. Also, the INAP for IN CS-1 has a modular structure to accommodate a variety of deployment scenarios.

g) *Standardized communications between network functions via service independent interfaces*

The INAP for IN CS-1 is a service independent interface derived from the Distributed Functional Plane modelling.

The Recommendations for IN CS-1 describe models, concepts, and interfaces that address a portion of the IN characteristics. IN CS-1 does not address the service creation, service execution, or service management related IN characteristics [a), f), h) through j)]. However, IN CS-1 provides a foundation for meeting these characteristics in future capability sets. For example, the concept of SIBs may be useful for service creation and the SIBs, Information Flows, and INAP for IN CS-1 may be useful for defining an application programming interface.

The implementation of the IN architecture will facilitate the rapid introduction of new services. Its architecture can be applied to various types of telecommunications networks, including, but not limited to: public switched telecommunications network (PSTN), public switched packet data network (PSPDN), mobile, and Integrated Services Digital Network (ISDN).

3 Capabilities provided by Capability Set 1

(Reference, Q.1211, Introduction to Intelligent Network Capability Set 1.)

IN CS-1 represents a subset of the Long Term capabilities of an IN structured network. While it is characterized in the subclauses that follow, it is important to note that IN CS-1 is structured in such a way as to allow existing network resources to be enhanced to support initial IN capabilities. This provides for re-use of equipment and rapid deployment of IN CS-1 capabilities to allow service providers to gain the benefits of IN as soon as possible, while allowing for a planned migration to future, longer term, IN capabilities.

3.1 Service implementation independence

(Reference, I.312/Q.1201, Principles of Intelligent Network Architecture.)

IN CS-1 allows service providers to define their own services independent of service specific development by equipment providers. A platform is provided which is not service specific, i.e. it will support the definition of many varied services, as opposed to a platform defined for a specific set of services.

3.2 Multi-Vendor capability

(Reference, I.312/Q.1201, Principles of Intelligent Network Architecture.)

The objective of IN CS-1 is to ensure that services defined within an IN structured environment will operate properly across IN equipment provided by multiple vendors. Meeting this capability objective will eliminate the dependence upon a specific equipment vendor for the provision of services to be provided in an IN structured network. However, it must be understood that it is possible to implement subsets of CS-1, the caveat in Q.1211, and stated in clause 1 above, address this scenario. Also, there may be additional signalling functionality that is needed to meet this objective, as described in clause 6.6 below. Further, the various implementation options provided by CS-1 indicate that to meet this objective will require close coordination between equipment providers and service/network providers.

3.3 Multi-Network capability

(Reference, I.312/Q.1201, Principles of Intelligent Network Architecture.)

IN CS-1 allows service providers to ensure that services defined within an IN structured environment will operate properly over multiple IN structured networks. This eliminates the dependence on a specific network for the provision of service provider defined services. However, IN CS-1 provides many options that must be coordinated between various network implementations, by way of bilateral agreements, etc., to ensure this multi-network capability. This could include such things as:

- implementation of common subsets of IN CS-1;
- closely working together to ensure a common interpretation of the CS-1 Recommendations;
- coordination of additional signalling functionality; and
- the implementation of common options in IN CS-1.

3.4 Rapid service delivery

(Reference, I.312/Q.1201, Principles of Intelligent Network Architecture.)

An IN CS-1 structured network allows the service provider to create their own services, using capabilities provided by IN equipment vendors, thus reducing the time interval from the definition of a service to the delivery of the service to the service provider.

3.5 Service deployment

(Reference, I.312/Q.1201, Principles of Intelligent Network Architecture.)

An IN CS-1 structured network allows the service provider to manage the deployment of services through their own service management capabilities. This removes the dependency of the service provider upon the deployment of services in network nodes for service deployment. In other words, if the network is an IN structured network, the service provider can deploy services at any network node(s) at their own schedule.

Service management functionality is used to provision and manage the service control functionality, service data functionality, and specialized resource functionality in the network, outside of the context of call/service processing. Standardized interfaces for this functionality are outside the scope of CS-1. However, the ability of a service subscriber to interact directly with subscriber-specific service management information will not be excluded or constrained for CS-1.

IN CS-1 Service management aspects primarily address the network operator's interaction with the SSF, SCF, SDF, and SRF. This interaction normally takes place outside the context of a particular call or service invocation.

However, CS-1 must neither exclude nor constrain the capability of service customers to interact directly with customer-specific service management information (e.g. a personal service profile).

The following points may be relevant to the CS-1 timeframe, but are not standardized in the CS-1 Recommendations:

- The SMF, SCEF, and SMAF may be used to add, change or delete CS-1 based service related information or resources in the SSF, SCF, SDF, and SRF. Such changes should not interfere with CS-1 based service invocations or calls that are already in progress.
- The network operator may, at its discretion, give the service customer the ability to add, change, or delete appropriate customer-specific information. The mechanisms and safeguards that are put into place by the network operator for this interaction may take advantage of CS-1 functions and capabilities.

The constraints placed on the CS-1 architecture have been put in place primarily to minimize and control feature interactions within single domains of responsibility.

The single-endedness of CS-1 based services means that all aspects of a call are under the control of one CCF/SSF and one SCF at any point in time (principle 4). The SSF is therefore responsible for the handling of interactions between CS-1 based SSF/CCF capabilities, and non-IN features already embedded in CCF software.

The SSF/CCF functionality is expected to be implemented through a closely-coupled, single vendor approach in CS-1. Therefore, this feature interaction problem will be constrained within single-vendor domains in CS-1, and will not require multi-vendor standards.

The ultimate responsibility for consistency of operations within a set of CS-1 based service features lies with the network operator. However, the software and data structures of the SCF, SDF, SMF and the tools provided by the SCEF, may be designed to aid the network operator in fulfilling this responsibility.

These are new areas for the telecommunications industry and CS-1 Recommendations should not seek to control or constrain market-driven implementations of SMF, SMAF, or SCEF.

4 Service aspects for CS-1

4.1 Basic Service Capabilities

(Reference, Q.1211, Introduction to IN Capability Set 1.)

IN CS-1 proposes targeted sets of services and service features to identify the capabilities that are supported by IN CS-1. These should not be considered as placing any limitation on the eventual set of services that might be offered with CS-1. Neither should they be considered minimum sets. In summary, different phases of IN evolution, i.e. as non-IN networks evolve to CS-1, are free to select their own sets of services and service features using CS-1 capabilities, or subsets thereof. Consideration should be given to possible implications of selecting different sets/subsets of services/service features which include limitations in interworking services in a multi-vendor/multi-network environments.

The following capabilities are provided to support these service aspects:

Flexible routing

IN CS-1 has as an objective the capability to allow the service provider to maintain control of routing decisions for services offered within an IN CS-1 structured network. These decisions may be based upon, time-of-day, day-of-week, authorizations codes, etc. The routing decision criteria will be under the control of the service provider within an IN CS-1 structured network.

Flexible charging

Similar to routing above, charging decisions in an IN CS-1 structured network will be under the control of the service provider. Charging decisions can be based on locations, destinations, authorization codes, etc. The service provider will have the capability to make, and implement, these decisions via his service management capability. While IN CS-1 does not directly provide this capability, it does not constrain the ability of a service provider to use such a vendor or network operator dependent capability.

Flexible user interaction

This objective provides the capability to change the amount, or degree, of user interaction for a specific service by the service provider. The service provider can vary the amount of user interaction on a service-by-service basis if desired, or may implement a range of interaction based on sets of services.

Recommendation Q.1211 contains a list of the targeted services and service features used to derive IN CS-1 capabilities.

4.2 Type-A service category

The services/service features provided in IN CS-1 fall into the category of “single ended”, “single point of control” services referred to as Type-A Services. The following definition applies to Type-A services.

Single ended service feature

A single-ended service feature applies to one and only one party in a call and is orthogonal at both the service and topology levels to any other parties that may be participating in the call. Orthogonality allows another instance of the same or a different single-ended service feature to apply to another party in the same call as long as the service feature instances do not have feature interaction problems with each other.

Single point of control

Single Point of Control describes a control relationship where the same aspects of a call are influenced by one and only one Service Control Function at any point in time.

Type-A services are characterised by a relatively simple control relationship between SSF and SCF. The SSF is a “client” for service-related information provided by the SCF, however, the switch retains connection control at all times. In contrast, the control relationship between SCF and SSF in Type-B services may require the sharing of connection control between the switch and external service logic. The information flows need to be rich in parameters to manage what is essentially a peer-peer, distributed processing relationship.

As there are considerable differences in operational, implementation, and control complexity between Type-A and Type-B services, CS-1 is targeted to support Type-A services only.

There are some circumstances in which it will be possible to apply “Type-A” IN technology to certain aspects of “Type-B” services. This applies to switch-based services in general, whether these services be of “Type-A” or “Type-B”, and to “Type-B” services in general, whether these be switch-based or CS-x based. Further detail can be obtained from Recommendation Q.1214. Service designers should also take into consideration the guidance provided in Appendix I/Q.1214 and in Appendix I/Q.1218 as they provide a good starting point for the study of the capabilities to support Type-B services in the context of CS-1.

Illustrative examples of Single Ended, Single Point of Control Service

Figure 1 shows the basic model to describe a single ended feature. Two Service Logic Program Instances (SLPIs) are independently active to the originating call segment and the terminating call segment in the same call. Though the SLPIs reside in separate SCFs, there are no interactions between the Service Feature Instances (SFIs) which are mapped to those SLPIs from the Service Plane to the Distributed Plane. Therefore, the model satisfies a single ended feature.

Figure 2 shows the model where one SFI is mapped to multiple SLPIs. However, there are no interactions between the SFIs. The interactions between SLPIs for one SFI are internally coordinated by the Interaction Manager, and cannot be recognized by the SSF. This figure, and Figure 4 b), are examples of a specific implementation of these capabilities and therefore represent implementation specific aspects of this feature.

Figure 3 shows a model where multiple SLPIs derived from different SFIs interact with each other in one SCF and are active to one half of a call through the Interaction Manager, which controls execution of multiple SLPIs. The interactions between SLPIs for different SFIs are internally coordinated by the Interaction Manager, and cannot be recognized by the SSF. Furthermore, there are no interactions between the SFI which is active to the originating call segment and the SFI which is active to the other call segment. Therefore, it satisfies a single ended feature.

Figures 4 a) and 4 b) show a model where a single SCF is active to multiple call segments of a call. Figure 4 a) shows the case of a single SLPI affecting multiple SSF sides. Figure 4 b) shows the case of a single SFI, which is mapped to multiple SLPIs, affecting multiple SSF sides, and, as noted earlier, is an implementation specific aspect of this feature. Figure 4 c) shows the case that multiple SLPIs derived from different SFIs interact with each other in one SCF through the Interaction Manager and affect multiple SSF sides. From the SSF viewpoint, it can be seen that a single instance in the SCF has the relationship to a call in SSF in all these cases since the Interaction Manager coordinates the interactions between multiple SLPIs, thus satisfying a single ended feature.

Figure 5 shows a model where multiple SFIs in different SCFs interact with each other, and the SLPIs derived from those SFIs are separately active to the originating and terminating call segments. This pattern violates a single ended feature and is not a Type-A Service/Service Feature.

Figure 6 shows a model where multiple SFIs in different SCFs affect one half of a call at the same time. This violates a single point of control feature and is not a Type-A Service/Service Feature.

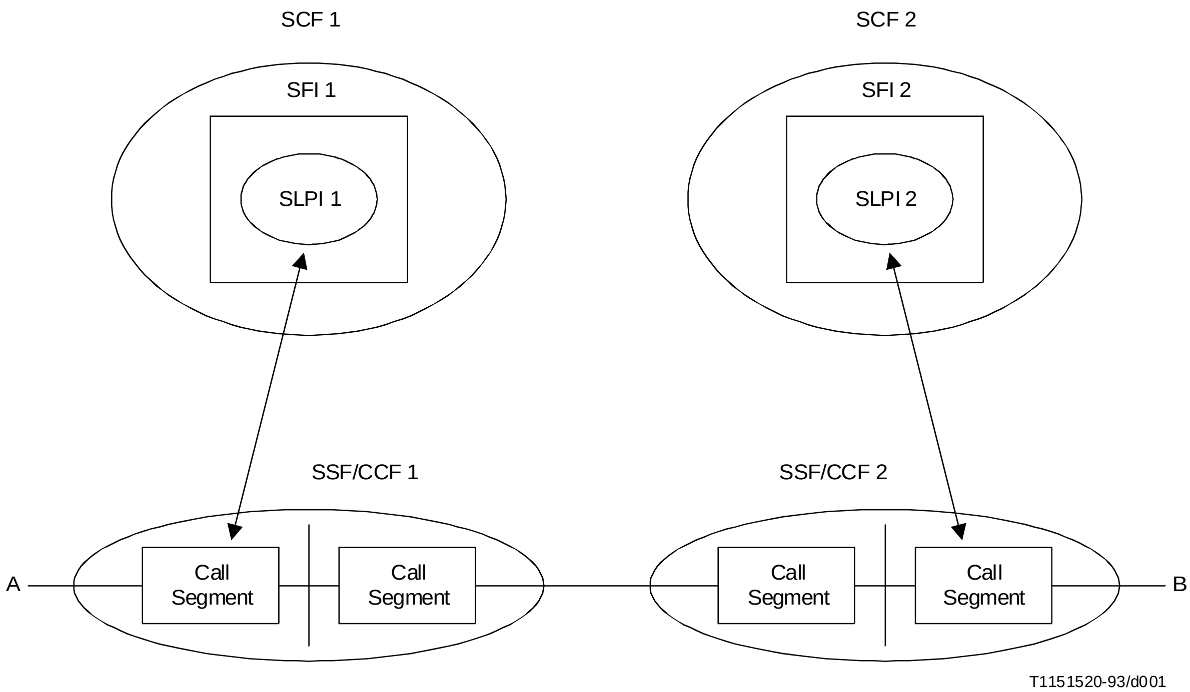


FIGURE 1/Q.1219

Basic Model

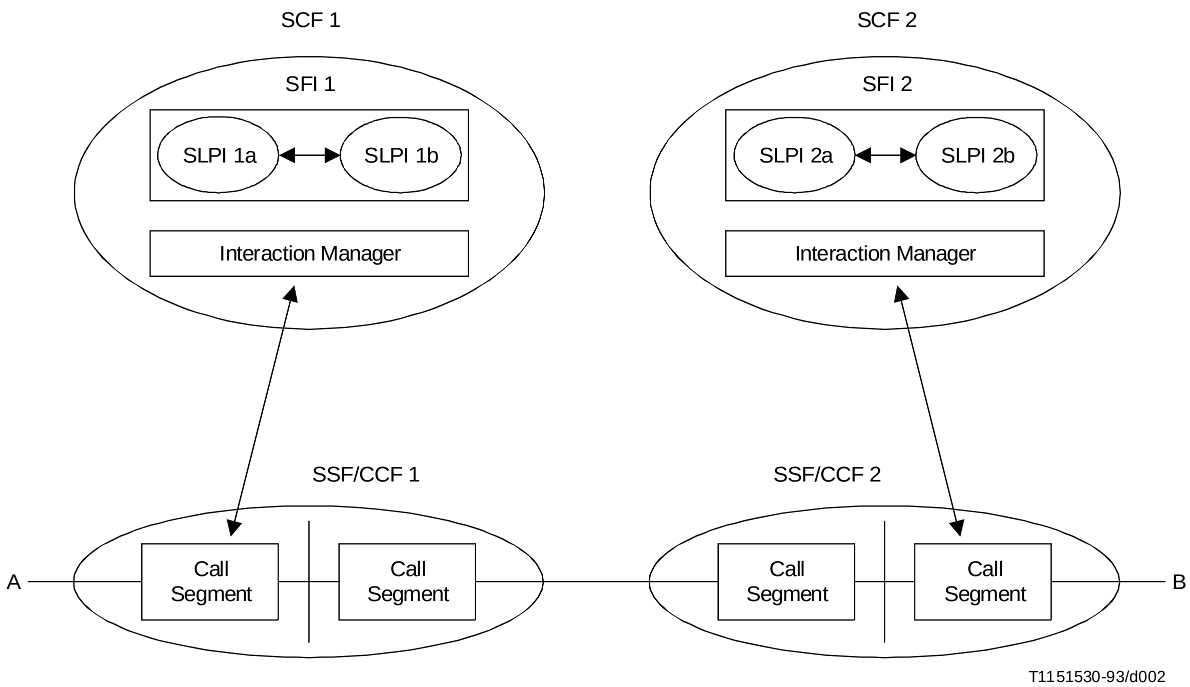


FIGURE 2/Q.1219

Interaction between SLPIs

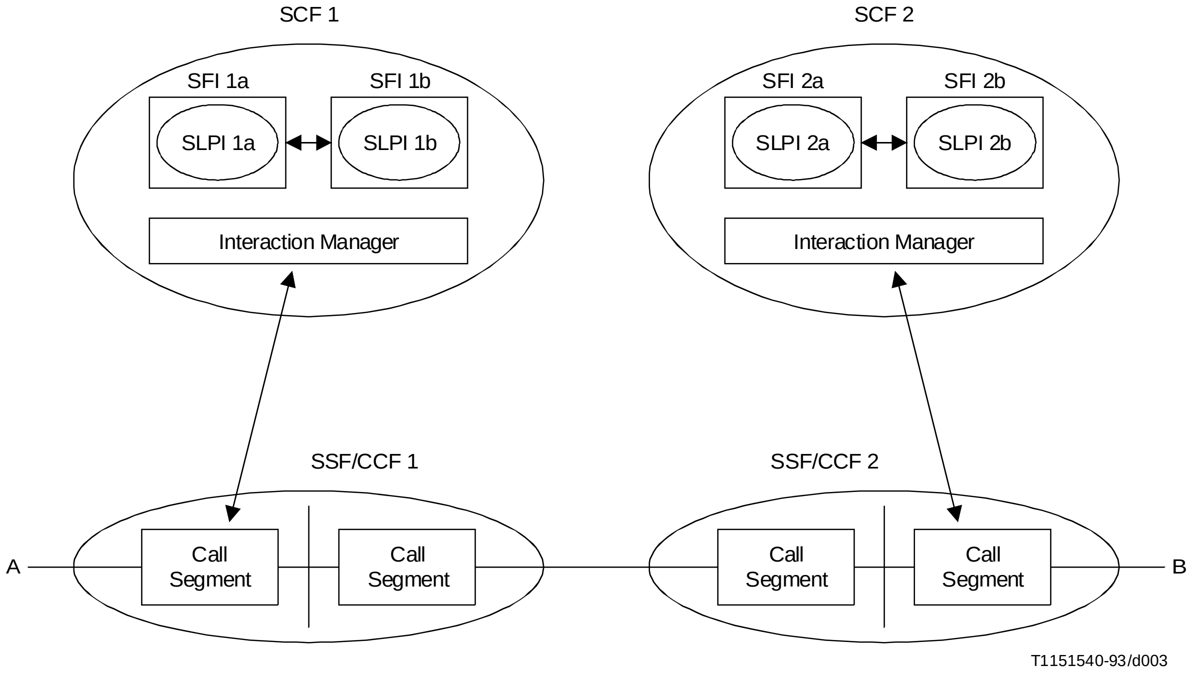
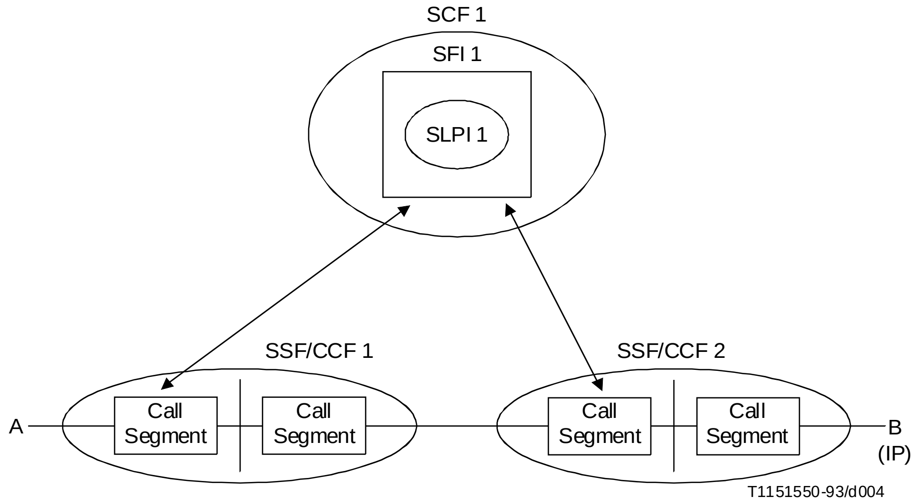
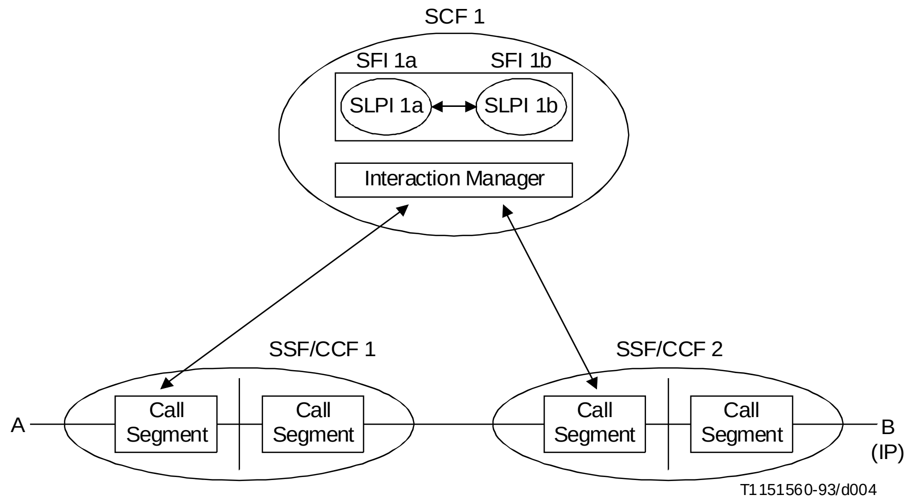


FIGURE 3/Q.1219

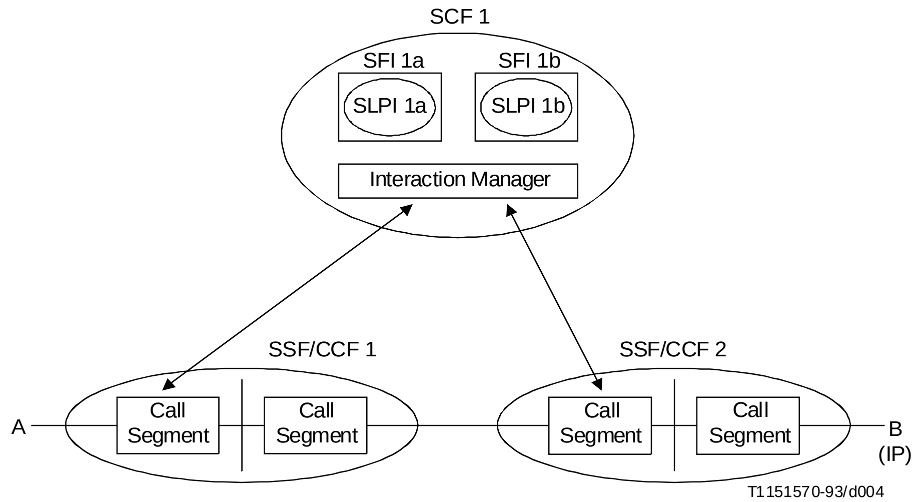
Interaction between SFIs



a) Basic Model for Affecting Multiple SSF/CCFs (e.g. assisting case)



b) Interaction between SLPIs



c) Interaction between SFIs

NOTE – From Figure 4 a) to Figure 4 c), the control relationship from SCF 1 is applied to the originating Call Segment in the SSF/CCF 2 because of illustrations of the assisting case. However, the relationship can be applied to the terminating Call Segment instead of the originating one.

FIGURE 4/Q.1219

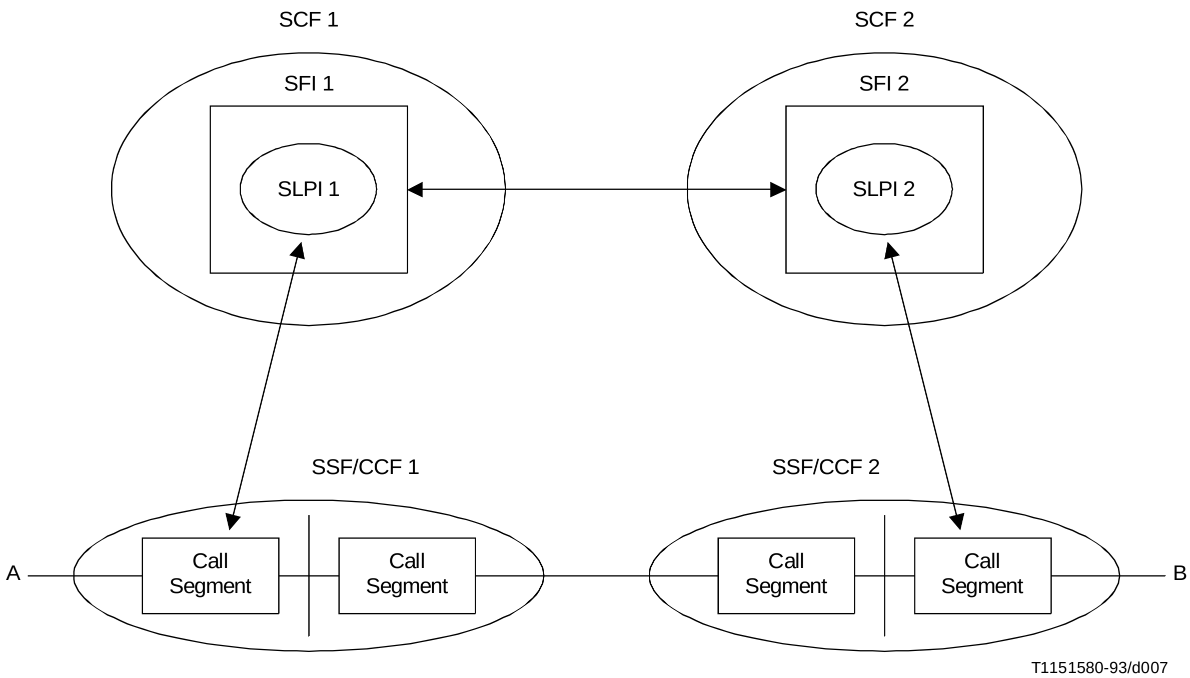


FIGURE 5/Q.1219

Interaction between SCFs

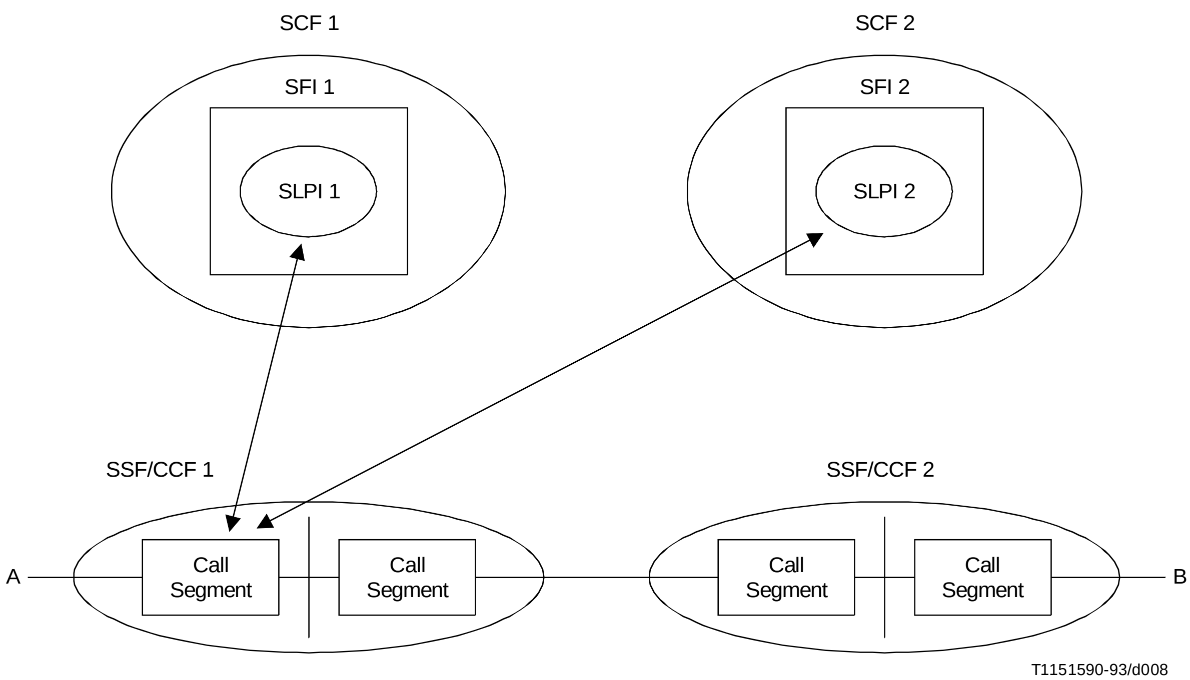


FIGURE 6/Q.1219

Two SCFs Affecting One Call Segment

4.3 Type-B service category

In Type-B services, several IN subscribers may be associated within a single call. During the call, subscribers may be added or dropped. These associations take place physically in the switches involved in the call (SSF/CCF functions) under the control of an SCF. The SCF will need rules to handle feature arbitration between subscribers involved in the call (e.g. incompatible screening lists). This may have to involve real-time consultations between the SCFs which "represent" the various parties involved in the call. Rules will also be required to handle topological decisions (e.g. which physical switches should be chosen to "join" groups of subscribers scattered around a network).

CS-1 standards do not encompass "Type-B" services because of their increased operational and control complexity. Type-B services are considered to be for further study (FFS).

4.4 Phases of deployed services

After deployment a service will pass through three stages: the subscription, the activation (and related deactivation), and the invocation. In order to reach a smooth and quick introduction of services (one of the goals of IN), there should be a notion of these stages throughout the planes of the IN CS-1 architecture. However, IN CS-1 focuses on invocation, with little attention paid to activation and no attention paid to subscription. CS-1 does not contain a structure to relate these stages to each other. Although it is expected that future Capability Sets will provide the linkage between these three stages.

5 CS-1 Architecture

5.1 Functions

The functions addressed in IN CS-1 are grouped into the following categories:

- Call Control related functions;
- Service Control related functions.

A brief description of each of these functions follows. A more detailed description can be found in Recommendation Q.1214, Distributed Functional Plane for IN CS-1.

Call control related functions

The Service Switching Function (SSF) provides an interface between a Call Control Function (CCF), to a Service Control Function (SCF). The SSF allows the CCF to be directed, from a service control view, by the SCF in order to realize a service in an IN CS-1 structured network.

The Specialized Resource Function (SRF) provides a pool of resources for access by other network entities. These resources include, but are not limited to, DTMF sending and receiving, speech recognition, synthesized speech provision, etc.

The Call Control Function (CCF) refers to call and connection handling in the classical sense, e.g. in a switching system (exchange). The CCF is responsible for establishing and controlling the connection related aspects of an IN CS-1 supported service under the control of the SCF, via the SSF.

The Call Control Agent Function (CCAF) provides user access to the IN CS-1 structured network.

Service control related functions

The Service Control Function (SCF) contains the IN CS-1 service logic and handles IN CS-1 service related processing activities. The SCF directs the CCF, via the SSF, in the realization of IN CS-1 supported services.

The Service Data Function (SDF) handles access to service-related data and network data, and provides consistency checks on that data. The SDF provides a logical data view to the SCF, thereby removing any dependency relative to the real data implementation in the SDF.

Figure 7 taken from Recommendation Q.1211, clause 7, describes the relationship of these functions:

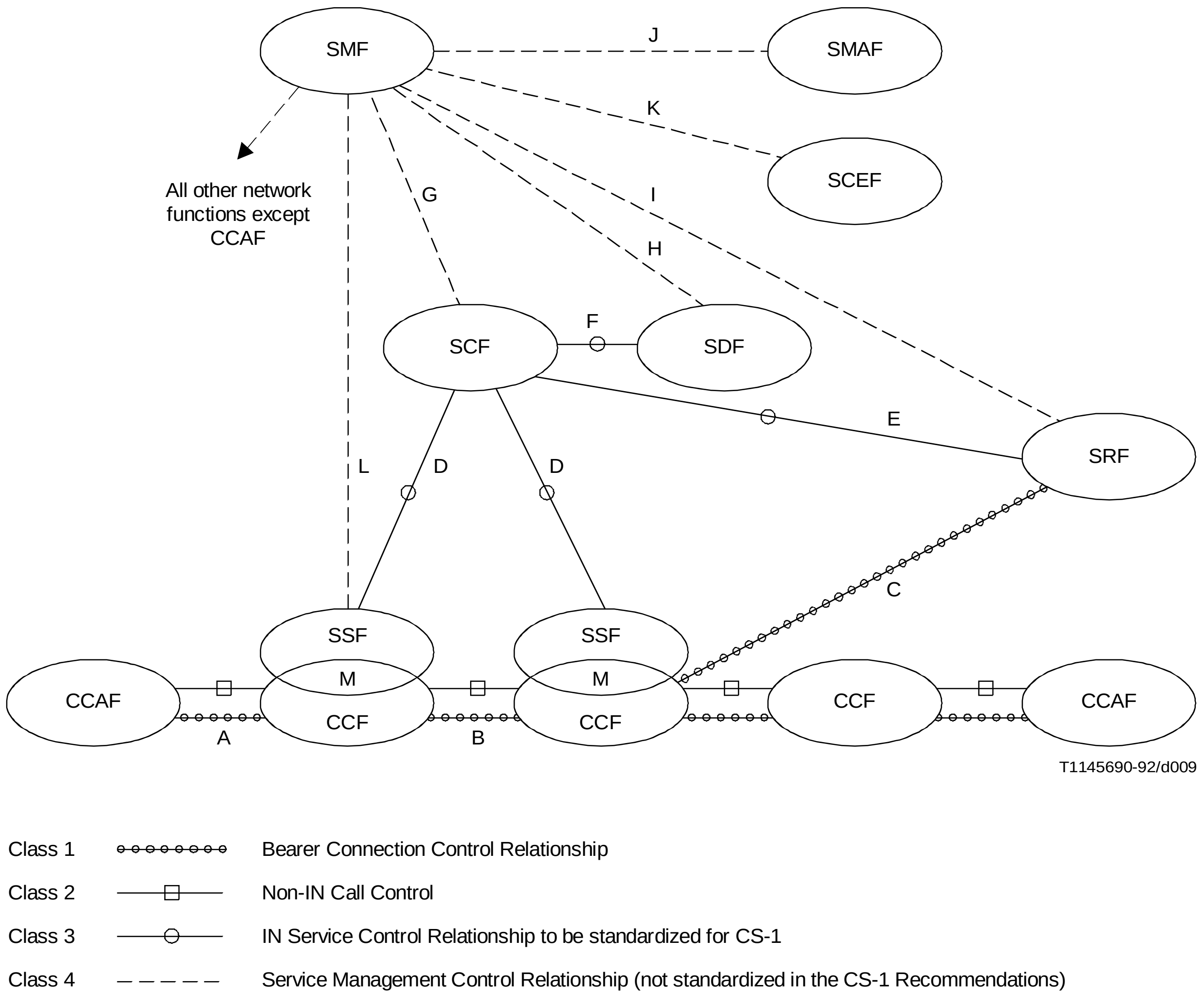


FIGURE 7/Q.1219

Functional relationships and reference points for CS-1

5.2 IN CS-1 plane relationships

(Reference, I.312/Q.1201, Principles of Intelligent Network Architecture.)

The following discussion is intended to point out to implementors areas where care should be exercised in developing service for IN CS-1 and some of the areas already identified for further study in future IN Capability Sets.

The IN Conceptual Model is a tool for describing the capabilities and characteristics of an IN structured network. As such it represents an abstraction of an IN structured network. These planes are described in the corresponding Recommendations, which contain references and details regarding the use of the IN Conceptual Model: Q.1213, Global Functional Plane for CS-1; Q.1214, Distributed Functional Plane for CS-1; and Q.1215, Physical Plane for IN CS-1 The Service Plane is not described in the IN CS-1 Recommendations, since it will be realized in future Capability Sets. The discussion here is related to the relationship between the four planes in CS-1.

There are four planes in the IN Conceptual Model that drives the architecture for IN CS-1. These planes are explained more completely in their corresponding Q.1210-Series Recommendations. The discussion here is relative to their relationship in IN CS-1. The four planes included in the model architecture are the Service, Global Functional, Distributed Functional, and Physical planes.

5.2.1 IN CS-1 Service Plane

For IN CS-1, the Services Plane is nearly empty with no attention paid to the general data structures on which services operate. This gap can lead to some difficulties in mapping the desired data on predescribed data structures in the Physical Plane, which occur in operations. A set of targeted Service Features (SFs) have been identified for IN CS-1, but are not realized in the Service Plane. They were used to ensure that the required functionality would be available to support the set of targeted Services for IN CS-1. In future Capability Sets, Service Features will be realized on the Service Plane. Recommendation Q.1211 defines the set of targeted services and service features for CS-1. These sets were identified in order to determine the capabilities and functionalities to be provided in IN CS-1, since this is the first set in the evolution of IN Recommendations. The implementor must use caution when considering the coupling and interaction of these service features in IN CS-1. Currently, no mechanism exists to resolve this issue relative to the service plane, however, there are applicable considerations and mechanisms for the distributed functional plane (see Recommendation Q.1213) that are of direct use to implementors of IN CS-1.

5.2.2 IN CS-1 Global Functional Plane

The Global Functional Plane contains the Service Independent Building Blocks that were used in IN CS-1 to develop information flows and operations. These SIBs are fully defined in Recommendation Q.1213 which used the services and service features defined in Recommendation Q.1211 to ensure that the needed information flows and operations were included in IN CS-1.

Although these building blocks have been identified, their cooperation and interaction have not yet been fully defined. In other words, the Global Service Logic needed to link the SIBs has yet to be fully defined. Further, an overall data structure on which SIBs act is not available in IN CS-1. Guidance on Global Service Logic is provided in clauses 2/Q.1203 and 5/Q.1203, and clause 4/Q.1213 and guidance on data associated with SIBs is contained in 3.3/Q.1203, and is applied in clause 2/Q.1213. The SIBs provided in IN CS-1 are to be used to describe services, to aid in decomposing services, and to provide a common vehicle for the discussion and understanding of the components of services.

Additional considerations in this plane for IN CS-1 are: activation of services, monitoring functions, and addressing. The Global Functional Plane offers the possibility to express the activation and deactivation of a service by means of the Service Data Management SIB. In the decomposition, however, the possibility of setting a Detection Point – Response at the Distributed Functional Plane has been omitted. In the Distributed Functional Plane some assumptions have to be made for handling on-hook triggers and error situations. It is suggested that the incorporation of a special monitor function would cover these situations. Regarding addressing, a lot of parameters in the information flows have physical meaning. Examples are the database and SRF addresses that have to be specified in a query and the Play Announcement message. It is proposed that SLPs be independent of physical implementation.

5.2.3 IN CS-1 Distributed Functional Plane

The Distributed Functional Plane contains the Functional Entities which provide service capabilities in IN CS-1. Most information flows on the Distributed Functional Plane map directly on operations in the physical plane. Also, the relationship between the Basic Connection State Model in the Distributed Functional Plane and the Legs in the Physical Plane require further study.

The IN DFP architecture for CS-1 encompasses only the functional entities related to IN service execution (see Recommendation Q.1204 for a full listing of all the functional entities on the generic DFP). A review of Recommendation Q.1214 (IN DFP Architecture for IN CS-1) provides static and dynamic models of these functional entities. These models are used to define how IN service control interacts with basic call processing and to understand the nature of the functional entity actions required for CS-1. Recommendation Q.1214 also provides the SIB stage 2 information flows and functional entity actions for CS-1 and detailed information flow descriptions as the basis for specifying IN protocols.

5.2.4 IN CS-1 Physical Plane

The Physical Plane models the physical aspects of an IN-structured network. The model identifies the different physical entities and protocols that may exist in deployed IN-structured networks. Each physical entity reflects the functionality of one or more Functional Entities, and each protocol reflects the functionality of the information flows and information elements described in the Distributed Functional Plane.

Recommendation Q.1215 describes the Physical Plane for IN CS-1 and provides an illustrative mapping of Functional Entities to physical entities for implementations of IN CS-1. Recommendation Q.1218 describes the protocol and procedures for IN CS-1 and provides a complete mapping of the information flows to application layer protocol data units (i.e. operations). Recommendation Q.1218 focuses primarily on the syntax of the application layer protocol. The semantics of the INAP for CS-1 are presented in clause 3/Q.1218 and in the SIB Stage 2 descriptions and Functional Entity models within Recommendation Q.1214.

Recommendation Q.1218 defines the syntax of several error types for failure scenarios and describes general error procedures; however the specific error handling procedures and semantics are not completely described in the IN CS-1 Recommendation series. Network operators should complete the specification of error handling procedures and semantics. General guidelines for error handling procedures are described in 12.4.3/Q.1400 and in 2.4/Q.775 and 3.2.1.4/Q.775. Refer to A.2.7 (Time Handling Aspects) for service examples with errors and failures.

The concepts of Application Service Element and Application Context are used within the INAP for CS-1 to support flexibility and evolution of implementations of IN CS-1. See 6.5 for specific guidelines in these areas.

5.3 Interfaces and relationships

The relationships between the entities on the DFP are maintained via a set of defined interfaces for the IN Service Control relationship in CS-1. The possible interfaces between these functions are defined by letter codes for ease of tracking and are as follows:

CS-1 Interface (see Note)	Description
D	SSF-SCF
E	SCF-SRF
F	SCF-SDF

NOTE – A complete listing and discussion of all of the interfaces defined for an IN Structured Network may be found in Recommendation Q.1211.

6 Infrastructure in CS-1

The following subclauses provide a description of the foundational components (infrastructure) of IN CS-1.

6.1 Service Independent Building Blocks (SIBs)

A service independent building block is composed of a set of Functional Entity Actions (FEAs) that either independently, or in combination with other SIBs, are used to model service entities. SIBs are used by service logic and they provide a global view of an IN-structured network as a single entity. SIBs are independent of the distribution of the network capabilities within the physical. SIBs are re-usable and reflect network-wide capabilities.

The reader should note that although SIBs are defined as being used to build service entities, they are not implementable in a physical entity. They are abstract representations of network capabilities that exist in an IN structured network. Thus they exist only in the global functional plane of the INCM and not in a physical network.

While SIBs only exist in the GFP, they are ultimately realized in physical entities via SIB stage 2 decomposition into FEAs and information flows in the DFP. They are then mapped into operations and procedures in the Physical plane. In this manner the SIBs are useful in their utility in decomposing/describing services and their potential utility as a part of service creation.

For a complete description of the following SIBs, refer to Recommendation Q.1213.

The following SIBs are defined for IN CS-1:

1 – Algorithm

This SIB applies a mathematical algorithm to data to produce a data result.

2 – Charge

This SIB determine if there is any special charging treatment for the call, where special refers to any charging in addition to that normally performed by the basic call process. In general this involves identifying:

– the resources for which charging is to occur;

– the charging structure that is to apply for each such resource;

– where the charges are to be directed.

This SIB is not responsible for the subscriber billing process.

3 – Compare

This SIB performs a comparison of an identifier against a specified reference value. Three results are possible:

– identifier is GREATER THAN the value;

– identifier is LESS THAN the value;

– identifier is EQUAL TO the value.

4 – Distribution

This SIB allows the user to distribute calls to different logical ends of the SIB dependent on user specified parameters.

5 – Limit

This SIB limits the number of calls related to IN provided service features. Such limiting will be based on user specified parameters. This SIB is not used for network congestion management functions.

6 – Log Call Information

This SIB logs detailed information for each call into a file. The collected information may be used by management services (e.g. statistics, etc.) and not by call related services.

7 – Queue

This SIB provides sequencing of calls to be completed to a called party.

8 – Screen

This SIB performs a comparison of an identifier against a list to determine whether the identifier has been found in the active list.

9 – Service Data Management

This SIB enables end users specific data to be replaced, to add, change, retrieved, incremented, or decremented.

10 – Status Notification

This SIB provides the capability of inquiring about the status and/or status changes of network resources.

11 – Translate

This SIB determines output information from input information.

12 – User Interaction

This SIB allows information to be exchanged between the network and a call party, where a call party can be either a calling or a called party.

13 – Verify

This SIB provides confirmation that information received is syntactically consistent with the expected form of such information.

14 – BCP

For CS-1 the Basic Call Process has been defined as a specialized SIB which provides the basic call capabilities. These capabilities enable the use of global service logic as well as other SIBs to completely describe CS-1 services and service features. Nine Points of Initiation (POIs) for Global Service Logic (GSL) interfaces have been specified. In addition, six Points of Return (PORs) for GSL have been identified. The POIs and PORs are listed below:

POI

Call Originated
Address Collected
Address Analysed
Prepared to complete Call
Busy
No Answer
Call Acceptance
Active State
End of Call

POR

Continue with existing data
Proceed with new data
Handle as transit
Clear Call
Enable call party handling
Initiate Call

6.2 Service logic

Service Logic is a set of routines and rules that interact with network capabilities and the basic call state model to develop and implement services in an IN-structured network. Service logic in IN CS-1 falls into two categories, Global and Distributed.

Global Service Logic (GSL) has been defined as the "glue" that describes the order in which SIBs can be chained together to accomplish services. For a given CS-1 service/service feature, GSL is used to describe:

i) A specific POI which will define the functional launching point from the BCP to the SIB chain.

ii) A specific set of PORs where the SIB chain can logically return to the BCP.

iii) The pattern and order of SIBs which are to be chained together.

iv) Data parameters for each SIB in the SIB chain.

Distributed Service Logic (DSL) exists on the DFP. There is one set of DSL per SIB and it uses the Functional Entity Actions and Information Flows for service execution.

6.3 Functional Entity Call/Service Logic Processing Models

Functional Entity call/service logic processing models provide a tool used by IN architects to model a call and to understand and describe the distribution of functions between functional entities and functional entity relationships. IN call/service logic processing encompasses call and connection processing in the SSF/CCF, service logic execution in the SCF, and the use of supporting resources in the SRF and supporting data in the SDF. Recommendation Q.1214 describes this IN call/service logic processing for IN CS-1 in terms of call modelling and modelling of service logic processing.

– Call modelling provides a high-level service and vendor/implementation independent abstraction of IN call and connection processing in the SSF/CCF. This abstraction provides an observable view of SSF/CCF activities and resources to the SCF, enabling the SCF to interact with the SSF in the course of executing service logic.

– The modelling of service logic processing provides an abstraction of SCF activities and resources needed to support this service logic execution, as well as an abstraction of SRF and SDF activities and resources accessible to the SCF.

The IN CS-1 modelling in Recommendation Q.1214 is based on the general modelling objectives, assumptions, and architecture described in clause 3/Q.1204, and makes use of the tools identified in its annexes, as applicable to IN CS-1.

6.3.1 Call Modelling for IN CS-1

To provide an observable view of the SSF/CCF to the SCF, and to enable the SCF to interact with the SSF, call modelling for IN CS-1 provides the following:

- a foundation based on the existing base of evolvable network technology;
- single-ended view of SSF/CCF call processing in terms of both Originating and Terminating Basic Call State Models (BCSMs);
- a framework for defining triggering requirements in the BCSMs to invoke IN service logic and to report call processing events to IN service logic in terms of Detection Points (DPs), which can be used in combinations by the implementor to provide network services;
- a framework for ensuring correct sequencing of functions within an SSF/CCF in terms of BCSM Points in Call (PICs) and transitions;
- rules of representing and handling service logic instance interactions; and
- a framework for defining the information flows (relationships) between an SSF and an SCF.

Examples of call/connection processing functions accessible to the SCF from the SSF/CCF as reflected in the related IN CS-1 information flows include functions to:

- influence the flow of call processing (e.g. rerouting a call, clearing a call, or providing serial calling);
- access and change information related to call processing (e.g. address translation, routing information);
- manipulate the connectivity of the call (e.g. forwarding and other capabilities for further study);
- monitor for events related to call processing and connectivity manipulation (e.g. no answer, busy, disconnect).

6.3.2 Modelling of Service Logic Processing for IN CS-1

To provide an abstraction of SCF activities and resources, as well as SRF and SDF activities and resources accessible to the SCF, modelling of service logic processing for IN CS-1 provides the following:

- a high-level vendor/implementation independent abstraction of service logic processing in the SCF, specialized resources in the SRF, and service data in the SDF;
- a characterization of the capabilities of an SRF and SDF made available to an SCF;
- a framework for defining the information flows (relationships) between an SRF and an SCF and between an SDF and an SCF.

Note that the SRF, SCF and SDF modelling only provides high-level modelling of necessary functionality, but makes no recommendations on specific mechanisms to implement this functionality (e.g. no recommendations on service logic invocation, management of service logic instance interactions, reservation and allocation of specialized resources, data architecture and access to data). Also, note that the modelling primarily addresses the functionality for normal call processing scenarios.

Examples of specialized resource functions accessible to the SCF from the SRF as reflected in the related IN CS-1 information flows include functions to:

- send information to users participating in a call (e.g. prompts for information, announcements);
- receive information from users participating in a call (e.g. authorization codes);

– modify user information (e.g. text to speech synthesis, protocol conversion); and

– provide specialized connection resources (e.g. audio conference bridge, information distribution bridge).

Examples of service data processing functions accessible to the SCF from the SDF as reflected in the related IN CS-1 information flows include functions to:

– access service information (e.g. subscription data parameters); and

– update service information (e.g. sum of charging).

6.3.3 General Considerations

6.3.3.1 Evolutionary Aspects

The call/service logic processing models, in their continuing evolution, should be able to support the needs described in 1.3/Q.1201 (Evolution of IN), such as more granular basic call states, additional triggering requirements, call/connection separation, and robust service feature interactions mechanisms. These evolutionary aspects are not addressed in IN CS-1, and are for further study in future capability sets. Annex A/Q.1204 provides a starting point for the study of two of these aspects: more granular basic call states and additional triggering requirements.

6.3.3.2 Relation of Modelling to Products

Since call modelling and modelling of service logic processing only provides an observable (i.e. external) view of SSF/CCF, SCF, SRF, and SDF activities and resources, this modelling does not imply an obligation to vendors to implement functional entities into products as a one-to-one mapping of functional entity model components.

6.3.3.3 BCSM Aspects

The following clarifications are provided with respect to the CS-1 BCSM Description in 4.2.2.2/Q.1214:

a) *Collect_Information Point in Call (PIC 2)* – In the case of ISDN *en bloc* sending, the receipt of a SETUP message detected at the Origination_Attempt_Authorized Detection Point (DP 1) causes the BCSM to pass through PIC 2 to the Collected_Information DP (DP 2), without further processing in PIC 2. Note that the BCSM transitions to DP 2 when the *initial* information package/dialling string is received from the calling party – this occurs when enough information is received to proceed with call processing (e.g. as in the case of ISDN overlap sending or MF outpulsing). If DP 2 is armed as a Trigger Detection Point-Request (TDP-R), the SSF sends an initiating DP request (i.e. InitialDP or CollectedInformation information flow) to the SCF when enough information is received to determine if the TDP criteria are met, but does not suspend BCSM processing until the *complete* information package/dialling string is received.

b) *Analyse_Information (PIC 3)* – One of the results of processing in this PIC is determination of the routing address. One of the following constitutes the routing address:

   i) called party number only (called party number is served by the SSF);

   ii) called party number and route index, where the route index is a pointer to a trunk group to route an outgoing call attempt on (called party number is served by another SSF);

   iii) called party number and route index, where the route index is a pointer to a list of trunk groups to route an outgoing call attempt on (called party number is served by another SSF).

c) *Mapping to Recommendation Q.71* – Annex B provides a Specification Description Language (SDL) description of the BCSM consistent with the CCF basic call processing description for ISDN defined in Recommendation Q.71. To do so, it is assumed that BCSM processing is performed in the CCF.

d) In May 1993, modifications to Recommendation Q.71 to align it and the BCSM have been proposed but have yet to be reviewed. Relevant material on the insertion of DP Hooks in Recommendation Q.71 is FFS.

With respect to the DP criteria listed in 4.2.2.4/Q.1214, note that these DP criteria only apply to TDPs. DP criteria for Event Detection Points (EDPs) are addressed by the RequestReportBCSMEvent information flow. In addition, note that one or more DP criteria may be applicable at a given DP. The assignment of DP criteria to a TDP and the combinations of DP criteria applicable at a given DP are for further study. Note further that the assignment of DP to a TDP on either a line/trunk, group or office basis may have an impact on the memory and real-time performance requirements of the SSF/CCF. The DP criteria for CS-1 are defined below, as applicable to a given TDP.

Note that the applicability of DP criteria at a given DP depends on when call processing information is available and how long it is retained. IN CS-1 places no requirements on equipment suppliers in this area. If network and service providers plan to implement IN CS-1 services in a multi-supplier environment, they should consider formulating such requirements to ensure consistent implementations across supplier equipments. Such requirements should be considered carefully so as not to adversely impact memory and real-time performance aspects of SSF/CCF processing.

1) *Trigger assigned* (unconditional/conditional on other criteria) – An indicator of the armed/disarmed status of a TDP assigned on a line/trunk, group, or office basis.

   The trigger assigned criterion can be used by itself or in conjunction with other criteria at a TDP. If the trigger assigned criteria criterion is unconditional at a TDP, then it is used by itself – no other DP criterion need to be satisfied at the TDP before informing the SCF that the TDP was encountered. If the trigger assigned criterion is conditional at a TDP, then it is used in combination with other criteria at the TDP – all of the other DP criteria in the combination need to be satisfied before informing the SCF that the TDP was encountered.

   Applies at all DPs (all DPs can be provisioned as TDPs).

2) *Class of Service* – This is either a i) customer class of service, ii) trunk class of service, or iii) private facility class of service; i) is a code that identifies all attributes of a line that require distinctive call processing treatment (e.g. for party lines and coin lines); ii) is a code that identifies attributes of a trunk group such as type of signalling used, and iii) is a code that identifies attributes of a private trunk group such as type of signalling used and flash repeat capability.

   Originating access (user/network) class of service is available at DP 1 and could be applicable at DPs 1-10. Terminating access (user/network) class of service is available at DP 12 and could be applicable at DPs 12-18.

3) *Specific B-channel identifier* – An identifier of the specific B-channel on an ISDN BRI or PRI from which a call attempt has originated or to which a call attempt is to be terminated.

   A-party B-channel identifier is available at DP 1 for a party served by an ISDN BRI or PRI only and could be applicable all DPs 1-10. B-party B-channel identifier is available during PIC 8 after an idle terminating facility has been selected for a party served by an ISDN BRI or PRI only and could be applicable at DPs 14-17 and 18 (only after an idle terminating facility has been selected).

4) *Specific digit strings* – A string of digits that must match collected digit strings for dialling plans in which a variable number of digits are to be collected. It could be zero or more digits (e.g. to trigger on “off-hook delay”).

   The string of digits should be consistent with the structure of the dialling plan and should be administerable. For example, the network provider may specify the first N digits where N is consistent with the structure of the E.164 numbering plan, or any other appropriate numbering plan.

   Collected digit strings can be available at DP 1 for a party served by an ISDN BRI or PRI using en bloc sending and at DP 2 for a party served by a non-ISDN line. Since collected digit strings are not analysed until PIC 3 (except to determine if a sufficient number of digits have been collected), this criteria could be applicable at DPs 3-10. DP 3 (mandatory) and DPs 4-10 (optional) is proposed since not all SSP suppliers may retain this information for the duration of the call/attempt:

   – Collected digit string can be available at DP 1 for an SS7 trunk.

– Collected digit string can be available at DP 2 for a party served by a conventional trunk, ISDN BRI or PRI using overlap sending, and private facilities.

5) *Feature codes* (e.g. *XX,#) – A vertical service code, such as a "#" or a two-digit or three-digit code preceded by "*" or "11," that precedes any subsequent digit collection (e.g. according to the "normal dialling plan").

Feature codes can be available at DP 1 for a party served by an ISDN BRI or PRI using en bloc sending or for an SS7 trunk, and can be available at DP 2 for non-ISDN lines and private facilities. Since collected digit strings are not analysed until PIC 3 (except to determine if sufficient information has been collected), this criteria could be applicable at DPs 3-10. DP 3 (mandatory) and DPs 4-10 (optional) is proposed since not all SSP suppliers may retain this information for the duration of the call/attempt:

– Feature codes can be available at DP 2 for a party served by an ISDN BRI or PRI using overlap sending.

6) *Prefixes* (e.g. 0+, 00+, 011, 01, 1+) – A string of digits that are not feature codes or access codes and which precede any subsequent digit collection (e.g. according to the "normal dialling plan")

Prefixes can be available at DP 1 for a party served by an ISDN BRI or PRI using *en bloc* sending, and can be available at DP 2 for non-ISDN lines, conventional trunks, and private facilities. Since collected prefix information is not analysed until PIC 3 (except to determine if sufficient information has been collected), this criteria could be applicable at DPs 3-10. DP 3 (mandatory) and DPs 4-10 (optional) is proposed since not all SSP suppliers may retain this information for the duration of the call/attempt:

– Prefixes can be available at DP 2 for a party served by an ISDN BRI or PRI using overlap sending.

7) *Access codes* (e.g. 8+) for customized numbering plan – A string of digits in a customized numbering plan that matches access codes such as attendent access codes, access codes to escape to the public network, access codes to access a private facility, access codes to access a private network, and feature access codes.

Access codes can be available at DP 1 for a party served by an ISDN BRI or PRI using *en bloc* sending, and can be available at DP 2 for non-ISDN lines and private facilities. Since collected access codes are not analysed until PIC 3 (except to determine if sufficient information has been collected), this criteria could be applicable at DPs 3-10. DP 3 (mandatory) and DPs 4-10 (optional) is proposed since not all SSP suppliers may retain this information for the duration of the call/attempt:

– Access codes can be available at DP 2 for a party served by an ISDN BRI or PRI using overlap sending.

8) *Specific abbreviated dialling strings for customized numbering plan* – An abbreviated called party number in a customized numbering plan that must match collected address information.

Abbreviated address information can be available at DP 1 for a party served by an ISDN BRI or PRI using *en bloc* sending, and at DP 2 for a party served by a non-ISDN line or private facility. Since collected address information is not analysed until PIC 3 (except to determine if sufficient information has been collected), this criteria could be applicable at DPs 3-10. DP 3 (mandatory) and DPs 4-10 (optional) is proposed since not all SSP suppliers may retain this information for the duration of the call/attempt:

– Specific abbreviated dialling strings can be available at DP 2 for a party served by an ISDN BRI or PRI using overlap sending.

9) *Specific calling party number strings* – A string of digits that must match the calling party number, which is a local, national, or international E.164 number or a number in a customized numbering plan. If a call has been forwarded, the calling party number is the number of the original calling party.

The calling party number is available at DP 1 in the originating BCSM and DP 12 in the terminating BCSM for a call originating from a non-ISDN line, ISDN BRI or PRI, and can be available at DP 1 and DP 12 for SS7 trunks. This criterion could be applicable at all DPs.

10) *Specific called party number strings* – A string of digits that must match the called party number, which is either a local, national, or international E.164 number, or a number in a customized numbering plan; the latter is not supported by SS7 or conventional trunks. If a call has been forwarded, the called party number is the number of the party that the call is forwarded to.

    The called party number can be available at DP 1 for a party served by an ISDN BRI or PRI using *en bloc* sending or for an SS7 trunk, and can be available at DP 2 otherwise. Since collected address information is not analysed until PIC 3 (except to determine if sufficient information has been collected), this criteria could be applicable at DPs 3-10 and 12-18. In the originating BCSM DP 3 (mandatory) and DPs 4-10 (optional) is proposed. No specific proposals are made for DPs 12-18 in the terminating BCSM.

11) *Nature of address* (e.g. Subscriber Significant Number, National Significant Number, International number) – An indicator of whether the called party number is a private, local (or subscriber), national, or international number.

    The nature of address is available at DP 3. This criteria could be applicable at DPs 3-10. DP 3 (mandatory) and DPs 4-10 (optional) is proposed since not all SSP suppliers may retain this information for the duration of the call/attempt.

12) *Bearer capability* – An indicator of the bearer capability as defined in Recommendation Q.1218.

    The bearer capability information is available at DP 1. This criteria could be applicable at all DPs.

13) *Feature activation/indication* (unconditional/conditional on specific feature patterns) – In a local exchange only, a feature activation/indication on an ISDN BRI or PRI or that is detected at the Mid_Call DP (e.g. "hook-flash," #, etc.) for ISDN and non ISDN lines that can be sent in conjunction with or preceding other address/digit collection.

    A feature activation/indication can be available at DP 1-10 in the originating BCSM for a party served by an ISDN BRI or PRI, and can be available at DP 8 in the originating BCSM for a party served by a non-ISDN line. A feature activation/indication can be available at DP 14-18 in the terminating BCSM for a party served by an ISDN BRI or PRI, and can be available at DP 16 in the terminating BCSM for a party served by a non-ISDN line. Since collected feature activation information is not acted upon before PIC 3 in the originating BCSM and before PIC 9 in the terminating BCSM, this criterion could be applicable at DPs 3-10 and DPs 14-18.

14) *Facility information* (unconditional/conditional on specific facility information patterns) – A match on the Facility Information Element contained in a signalling message as defined in DSS 1 and ISUP.

    Applicable DPs can be determined by mapping signalling messages to the BCSM (see 4.2.2.2/Q.1214 and Appendix II of this Recommendation) and are for further study.

15) *Cause* (unconditional/conditional on specific cause patterns) – A match on the Cause IE contained in a signalling message as defined in DSS 1 and ISUP or an indicator of the cause of specific events of interest. Further study is required to identify the cause values needed as DP criteria for CS-1 services from the complete list of cause values specified by Recommendation Q.1218.

    Route selection failure information is available at DP 4, busy cause information is available at DPs 5 and 13, and release cause information is available at DPs 9, 10, 17, and 18. This criteria is applicable at the identified DPs.

With respect to BCM DP processing described in 4.2.2.5/Q.1214, note that DP processing spans the CCF and SSF, with the CCF detecting events in the BCSM and passing them on to the SSF, where EDP and TDP criteria are checked. This separation of functionality between the CCF and SSF is not defined in Recommendation Q.1214 since the CCF and SSF are considered as a unit in CS-1, but is useful to understand from the perspective of mapping to Recommendation Q.71, which only addresses CCF processing, and from an evolutionary perspective, in which the CCF and SSF may ultimately be separated. As such, Annex B provides an SDL description of DP processing in the SSF to complement the SDLs on BCSM processing in the CCF. These SDLs are based on the SDL description in Recommendation Q.1214, assuming the separation of functionality between the CCF and SSF described above rather than treating the SSF/CCF as a unit.

Reset of Resources in Multiple Request Sessions

In session calling (e.g. UPT follow-on and Credit Card) and more generally when the user does not hang up between IN service requests, the O-BCSM is used two or more times.

As a consequence the O-BCSM may be reset by SCF operations like CollectInfo, Connect, etc., from a DP to the same or some other DP to restart processing (even a DP already passed in previous processing).

NOTE 1 – An inconsistency exists between the SIB descriptions (including the BCP SIB) and the information flows of Recommendation Q.1214. In the former case, the functionality as described in this text is possible, but the functionality is not included in the information flow descriptions.

Concurrently, resources in the CCF must also reset in a way that enables switch logic to process the new user request. This includes the need to release physical resources like outgoing legs, etc., and the need to retrieve essential data like calling user identity and more generally to reset the software resources in a status coherent with the DP at which processing will restart.

As a first attempt to clarify procedures, the following set of actions could be assumed:

– The CCF monitors and controls the state of all CCF related physical resources.

– The CCF stores any essential data for all passed PICs in IN calls.

– This data is maintained until the calling (controlling) user releases.

– The SCF sends the right operation to reset the O-BCSM and to arm/rearm relevant DPs.

– When the O-BCSM returns to the indicated DP, the CCF independently resets the CCF related physical and logical resources in coherence with the PIC.

These actions would ensure independence when two IN services are invoked in sequence.

It could be useful to define, for each PIC, the data items that should be kept available.

NOTE 2 – Modification to Recommendation Q.71 to support the required CCF actions have been proposed but as of May 1993 they have not been reviewed by SG 11.

6.4 Information Flows

An information flow is a message that exists between two Functional Entities (FEs) and is defined and scoped in terms of "logical objects", and the operations which can be performed on those objects. These objects represent an abstraction of resources contained within an FE which another FE can manipulate.

For each relationship the following information is given:

1) The conditions under which the relationship can be established and terminated.

2) The information flows between the two entities concerned in the relationship, *in alphabetical order*.

Information flows between two FEs consist of either a clientrequest/server response pair, or a client request alone.

Information flows between two FEs generally takes place independently of information flows between other FEs.

6.4.1 Requirement for an Information Flow from SCF to SSF to Initiate "Call Follow-On"

It has been determined that follow-On Calls as required by, for example, UPT, GVNS, and Charge Card services, cannot be fully supported by the information flows currently contained in clause 6/Q.1214. Specifically, an information flow from SCF to SSF is required to allow the SCF to initiate forward clearing of the "B" party by the SSF/CCF. Proposals for such an information flow include forms of Release Call information flow or Release Call Party Connection. The latter was intended for such a purpose but is currently in Appendix I/Q.1214, since it has not yet been fully defined.

For purposes of Stage 2 descriptions of those IN supported services which require follow-on call, it is suggested that Release Call Party Connection be used, provided it is noted that the information flow is not fully defined and is subject to further study. The study of the information flows in Appendix I/Q.1214, and the development of the corresponding INAP operations, are under consideration in Study Group 11.

6.5 Intelligent Network Applications Protocol (INAP)

IN CS-1 defines the Intelligent Network Applications Protocol (INAP) required to support the services and service features defined for IN CS-1. It supports interactions between the following four FEs, as defined in the IN functional model:

- Service Switching Function (SSF);
- Service Control Function (SCF);
- Specialized Resource Function (SRF);
- Service Data Function (SDF).

The definition of the protocol can be split into three sections:

1) the definition of the primitive service which the protocol provides;
2) the definition of the messages transferred between entities;
3) the definition of the actions taken at each entity.

A mapping between the Information Flows and Operations is provided in Recommendation Q.1218. The operation definitions are in Abstract Syntax Notation (ASN.1), and the actions are defined in terms of state transition diagrams.

6.5.1 General ASE Discussion

An Application Service Element (ASE) defines a function or set of functions to facilitate application communications. An ASE may be general and used by various application protocols, or an ASE may be specific to an application and used within one or more contexts of an application protocol. The general ASEs that INAP uses include Transaction Capabilities (TC) and Remote Operations Service Element (ROSE). Twenty-five specific ASEs have been defined for INAP (CS-1). These additional ASEs are based on the criteria for grouping operations into ASEs as described in 11.5.1/Q.1400, which include functional distribution, modular re-use, and future evolution. For INAP, these additional ASEs are not strictly defined (i.e. through using the APPLICATION-SERVICE-ELEMENT macro as defined in Remote-Operations-Notation-extension {joint-iso-ccitt remote-operations(4) notation-extension(2)}) but are loosely defined by comments in 2.4/Q.1218.

An Application Context consists of a combination of ASEs and the relationship between the ASEs. The INAP for IN CS-1 has a modular structure to support a variety of application contexts. For example, between the SSF and SCF FEs a possible application context would bind, for a given transaction, the ASEs for charging, status reporting, basic call control, etc. Similarly, an application context could be established between the SCF and SRF FEs for user interaction. Thus, an application context is typically a subset of the total INAP and specifies that portion of the protocol needed for communicating between the two types of FEs. No specific application contexts are defined for IN CS-1.

For an implementation of INAP for IN CS-1, the required ASE(s) and application contexts should be determined by the service needs to be supported and the Functional Entities supported by the Physical Entities in the network. An implementation of INAP need not support all of the ASEs in Recommendation Q.1218. Also, definitions of application contexts are implementation specific.

For additional information on ASEs, application contexts, and application layer structuring, see 4.3/Q.1400, “Architecture Framework for the Development of Signalling and OAM Protocols Using OSI Concepts.”

It is anticipated that minor extensions to the INAP for IN CS-1 may be needed to accommodate implementation experiences and for protocol evolution purposes. An abstract syntax is extended if an associated type is extended (e.g. a SEQUENCE type is extended by adding a new type definition to the sequence). The operations for CS-1 include a generic extension mechanism. Refer to 12.5/Q.1400 for the definition of this extension mechanism and guidelines for its use.

Several data type definitions in the INAP for IN CS-1 are OCTET STRING and have no further syntax description. Network operators should complete the definition of the syntax and semantics for these data types.

The INAP for CS-1 defines the syntax of several error types; however, the IN CS-1 Recommendations do not completely describe error handling procedures or semantics associated with these error types. Network operators should complete the specification of error handling procedures and semantics. General guidelines for error handling procedures are described in 12.4.3/Q.1400 and in 2.4/Q.775 and 3.2.1.4/Q.775. Additional information on the error handling semantics for IN CS-1 can be found in 4.2.2.5/Q.1214, which identifies error cases and SSF/CCF application processing procedures associated with DP processing and response processing from the SCF; Annex B/Q.1214, which identifies SSF/CCF – SCF information flow scenarios for error cases; and clause 3/Q.1218, which defines general INAP procedures for error handling and timer expiry for the SSF, SCF, SRF, and SDF protocol state machines (3.1.1.5, 3.1.2.5, 3.1.3.4, and 3.1.4.4, respectively). Also, refer to A.2.7 (Time Handling Aspects) for service examples with errors and failures.

There is common functionality among several of the defined operations to provide for the graceful evolution from different existing IN implementations to a common standardized IN. Operations that have common functionality include:

1) Connect (within the Connect ASE) and AnalyseInformation or SelectRoute (within the SSF call processing ASE); and

2) InitialDP (within the SCF activation ASE) and the basic BCP DP ASE operations.

Network operators should implement one of the two ASEs for the two areas within a given Application Context.

Additional guidance on the semantics of the operations is provided in the SIB Stage 2 and information flow descriptions in clauses 5/Q.1214 and 6/Q.1214, which complement the state transition diagrams.

6.5.2 General Application Context Discussion

In the CS-1 Recommendations, the Application Context (AC) definitions are not included. However, the guideline for ACs will be informative when implementing based on the CS-1 Recommendations. Therefore, the guideline is shown here. It also shows the framework for selection of optional features in CS-1.

The ASEs which are identified in Recommendation Q.1218 are numbered as described below:

1) SCF activation ASE;

2) Basic BCP DP ASE;

3) Advanced BCP DP ASE;

4) SCF/SRF activation of assist ASE;

5) Assist connection establishment ASE;

6) Generic disconnect resource ASE;

7) Non-assisted connection establishment ASE;

8) Connect ASE (elementary SSF function);

9) Call handling ASE (elementary SSF function);

10) BCSM event handling ASE;

11) Charging event handling ASE;

12) SSF call processing ASE;

13) SCF call initiation ASE;

14) Timer ASE;

15) Billing ASE;

16) Charging ASE;

17) Status reporting ASE;

18) Traffic management ASE;

19) Service management ASE;

20) Call report ASE;

21) Signalling report ASE;

22) Specialized resource control ASE;

23) User data manipulation ASE;

24) Cancel ASE;

25) Activity Test ASE.

Using these notations, the following possible sub-grouping of ASE combinations are used.

(1) SSF Call Processing relating ASEs

(1-1) Call Model Bases

S0 (Null)	:
S1 (Minimum Set 1)	:2,9,10,12,14
S2 (Initiate)	:2,9,10,12,14,13
S3 (Mid-Call)	:2,9,10,12,14,3
S4 (Int., Mid.)	:2,9,10,12,14,13,3

(1-2) Non Call Model Bases

S0 (Null)	:
S1 (Minimum Set 1)	:1,8,9,10,14
S2 (Initiate)	:1,8,9,10,14,13

(2) SRF Connect Control relating ASEs

RCC0 (Null)	:
RCC1 (Internal)	:6,7,22
RCC2 (External)	:5,6
RCC3 (Both)	:5,6,7,22

(3) SRF Control relating ASEs

R0 (Null)	:
R1 (SSF Relay)	:22
R2 (SRF Direct, Both)	:4,22

(4) Assisted/Handed-off SSF relating ASEs

A/H0 (Null)	:
A/H1 (Assisted/Handed-off)	:4

(5) Charging relating ASEs

CH0 (Null)	:
CH1 (Furnish)	:15
CH2 (Apply)	:16
CH3 (Furnish, Apply)	:15,16

(6) Charging Event relating ASEs

CE0 (Null)	:
CE1 (Notification)	:11

(7) Resource Status Relating ASEs

RS0 (Null)	:
RS1 (Status)	:17

(8) Traffic Management Relating ASEs

T0 (Null)	:
T1 (Gap)	:18
T2 (Blocking)	:19
T3 (Gap, Blocking)	:18,19

(9) Call Information Relating ASEs

CI0 (Null)	:
CI1 (Call Info)	:20

(10) Generate Event Relating ASEs

G0 (Null)	:
G1 (Charging Info)	:21

(11) Service Data Relating ASEs

D0 (Null)	:
D1 (Data Access)	:23

(12) Cancel Relating ASEs

C0 (Null)	:
C1 (Cancel)	:24

(13) Activity Test Relating ASEs

A0 (Null)	:
A1 (Test)	:25

And the following are the potential AC names for ASE, for each physical entity.

(1) For SSP

(Sx, RCCx, R0, A/Hx, CHx, CEx, RSx, Tx, CIx, Gx, D0, Cx, Ax)

(2) For SCP/AD

(Sx, RCC0x, Rx, A/Hx, CHx, CEx, RSx, Tx, CIx, Gx, Dx, Cx, Ax)

NOTE – The AC name used by the same SCP/AD should be changed based on the type of remotely communicating PEs, except SN.

(3) For IP

(S0, RCC0, Rx, A/H0, CH0, CE0, RS0, T0, CI0, G0, D0, Cx, A0)

(4) For SN

(Sx, RCCx, R0, A/H0, CHx, CEx, RSx, Tx, CIx, Gx, Dx, Cx, Ax)

In the communication between the SSP and SCP, a control relationship can be invoked by both entities. There are two possibilities for assigning LegID. One is the case that LegID is assigned by both entities. In this case, LegID may consist of a sending side ID and a receiving side ID, with each locally assigned within each physical entity. The other case is that LegID is assigned by a single side entity. In this case, only one ID may be assigned by either of the entities. Therefore, choice of the sending SideID or receiving SideID is used. This is network operator specific, so that Recommendation Q.1218 specifies that one of the cases may optionally be selected.

6.5.3 Service Filtering

6.5.3.1 Use of Filtering Criteria

The current version of CS-1 Recommendation allows to perform service filtering using three mutually exclusive criteria, contained in the FilteringCriteria parameter:

- service key;
- dialled digits – This refers to the whole dialled number or only a part of this number (first digits);
- calling line identity.

The FilteringCriteria parameter is optional in the ActivateServiceFiltering (ASF) operation. Nevertheless, this parameter can be missing only when ASF operation is used within the context of a call. In this case, the implicit value is filtering on the Call-associated ServiceKey.

The use of the "dialled digits" parameters for service filtering is seen as the most common application. In such a case, service filtering can be used as follows.

The simplest way is to assign as filtering criteria a complete destination number and then count all calls to this destination number. The parameter "MaximumNumberofCounters" in the ASF operation has the value 1.

To apply filtering onto several destination numbers "dialled digits" has also to be provided as a complete destination number. The parameter "MaximumNumberofCounters" is set to n > 1; this parameter specifies the number of counters as well as the number of destination number for filtering. Service filtering is applied to the destination number specified by the "dialleddigits" parameter and to the following (n – 1) destination numbers. One can defines this way a simple televoting service.

Note that this is the only case where several counters can be used in service filtering.

6.5.3.2 Relations of service filtering operations with call context and TCAP transactions

The ActivateServiceFiltering (ASF) operation can be sent by SCF to SSF either within the context of a call or outside the context of a call. Within a single network, it is recommended to use only one of these two methods.

A given TCAP transaction (either within the context of a call or outside the context of a call) can only contain one ASF operation.

The ServiceFilteringResponse (SFR) can be sent by SSF to SCF either within the context of a call or outside the context of a call. In the first case, new incoming (not filtered) calls will be used to send SFR. For a given operator, it is recommended to use only one of these two methods, and the same method as for ASF.

When ASF and SFR are sent outside the context of any call, two solutions are possible with respect to the TCAP transactions either:

- use one single long TCAP transaction for the request ASF and all resulting responses SFR; or
- use several short TCAP transactions, one for the request ASF and one for each resulting response SFR.

Each solution has pros and cons: the first solution decreases the number of signalling message between SCF and SSF, but the second solution avoid to maintain a TCAP dialogue open for a long time between SCF and SSF, then decreases the memory occupation.

It is recommended that within a single network, only one solution would be used.

6.5.3.3 Acknowledgement of ASF

In some cases (for example before a big televote starts), it could be felt necessary to get an immediate acknowledgement by the SSF of the ASF request. As ASF is of class 2, this cannot be done by a TCAP result of this operation, but it can be done by a SFR operation with counters values = 0 immediately sent from SSF to SCF (in the same TCAP transaction where ASF was sent).

In the current CS-1 protocol, there is no way to indicate in the ASF invocation whether an immediate acknowledgement is required or not. Hence this can be achieved on an network dependent basis: within a single network one can choose to systematically acknowledge (or not) every ASF operation.

6.5.3.4 User Interaction

Interaction with A or B party

It is possible in CS-1 to have interaction (with the PlayAnnouncement or the PromptandCollect operation) with A or with B party. The choice of the A or B party is made by the LegId parameter in the ConnectToResource operation. Note that this ConnectoResource operation must always be executed before the PlayAnnouncement or the PromptandCollect operation.

However, it is not possible in the present CS-1 recommendation to have simultaneous interactions with A and B parties, for this would require:

– several ConnectToResource active at the same time;

– a new LegId parameter in the PlayAnnouncement and PromptandCollect operations, in order to determine the party concerned by the interaction (A or B).

These features are not presently supported by CS-1.

For example, it is not possible to play an announcement to A and at the same time to prompt B for authentication.

This point is for further study.

Error during the collection of digits in PromptandCollect operation

In PromptandCollect operation, the only possible error during the collection of digits correspond to "not enough digits received". This is detected at SRF level by the firstDigitTimeout parameter or the interDigitTimeout parameter. When such an error occurs, one the following treatment can be applied by SRF according to the value of the ErrorTreatment parameter: (stdErrorAndInfo): either send the received digits to SCF, or send the error ImproperCallerResponse to SCF. SCF will then decide itself on the subsequent actions (prompt again, release the call, ...). It seems preferable to send an error to avoid SCF to again check the number of digits received. (repeatPrompt): Repeat the prompting announcement from the beginning. (help): Play to the user a special help announcement. This announcement is the same for all services since the SRF does not know the precise service used.

Use of speech synthesis

In PlayAnnouncement and PromptandCollect operations, the MessageId parameter is used only for inband information. Therefore, when MessageId contains the subparameter "Text", which is a sequence of IA5String and attributes, IA5string must be converted into speech by SRF using speech synthesis. Attributes define the synthesis type (language, male or female voice, ...)

This speech synthesis feature can be useful to avoid the storage by SRF of a lot of predefined messages in vocal form (such as the messages used for indicating a telephone number).

6.5.4 Optional Parameters

At present Recommendation Q.1218 identifies a list of optional parameters associated with various Operations. Due to the optional nature of these parameters, various implementations may support all or some of these optional parameters. This raises the possibility of protocol messages from various implementations carrying varying number of optional parameters as a result of some trigger, at the same TDP.

For instance one implementation may send 1 – n optional parameters and another may send 1 to n – 2 optional parameters from the same TDP. This raises the possibility of inconsistent information being sent to the SCP.

In addition, if all the optional parameters are sent to the SCP as a result of a trigger, there could be an impact on the protocol message length. The sending of appropriate optional parameters can have a positive impact on the message length and the SS7 network. In order to do this, there is a need for specifying rules/conditions (as was done for Recommendation Q.931) under which certain optional parameters can be sent. At present CS-1 Recommendations do not specify any such rules and conditions and this could raise the following issues.

– Longer protocol messages;

– Inconsistent information from the SSP to the SCP from the same TDP.

Figure 8 depicts the Rules/Conditions for including appropriate parameters in messages.

6.5.5 Considerations for the use and understanding of various operations and procedures in Recommendation Q.1218

The following considerations are provided with respect to the use and understanding of various operations and parameters defined in Recommendation Q.1218:

a) RequestReportBCSMEvent operation – Note that though the response to RequestReportBCSMEvent can be either EventReportBCSM or one of the family of DP specific operations, it is up to the network operator to specify which applies in their network. This may cause an adverse impact on network interworking if this operation flows across network boundaries, though such network interworking is not within the scope of CS-1.

b) Connect operation – Note that when address information only is included in the Connect operation, call processing resumes at PIC 3 in the O-BCSM; when address information and routing information is included, call processing resumes at PIC 4.

c) DialledDigits parameter vs. CalledPartyNumber parameter – These two parameters are distinguished because

   i) dialled digits and called party number may be available at different DPs (e.g. dialled digits collected from the user at DP 2 and called party number resulting from the analysis of dialled digits at DP 3);

   ii) dialled digits and called party number may or may not be equivalent (e.g. dialled digits may be an abbreviated dialling string or freephone number and called party number is the translated number).

d) Clarification of INAP Procedures by SDL Diagrams

   SDL diagrams derived from the FSMs and associated text of Q.71 have been included in the CS-1 Maintenance Document. These are intended for clarification of INAP procedures and are still to be reviewed by Study Group 11.

e) The mechanism by which an instance of a SSF FSM is created requires identification (see 3.1.1.3/Q.1218). Since the SSF FSM itself handles DP processing (as currently described), an instance of the SSF FSM must be created whenever a DP is encountered in BCSM processing. This is an issue which requires further study.

f) In the description of state b, trigger processing, for the SSF FSM, one failure condition is "insufficient information to proceed", with subsequent call termination. To avoid ambiguity, failure on "insufficient information to proceed" should be treated the same way as "Trigger criteria match not found", with control returning to the CCF.

g) Clarification of Initiate Call Procedure

Explanation of the Problem and Solution

The Initiate Call Attempt Operation followed immediately by the Request Report BCSM Event Operation leaves the SCF Call State Model (SCSM) and the SSF Finite State Machine (SSF-FSM) in improper states for the handling of subsequent Event Detection Points (EDP). The following is a detailed explanation, with a solution which is entirely within the scope of CS-1.

Step 1 – The Initiate Call Attempt Operation is sent from the SCP to the SSP. The SCSM transits from State 1 (Idle) to State 2.1 (Prepare SSF Instruction). The SSF FSM transits from State a (Idle) to State c (Wait For Instruction). See Figure 8.

Also, upon receipt of the Initiate Call Attempt Operation by the SSP, a Basic Call State Model (BCSM) instance is created and call processing proceeds.

Step 2 – The Request Report BCSM Event Operation is sent from the SCP to the SSP. The SCSM remains in State 2.1, since this is a non-call processing instruction. The SSF-FSM also remains in State c. See Figure 9.

Problem – The SCSM is not in the proper state to handle a report of a DP being encountered, from the SSF. Also, the SSF-FSM is not even in the proper state to report to the SCF that the DP was encountered.

Solution – Service logic within the SCF could generate a Continue Operation at this time. The transmission of a Continue Operation will cause the SCSM to transit from State 2.1 to State 2.3 (Waiting for Notification or Report). Also, the receipt of a Continue Operation will cause the SSF-FSM to transit to State f (Monitoring). See Figure 10.

After the Continue Operation, both the SCSM and the SSF-FSM are in the correct states to handle the occurrence of the requested DP. Therefore, the following sequence of Operations could be used.

Initiate Call Attempt Operation
Request Report BCSM Event Operation
Continue Operation

Also, note that this sequence could be followed when the corresponding Information Flows are used to illustrate a service or service feature. In addition to Request_Report_BCSM_Event any other non-call processing operation can be processed, after Initiate_Call_Attempt and before Continue.

It may not be possible to request that DP 3 (Collected_Information) be armed while using this sequence of operations. This issue is FFS.

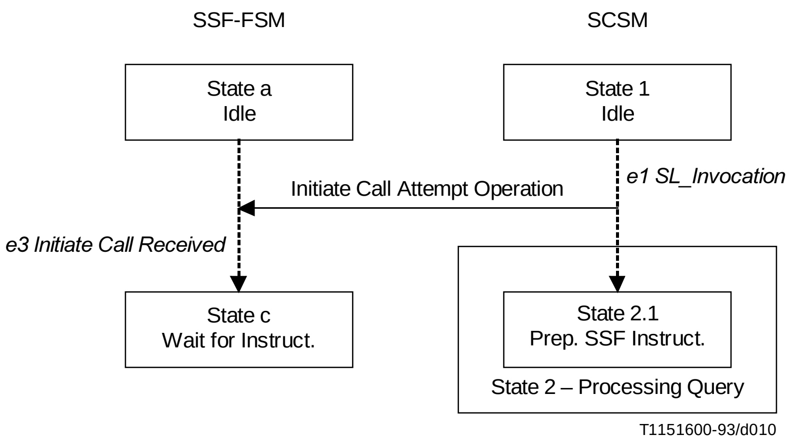


FIGURE 8/Q.1219

Step 2 – Initiate Call Attempt Operation

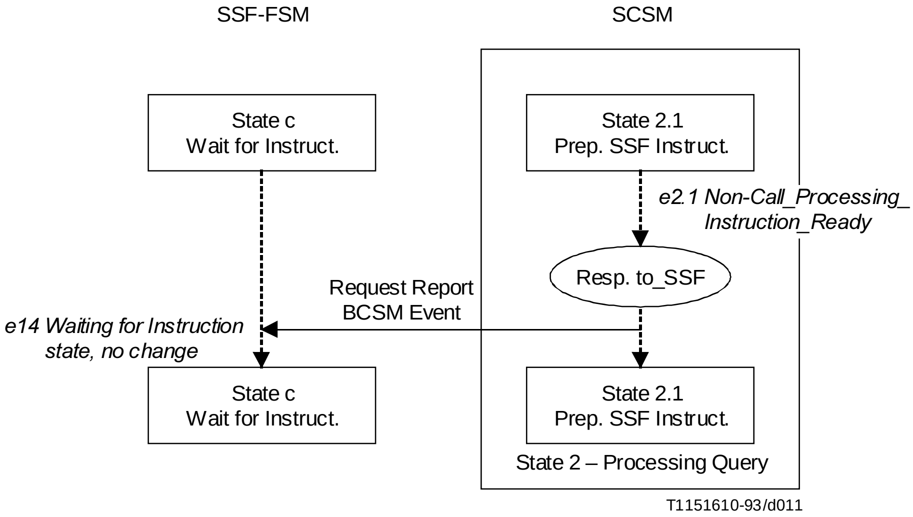


FIGURE 9/Q.1219

Step 2 – Event Report BCSM Event Operation

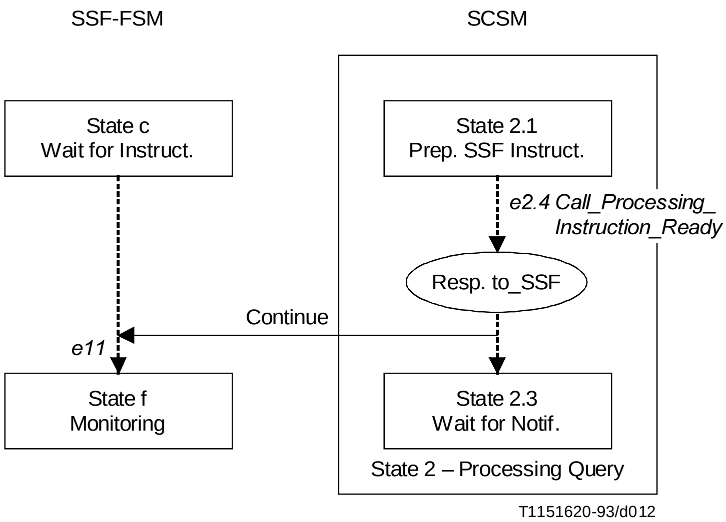


FIGURE 10/Q.1219

Solution – Continue Operation

6.6 Requirements on Inter-Exchange and User-Network Signalling

6.6.1 General

IN CS-1 is defined to minimize its impact on the existing inter-exchange signalling and user-network signalling protocols including the dedicated signalling systems being used in analogue networks. However, there exist some additional features required for them depending on the applications, in particular in the following aspects.

All CS-1 capabilities can be provided without modification to Recommendations Q.763, Q.931 and Q.932 (*White Book*) bearer signalling protocols.

1) Inter-exchange signalling or user-network signalling may have to transport some information (e.g. addressing information or digit information) to be used for detection of triggering conditions to initiate IN features.

2) Inter-exchange signalling may have to transport some information to initiate an appropriate action at the assisting SSP or hand-off SSP for user interaction. This typically includes the information to co-relate the transaction, to be established between the assisting/hand-off SSP and the initiating SCP, with the original transaction (the CalledPartyNumber parameter in Recommendation Q.763 may be used to transport information from the initiating SSP to the assisting/hand-off SSP).

3) Inter-exchange signalling or user-network signalling may have to transport some information from an SSP to an IP for initiation of an appropriate action at the IP for user interaction.

How to implement these features is network dependent and it is not specified in the IN CS-1 Recommendations.

6.6.2 Interworking INAP and Network or Access Signalling

CS-1 recommendations allow for interworking between INAP and network or access signalling on SSF. Nevertheless, not all parameters present in ISUP or DSS 1 message can be conveyed by INAP protocol. This means that these parameters are not available at SCF level in DP operations (SSF –> SCF), and that they cannot be set by the SCF in call proceeding and connect operations (SCF –> SSF).

Below is a (non-exhaustive) list of concerned parameters:

For ISUP:

ForwardCallIndicators;

OptionalForwardCallIndicator;

GenericNumber;

RedirectingNumber;

RedirectionInformation;

CugInterLockCode.

For DSS 1:

IndexCode;

BearerCapability.

NOTE – BearerCapability parameter is available in DPspecific operations, but not in InitialDP operations, neither in call proceeding and connect operations.

The absence of these parameters will entail some limitations on the service capabilities, in particular for ISBN subscribers. For example, for freephone service it would not be possible to send the freephone number dialled by the calling subscriber to the called subscriber by using the GenericNumber ISUP parameter

This point is for further study. However, if it is felt immediately necessary to introduce such parameters in the INAP protocol, it can be done within the present CS-1 recommendations by using the “extensions” capability which is available for all INAP operations.

6.6.3 Terminal Type and Access Type of User

It is interesting (in particular for interaction with the user) to know what the terminal type of the user is. The optional parameter "TerminalType" in DP operations can be used for this, with the possible specific values: dialPulse, dtmf, isdn. However, the following limitations apply:

- the type of the terminal is known only on local exchanges (originating or terminating exchange);
- even in a local exchange, in some cases it is not possible to know the real type of the terminal (for example for ISDN PABX with analogue lines).

If the type of terminal is not known, then the optional parameter "TerminalType" can be either absent or set to "unknown".

Another related information is the "access type" (e.g. analogue, digital, ISDN) of the user. This access type is always known by the local exchange. As this information is conveyed in the ISUP parameter ForwardCallIndicator, it may also be known at a transit exchange level.

However, in the present CS-1 recommendation there is no specific way to convey this information to SCF. This point is for further study.

6.6.4 Optional Parameters in DP Operations

A particular case occurs with the CallingLineIdentity or CallingPartyCategory Parameters: these parameters may not be available at the beginning of the call (e.g. when MF signalling is used), but the SSP may be able to get them by some additional signal exchanges on the incoming trunk.

It is recommended that at the SSF level it would be possible to decide whether these parameters are required or not before calling SCF. This will avoid to have systematically the additional signal exchanges, even if calling line information is not needed for the service.

For example, calling line information may be necessary for freephone, to allow SCF to compute the charging of the call, but may be useless for televoting.

6.6.5 Miscellaneous

The cancel operation applies to only one operation. Several cancel must be sent when several operations must be cancelled.

7 Service example

7.1 Utilizing CS-1 capabilities

An explanation, using the IN Conceptual Model as a guide, will help to provide a detailed view of service modelling in CS-1. The Service Plane of the INCM contains services and service features that are usable by the service users. The Global Functional Plane provides a high level view of service modelling in terms of the required network-wide functionalities. The fact that such modelling is closely analogous to the Stage 1 service description methodology for ISDN services aids in the understanding of this plane.

At the next level of complexity in the decomposition of services, as modelled on the GFP in terms of chains of (SIBs) and BCP POIs and PORs, the (DFP) representations of the GFP modular network capabilities necessary to support the services can be further distributed into the network. The functional entities, containing necessary functional entity actions, and the information flows between FEs can be presented in terms of the relationship to the services being modelled.

The physical entities and protocols of the Physical Plane on the INCM can be better understood after the progression of the explanation of service modelling as described in previous clauses. The physical network capabilities can, in this way, be shown to be directly related to the SIBs necessary to support the targeted services, and allow the flexible implementation of other new services, and allow the flexible implementation of other services which can be "constructed" from these modular capabilities.

An example of modelling a service using the CS-1 capabilities, i.e. the use of Service Independent Building Blocks (SIBs) and INAP, is shown in Annex A.

In IN CS-1, services are specified using service logic and information flows. Because of the possible complexity of services to be specified, as well as the relatively low level of information flow functionality, the service specification process can be difficult and error prone. Recommendation Q.1218 provides the protocol SACF/MACF rules and finite state machines to help clarify these issues.

One approach to manage this complexity is through the use of a modelling procedure using SIBs. Although the formal use of SIBs in IN CS-1 is restricted to determining and validating network capabilities, enough information is available in the SIB Stage 2 descriptions to make them useful in the service specification process. This can be accomplished by the following methodology:

1) Using the service description, model the necessary functionality using, when appropriate, sequences of SIB functionality and service specific functionality, as described in 7.2 below.

2) Identify and develop any areas which require additional functions or where an overlap exists.

3) Replace each SIB functional description with the sequence of information flows from its Stage 2 description and replace each service specific functional description with service logic.

4) Add, delete, or reorder information flows and service logic to achieve the desired service operation.

Structuring the service specification in this way can significantly reduce development time and make the entire process more stable. It can also help to identify additional SIBs which may be useful in modelling other services.

An additional source for an IN supported service, for an example, is Recommendation Q.76. This Recommendation provides a stage 2-like description of UPT Service Set 1, using IN CS-1 capabilities. A stage 1-like description of this service is provided in Recommendation F.851.

It is felt that this service example is highly valuable in that:

– It provides a fairly comprehensive example of usage of CS-1 capabilities, UPT being a complex service and including several features.

– It is aligned with the service description methodology provided by Recommendation Q.65.

– It shows interworking for service execution between 2 IN-structured networks, i.e. the "originating" and the "home" network.

7.2 Guidelines for service scenarios

The following guidelines apply to the service scenarios included in this Recommendation. The guidelines relate to the underlying scope of the User's Guide as stated in clause 1 of the Guide.

1) *Service scenario method*

   Service scenarios generally follow the 4 planes of the IN Conceptual Model (INCM), as described in 3.1/I.312/Q.1201.

2) *Service scenarios illustrate CS-1 capabilities*

   One purpose of the User's Guide is to serve as a means to explain and demonstrate how to use the capabilities defined for CS-1. The service scenarios are tools which serve this purpose.

3) *Service scenarios are not service specifications*

   Service scenarios are not intended as service specifications. That is, service scenarios do not describe all aspects of a service. Service scenarios are described to a level of detail such that users can form a common understanding of the capability to be demonstrated/explained. This includes a textual description of the service which provides an overview of the service from the user's point of view. In addition, the service scenarios represent the network's point of view by depicting the Basic Call State Model (BCSM), Information Flows, and Functional Entity Actions (FEA).

4) *Service scenarios are depicted graphically*

   Service scenarios are clear and concise. They provide a place in the Recommendation set where a user can acquire a quick understanding of the CS-1 capabilities. Graphical representations have the advantage of being rapidly and easily assimilated. Such representations are the basis of the service scenarios.

5) *Service scenarios do not imply a recommended implementation*

   In order to demonstrate a particular capability, implementation assumptions may need to be made (e.g. use of an IP in a remote SSP to demonstrate CS-1's handoff/assist capability). These assumptions are not meant to imply a recommended implementation of a specific service but rather to demonstrate the implementation flexibility inherent in the definition of CS-1.

6) *Only IN CS-1 related interfaces are completely depicted*

   The interfaces described in the service scenarios are generally limited to those identified for the IN CS-1. If the inclusion of other interfaces (e.g. between exchanges) is mandated to clarify some aspect of the service, then an appropriate general reference (e.g. Generic Signalling Network) is used.

7) *CS-1 Information Flows are illustrated*

   Service scenarios illustrate a variety of IN CS-1 Information Flows.

8) *A common format is used for all service scenarios*

   A common format allows the user to focus on understanding CS-1 concepts and capabilities. This format is described 7.3.

7.3 Format for Service Scenarios

The service scenarios included in this Recommendation are intended to promote a common understanding of CS-1 capabilities. They are not intended to be service specifications. The format of each service scenario consists of four items to reflect the four planes of the IN Conceptual Model (INCM) which consists of the Services/Features, Global Functional, Distributed Functional and Physical views.

1) *Capability statement*

   A short statement of the CS-1 capability that the scenario is describing is provided.

2) *Textual description of service*

   This item provides a short textual description of the service, from the user's perspective. It is included to provide an overview of the intended service. Where appropriate, the service and service feature descriptions of Recommendation Q.1211 Introduction to IN CS-1 are used.

3) *Global View*

This item provides a description of the service from the Network's perspective. At this level of abstraction, the Network is viewed as a single entity which is capable of performing the required functionality. The service is described in terms of chaining a small set of generic capabilities in the proper sequence. These generic capabilities are described as Service Independent Building Blocks (SIB) in Recommendation Q.1213 Global Functional Plane for IN CS-1. Sub-items included in the model are:

- *Service Independent Building Blocks (SIB)*, including:
  - Service Support Data (SSD);
  - Call Instance Data (CID).
- *Global Service Logic (GSL)*, including:
  - Basic Call Process (BCP);
  - Points of Initiation (POIs);
  - Points of Return (PORs)

4) *Distributed View*

This item provides a more detailed Network description of the service than in 2). The network is no longer viewed as a single entity but as discrete logical groupings of functionality, Functional Entities (FE). Recommendation Q.1214 Distributed Functional Plane for IN CS-1 describes the items at this level.

A) *Call Segment Diagram*

The pertinent connection segments for the scenario (originating or terminating) are graphically represented in relation to the call parties.

B) *Functional Entity Interfaces Diagram*

The Functional Entity interfaces used in the scenario are graphically represented, in relation to the call parties.

C) *Information Flow Sequence Diagram*

The Information Flow sequence diagram allows the interactions between Functional Entities to be captured. The items included in the diagram are:

- The Functional Entities;
- Basic Call State Model transitions;
- Detection Point instances;
- Information Flows;
- FEA reference numbers.

D) *List of Functional Entity Actions (FEA) Descriptions*

The description of the identified FEAs are included in this item. The FEAs describe the actions in a service independent manner. The FEA reference number is used as a cross reference into the Information Flow Sequence diagram, to determine the point in time when the FEA is invoked. This description method corresponds to the SIB Stage 2 descriptions of Recommendation Q.1214, which is the source for the reference numbers.

5) *Physical View*

This item provides the most detailed Network descriptions in the scenario. The physical locations, or Physical Entities (PE), for the FEs are identified. The protocol used between the PEs is also identified. The items in this section are described in Q.1218 Intelligent Network Interface Recommendation. Sub-items included are:

A) *Physical Entity Interface Diagram* consisting of:

- The Physical Entities
- Location of Functional Entities
- The interface protocol

B) *Time Sequence Diagram* consisting of:

– Operations

– Parameters

– Instances of TC_Begin

– Instances of TC_End

C) *Application Entity (AE) Procedures*

The procedures for each AE, within each FE, shall be itemized showing transition events and states.

8 Physical deployment scenarios

8.1 Mapping FEs to PEs

The mapping of IN functional entities to physical entities is based upon the following assumptions:

1) the IN Conceptual Model is used as a tool to develop the IN physical architecture;

2) existing and new technologies can be used to develop the physical entities;

3) the specification of FEs in the distributed functional plane and standard interfaces in the physical plane will make the network vendor independent and service independent;

4) for CS-1, a sufficient number of interfaces for support of services. Service creation and OAM functions will not be addressed.

This clause gives a mapping of Functional Entities (FEs) into Physical Entities (PEs) for CS-1, and describes the relationships between the PEs. In so doing, an appropriate distribution of functionality for CS-1 is identified, and functional interfaces suitable for standardization are highlighted. The PEs described in this clause are for illustrative purposes only, and do not imply the only possible mapping of functionality for CS-1.

This clause describes a flexible physical architecture made up of several PEs. Each PE contains one or more FEs, which define its IN functionality. PEs included in the physical architecture shown in Figure 1/Q.1215 are Service Switching Points (SSPs), Network Access Points (NAPs), Service Control Points (SCPs), Intelligent Peripherals (IPs), Adjuncts (ADs), Service Switching and Control Points (SSCPs), Service Data Points (SDPs), and Service Nodes (SNs).

Typical mapping of FEs to PEs are shown in Table 2.

NOTE – Table 2 is a typical mapping and does not address all of the PEs identified in Table 2.

There is no intention that Table 2 should disallow any other combination of FEs in a PE.

The above mappings are shown in Figure 1/Q.1215. The solid lines on the figure show transport paths that may exist between the PEs, and the dotted lines show signalling paths that can carry application layer messages for IN services.

8.2 Mapping of FE-FE relationships to PE-PE relationships

The FE-FE interfaces that fall within the scope of CS-1 are:

1) SSF-SCF;

2) SCF-SDF; and

3) SCF-SRF.

A mapping to the PE-PE interfaces is provided in Table 3.

The references to the PE-PE relationships correspond to the interfaces identified in Figure 1/Q.1215.

TABLE 2/Q.1219

Typical Scenarios of FE to PE Mapping

PEs	FEs			
	SCF	CCF/SSF	SDF	SRF
SCP	C	–	C	–
SN	C	C	C	C
AD	C	–	C	–
SSP	O	C	O	O
IP	–	–	–	C
SDP	–	–	C	–
SSCP	C	C	C	O
NAP	–	C (CCF only)	–	–
C Core O Optional – Not Allowed				

TABLE 3/Q.1219

Typical FE-FE – PE-PE Relationships

FE-FE →	PE-PE
SSF-SCF	SSP-SCP
	SSP-AD
	SSP-SN
SCF-SDF	SSP-SCP
	SCP-SDP
SCF-SRF	SCP-IP
	SCP-SSP-IP
	AD-IP

9 Future IN Capability Sets

9.1 Generic plans

IN CS-1 is the first standardized stage of the IN as an architectural concept for the creation and provision of telecommunication services.

The implementation of the IN architecture will facilitate the rapid introduction of new services. Its architecture can be applied to various types of telecommunications networks.

The ultimate IN is an evolving target, therefore, in order to take full advantage of the technological possibilities at a given point in time it is necessary to define phases, to define specific phases in the evolution to a target architecture. This approach is defined in the following subclauses.

9.2 "Stretch forward/ease back"

The specific concept applied to the evolutionary aspects of the IN are included in the "stretch forward/ease back" scenario. This scenario allows for a view of the long term IN aspects, i.e. stretching forward, and then "easing back" to a view of the existing network architecture. The next step is to decide the potential for the existing network architecture to evolve to the long term view. This approach allows for the maximization of existing resources while planning for future evolutionary stages of IN development.

9.3 Evolvable capabilities

IN CS-1 is the first of many planned sets of capabilities to provide increased service, network, and vendor independent capabilities for the provision of telecommunication and information services. Each set of capabilities is planned to enhance the prior set of capabilities, while still targeting the long term IN capabilities. Interworking and compatibility constraints will control the timing and content of future capability sets.

9.4 Evolvability concepts

The concepts which support the evolution of IN capability sets include a process called "harmonization". Harmonization is a process whereby several views of evolution are reconciled. A view is developed as to where IN should be in the long term time frame, i.e. what capabilities are envisioned. Then, a view is taken as to how this evolution can be accomplished using the existing network resources. These two, or more, views are then reconciled, or harmonized, to define what is achievable within the next planning and development cycle for the affected network resources.

The CS-1 architecture takes into account the evolution requirement, i.e. it supports the CS-1 targeted services but its functionalities are designed to evolve towards the future Capability Sets (CS-2 and beyond). Therefore, the CS-1 capabilities are defined without any assumptions that are known to limit their ability to evolve into future capability sets.

Annex A

IN CS-1 service scenario examples

(This annex forms an integral part of this Recommendation)

A.1 Introduction

To allow for a more complete understanding of the service scenarios provided in this annex, the following terms are defined:

A.1.1 travelling class mark: This is a feature that allows a customer in one business group of a Virtual Private Network (VPN) to have his service capabilities identified when he calls another business group location, via the "Travelling Class Mark". This capability allows the customer to have the same service features, no matter which location he calls to in the VPN.

A.1.2 extension parameters: These parameters are provided to allow the addition of new parameters to operations without having to change the protocol machine, but modifying the applications for CS-1.

A.1.3 use of the correlation id in the connect operation: This is to be used for the hand-off procedure, for networks able to carry the correlation ID in their SS7 signalling system to the next SSP.

A.1.4 connect To resource: Three possibilities are allowed in CS-1:

– *Usage of IPRoutingAddress alone* – When the IP is remote from the SSP and when there is only one party.

– *Usage of LegId alone* – When the IP is integrated with the SSP, for one or several parties.

– *Usage of both* – When the IP is remote, for one or several parties.

A.1.5 Service scenarios may be described using both generic operations and DP specific operations.

A.1.6 The service scenarios contained in this annex may include capabilities not explicitly covered by the procedures in Recommendation Q.1218, but which are supported in an implementation dependent manner in the SSP, while still remaining in line with clause 2/Q.1218.

The following examples are included in this annex:

- Example INAP operations and parameters;
- Example Automatic Alternative Billing Service;
- Example "UPT" Service;
- Example Call Forwarding Unconditional with Announcement Service;
- Example Service Assist Service;
- Example Malicious Call Identification Service.

A.2 INAP operations and parameters

The following "scenarios" are only examples on how some INAP operations and parameters may be used. They are not complete, in the sense that procedures, parameters, are not fully described. Their purpose is rather to focus on some specific aspects and show how INAP operations may be used to provide some specific services with some specific network configuration.

Legend:

→: The TCAP message is sent from the SSP to the SCP

←: The TCAP message is sent from the SCP to the SSP.

A.2.1 Translation aspects

The two following scenarios are some of the simplest in IN, with only two messages needed between the SSP and the SCP. Many major IN-provided services (such as freephone, credit card, premium charge, UPT or VPN) are based on variants of such scenarios.

Simple Translation handled by IN (Figure A.2-1)

- This first example also shows signalling messages used by the network. The other examples will not show these messages.
- Party A dials a number; if local exchange is not a SSP, call is routed to a transit exchange SSP.
- On receipt of the number, the SSP detects that triggering conditions are met at detection point "Collected_Info".

-- *A Party off hook; dial tone; dialling(dialled number); if the SSP is at transit level, an Initial address message is sent to the SSP; triggering occurs.*

→ 1-Initial DP(dialled number), TC_Begin (see Note 1)

-- *The SCP performs the Translation of the dialled number into the B number, by looking to its database or to a remote database (SDP).*

← 2-Connect(B number), TC_End (see Note 2)

-- *The SSP establishes the call to the B party, with an Initial address message.*

NOTE 1 – The Initial DP operation is put in the TCAP message from the SSP to the SCP when TC_Invoke is used. As this is to be done for each operation, this will not be mentioned in the other examples. The message is sent upon invocation of TC_Begin, which begins a "transaction" with the SCP, or a DP specific operation may be used (e.g. Collected Information).

NOTE 2 – In the same way, the Connect operation is put in the TCAP message with a TC_Invoke. The TC_End closes the transaction between the SCP and the SSP. Then the SSP becomes a normal switch. The Connect is usually a class 2 operation (that is, some error report is reported to the SCP if the execution cannot be done successfully). But in this case, the Connect operation is used as a class 4 operation (no error from the SSP is expected). If the SCP wants to be aware of errors, the scenario would look like (see Figure A.2-2):

← 2-Connect(B number), TC_Continue
-- *The SCP is informed of the expiration of the TCAP timer of the Connect operation (through a TC_Cancel indication).*
← 3-TC_End

Simple Translation with specific announcement to A party (see Figure A.2-1)

A specific announcement (such as "you are being connected", some customized announcement, special tone) is sent to Party A, while the network is establishing the call.

→ 1-Initial DP, TC_Begin
← 2-Connect To Resource, Play Announcement, Connect(B number), TC_End (see Note 3)

NOTE 3 – By default, the Connect To Resource is to the A party. As a general rule, the first time a Play Announcement is used, there must be a Connect To Resource before. In this specific example, the announcement duration has a typical value of several minutes (this is to cater for international call set up time). At the same time, the Connect operation is executed immediately. Announcement interruption and disconnection of the resource will occur upon through-connection to the B party. Therefore, the announcement duration will depend on the call set-up time. Switchpath from B to A party is established by the SSP when the B party is in alerting phase, that is when the appropriate signal message from the B side (for instance, the address complete) is received by the SSP. This example is not supported by the procedures described in clause 3/Q.1218.

A.2.2 Freephone – Simple translation and specific charging

Several implementations are possible, depending on network-specific mechanisms for charging.

A.2.2.1 SSP with charging capability

Freephone – The SSP has online charging capability (such as counters) (see Figure A.2-1)

→ 1-Initial DP (freephone number), TC_Begin
← 2-Apply Charging(To B Party), Connect(B number), TC_End (see Note 1)

NOTE 1 – Connect and Apply Charging are within the same TCAP message (sequence of components) because they must be synchronized. The SSP will start charging according to the Apply Charging parameters (for instance, when an Answer message from B is seen by the switch). The Apply Charging could be sent after the Connect.

Freephone – The SSP has offline charging capability (based on collected information) (see Figure A.2-1)

→ 1-Initial DP (freephone number), TC_Begin.
← 2-Furnish Charging information(B Party), Connect(B number), TC_End (see Note 2)

NOTE 2 – The Furnish Charging Information operation may be invoked after the Connect operation.

A.2.2.2 SSP with charging capability but no storage facility

The SSP is able to compute (or to collect relevant information) the charge for the call, but it cannot store the information.

Freephone – If the SSP has online charging capability (such as counters) (see Figure A.2-3)

→ 1-Initial DP (freephone number), TC_Begin
← 2-ApplyCharging(To B Party, Info to be sent to SCP), Connect(B number), TC_Continue
→ 3-ApplyCharging Report(charge), TC_Continue (see Note 1)
← 4-TC_End (see Note 2)

NOTE 1 – This is sent at the end of the call.

NOTE 2 – This message does not have any operation. It indicates to the SSP that the transaction is closed and that no further operation is to be sent by the SCP to the SSP. Except for some exceptional situations (such as the SCP has failed), the SSP never takes the initiative to end the transaction as it is not able to know what the service logic is.

Freephone – If the SSP has offline charging capability (based on collected information) (see Figure A.2-3)

→ 1-Initial DP (freephone number), TC_Begin
← 2-Furnish Charging Information(B Party), Connect(B number), Call Information Request (see Note 5), TC_Continue (see Note 3)
→ 3-Call Information Report, TC_Continue (see Note 4)
← 4-TC_End

NOTE 3 – Here, although we have three operations within the same TCAP message, only the first two must be synchronized. The third one may be executed after as it is independent of call processing.

NOTE 4 – This is sent at the end of the call.

NOTE 5 – In this example, the Call Information Request contains collected charging information.

A.2.2.3 SSP without charging capability

Freephone – Charge is computed by the SCP (see Figure A.2-4)

→ 1-Initial DP (freephone number), TC_Begin
← 2-Connect (B number), Request Report BCSM Event (O_Answer-EDP/N, O_Disconnect-EDP/N), TC_Continue (see Note 1)
-- *The SSP forward the call to B, with an IAM message.*
-- *The SSP receives an Answer message from the B party.*
→ 3-Event Report BCSM (O_Answer), TC_Continue (see Note 2)
-- *The SSP receives a Release message from the A or B party.* (See Note 3)
→ 4-Event Report BCSM (O_Disconnect), TC_Continue
← 5-TC_End

NOTE 1 – It is possible to request notification of several BCSM events in one operation. The Request Report BCSM operation is synchronized with the Connect operation (the SSP has to start monitoring once the connect is executed). The Answer and Disconnect are requested on the Notify And Continue mode, in the Monitor Mode parameter.

NOTE 2 – For instance, as an operator specific choice, there would be a timestamp parameter in the Event Report BCSM (in the Event Specific Information BCSM, the coding of which is defined by a network operator).

NOTE 3 – If the release comes from the A party, event notification is immediately done. If the release comes from the B party, there are several cases to consider, depending on the monitor mode (interrupt, or notify and continue) and on the signalling system. In the present case (notify and continue), if the signalling system is ISUP, and unless there is some interworking with PSTN, the release from the B party is a confirmed one. Then, the event is notified to the SCP. In other signalling systems , a new answer message may be received from B. In this case, the SSP must arm the new-answer timer and notify the release upon expiration of the timer or upon receipt of a release from party A. In the interrupt case, what is notified to the SCP is the first release message from B (confirmed or not).

Freephone – Charge is computed by the SCP from tariff messages received from the network (see Figure A.2-5)

This example assumes that the network handles tariff messages in its signalling system (such capability is network operator specific and described in Annex A of CCITT Recommendation on TUP).

→ 1-Initial DP (freephone number), TC_Begin

← 2-Connect(B number), Request Report BCSM Event(O_Answer, O_Disconnect), Request Notification Charging Event (tariff message), TC_Continue (see Note 4)

-- *The SSP forwards the call to B, with an IAM message.*

-- *The SSP receives a tariff message from the network (B party side).*

→ 3-Event Notification charging (tariff message), TC_Continue

← 4-Send Charging Information (Ack to B party side), TC_Continue (see Note 5)

-- *The SSP sends an ACK to the tariff message.*

-- *Continue from here with message 3 of the case "Freephone – Charge is computed by the SCP".*

NOTE 4 – The Request Report BCSM Event and the Request Notification Charging Event operations are synchronized with the Connect operation. The Answer and Disconnect are requested on the Notify And Continue mode, in the MonitorMode parameter. A tariff message is monitored by the request notification of charging events. The tariff message must be monitored in Interrupt mode.

NOTE 5 – As the tariff message from the B side of the network is monitored in interrupt mode, the message will not be forwarded to the A side of the network. As tariff messages are acknowledged (they represent money), the SCP must be able to request this acknowledgement with the Send Charging Information operation.

A.2.2.4 SSP with charging capabilities with SCP interaction

Freephone – SCP is not performing charging, but must change the tariff (see Figure A.2-6)

This example assumes that the network handles tariff messages in its signalling system . The example below is for the case when the tariff is being changed at the beginning of the call, but it could be done during the call if a tariff message is sent during conversation phase (not shown in scenarios).

→ 1-Initial DP (freephone number), TC_Begin

← 2-Connect, Request Notification Charging Event (tariff msg), TC_Continue (see Note)

-- *The SSP sets up the call to the B party (IAM is sent).*

-- *The SSP receives a tariff message from the B party side.*

→ 3-Event Notification Charging(tariff msg), TC_Continue

← 4-Send Charging Info(modified tariff msg, to A party), TC_Continue or TC_End

-- *The SSP sends the modified tariff msg to the A party.*

...

NOTE – Connect and Request Notification Charging Event are synchronized. The tariff message is monitored in the interrupted mode.

A.2.5 User interaction aspects

A.2.5.1 User interaction with A party and Translation

In band user interaction with A party is used to get the destination number to be translated. It is assumed that the SSP has an integrated IP.

User interaction with A party with translation – A has to dial a number first (see Figure A.2-7)

- Party A dials a number; if local exchange is not a SSP, call is routed to a transit exchange SSP.
- On receipt of the number, the SSP detects that triggering conditions are met at detection point "Collect_Info".

→ 1-Initial DP(dialled number), TC_Begin
← 2-Connect To Resource, Prompt and Collect User Information, TC_Continue (see Note 1)
→ 3-Specialized Resource Report (Number), TC_Continue
← 4-Connect(B number), TC_End (see Note 2)

NOTE 1 – For prompting and collecting, the SSP must have a complete switchpath to the A party in both backward and forward directions. In PSTN, if the SSP is a transit exchange, this would usually require that an ACM message must have been sent before prompting; this would depend on how a network handles switch path completion, which is network dependent. In ISDN, if the A party is an ISDN user, an answer message is also necessary to have speech path [in Recommendation Q.764, for successful ordinary call (*en bloc* operation), backward switchpath is completed at setup while forward switchpath is at answer. This asymmetrical bidirectional handling of switchpath was designed for preventing terminals sending data on a 64 kbit channel before charging starts].

NOTE 2 – If the connect is for an ISDN call, as there is complete bidirectionnal switch path from A to the SSP, the SSP must implement an asymmetrical switchpath completion from the SSP to the B party (that is backward switch path upon receipt of the ACM, forward switch path upon receipt of the ANM). This is to ensure that from A to B, there is an asymmetrical switchpath completion to prevent fraud in ISDN calls handled by IN.

User interaction with an ISDN A party (see Figure A.2-1)

This scenario is the same as the previous one, except that A receives in band announcement and ISDN information.

→ 1-Initial DP(dialled number), TC_Begin
← 2-Connect To Resource, Play Announcement(ISDN info), Play Announcement(In band info), TC_Continue (see Note 3)
...

NOTE 3 – Two consecutive Play Announcements are sent, the first one sending the display information to the ISDN terminal. Since the execution of the display information by the SSP should take a limited amount of time, the in-band information will be immediately sent after to the end-user by the SSP, in sequence with the display information.

User interaction with A party with translation: A just goes off hook (see Figure A.2-7)

- Party A goes off hook; the local exchange must be an SSP.
- On receipt of the number, the SSP detects that triggering conditions are met at detection point "Origination_Attempt_Authorized".

→ 1-Initial DP, TC_Begin (see Note 4)
← 2- ... (see Note 5)

NOTE 4 – The Initial DP has no called party number parameter.

NOTE 5 – The rest as in the first example.

Credit card calling from a credit card payphone (see Figure A.2-8)

Here detection point is at off hook. Thus, we assume that the SSP is a local exchange. The SSP has an integrated SRF.

→ 1-Initial DP, TC_Begin
← 2-Connect To Resource, Prompt And Collect User Information(Disconnect from IP forbidden), TC_Continue (see Note 6)
→ 3-Specialized Resource Report (Credit card Number, PIN Code), TC_Continue
-- *Authentication is performed, with a query to an SDP.*
← 4-Prompt And Collect User Information(Disconnect from IP allowed), TC_Continue
→ 5-Specialized Resource Report (B Number), TC_Continue
← 6-Connect(B number), TC_End

NOTE 6 – The IP is not allowed to disconnect as a subsequent usage of the resource will be requested with a second prompt and collect.

In-band user interaction with a called party (see Figure A.2-9)

This is an example of "out of the blue" call, for sending some information to the called party. This could be used for a wake up call. It is assumed that the SSP has the SRF integrated. The SSP may be any network exchange, but the SCP may optimize circuit usage by choosing an SSP close to the called party.

There may be a performance problem which causes a long interval between the called party's off-hook and the sending of an announcement to the called party. Therefore, Connect To Resource and Play Announcement should be sent with Initiate Call Attempt within the same TCAP message, and the SSP needs to queue them.

← 1-Initiate Call Attempt (B number), Connect To Resource (B Party), Play Announcement, TC_Begin (see Note 7)

NOTE 7 – When the SSP receives this message, it needs to queue Connect To Resource and Play Announcement because it cannot process them during processing Initiate Call Attempt. This point is consistent with 3.1.1/Q.1218.

NOTE 8 – The Play Announcement could be replaced by a Prompt And Collect. In this latter case, the transaction cannot be closed.

NOTE 9 – In this scenario, it does not matter whether the B party is an originating side (i.e. a controlling leg) or a terminating side (i.e. a passive leg) for O_BCSM. However, if the service logic arms an EDP-R at O_Mid_Call for the B party, for example, then the B party must be an originating side.

This is an example of "out of the blue" call (when there are no performance problems), for sending some information to the called party. This could be used for a wake up call. It is assumed that the SSP has the SRF integrated. The SSP may be any network exchange. The SCP may optimize circuit usage by choosing an SSP close to the called party.

← 1-Initiate Call Attempt (B number), Request Report BCSM event (O_Answer from B), TC_Begin (see Note 10)
→ 2-Event Report BCSM (O_Answer from B), TC_Continue
← 3-Connect To Resource (B party), Play Announcement, TC_End (see Note 11)

NOTE 10 – The Initiate Call Attempt and Request Report BCSM Event must be synchronized by the SSP, as they are sent within the same TCAP message. An Initiate Call Attempt must always be followed by a Request Report event (either BCSM or charging).

NOTE 11 – The Play Announcement could be replaced by a Prompt And Collect. In this latter case, the transaction cannot be closed.

Authentication of the called party (see Figure A.2-10)

This capability is not supported by clause 2/Q.1218. This is an example of interaction with the called party (which is needed for UPT for instance for authenticating the B party, or for services such as automatic alternate billing). It is assumed that the SSP has the SRF integrated. Charging aspects are not treated here.

→ 1-Initial DP (UPT number), TC_Begin

-- *The SCP has to access to some database in order to be able to translate the UPT number into a B party number.*

← 2-Connect To Resource, Play Announcement, Initiate Call Attempt(B number), Request Report BCSM Event(Answer), TC_Continue (see Note 12)

→ 3-Event Report BCSM (O_Answer from B), TC_Continue

← 4-Connect To Resource(to B party), Prompt And Collect (PIN code), TC_Continue

→ 5-Specialized Resource Report (PIN Code), TC_Continue

-- *Successful authentication.*

← 6-Reconnect, TC_End (see Note 13).

NOTE 12 – The calling party receives some waiting announcement while the B party is authenticated. The announcement to the A party is done by the Connect To Resource and Play Announcement operations. At the same time, the switch initiates the call to the B party. Such capability is not described by the SSF-FSM of Recommendation Q.1218.

NOTE 13 – The SSP performs through connection to allow conversation between the A and B party. The Reconnect operation is described in the Appendix I/Q.1218.

A.2.6 Traffic aspects

Televoting or mass calling services (see Figure 9)

As an example, it was assumed that service filtering was to be reported every Nth call and that it was based on some dialled number). Service filtering may be invoked outside of a call context (as shown in the following example), or within a call context.

← 0-Service Filtering (Starttime, Stoptime, every n call,) TC_Begin (see Note 1)

→ 1-Service Filtering response, TC_Continue (see Note 2)

← 2-TC_End

→ 3-Initial DP(dialled number), Service Filtering Response, TC_Begin (see Note 3)

← 4-Connect To Resource, Play Announcement, Release Call, TC_End (see Note 4)

3-4 may be repeated a number of times

→ 5-Service Filtering Response, TC_Begin(see Note 5)

← 6-TC_End

NOTE 1 – This Service Filtering is sent out of any call context. The stop time may occur although the number of calls has not exceeded n.

NOTE 2 – The Service Filtering Response contains counters equal to zero. This message is used as an acknowledgement to the SCP.

NOTE 3 – n – 1 calls have been filtered. The Service Filtering Response will contain some counter values.

NOTE 4 – 3-4 sequence will occur a number of times until the stop time defined in message number 0. In this example, as both the Play Announcement and the release call operations are related to the A party, the release call operation is executed after the end of the Play Announcement, that is when the SRF decides to disconnect the connection from the IP to the SSP. If the A party decides to disconnect before the end of the announcement, then the Release Call operation will not need to be executed.

NOTE 5 – This TCAP message is independent of any call and is used for reporting counter values when the filtering stoptime occurs.

Call gapping (see Figure A.2-1)

Call Gap is used for reducing the number of requests to the SCP for specific type of calls (such as calls towards one freephone number) for a specified amount of time. The following scenario is one example of usage of the Call Gap operation, within the context of one call, for a freephone call.

→ 1-Initial DP (freephone number), TC_Begin
← 2-Connect, Call gap (freephone number), TC_End (see Note 6)

NOTE 6 – Gapping will not be applied to this call, but to all subsequent calls coming from the **same SSP** and for the same freephone number.

Call queuing (see Figure A.2-11)

In this example, there are several calls to the same service, where only a limited number can be served at once. Other calls have to wait in a queue. The first call is an example of a call that passes directly and gets connected. The second call is an example of a call that is first connected to an announcement machine (put in a queue in the SCP). We assume that the SSP has an integrated SRF. Messages described hereafter are related to two TCAP transactions (first transaction: 1, 2, 5, 6; second transaction: 3, 4, 7, 8 and 9). In this example, it was assumed that all the calls to be queued are handled by a single SCP.

→ 1-Initial DP, TC_Begin
← 2-Connect, Request Report BCSM Event (O_Disconnect), TC_Continue
→ 3-Initial DP, TC_Begin

← 4-Connect To Resource, Play Announcement, TC_Continue (see Note 7)

→ 5-EventReportBCSM(O_Disconnect), TC_Continue
← 6-TC_End

← 7-Disconnect Forward Connection , Connect, Request Report BCSM Event(O_Disconnect), TC_Continue (see Note 8)
→ 8-EventReportBCSM(O_Disconnect), TC_Continue
← 9-TC_End
... (see Note 9)

NOTE 7 – The Play Announcement means that the SSP has to ensure the speech path from the switch to the calling party (this may already be the case if the SSP has sent an answer message to the calling party). If the announcement is charged to the A party, a charging-related operation should be added (Apply Charging or Furnish Charging Information).

NOTE 8 – 6 and 7 may be executed in parallel.

NOTE 9 – Dequeue another call (if there is one).

The next example is the case where the second call is an example of a call that is not connected to an announcement.

→ 1-Initial DP, TC_Begin

← 2-Connect, Request Report BCSM Event (O_Disconnect), TC_Continue

→ 3-Initial DP, TC_Begin

← 4-Reset Timer (SSF interoperation timer), Hold Call In Network, TC_Continue (see Note 10)

→ 5-Event Report BCSM (O_Disconnect), TC_Continue

← 6-TC_End

← 7-Connect, Request Report BCSM Event (O_Disconnect), TC_Continue (see Note 11)

→ 8-Event Report

BCSM (O_Disconnect), TC_Continue

← 9-TC_End

.............(see Note 12)

NOTE 10 – The reset timer is necessary, as the call is going to be queued and the next instruction will come from the SCF when this latter decides when to dequeue the call. The new value of the interoperation timer must be greater than Tscf2 (see section on time handling aspects). Upon receipt of the Hold Call In Network, if the SSP is a local exchange, it also resets the answer timer. If the SSP is a transit exchange, it has to send an answer message to the local exchange.

NOTE 11 – 6 and 7 may be executed in parallel.

NOTE 12 – Dequeue another call (if there is one).

A.2.7 Time handling aspects

The SCP is slow to respond, or has failed, while the SSF is waiting for instructions (see Fig A.2-12)

→ 1-Initial DP, TC_Begin

-- *Call is suspended at the SSP, which is waiting for instructions. Tssf1 is armed (typical value: a few seconds).*

-- *Tssf1 expires. Then, the SSP performs BCSM-state dependent actions and establish the call if possible. Otherwise it acts as if the call was congested with an SSF-FSM transition to the IDLE state.*

The SCP fails after instructions were sent to the SSF (see Figure A.2-13)

→ 1-Initial DP, TC_Begin

-- *Call is suspended at the SSP, which is waiting for instructions. Tssf1 is armed.*

← 2-any operations with connection, TC_Continue (see Note 1)

-- *Tssf1 is disarmed. Tssf2 is armed (typical value: 1000 seconds).* (See Note 2)

-- *Tssf2 expires. Then, the SSP performs BCSM-state dependent actions as if the call was to be released, with an SSF-FSM transition to the IDLE state.*

NOTE 1 – Connect, Connect To Resource, operations are examples of acceptable operations.

NOTE 2 – This assumes that conversation phase may last 1000 seconds, without any instructions from the SCP.

The SSP fails during a call (see Figure A.2-14)

→ 1-Initial DP, TC_Begin
← 2-any operations with connection, TC_Continue
-- *as the previous TCAP message is sent by the SCP, Tscf2 is armed (for instance 500 seconds, less than Tssf2).*
-- *When the SSP receives the previous message, Tssf2 is armed (typical value: 1000 seconds).*
-- *SSP fails and restarts later.* (See Note 3)
-- *Tscf2 expires.*
← 3-Activity Test (see Note 4)
→ 4-TC_Reject (see Note 5)
-- *The SCP must clear the SLP instance.*

NOTE 3 – Then, the transaction does not exist any longer in the SSP.

NOTE 4 – Tssf2 is normally to be rearmed upon receipt of the Activity Test.

NOTE 5 – Upon receipt of the activity test from the SCP, as the transaction does not exist any longer, the SSP rejects the operation.

While the SSP is waiting some instructions, the SCP is waiting for some result (such as a query to a database) that does not come (see Figure A.2-15)

→ 1-Initial DP, TC_Begin
-- *the SSP arms Tssf1 (for instance 5 seconds).*
-- *upon receipt of the TCAP message, the SCP arms Tscf1 (typical value, a few seconds, for instance 3 seconds, less than Tssf1).*
--*Tscf1 expires. The SCP rearms Tscf1.*
← 2-Reset Timer(Tssf1) (see Note 6)
--*The SSP rearms Tssf1.*
--*Tscf1 expires. Local abort in the SCP.*
--*Tssf1 expires (after Tscf1). Local abort.*

NOTE 6 – This message is sent when Tscf1 is rearmed. The SCP asks for some more delay in the SSP.

A.2.8 Non-call processing aspects

Simple translation and call information request

Some information is requested by the SCP, for charging or statistical purposes.

→ 1-Initial DP, TC_Begin
← 2-Connect, Call Information Request (correlation Id), TC_End (see Note 1)

→ 3-Call Information Report (correlation Id), TC_Begin
← 4-TC_End

NOTE 1 – The SCP has decided to close the transaction. This means that the report is to be sent by the SSP in a new TCAP transaction. The correlation Id is **network operator specific.**

Continuous resource status monitoring (see Figure A.2-16)

A physical resource is monitored continuously (such as a line status). The following shows one scenario where the request is sent in one call context, and the reports are sent in another transaction. The duration parameter must then be used. If reporting must occur only within a transaction, the duration parameter would not be used.

→ 1-Initial DPs, TC_Begin
← 2-Connect, Request Every Status Change (resource Id, Duration Report), TC_End
→ 3-Status Report (resource Id), TC_Begin
← 4-TC_Continue (see Note 2)
→ 5-Status Report (resource Id), TC_Continue
→ 6-Status Report (resource Id), TC_Continue
→ --
→ n-TC_End (see Note 3)

NOTE 2 – The empty TC_Continue is necessary to establish the transaction to the SSP.

NOTE 3 – When the monitor duration expires, the transaction is closed by the SSP. **This is the only case where the SSP may close a transaction**.

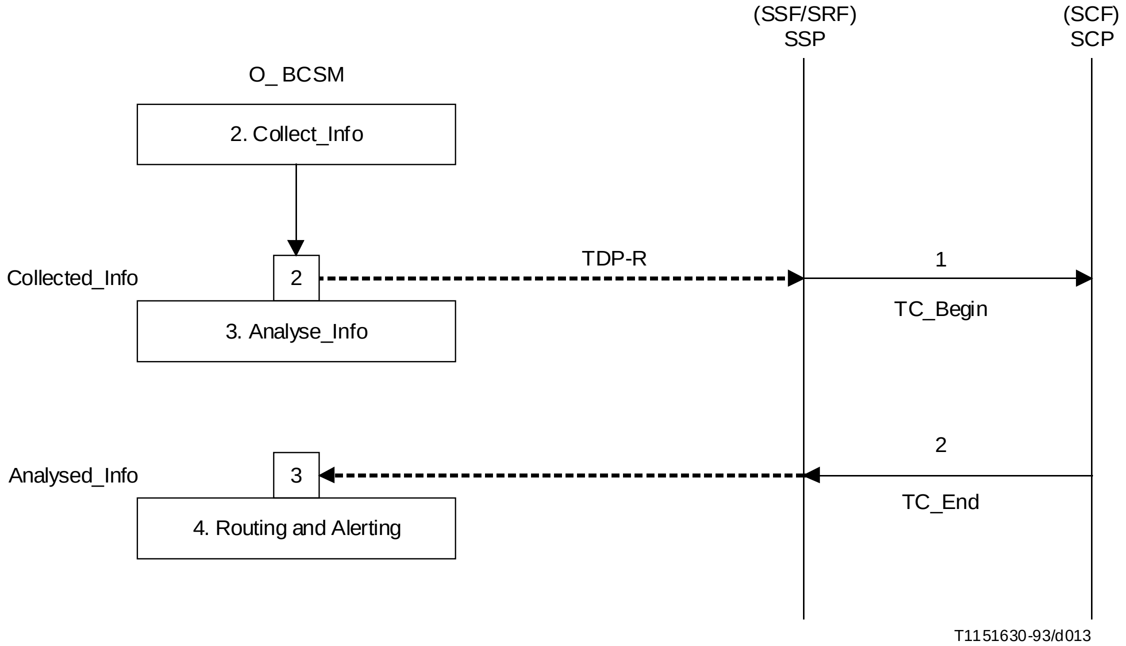


FIGURE A.2-1/Q.1219

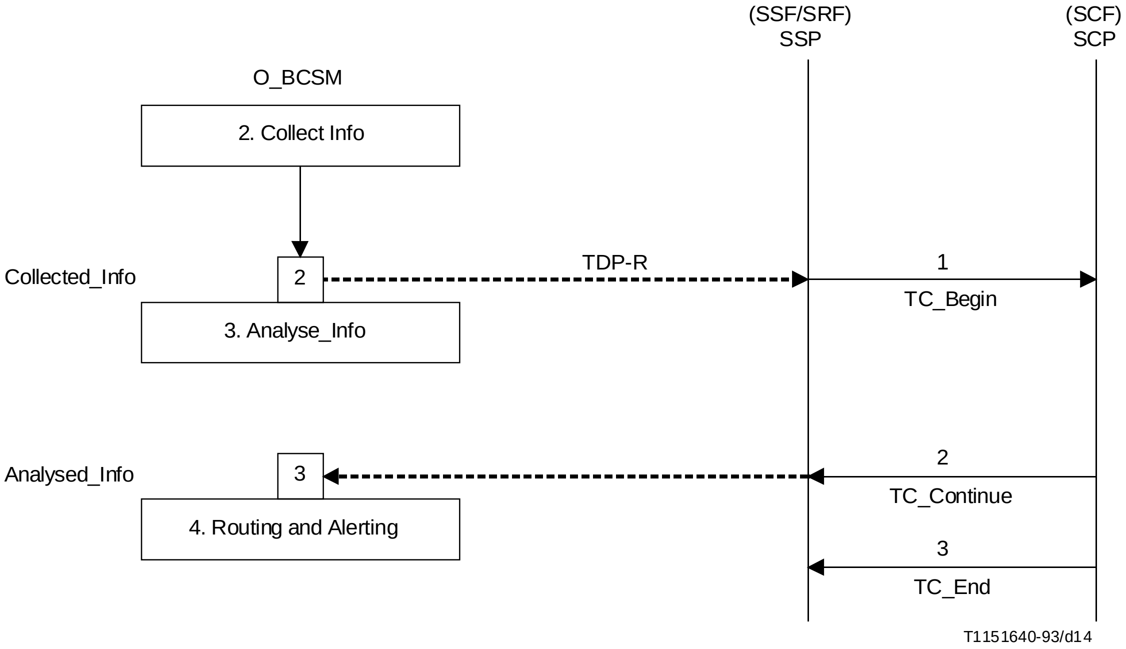


FIGURE A.2-2/Q.1219

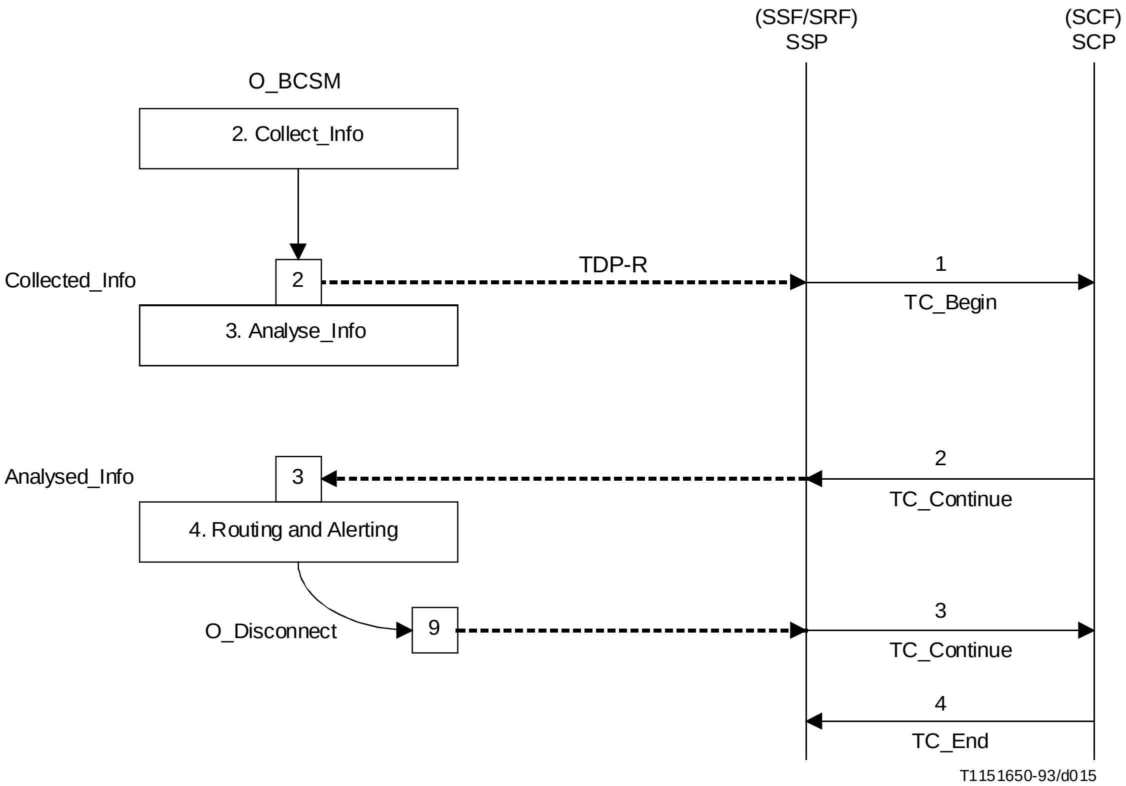


FIGURE A.2-3/Q.1219

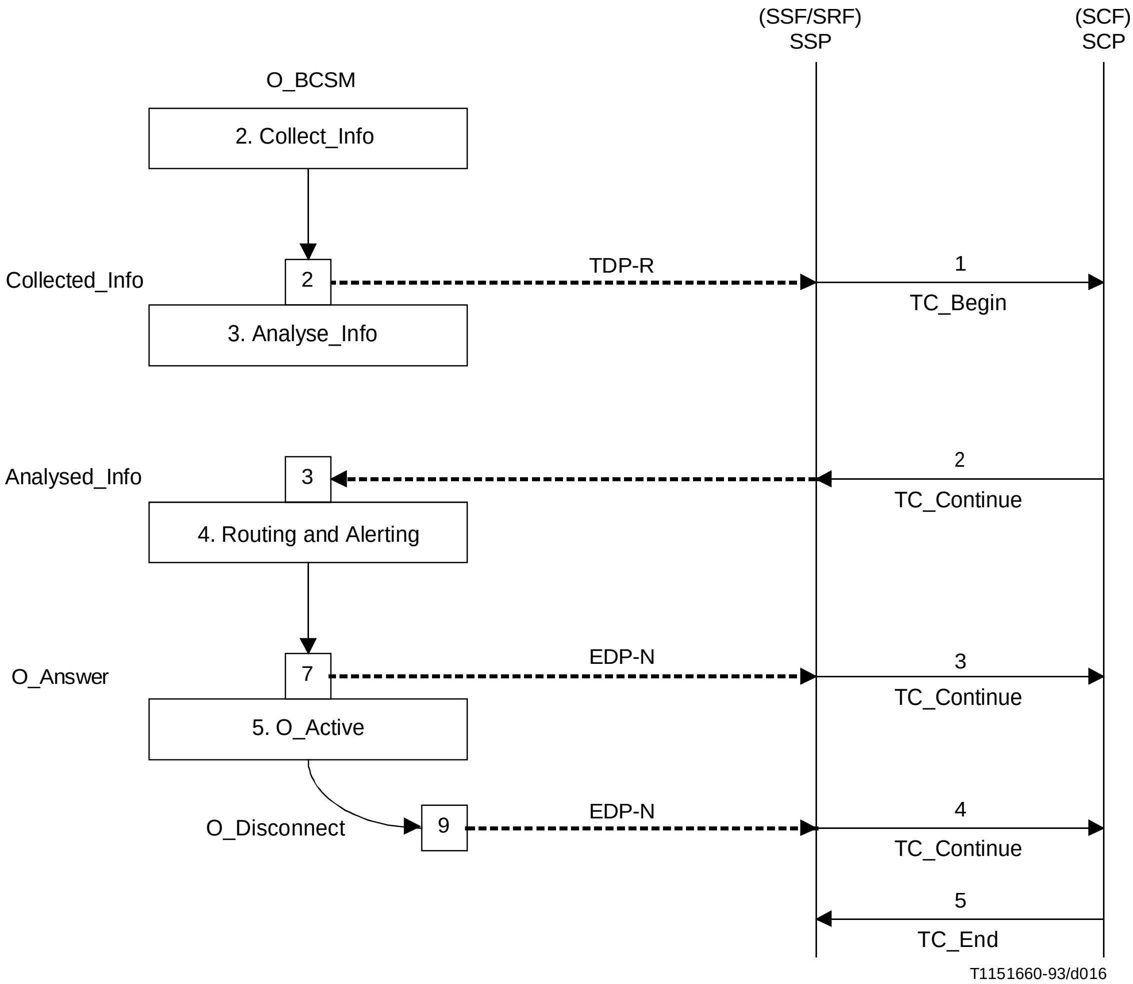


FIGURE A.2-4/Q.1219

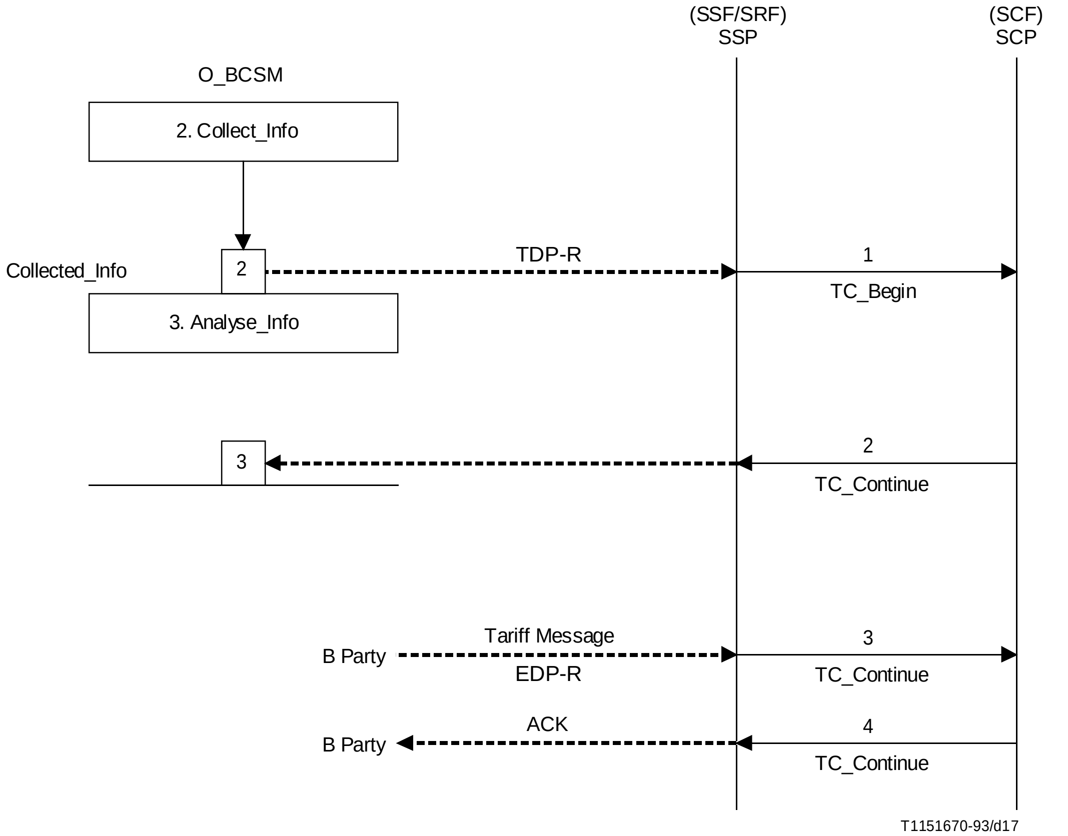


FIGURE A.2-5/Q.1219

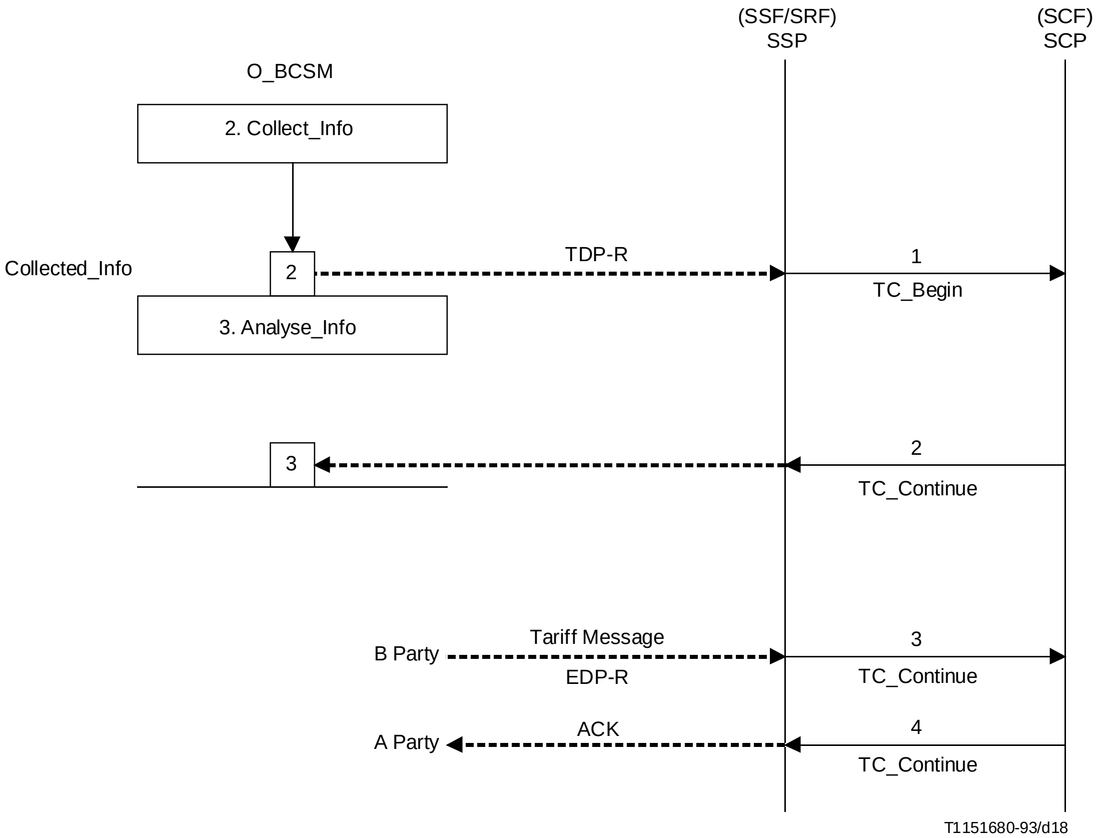
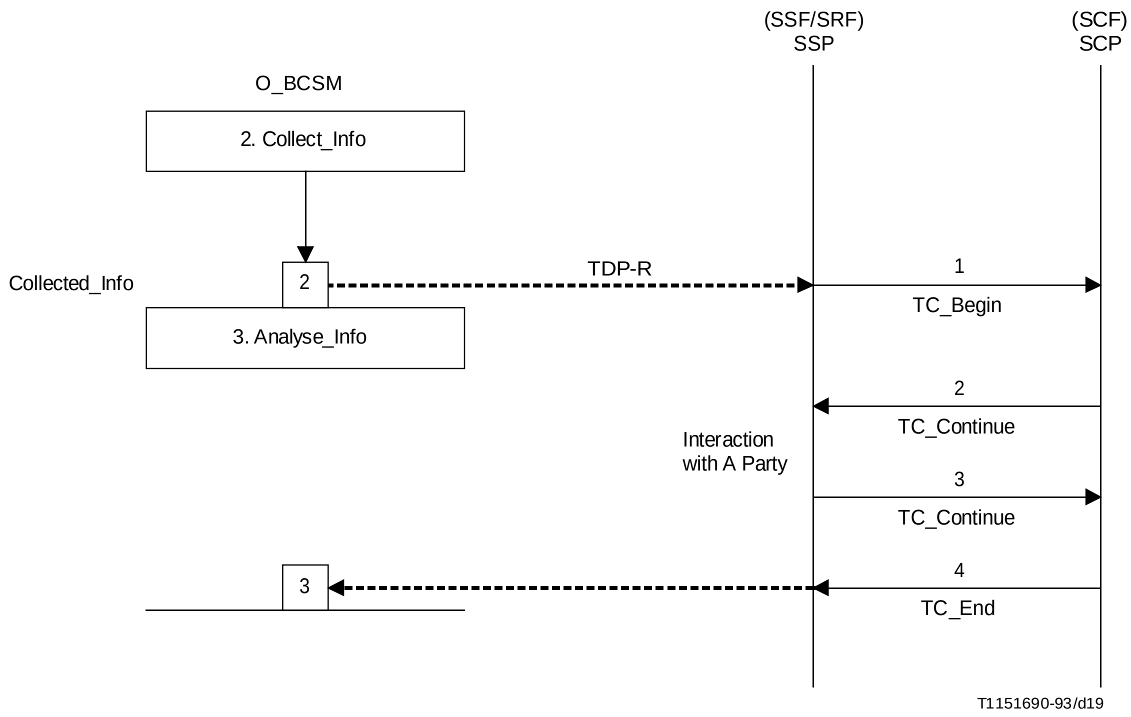


FIGURE A.2-6/Q.1219



NOTE – TDP-R comes from DP 1 (Orig. Attempt_Authorized) if it is the off-hook case.

FIGURE A.2-7/Q.1219

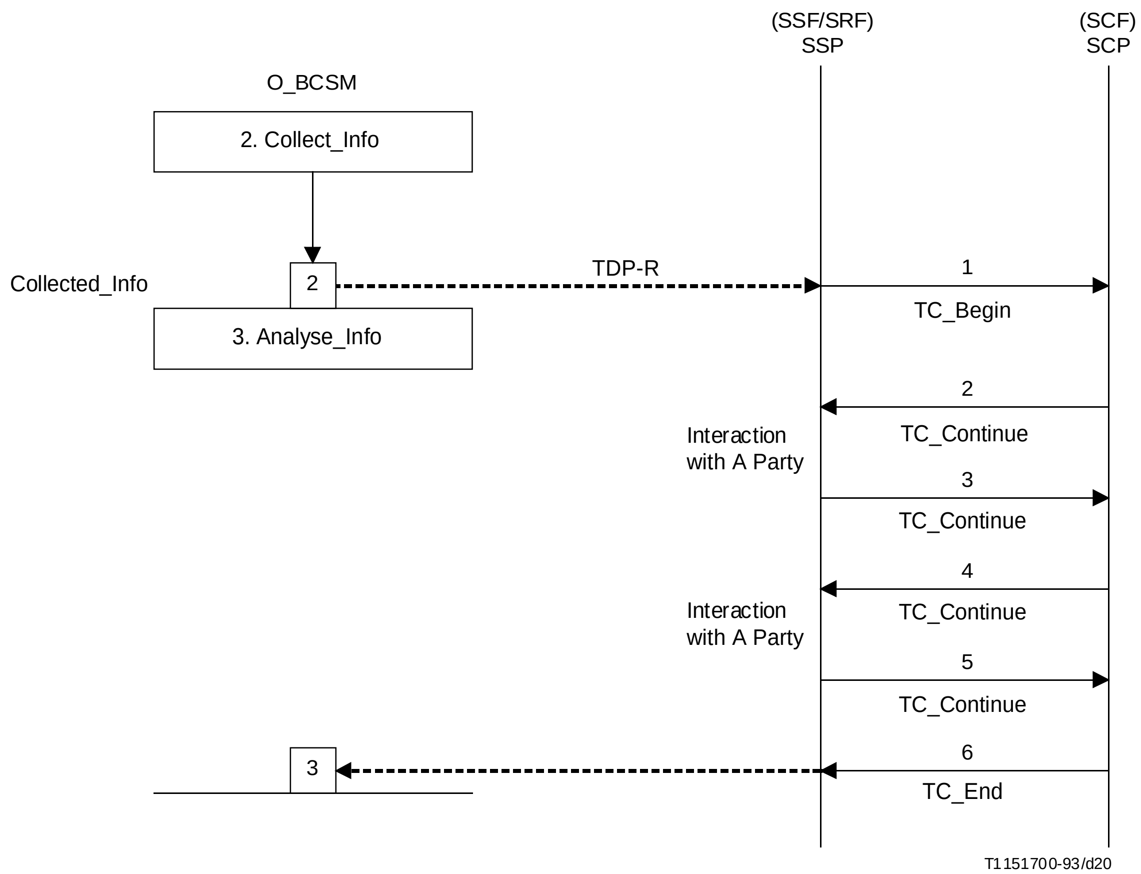


FIGURE A.2-8/Q.1219

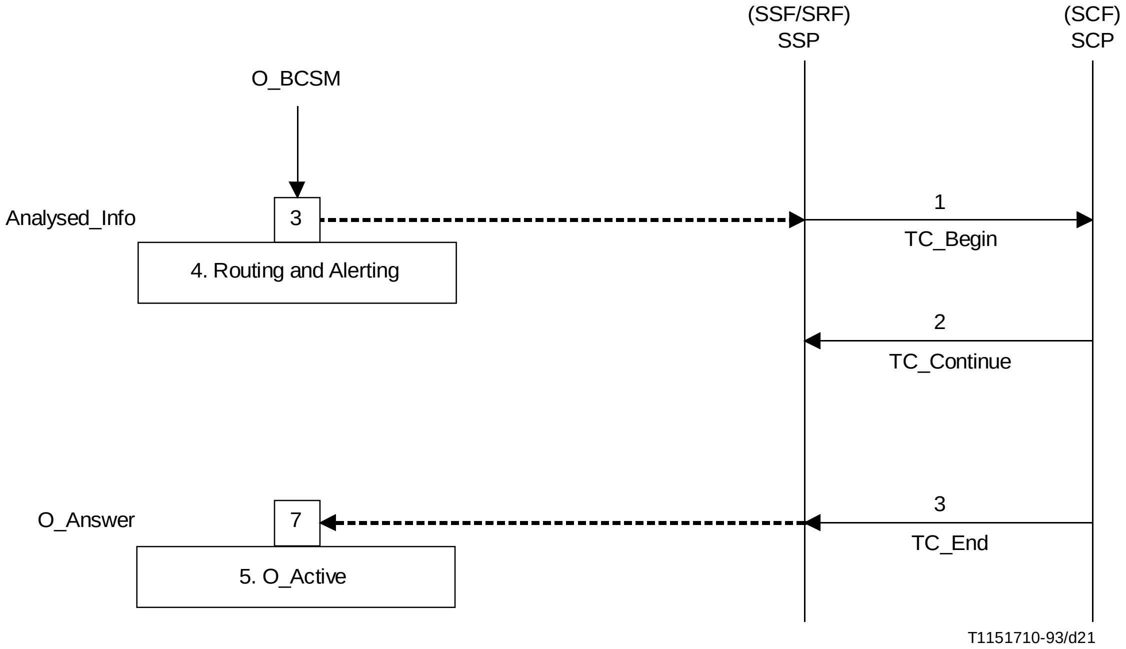


FIGURE A.2-9/Q.1219

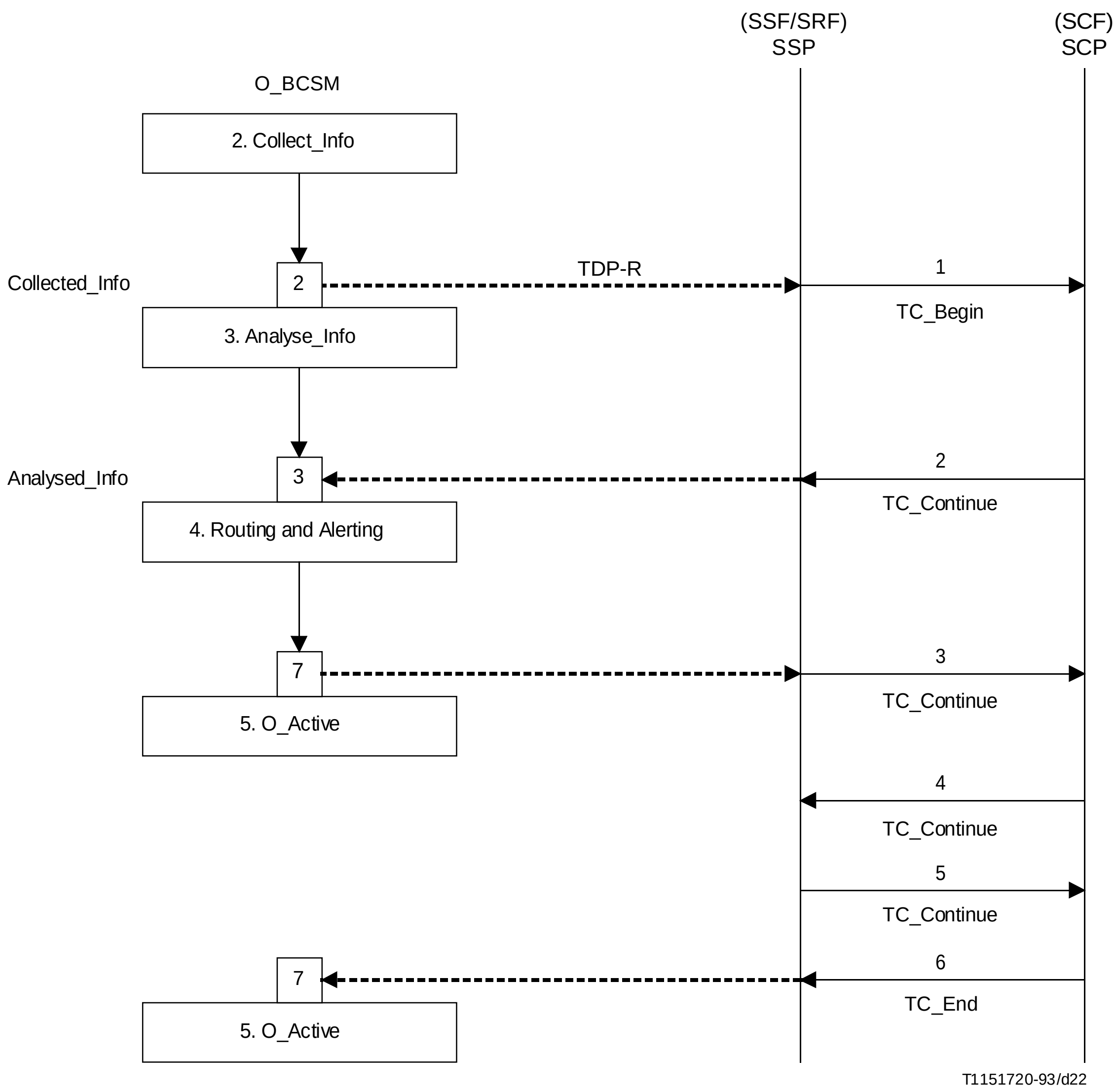


FIGURE A.2-10/Q.1219

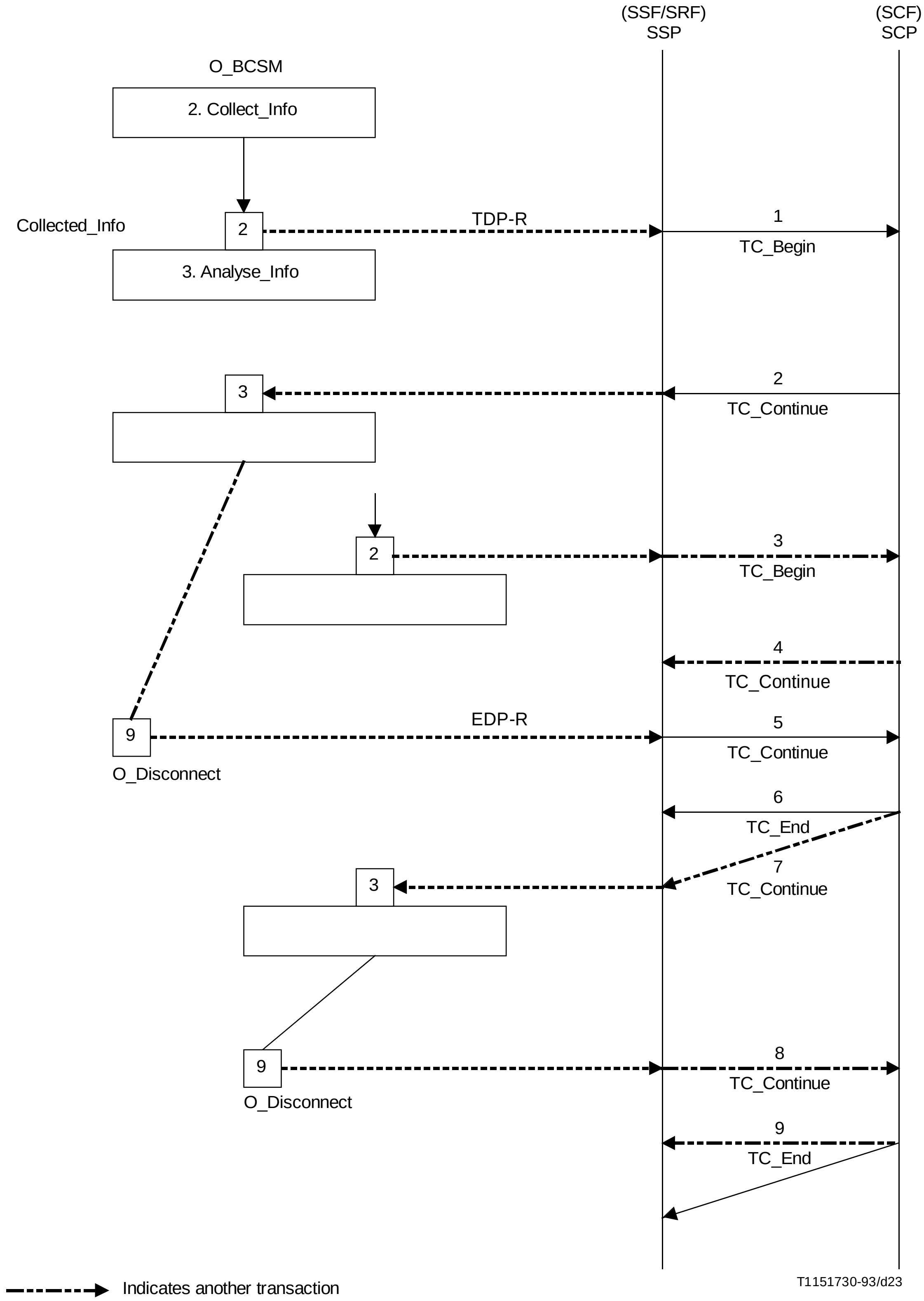


FIGURE A.2-11/Q.1219

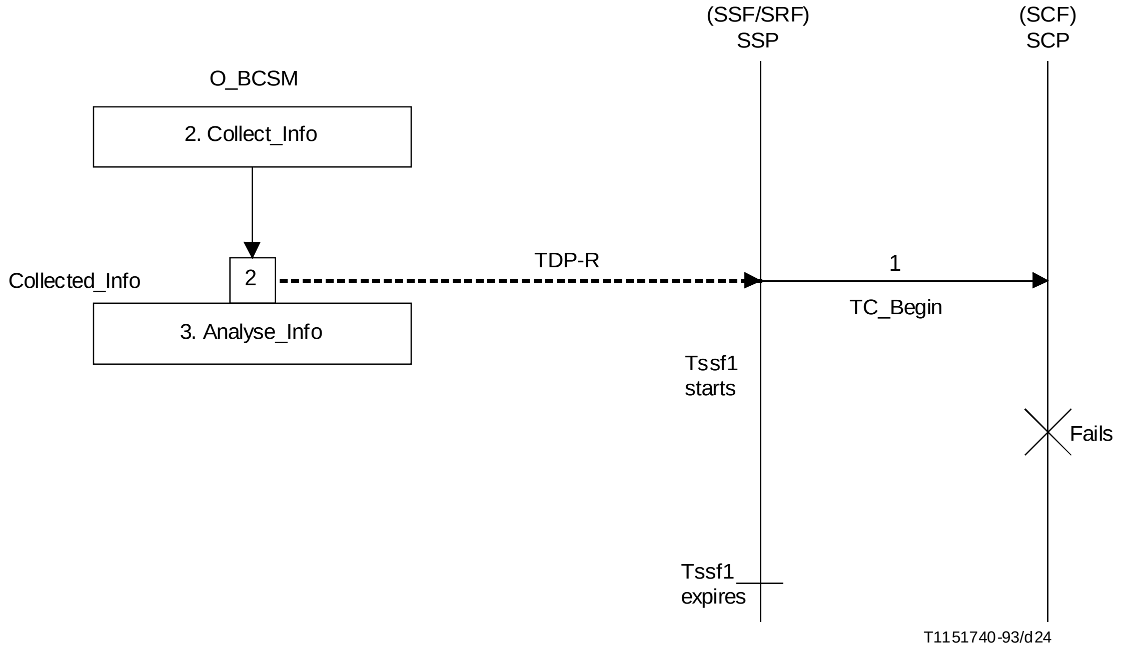


FIGURE A.2-12/Q.1219

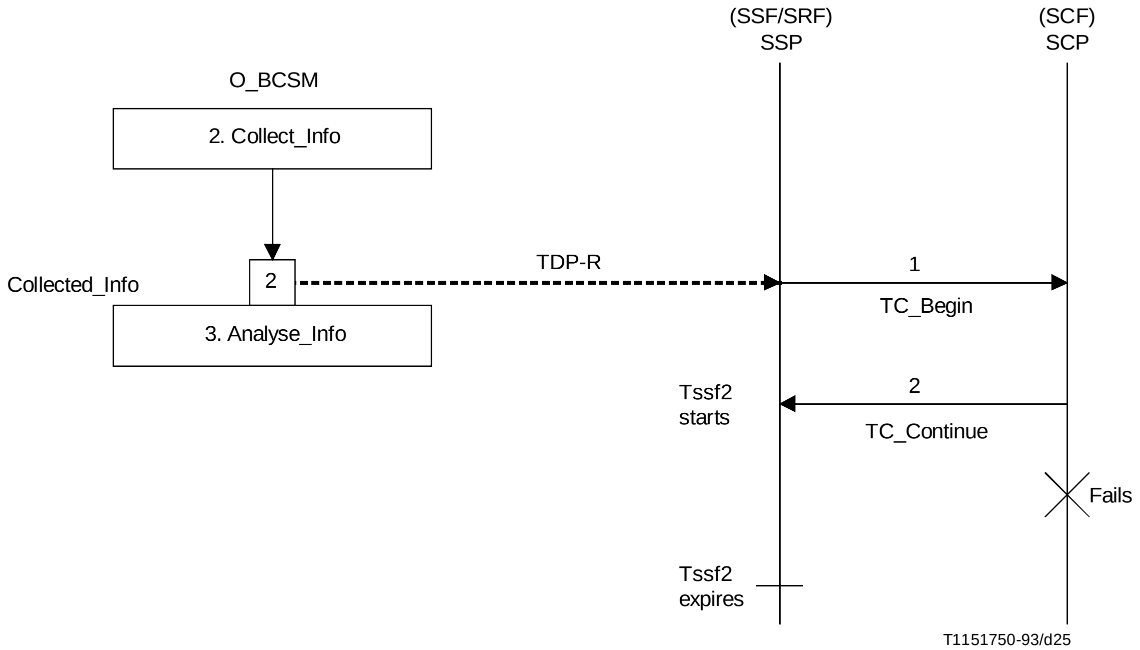


FIGURE A.2-13/Q.1219

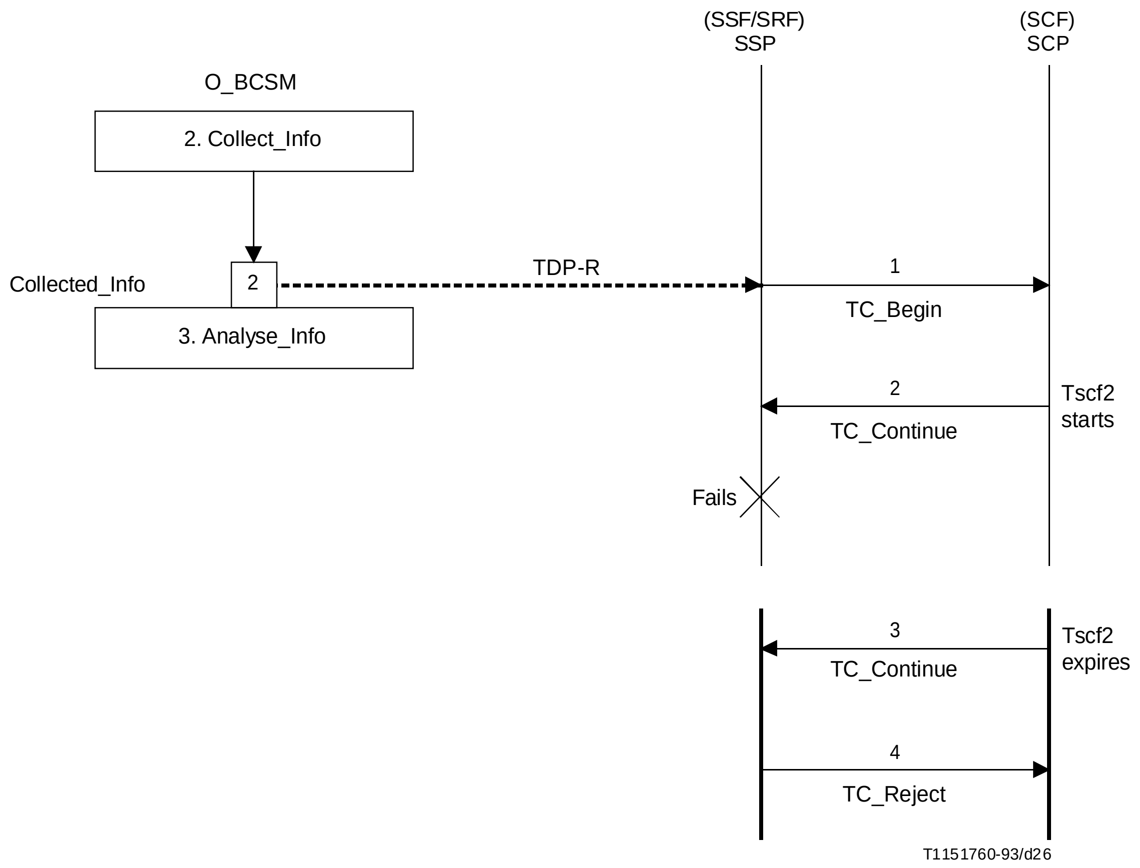


FIGURE A.2-14/Q.1219

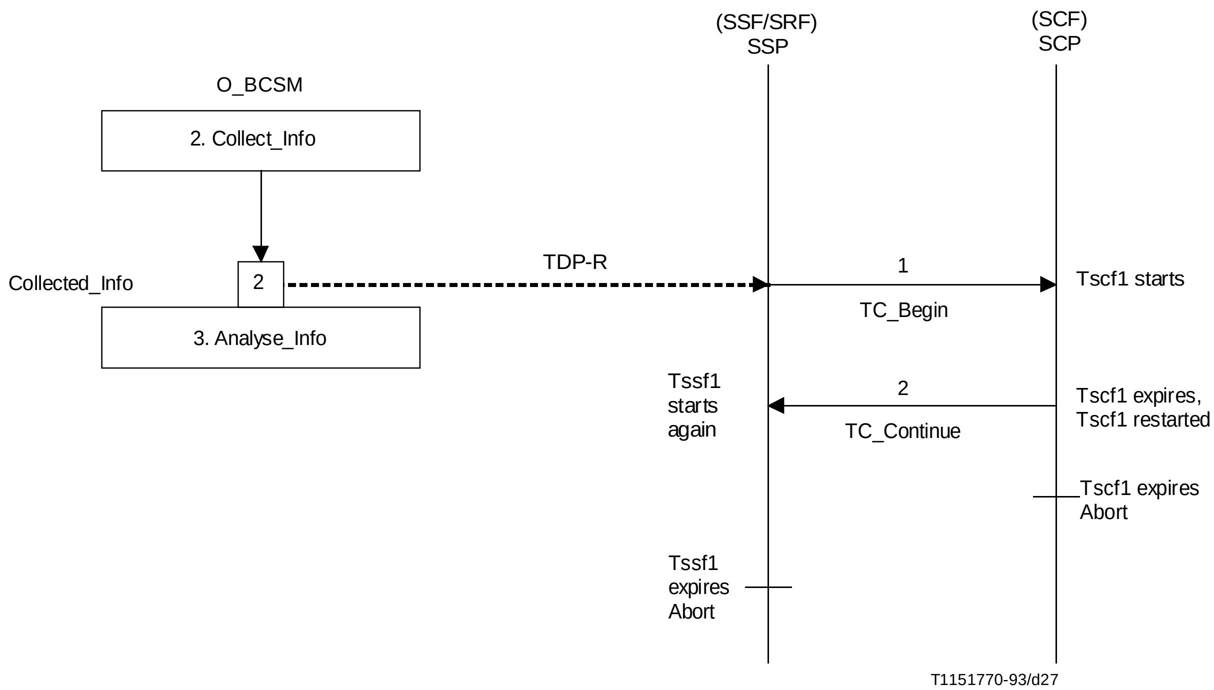


FIGURE A.2-15/Q.1219

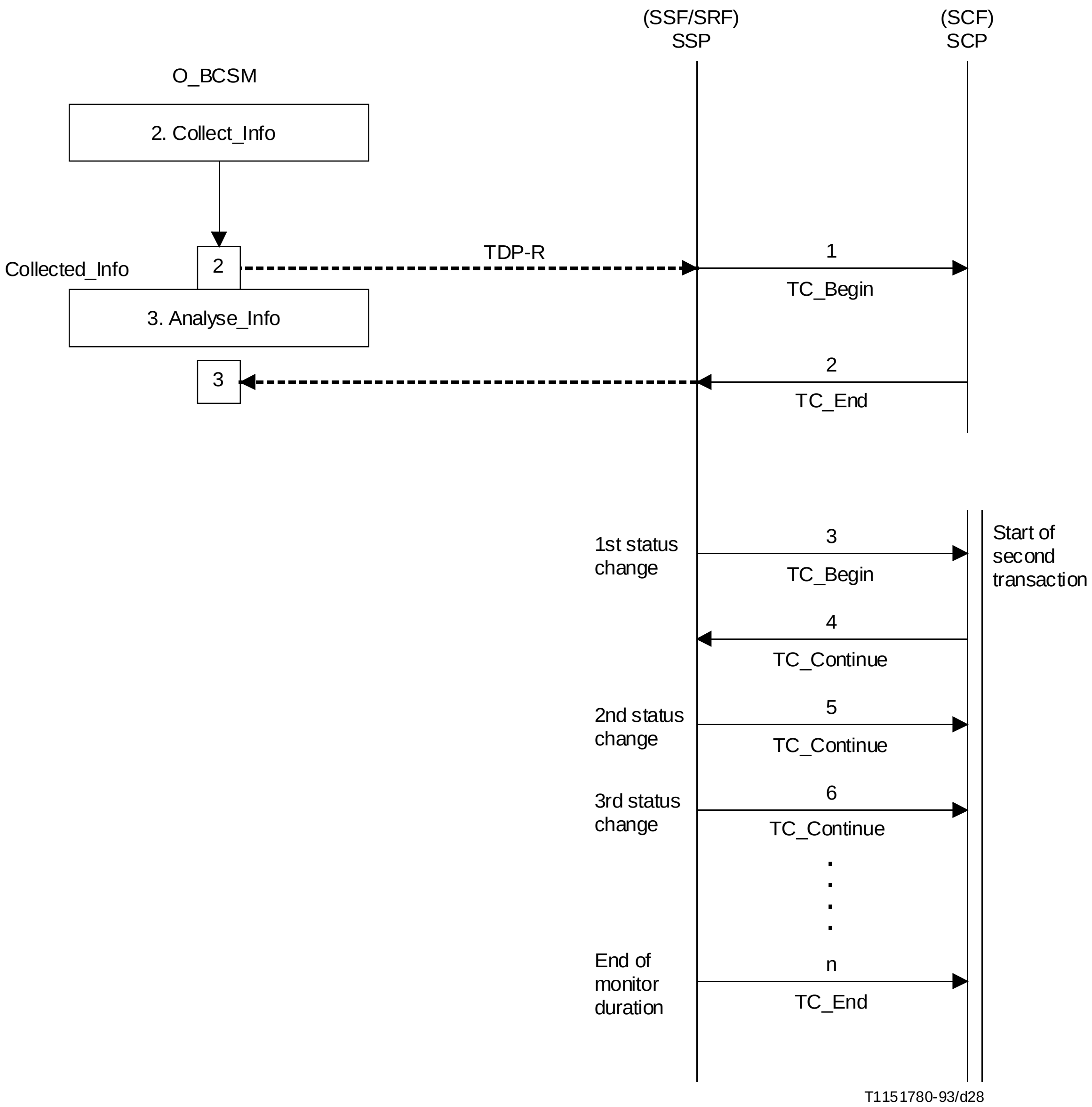


FIGURE A.2-16/Q.1219

A.3 Automatic alternative billing service scenario example

1) Service name: automatic alternative billing

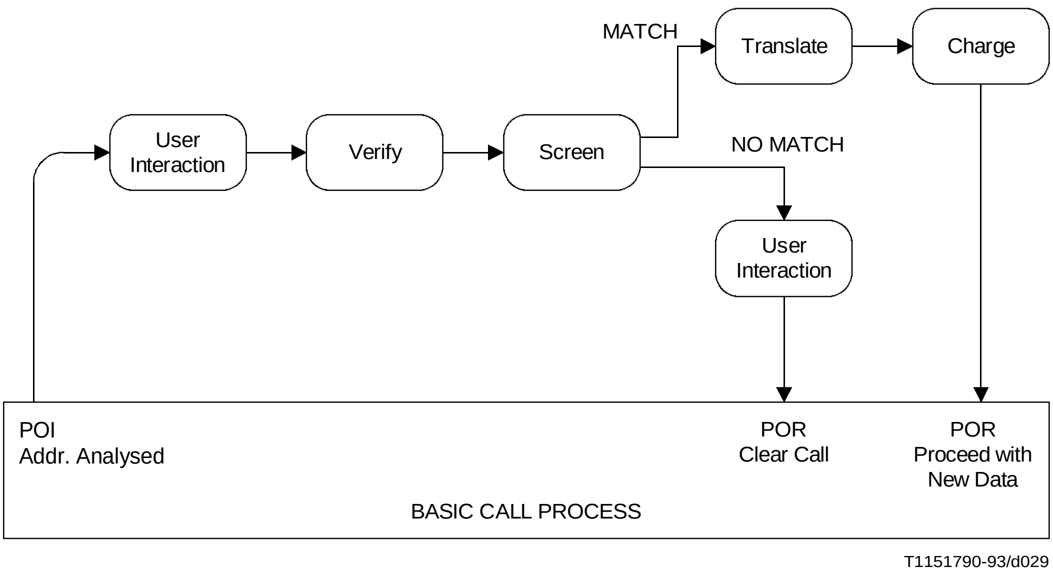
2) Description of service (description derived from Annex B/Q.1211)

The AAB service enables a user to make a call from any telephone and for the call charge to be billed to the user's account which is specific to this service, and which does not refer either to the calling line or to the called line.

An Account Code and Personal Identification Number (PIN) are allocated to a service user by the service management procedure.

To invoke the service, the user dials an access code followed by destination number, as a free call. The user then receives announcements asking for Account Code and PIN number. The Account Code and PIN are validated.

3) Global view (refer to Recommendation Q.1213)



Basic Call Process

POI – Address Analysed
CID – Dialled Number (1)
– Calling Line Identity (CLI) (2)

User Interaction SIB

input:

SSD – Announcement Parameters
– Collect Info Parameters
CID – Calling Party ID (CLI) (2)

output:

CID – Collected Data (3)
Logical End – Success

Verify SIB

input:

SSD – Maximum and Minimum number of Characters
– Format
CID – Identifier (Collected Data) (3)

output:

Logical End – Pass

Screen SIB

input:

SSD – Screen List Indicator
CID – Identifier (Collected Data) (3)

output:

Logical End – Match
– No Match

SCREEN RETURNS NO MATCH

User Interaction SIB

input:

SSD – Announcement Parameters
CID – Calling Party ID (CLI) (2)

output:

Logical End – Success

Basic Call Process

POR – Clear Call
CID – Calling Party ID (CLI) (2)

SCREEN RETURNS MATCH

Translate SIB

input:

SSD – Type
– File Indicator
CID – Information (Dialled Number) (1)

output:

CID – Translated Data (Destination Number) (4)
Logical End – Success

Charge SIB

input:

SSD – Number of Accounts (Unique Account List)
CID – Account (Collected Data) (3)

output:

Logical End – Success

Basic Call Process

POR – Proceed with New Data
CID – Destination Number (4)

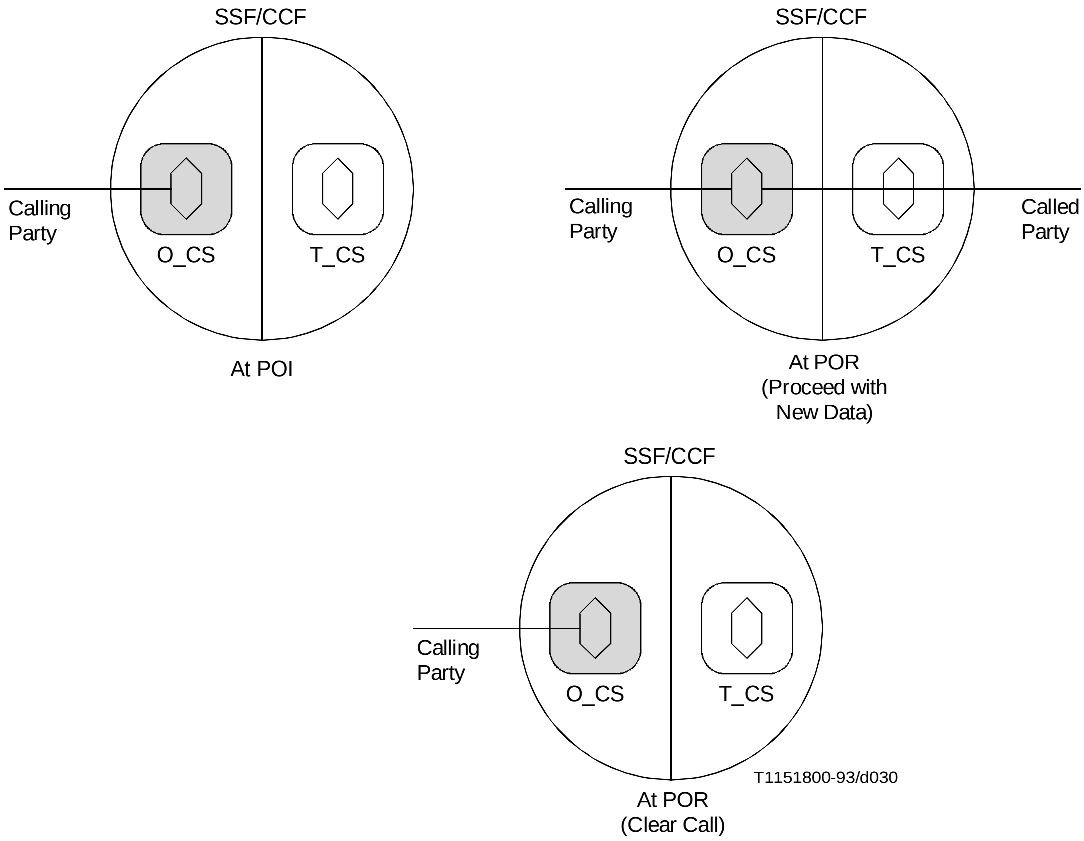
NOTES

1 Not all SSD/CID items for an SIB are necessary for this service.

2 Error conditions corresponding to the "Error" logical ends are not included here. In such a case, the User Interaction SIB is invoked to provide the user with an appropriate announcement, and the chain of SIBs ends with "POR = Clear Call" (refer to the "No Match" case after the Screen SIB.)

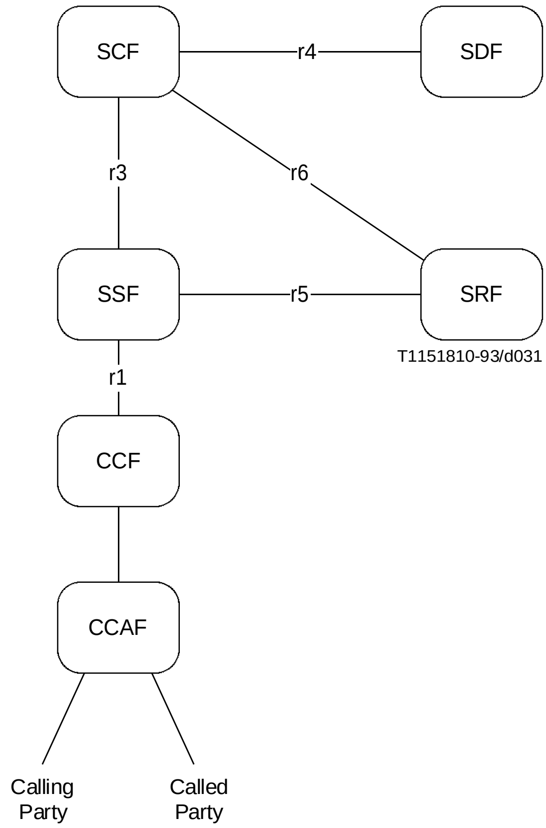
4) Distributed view (refer to clause 4/Q.1214)

A) Call Segment Diagram

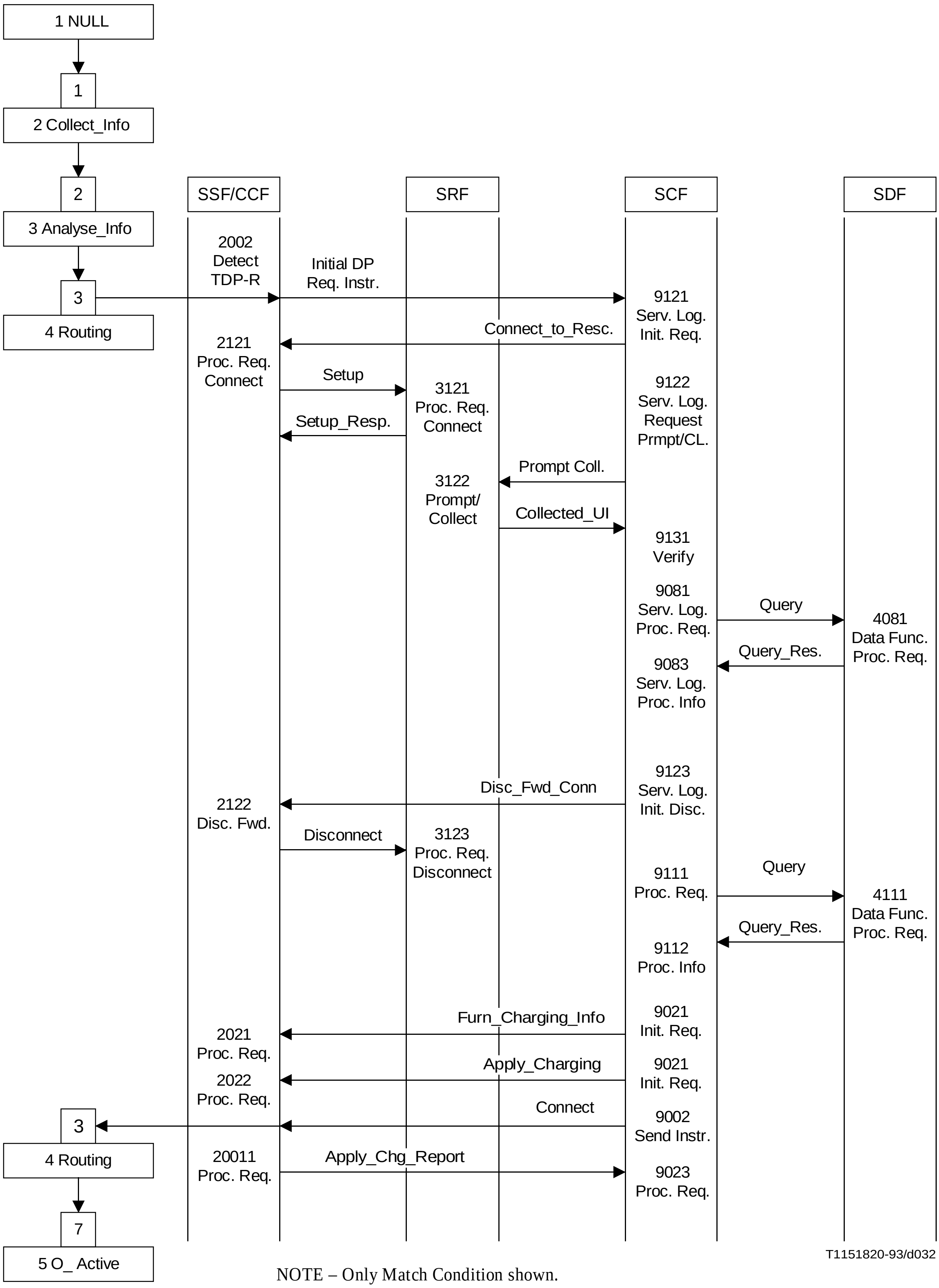


NOTE – Shaded Call Segment is the only Call Segment viewed by the SFC for this service instance.

B) FE Interfaces Diagram (IN interfaces included only)



C) Information Flow Sequence Diagram



NOTE – Only Match Condition shown.

D) FEA Descriptions (Descriptions clause 5/Q.1214)

2002 Detect Trigger Detection Point – Request
Send initial DP information and suspend call processing.

9121 Initiate Request
Request a connect to resources.

2121 Process Request
Analyse information and establish connection to SRF.

3121 Process Request
Connect incoming resource to specialized resource.

9122 Request Prompt/Collect Information

3122 Prompt and Collect – Analyse received information, apply announcement resource toward user, collect and formulate response.

9131 Perform Verify

9081 Process Request

4081 Data Function Processing Request – Screen data in base against returned information, return MATCH or NO MATCH response.

9083 Processing Information – Disconnect specialized resources. On MATCH; Charge call and request connection. On NO MATCH; Clear Call.

9123 Service Logic Initiate Disconnect

2122 Disconnect Forward

3123 Process Request
Disconnect specialized resource.

9111 Service Logic Processing Request

4111 Data Function Processing Request

9112 Service Logic Processing Information

9021 Initiate Request
Initiate a furnish charging or apply charging information request.

2021 Process Request
Process furnish charging information.

2022 Process Request
Process apply charging information and return.

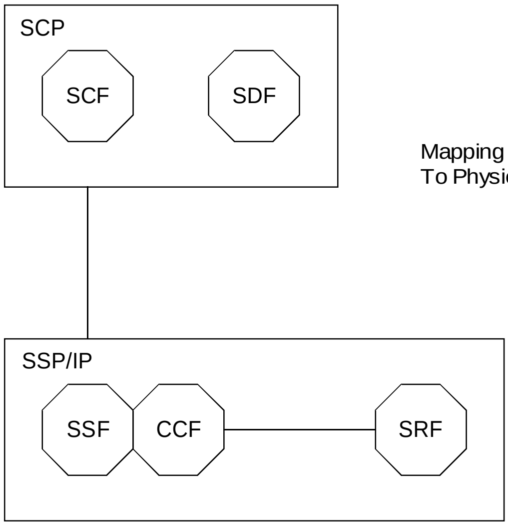
9002 Process Request and Send Immediate Instruction

20011 Process Connect

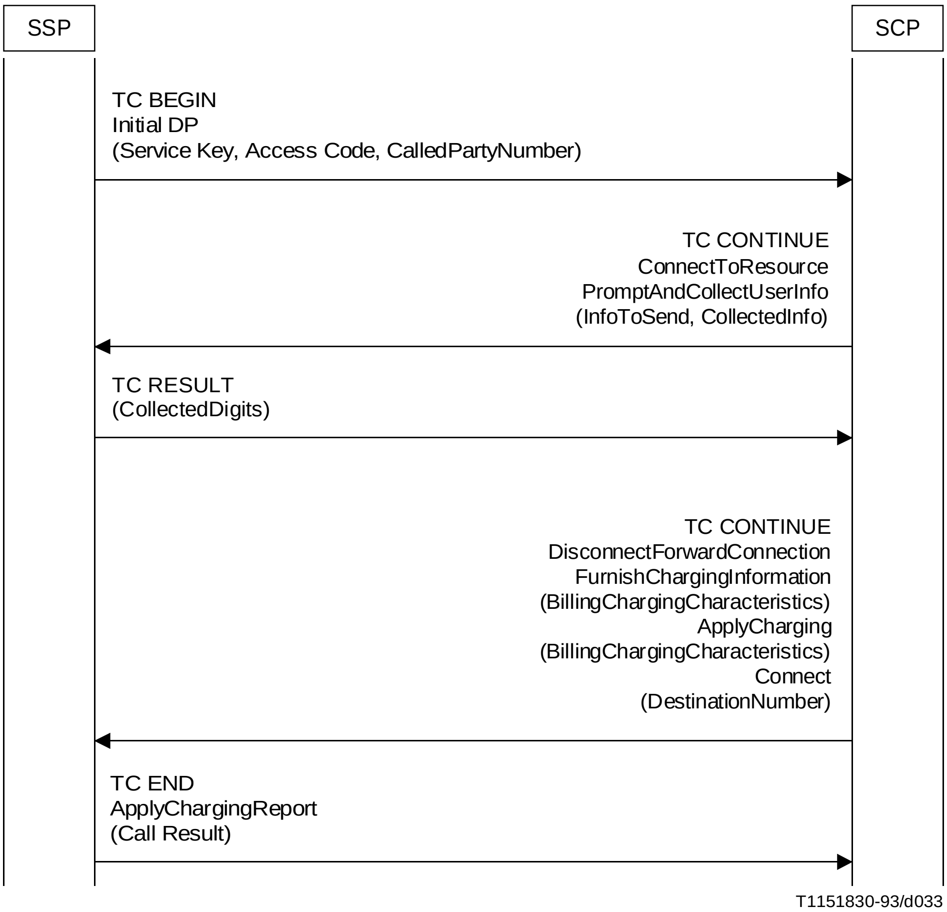
9023 Process Request
Process apply charging report information.

5) Physical view

A) Physical Entity Diagram (Refer to Recommendation Q.1215)



B) Time Sequence (Refer to Recommendation Q.1215)



C) AE Procedures (Refer to clause 3/Q.1218)

SSF:

Start IN Processing
State: Idle

transition event: e1
State: Trigger Processing

transition event: e4 (send Initial DP operation)
State: Waiting for Instructions

transition event: e5 (User Interaction requested)
State: Waiting for End of User Interaction

transition event: e6 (User Interaction ended)
State: Waiting for Instructions

transition event: e9 (receive Connect operation)
State: Idle

End of IN Processing

SCF:

Start IN Processing
State: Idle

transition event: e1 (receive Initial DP operation)
State: Process Query

transition event: e5 (Specialized Resource Facility Needed)
State: Routing to Resource

transition event: e7 (resource attached)
State: User Interaction

transition event: e11 (continue SCF processing)
State: Process Query

transition event: e13 (send Query)
State: Waiting for SDF Response

transition event: E14 (receive Query response)
State: Process Query

transition event: e4 (send Connect operation)
State: Idle

End of IN Processing

SRF:

Start IN Processing
State: Idle

transition event: E1 (Connect Request from SSF)
State: Connected

transition event: E2 (Prompt and Collect from SCF)
State: User Interaction

transition event: E10 (Call Abandoned from SSF)
State: Idle

End of IN Processing

SDF:

Start IN Processing
State: Idle

transition event: E1 (SCF Request)
State: Processing SCF Request

transition event e2 (Processing Completed)
State: Idle

End of IN Processing

A.4 Service assist service scenario example

A.4.1 Introduction

This example describes the use of the Service Assist capability as an extension of the *Automatic Alternative Billing* service described in Appendix I (ITU-T IN User's Guide), but, has far broader implications since it provides a model for how the Service Assist capability would be involved in any other case of user interaction when an Intelligent Peripheral is embedded within the SSP.

A.4.1.1 Need for service assist

If the SSP requesting instructions from the SCP does not have the required resources either within the SSP or in a directly attached IP, then remote resources need to be accessed. These remote resources can be accessed by a Service Assist procedure[2)] whereby the SCP requests that the call be transferred to another SSP, which has direct access to the required resources; this SSP then establishes a dialogue with the SCP. Eventually, control of the call is passed back to the initiating SSP.

A.4.1.2 Document map

The rest of this contribution is as follows:

- Subclause A.4.2 modifies a configuration in the description of *Automatic Alternative Billing* service to demonstrate the use of Service Assist. This subclause also provides a brief description of the Service Assist procedure;
- Subclause A.4.3 describes the example at the IN Distributed Functional Plane;
- Subclause A.4.4 describes the example at the IN Physical Plane; and
- Subclause A.4.5 is the conclusion.

Note that within the IN model, the Service Assist procedures are not visible at any other plane than described in this Recommendation.

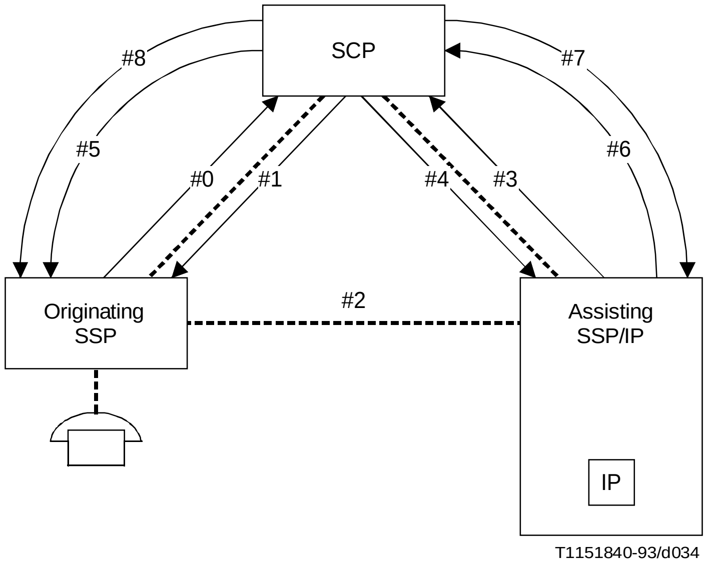
A.4.2 Configuration

The Physical Configuration associated with the need to use the Service Assist Capability is represented in Figure A.4-1. We assume that the SSP that sent the initial request for instructions[3)] does not have the IP that meets the demands of the *Automatic Alternative Billing* service, but that another SSP in the network[4)] does have such an IP

2) The procedures for Service Assist (with various flavours) are described in 3.1.3/Q.1218. The description of the relevant information flows and their semantics can be found in Recommendation Q.1214, while the respective operations are specified in clause 2/Q.1218. Finally, the formal specification of the Service Assist procedures is located in the description of the State Transition Models of clause 3/Q.1218

3) This SSP is called Originating SSP.

4) That SSP is called Assisting SSP.



#0 DP-specific Request Instructions
#1 Establish Temporary Connection
#2 (Transport)
#3 Assist Request Instructions
#4 Prompt and Collect User Information
#5 Reset Timer
#6 Return Result for Prompt and Collect UI
#7 End Assist Transaction
#8 Disconnect Forward Connection

FIGURE A.4-1/Q.1219

Augmented physical configuration

The following is a quick summary of the Service Assist procedure; it is illustrated in Figure A.4-1.

1) The SCP, during the processing of the initial request for instruction, determines that a resource that is remote from the SSP is required. Consequently, the *Establish Temporary Connection* operation is sent to the Originating SSP.

2) The Originating SSP routes the call to the Assisting SSP.

3) The Assisting SSP uses the *Assist Request Instructions* operation to establish communications with the SCP.

4) The SCP sends instructions to Play Announcements and Collect Digits to the Assisting SSP based on service logic control. The SSP establishes the connection between the IP and the call party; the announcements are played, data collected by the IP, etc.

5) The SCP may need to send *Reset Timer* operations to the Originating SSP so that it does not time out the transaction.

6) The collected information is received by the SCP.

7) The SCP terminates the transaction with the Assisting SSP; and

8) The *Disconnect Forward Connection* operation is sent to the Originating SSP to disconnect the temporary connection. (To avoid the obvious racing condition, it is important that the SSP receives this message after the Assisting SSP is instructed to end the Assist function.)

A.4.3 In-DFP-level description

This description is accompanied by Figure A.4-2, which replaces the sequence of IFs and FEA between SSF/CCF, SRF, and SCF of the IF diagram in the description of Automatic Alternative Billing, starting with the *Connect to Resource* IF and ending with the *Disconnect* IF, by the IFs and FEAs relevant to Service Assist.

1) In response to the initial query (Initial DP Request Instructions, see 5.3.2.2) from the originating SSF/CCF, the SCF sends the *Establish Temporary Connection* information flow. In this example, we select an option specified in 6.4.2.20/Q.1214, by which the *SCF Id* and *Correlation Id* are embedded within the *Assisting SSF/SRF Routing Address* information element.

2) The Originating SSF/CCF passes necessary information to the Assisting SSF/CCF via transport mechanism (through interface *r2* of Recommendation Q.1214).

3) The Assisting SSF/CCF issues the *Assist Request Instruction* information flow, using two mandatory information elements (*Call Id* and *Correlation Id*) and an optional information element *SSF/SRF Capabilities*.

4) Upon reception of *Assist Request Instruction*, the SCF proceeds with the information exchange relevant to playing announcements and collecting digits activities as described in the Automatic Alternative Billing example in Appendix I. During this activity, the SCF resets timer T_{SSF} via the *Reset Timer* information flow. At the end of this activity, the SSF sends the *Collected User Information* information flow, which stimulates the SCF to send the *Disconnect Forward Connection* information flow to the Originating SSF.

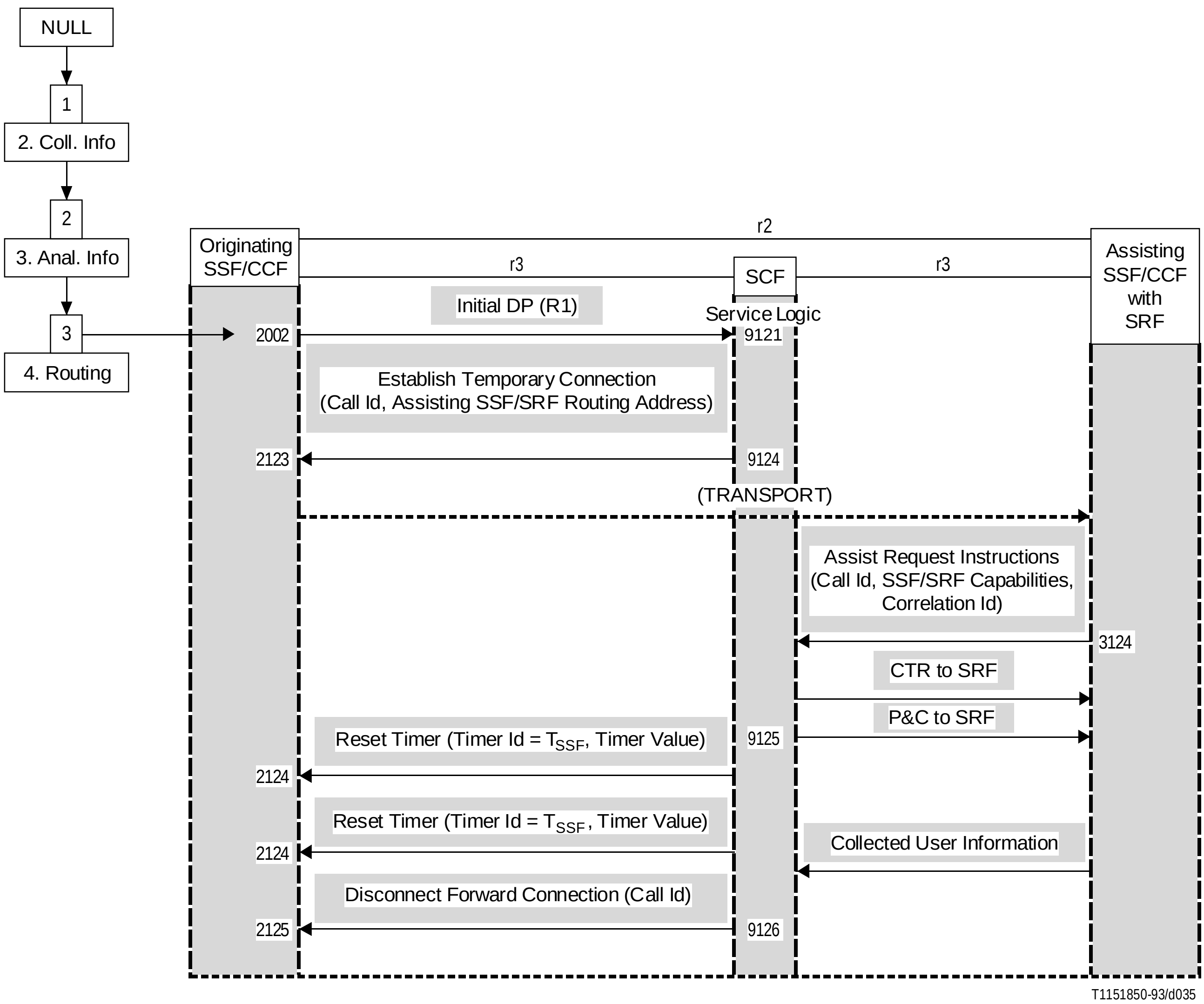


FIGURE A.4-2/Q.1219

Service assist information flows

A.4.4 In-physical-plane-level description

This description is accompanied by Figure A.4-3. The initial state of the system, as far as the specification of procedures of Recommendations Q.1218 is concerned, is as follows:

- The SSF of the Originating SSP is in it's State c, "Waiting for Instructions";
- The SCF of the SCP is in its State 3.1, "Determine Mode," which is, in turn, a substate of State 3, "Routing to Resource"; and
- The SSF of the Assisting SSP is in its State a, "Idle."

Then the above system works in the following manner:

1) When the service logic has determined that Service Assist should be invoked, the internal event *(e3.2) Assist_Needed* takes place, causing the **EstablishTemporaryConnection[assistingSSPIPRoutingAddress]** operation to be issued to the Originating SSP. This event causes the transition to State 3.2, "Waiting for Assist Request Instructions".
2) When the Originating SSP receives the above operation, it establishes the connection with the Assisting SSP. The following events take place:
   - *(ea1) Assist/Hand-off Needed* at the Assisting SSF – This event causes the **AssistRequestInstructions [correlationID, iPAvailable, iPSSPCapabilities]** operation to be sent to the SCP, and a transition to state e, "waiting for instructions"; and
   - *(e7) Temporary_Connection_Created* at the Originating SSP – Which causes the transition of the Originating SSF into State e, "Waiting for End of Temporary Connection";
3) The reception by the SCP of the **AssistRequestInstructions** operation causes the SCF to transit to State 4, "UserInteraction".
4) At this point, the SCP and SSP (with its internal IP) start the user interaction procedure, the details of which are outside the scope of this Recommendation. The relevant states of the SCF and Assisting SSF are shown, however (within the dotted rectangle). Note, that when the SCF timer expires, the **ResetTimer [timerValue]** operation is sent to the Originating SSP, but it causes no state change in the Originating SSF.
5) When the service logic detects the end of user interaction (event *(e11) Continue_SCF_Processing*), the SCP closes the transaction with the Assisting SSP by issuing **TC_End Req**. Then the SCP issues the **DisconnectForwardConnection** operation to the Originating SSP and enters its State 2, "Preparing SSF Instructions".
6) The reception of **TC_End Ind** by the Assisting SSP causes the event *(ea4) User Interaction Ended* at the Assisting SSF; The Assisting SSF transits into State b, "Waiting for Instructions".
7) The reception of **DisconnectForwardConnection** by the Originating SSP causes event *(e8) TemporaryConectionEnded*, which results in disconnection of the Assisting SSF. With that, the Originating SSF transits to State c, "Waiting for Instructions," and the Assisting SSF transits to State a, "Idle."

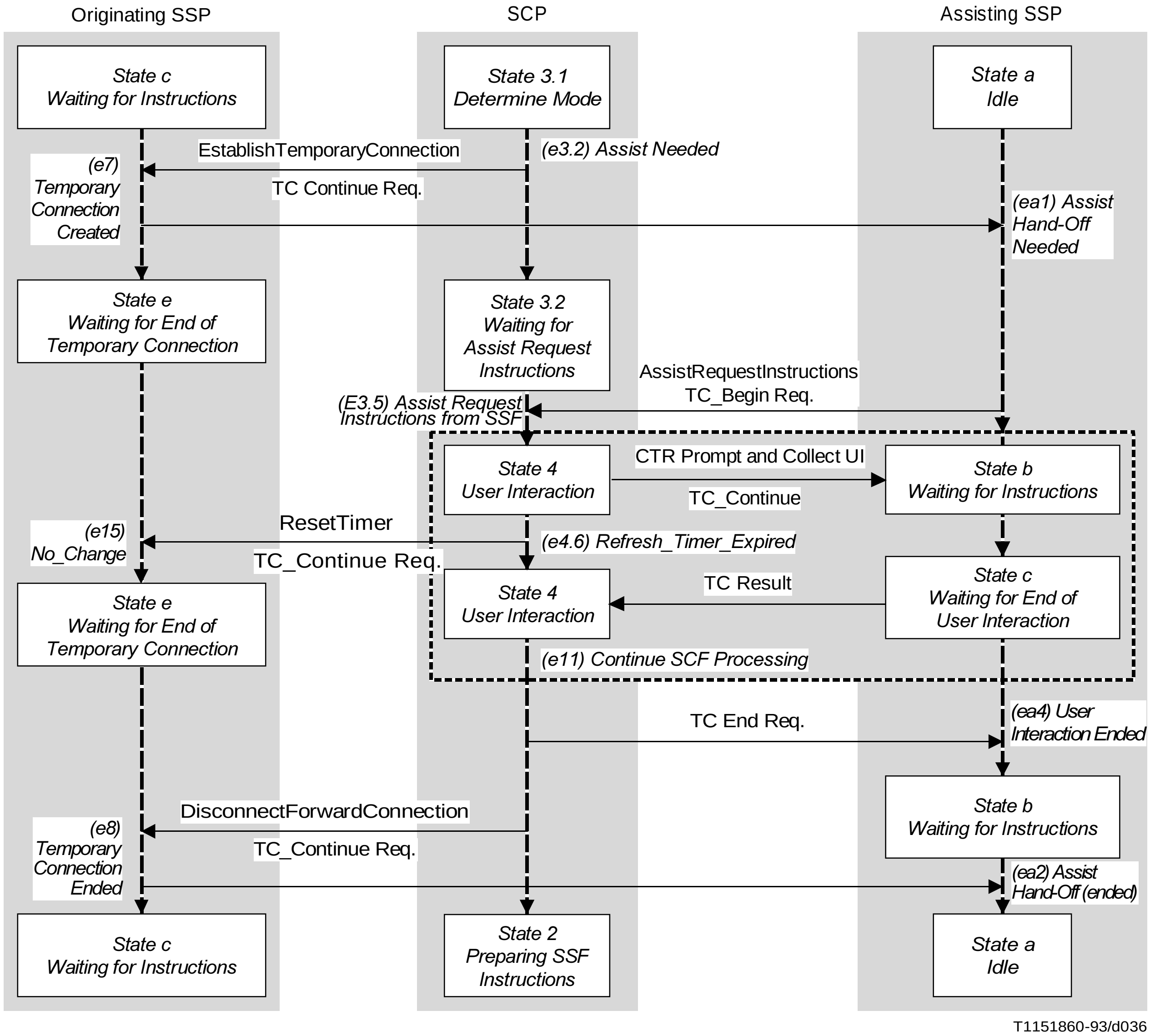


FIGURE A.4-3/Q.1219

Service assist operations

A.5 Universal Personal Telecommunication service

1) Service name: "Universal Personal Telecommunication"

2) Description of service

The prose description of this example service is based on the stage 1 description of UPT given in CCITT Recommendation F.851 (CCITT SG I/1). It does not exactly conform to UPT as described in Recommendation Q.76. Therefore, we will denote the service in this example by "UPT" to distinguish it from the UPT description in Q.76.

2.1) General characteristics

"Universal Personal Telecommunication Service" ("UPT") is a service that allows "UPT" users to move between terminals to make and receive calls on a global basis. The service can be divided into four functional parts:

1) registration;

2) deregistration;

3) outgoing calls;

4) incoming calls.

Parts 1 and 2 allow the user to (de)register himself for incoming calls. Part 3 allows the "UPT" user to place outgoing calls charged to his "UPT" account. Part 4 routes calls to the location the "UPT" user has registered himself at.

2.2) Short textual description

In this example Parts 1, 2, and 3 are considered a single procedure, therefore they are treated together. To authenticate himself to the "UPT" supporting network, the user dials a "UPT" 'specific service code' (e.g. UPT1). He is then prompted to enter his "UPT" Number and his Personal Identification Number (PIN). Networks capable of collecting "UPT" specific service code and "UPT" number in a single event, are allowed to do this and prompt the user only for his PIN. The combination PIN and "UPT" number is checked and a number of retries on the "UPT" number as well as on the PIN is allowed. If a maximum number of retries is exceeded, the user is prompted and the call is terminated. If the user enters his PIN and "UPT" number correctly, he is prompted to make a choice to either (de)registrate himself or to place an outgoing call.

If registration is selected the user is prompted for his location (the network address he is using), which is echoed back to him. The registration request is checked for authorisation and the network is checked for its capability to support the "UPT" incalls (e.g. terminate incoming calls). If all is correct the network address is stored in the "UPT" users home database. The user is then prompted if he would like to invoke additional facilities (e.g. place an outcall) and is left after the authentication procedure.

If deregistration is selected, the user is notified of the acceptance of deregistration and the network address is removed from the "UPT" users home database. The user is then prompted if he would like to invoke additional facilities (e.g. place an outcall) and is left after the authentication procedure.

If outcall is selected the user is prompted to enter the number of the party he would like to reach. This number is checked for authorisation of the user to call the number (e.g. with respect to charging). If allowed, the call is completed. If not the call is abandoned.

"UPT" – incalls are invoked by an A party dialling a second "UPT" 'specific service code' (e.g. UPT2) and the "UPT" Number (which for the purpose of this example consists of an Administration Number and a Subscriber Number). The home database of the "UPT" user is queried for the location of the "UPT" user and charging is settled. The user is notified that special "UPT" charging is applied to the call. The call is completed to the location on which the "UPT" user has registered himself.

3) Global view

3.1) Introduction

We give a description of Global Service Logic for the “Universal Personal Telecommunications” service. One part is graphical the other part consists of an individual SIB description. In the graphical description we will not depict the error exits of a SIB as we assume that these error exits lead to default error handling (release of call). Where SIBS are used in a monitor relation we depict them with dashed lines. The SIBs used for the control relationship are depicted (as usual) with solid lines.

3.2) “UPT” Outcall, (De)Registration

3.2.1) Service Independent Building Blocks description

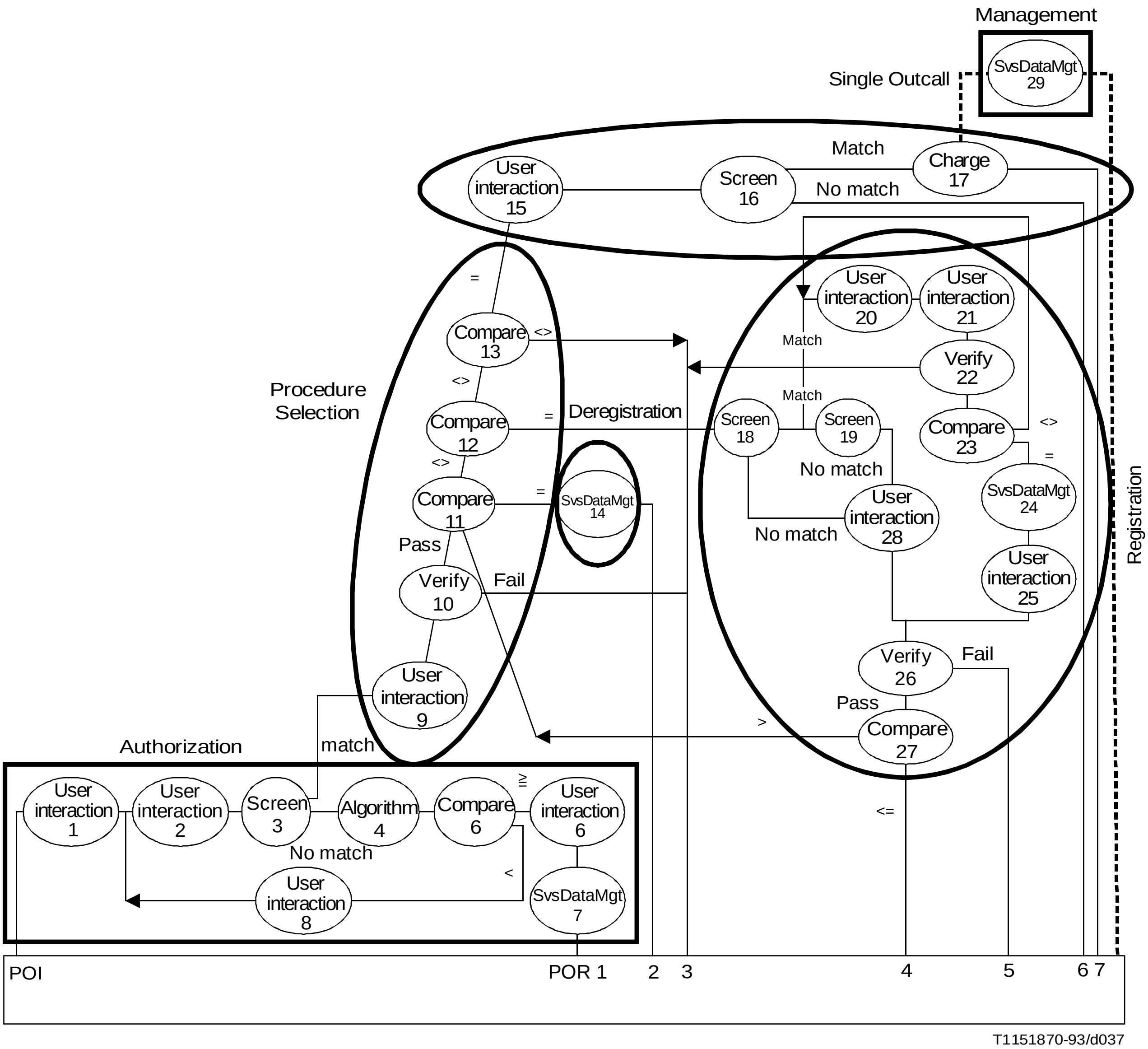


FIGURE A.5-1/Q.1219

Registration/Deregistration/Outcall of “UPT” user

Authorization

1. User Interaction[5)]

Input:

1.1	Service Support Data:			
	Announcement Parameters	Announcement ID	=	“Provide your identity”
	Repetition Requested		=	No
	Collect Information Parameters	User Interruptibility	=	Yes
		Voice Feedback	=	No
		Type	=	DTMF
		Maximum number of characters	=	13 * 12 digits + end delineator *\
		Minimum number of characters	=	4
		Initial Input waiting timer	=	10 * time in seconds *\
		Inter-character waiting timer	=	5
		End delineator	=	#
	CIDFP-Call party		=	
	CIDFP-Collected			
	CIDFP-Error			
1.2	Call Instance Data:	Call Party Identifier	=	CLI

Output:

1.3	Logical End:			Success Error
1.4	Call Instance Data:	Collected Data	=	UPT-number Error Cause (see Recommendation Q.1213)

2. User Interaction

Input:

2.1	Service Support Data:			
	Announcement Parameters	Announcement ID	=	“Provide your authentication code”
		Repetition Requested	=	No
	Collect Information Parameters	User Interruptibility	=	Yes
		Voice Feedback	=	No
		Type	=	DTMF
		Maximum number of characters	=	5 * 4 digits PIN + end delineator *\
		Minimum number of characters	=	5
		Initial Input waiting timer	=	10
		Inter-character waiting timer	=	5
		End delineator	=	#
	CIDFP-Call party		=	
	CIDFP-Collected			
	CIDFP-Error			
2.2	Call Instance Data:	Call Party Identifier	=	CLI

Output:

2.3	Logical End:			See SIB 1
2.4	Call Instance Data:	Collected Data	=	Dialled digits for authentication Error Cause (see Recommendation Q.1213)

---

5) Numbering of SIBs corresponds to Figure A.5-1.

3. Screen

Input:

3.1	Service Support Data:	Screen List Indicator	=	(UPT-service, home, PIN) * list at home database checked for PIN *\
	CIDFP-Screen CIDFP-Error			
3.2	Call Instance Data:	Identifier	=	Dialled digits for authentication (associated with CIDFP-Screen)

Output:

3.3	Logical End:			Match No Match Error
3.4	Call Instance Data:			Error Cause (see Recommendation Q.1213)

4. Algorithm

Input:

4.1	Service Support Data:	Type Value	= =	increment 1
	CIDFP-Algorithm CIDFP-Error			
4.2	Call Instance Data:	Identifier	=	number of retries

Output:

4.3	Logical End:			Success Error
4.4	Call Instance Data:	Identifier	=	number of retries Error Cause (see Recommendation Q.1213)

5. Compare

Input:

5.1	Service Support Data: CIDFP-Compare	Comparison Type	=	Identifier Value
		Reference value	=	maximum number of retries
	CIDFP-Error			
5.2	Call Instance Data:	Identifier	=	number of retries

Output:

5.3	Logical End:			GREATER THAN (>) LESS THAN (<) EQUAL TO (=) Error
5.4	Call Instance Data:			Error Cause (see Recommendation Q.1213)

6. User Interaction

Input:

6.1	Service Support Data:			
	Announcement Parameters	Announcement ID	=	“Wrong Authentication. Your number is now blocked. Please hang up”
		Repetition Requested	=	No
	Collect Information Parameters	Type	=	Null
	CIDFP-Call party CIDFP-Error			
6.2	Call Instance Data:	Call Party Identifier	=	CLI

Output:

6.3	Logical End:			See SIB 1
6.4	Call Instance Data:			Error Cause (see Recommendation Q.1213)

7. Service Data Management

Input:

7.1	Service Support Data:	File indicator	=	UPT-user data file * stored in the UPT home database *\
		Action	=	replace
		Element Indicator	=	Line Status data field * Line Status is "Normal" or "Blocked" *\
	CIDFP-Info CIDFP-Error			
7.2	Call Instance Data:	Information value	=	Blocked * Number of retries exceeded: line is blocked *\

Output:

7.3	Logical End:			Success Error
7.4	Call Instance Data:			Error Cause

8. User Interaction

Same as SIB 1 with	Announcement ID	=	"Wrong authentication: Please retry, provide your identity"

Procedure Selection

9. User Interaction

Input:

9.1	Service Support Data:			
	Announcement Parameters	Announcement ID	=	"Identify Procedure"
		Repetition Requested	=	No
	Collect Information Parameters	User Interruptibility	=	Yes
		Voice Feedback	=	No
		Type	=	DTMF
		Maximum number of characters	=	1
		Minimum number of characters	=	1
		Initial Input waiting timer	=	10
		Inter-character waiting timer	=	NULL
	CIDFP-Call party CIDFP-Collected CIDFP-Error		=	
9.2	Call Instance Data:	Call Party Identifier	=	CLI

Output:

9.3	Logical End:			See SIB 1
9.4	Call Instance Data:	Collected Data	=	ProcedureId Error Cause (see Recommendation Q.1213)

10. Verify

Input:

10.1	Service Support Data:	Maximum number of characters	=	1
		Minimum number of characters	=	1
		Format	=	n * any digit except 0 (1-10) *\
	CIDFP-Data			
	CIDFP-Error			
10.2	Call Instance Data:	Identifier	=	ProcedureId

Output:

10.3	Logical End:			Pass Fail Error
10.4	Call Instance Data:			Error Cause (see Recommendation Q.1213)

11. Compare

Input:

11.1	Service Support Data:	Comparison Type	=	Identifier Value
		CIDFP-Compare		
		Reference Value	=	1
	CIDFP-Error			
11.2	Call Instance Data:	Identifier	=	ProcedureId

Output: See SIB 5

12. Compare

Same as SIB 11 with Reference Value = 2

13. Compare

Same as SIB 11 with Reference Value = 3

Deregistration

14. Service Data Management

Input:

14.1	Service Support Data:	File Indicator	=	UPT-user data file
		Action	=	replace
		Element Indicator	=	Registered data field
	CIDFP-Info			
	CIDFP-Error			
14.2	Call Instance Data:	Information value	=	FALSE

Output: See SIB 7

Single Outcall

15. User Interaction

Input:

15.1	Service Support Data:			
	Announcement Parameters	Announcement ID	=	"Supply destination number"
		Repetition Requested	=	No
	Collect Information Parameters	User Interruptibility	=	Yes
		Voice Feedback	=	No
		Type	=	DTMF
		Maximum number of characters	=	13
		Minimum number of characters	=	4
		Initial Input waiting timer	=	10
		Inter-character waiting timer	=	5
		End delineator	=	#
	CIDFP-Call party			
	CIDFP-Collected			
	CIDFP-Error			
15.2	Call Instance Data:	Call Party Identifier	=	CLI

Output:

15.3	Logical End:			See SIB 1
15.4	Call Instance Data:	Collected Data	=	DestinationNumber Error Cause (see Recommendation Q.1213)

16. Screen

Input:

16.1	Service Support Data:	Screen List Indicator	=	(UPT-service, home, outcall) * list at home database checked for authorization of outcall *\
	CIDFP-Screen			
	CIDFP-Error			
16.2	Call Instance Data:	Identifier	=	DestinationNumber

Output: See SIB 3

17. Charge

Input:

17.1	Service Support Data:	Number of accounts to charge	=	1
		Account		
		Fixed Account	=	UPT-number
		Percentage	=	100
		Resource type	=	bearer type * extra charges for e.g. SDF usage are not considered in this example *\
		Units	=	1
		Service/Service Feature Identifier	=	UPT
	CIDFP-Error			
17.2	Call Instance Data:	Account	=	UPT-number

Output:

17.3	Logical End:			Success Error
17.4	Call Instance Data:			Error Cause (see Recommendation Q.1213)

Registration

18. Screen

Input:

18.1	Service Support Data:	Screen List Indicator	=	(UPT-service, local, registration) * list at local database checked for authorization of registration *\
	CIDFP-Screen			
	CIDFP-Error			
18.2	Call Instance Data:	Identifier	=	UPT-number

Output: See SIB 3

19. Screen

Input:

19.1	Service Support Data:	Screen List Indicator	=	(UPT-service, home, procedure) * list at home database checked for authorization of registration *\
	CIDFP-Screen			
	CIDFP-Error			
19.2	Call Instance Data:	Identifier	=	RegistrationId (= 2)

Output: See SIB 3

20. User Interaction

Input:

20.1	Service Support Data:			
	Announcement Parameters	Announcement ID	=	"Identify terminal"
		Repetition Requested	=	No
	Collect Information Parameters	User Interruptibility	=	Yes
		Voice Feedback	=	Yes
		Type	=	DTMF
		Maximum number of characters	=	13
		Minimum number of characters	=	4
		Initial Input waiting timer	=	10
		Inter-character waiting timer	=	5
		End delineator	=	#
	CIDFP-Call party		=	
	CIDFP-Collected			
	CIDFP-Error			
20.2	Call Instance Data:	Call Party Identifier	=	CLI

Output:

20.3	Logical End:			See SIB 1
20.4	Call Instance Data:	Collected Data	=	CurrentAddress Error Cause (see Recommendation Q.1213)

21. User Interaction

Input:

21.1	Service Support Data:			
	Announcement Parameters	Announcement ID	=	"Please acknowledge terminal identity"
		Repetition Requested	=	No
	Collect Information Parameters	User Interruptibility	=	Yes
		Voice Feedback	=	No
		Type	=	DTMF
		Maximum number of characters	=	1
		Minimum number of characters	=	1
		Initial Input waiting timer	=	10
		Inter-character waiting timer	=	NULL
		End delineator	=	NULL
	CIDFP-Call party			
	CIDFP-Collected			
	CIDFP-Error			
21.2	Call Instance Data:	Call Party Identifier	=	CLI

Output:

21.3	Logical End:			See SIB 1
21.4	Call Instance Data:	Collected Data	=	UserAcknowledgement * either # : terminal correct or *: terminal false *\ Error Cause (see Recommendation Q.1213)

22. Verify

Input:

22.1	Service Support Data:	Maximum number of characters	=	1
		Minimum number of characters	=	1
		Format	=	D * any digit (0-9) or delineator (#, *) *\
	CIDFP-Data			
	CIDFP-Error			
22.2	Call Instance Data:	Identifier	=	UserAcknowledgement

Output: See SIB 11

23. Compare

Input:

23.1	Service Support Data:	Comparison Type	=	Identifier Value
	CIDFP-Compare			
		Reference Value	=	#
	CIDFP-Error			
23.2	Call Instance Data:	Identifier	=	UserAcknowledgement

Output: See SIB 5

24. Service Data Management

Input:

24.1	Service Support Data:	File Indicator	=	UPT-user data file
		Action	=	replace
		Element Indicator	=	(Registered data field, CurrentAddress data field)
	CIDFP-Info			
	CIDFP-Error			
24.2	Call Instance Data:	Information value	=	CurrentAddress

Output: See SIB 7

25. User Interaction

Input:

25.1	Service Support Data:			
	Announcement Parameters	Announcement ID	=	“Your request has been processed. Identify new request or terminate”
		Repetition Requested	=	No
	Collect Information Parameters	User Interruptibility	=	Yes
		Voice Feedback	=	No
		Type	=	DTMF
		Maximum number of characters	=	1
		Minimum number of characters	=	1
		Initial Input waiting timer	=	5
		Inter-character waiting timer	=	NULL
	CIDFP-Call party CIDFP-Collected CIDFP-Error		=	
25.2	Call Instance Data:	Call Party Identifier	=	CLI

Output:

25.3	Logical End:			See SIB 1
25.4	Call Instance Data:	Collected Data	=	ProcedureId Error Cause (see Recommendation Q.1213)

26. Verify

Same as SIB 10 with	Format	=	N * any digit (0-9) *\

27. Compare

Same as SIB 11 with	Reference Value	=	0

28. User Interaction

Input:

28.1	Service Support Data:			
	Announcement Parameters	Announcement ID	=	“Your request cannot be processed. Identify new request or terminate”
		Repetition Requested	=	No
	Collect Information Parameters	User Interruptibility	=	Yes
		Voice Feedback	=	No
		Type	=	DTMF
		Maximum number of characters	=	1
		Minimum number of characters	=	1
		Initial Input waiting timer	=	10
		Inter-character waiting timer	=	NULL
	CIDFP-Call party CIDFP-Collected CIDFP-Error		=	
28.2	Call Instance Data:	Call Party Identifier	=	CLI

Output:

28.3	Logical End:			See SIB 1
28.4	Call Instance Data:	Collected Data	=	ProcedureId Error Cause (see Recommendation Q.1213)

Management

29. Service Data Management

Input:

29.1	Service Support Data:	File Indicator	=	UPT-user data file
		Action	=	increment
		Element Indicator	=	Account data field
		Inc/Dec Value	=	Charge for UPT-user
	CIDFP-Error			
29.2	Call Instance Data:		=	NULL

Output: See SIB 7

3.2.2) Global Service Logic

Output of BCP	Point of Initiation	=	Address Analysed
CIDFP-CLI			
CIDFP-Dialled			
Input of BCP	Point of Return 1	=	Clear Call
	Point of Return 2	=	Clear Call
	Point of Return 3	=	Clear Call
	Point of Return 4	=	Clear Call
	Point of Return 5	=	Clear Call
	Point of Return 6	=	Clear Call
	Point of Return 7	=	Proceed with new data
		=	(Destination Number)

3.3) UPT Incall

3.3.1) Service Independent Building Blocks description

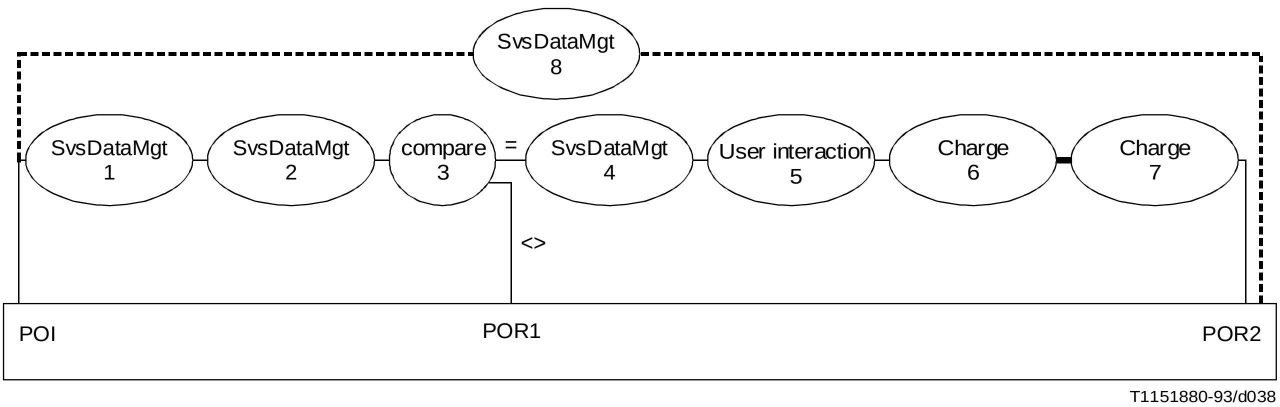


FIGURE A.5-2/Q.1219

SIB description of UPT Incall

1. Service Data Management

Input:

1.1	Service Support Data:	File Indicator	=	UPT-database(local) directory
		Action	=	retrieve
	CIDFP-Element			
	CIDFP-Error			
1.2	Call Instance Data:	Element Indicator	=	UPT-number

Output:

1.3	Logical End:			Success Error
1.4	Call Instance Data:	Data retrieved	=	UPT-database(home) location Error Cause

2. Service Data Management

Input:

2.1	Service Support Data:	File Indicator	=	UPT-user data file
		Action	=	retrieve
		Element Indicator	=	Registered data field
	CIDFP-Error			
2.2	Call Instance Data:			NULL

Output:

2.3	Logical End:			Success Error
2.4	Call Instance Data:	Data retrieved	=	Registered value Error Cause

3. Compare

Input:

3.1	Service Support Data:	Comparison Type	=	Identifier Value
		CIDFP-Compare		
		Reference value	=	true
	CIDFP-Error			
3.2	Call Instance Data:	Identifier	=	Registered value

Output:

3.3	Logical End:			GREATER THAN (>) LESS THAN (<) EQUAL TO (=) Error
3.4	Call Instance Data:			Error Cause (see Recommendation Q.1213)

4. Service Data Management

Input:

4.1	Service Support Data:	File Indicator	=	UPT-user data file
		Action	=	retrieve
		Element Indicator	=	CurrentAddress data field
	CIDFP-Error			
4.2	Call Instance Data:			Null

Output:

4.3	Logical End:			Success Error
4.4	Call Instance Data:	Data retrieved	=	Current address Error Cause

5. User Interaction

Input:

5.1	Service Support Data:			
	Announcement Parameters	Announcement ID	=	“UPT charging is applicable”
		Repetition Requested	=	No
		Collect Information Parameters	=	NULL
	CIDFP-Call party			
	CIDFP-Error			
5.2	Call Instance Data:	Call Party Identifier	=	CLI

Output:

5.3	Logical End:			Success Error
5.4	Call Instance Data:			Error Cause (see Recommendation Q.1213)

6. Charge

/* The accounting for calling a UPT-user is supposed to be done off-line, as the Charge SIB specified in Recommendation Q.1213 is not equipped to this end. We assume that no Pulse metering applies and no percentage is identified in this SIB. */

Input:

6.1	Service Support Data:	Number of accounts to charge	=	2
	Account 1: (UPT user)	CIDFP-Account	=	UPT-number
		Percentage	=	NULL * UPT charging is not proportional hence this SSD is not relevant *\
	Account 2: (caller of UPT user)	CIDFP-Line or CIDFP-Account		* dependent on CID: is the caller a UPT-user or not *\
		Percentage	=	NULL
		Resource type	=	bearer type
		Units	=	1
		Service/Service Feature Identifier	=	UPT
	CIDFP-Error			
6.2	Call Instance Data:	Account(s)	=	UPT-number(s)
		Line	=	UPT-caller CLI
		Pulse Metering	=	NULL

Output:

6.3	Logical End:			Success Error
6.4	Call Instance Data:			Error Cause (see Recommendation Q.1213)

7. Charge

Input:

7.1	Service Support Data:	Number of accounts to charge	=	1
	Account 1: (caller of UPT user)	CIDFP-Line or CIDFP-Account		* dependent on CID: is the caller a UPT-user or not *\
		Percentage	=	NULL
		Resource type	=	SDF * caller is charged for SDF usage *\
		Units	=	SDFunit * some operator specific unit *\
		Service/Service Feature Identifier	=	UPT
	CIDFP-Error			
7.2	Call Instance Data:	Account(s)	=	UPT-number(s)
		Line	=	UPT-caller CLI
		Pulse Metering	=	NULL

Output: See SIB 7 (Outcall)

8. Service Data Management

Input:

8.1	Service Support Data:	File Indicator	=	UPT-user data file
		Action	=	increment
		Element Indicator	=	Account data field
		Inc/Dec Value	=	Charge for UPT-user
	CIDFP-Error			
8.2	Call Instance Data:			NULL

Output:

8.3	Logical End:			Success Error
8.4	Call Instance Data:			Error Cause

3.3.2) Global Service Logic

Output of BCP	Point of Initiation	=	Address Analysed
CIDFP-CLI CIDFP-Dialled			
Input of BCP	Point of Return 1	=	Release Call
	Point of Return 2	=	Proceed with new data (Destination Number)

4) Distributed view

A) Call Segment Diagram

1) "UPT" outcall/ (de)registration

In Figure A.5-3 the call segment diagram for the "UPT" outcall/ (de)registration are shown. The shaded call segment is the only call segment viewed by the SCF for this service.

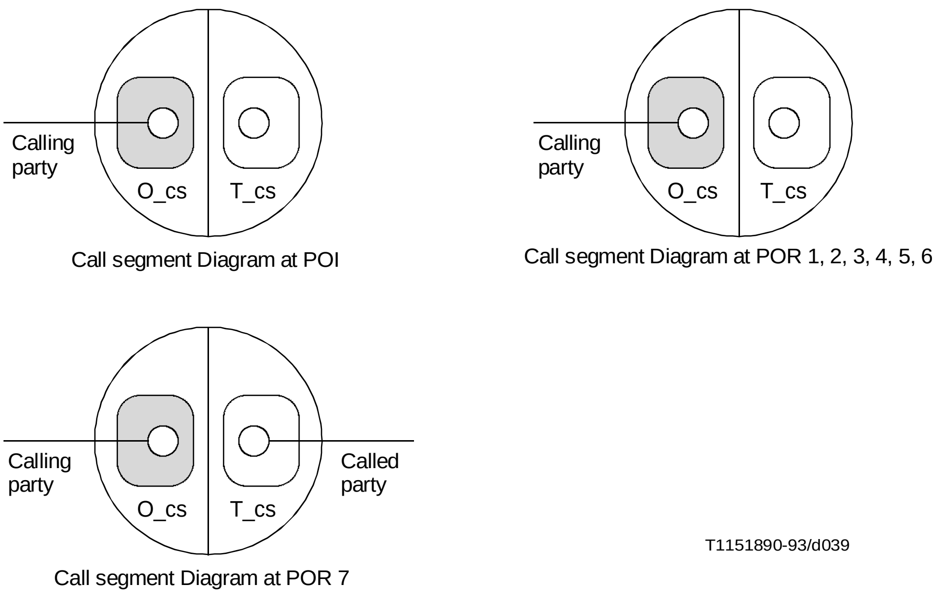


FIGURE A.5-3/Q.1219

Call segment diagrams for "UPT" outcall/(de)registration

2) "UPT" incall

In Figure A.5-4 the call segment diagram for the "UPT" incall are shown. The shaded call segment is the only call segment viewed by the SCF for this service.

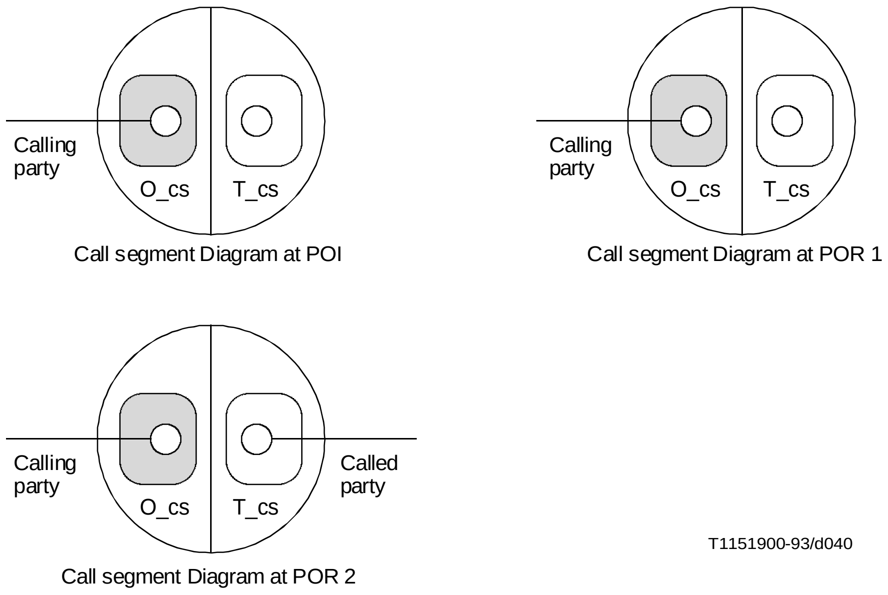


FIGURE A.5-4/Q.1219

Call segment diagrams for "UPT" incall

B) FE relationship diagram (IN relations only)

In Figure A.5-5 the Functional Entity relationships are shown.

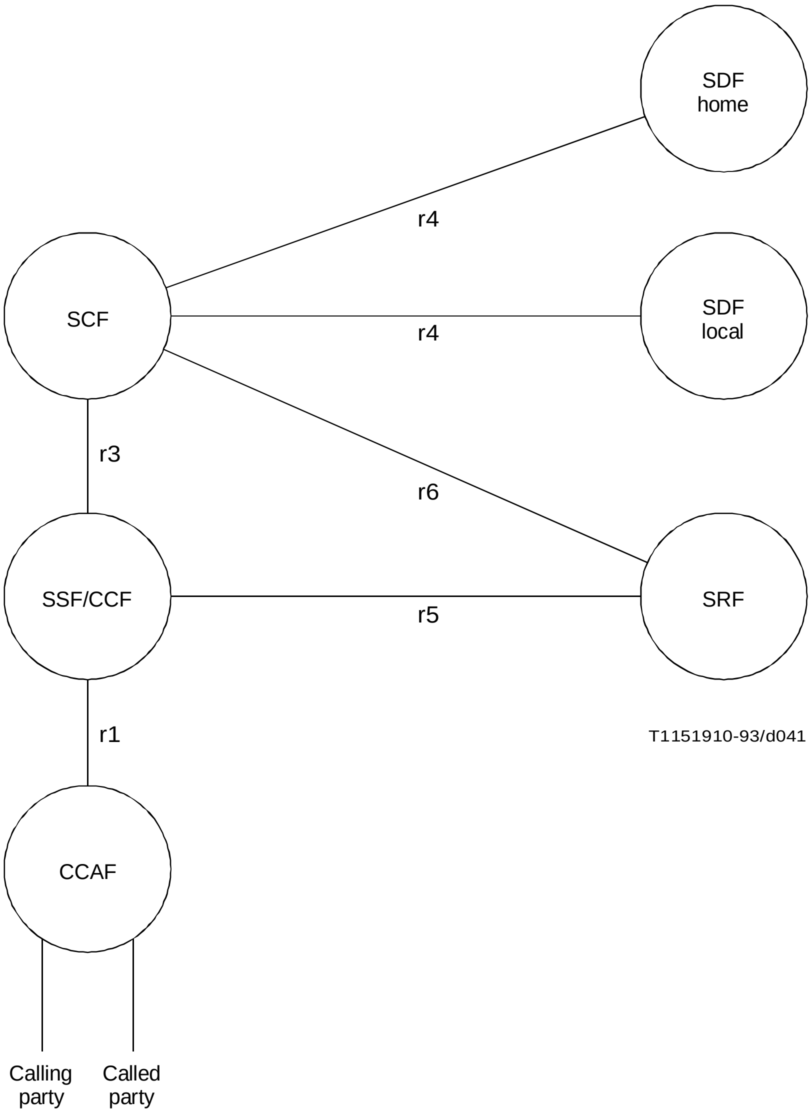


FIGURE A.5-5/Q.1219
FE relationship diagram (IN relations only)

C) Information flow diagram (IN IFs only)

The IFs of the “UPT” service are given below. Because the “UPT” service is an SDF intensive service, the pseudo PASCAL description of the structure of the data contained in the SDFs (local and home) is given below.

```
"UPT" Home SDF:             LIST OF Home Database Record;

Home Database Record:  STRUCTURE (
                               "UPT" Number:          STRING [ INTEGER (0:12) ] OF
                                                      Digits,
                               PIN:                   STRING (0:3) OF Digits,
                               Authorised service:    STRUCTURE (
                                                           Registration:   BOOLEAN,
                                                           Outcall:        BOOLEAN,
                                                           Incall:         BOOLEAN ),
                               Destination Number:    SET ( normal number, premium rate number ),
                               Current Location:      Routing Address,
                               Registered:            BOOLEAN,
                               Line Status:           SET ( Blocked, Normal ),
                               Timeout Count:         INTEGER,
                               Billing Record:        LIST OF Call Record );

Routing Address:            STRING [ INTEGER (0:12) ] OF Digits;

Call Record:                STRUCTURE (
                               Call connect elapsed time:   DURATION,
                               Call stop time:              TIME,
                               Calling Address:             Routing Address,
                               Called Address:              Routing Address );

"UPT" Local SDF:            LIST OF Local Database Record;

Local Database Record:      STRUCTURE (
                            "UPT" Number:             STRING [ INTEGER (0:12) ] OF Digits,
                            Supported Services:       STRUCTURE (
                                                           Registration:   BOOLEAN,
                                                           Outcall: BOOLEAN );
                            Home Location:            Database ID );
```

Figure A.5-6 shows the 'roadmap' the "UPT" IFs. The IFs are shown in the subsequent figures. The IFs and IEs are defined following the syntax stated below.

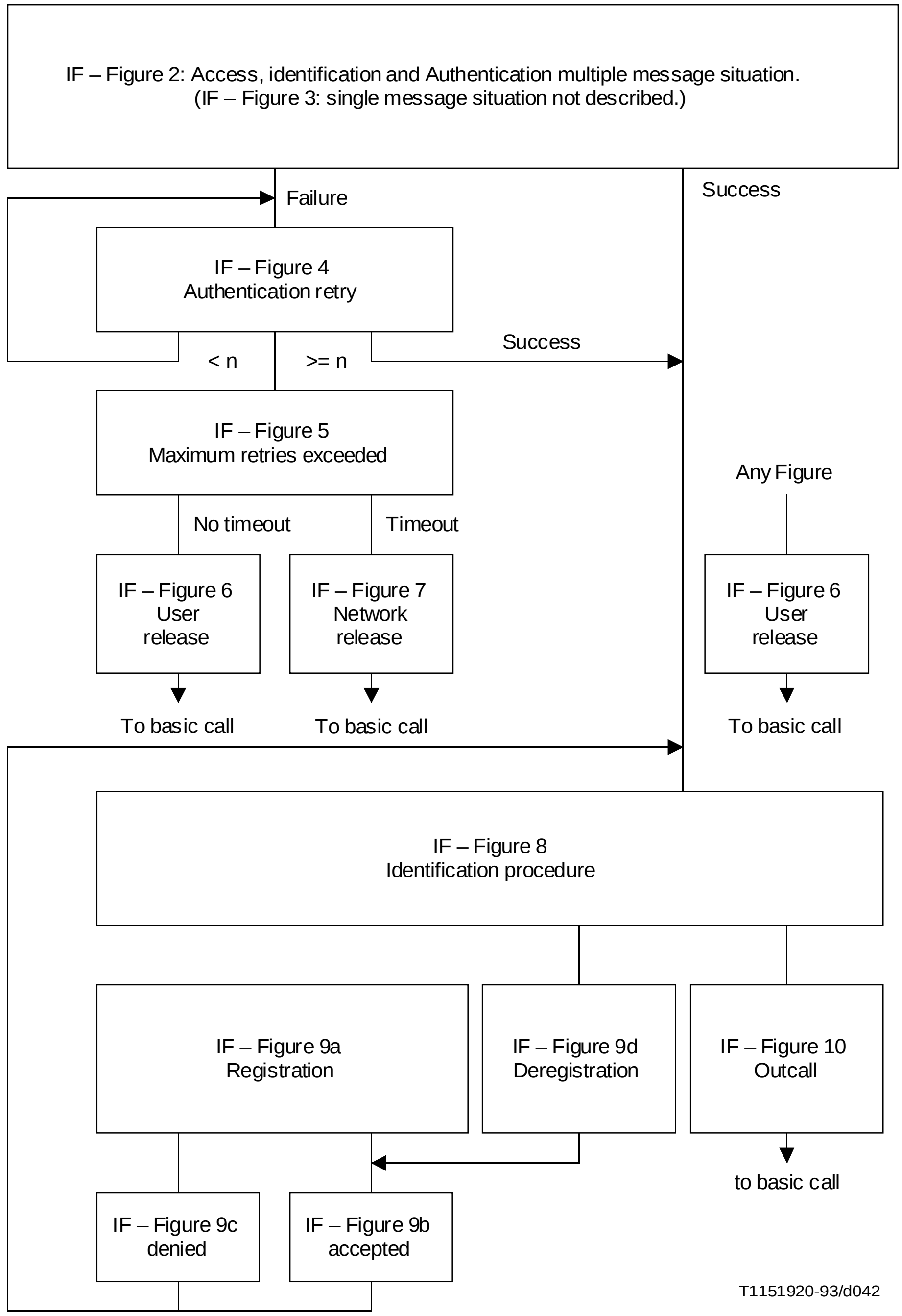


FIGURE A.5-6/Q.1219

IF – Figure 1: "Roadmap" of "UPT" Information Flows

Remarks on syntax of IFs and IEs:

1) The IFs are given according to the following scheme:

IF id = Information Flow type name:

Mandatory:	IE name	=	IE value;
	IE name	=	IE value;
Optional:	IE name	=	IE value;
	IE name	=	IE value.

2) The values of the IEs are given in the format ‘IE name = IE value’ unless ‘IE value’ has more than one component (e.g. Information to send is an IE with components Duration, Interval, Message ID, etc.). In the latter case the IE name is given in the format ‘IE1.Component 1 = value 1, IE1.Component 2 = value 2, etc’. In case of composed IEs, combinations of components are indicated between brackets [ ] (e.g. IE exists of Components 1 and 2 instead of the above indicated notation the following shorthand is used: ‘IE1.[Component 1 : value 1, Component 2 : value 2]’).

3) All type identifiers are denoted using a capital as the first letter while all variables are denoted using a small letter for the first letter of the name (e.g. ‘upt number’ of type ‘UPT Number’).

4) Only the relevant IE values are shown. Other IEs are not processed by the application, and are absent in the IFs.

Figure A.5-7 describes how a “UPT”-user gets access to the “UPT”- service by dialling a special service code and identifying himself in a prompt and collect sequence by his “UPT”-number and authentication code. The situation using a single message, as in Figure 3/Q.76 is not shown. For ease of cross referencing between documents the numbering in this Recommendation has not been adjusted. Thus IF – Figure 3 [Access, Identification, Authentication (using single message)] is not shown. In IF – Figure 2 the following Information Flows are used:

IDP1 = Initial DP:

Mandatory:	Call ID	=	A′ call,
	Service key	=	UPT1 (this Recommendation version),
Optional:	Terminal Type[6)]	=	DTMF phone.

RRBE1 = Request Report BCSM Event[7)]:

Mandatory:	BCSM Event List[1].Event Type	=	DP 10,
	BCSM Event List[1].Monitor Mode	=	Notify and Continue,
	Call ID	=	A′ call.

CTR1 = Connect To Resource[8)]:

Mandatory:	Call ID	=	A′ call,
Optional:	IP Routing Address	=	IP Line 1.

PCUI1 = Prompt and Collect User Information:

Mandatory:	Collected Info.Maximum Number of Digits	=	12,
	Collected Info.Minimum Number of Digits	=	4,
Optional:	Collected Info.End Of Reply Digit	=	#,
	Collected Info.Error Treatment	=	DEFAULT (i.e. send any collected digits),
	Collected Info.First Digit Timeout	=	10 seconds,
	Collected Info.Inter Digit Timeout	=	5 seconds,
Mandatory:	Disconnect from IP forbidden	=	TRUE,
Optional:	Information to Send.Elementary message ID	=	Message 1 (i.e. “Provide your identity”),
	Interruptable Announcement Indicator	=	TRUE,
Mandatory:	SRF Connect ID	=	SRF 1.

---

6) In this service example the SCF is assumed to have knowledge of SSF/ SRF capabilities. An SRF is default present and co-located with the SSF.

7) The basic call process is requested to report call abandoning by UPT-user, in order to stop SCF processing.

8) This is only one of the ways to handle user interaction. The other is to use an Establish Temporary Connection IF and an Assist Request Instructions IF (see 3.1.3.5.1/Q.1218).

CUI1 = Collected User Information:				
	Mandatory:	SRF Connect ID	=	SRF 1,
		Received Info	=	upt number (i.e. d_1 d_2 d. . .d_{11} d_{12}).
PCUI2 = Prompt and Collect User Information:				
	Mandatory:	Collected Info.Maximum Number of Digits	=	4,
		Collected Info.Minimum Number of Digits	=	4,
	Optional:	End Of Reply Digit	=	#,
		Collected Info.Error Treatment	=	DEFAULT (i.e. send any collected digits),
		Collected Info.First Digit Timeout	=	10 seconds,
		Collected Info.Inter Digit Timeout	=	5 seconds,
	Mandatory:	Disconnect from IP forbidden	=	TRUE,
	Optional:	Information to Send.Elementary message ID	=	Message 2 (i.e. "Provide your authentication code"),
		Interruptable Announcement Indicator	=	TRUE,
	Mandatory:	SRF Connect ID	=	SRF 1.
CUI2 = Collected User Information:				
	Mandatory:	SRF Connect ID	=	SRF 1,
		Received Info	=	pin (i.e. d_{13} d_{14} d_{15} d_{16}).
Qy1 = Query:				
	Optional:	Database ID	=	"UPT" Local SDF 1,
	Mandatory:	Information Key	=	["UPT" Number:upt number],
	Optional:	Requested Info Type	=	[Home Location, Supported Services][9].
QyR1 = Query Result:				
	Mandatory:	Requested Info	=	[Home Location: "UPT" Home SDF 1, Supported Services. Registration: TRUE / FALSE, Supported services. Outcall: TRUE / FALSE].
Qy2 = Query:				
	Optional:	Database ID	=	"UPT" Home SDF 1,
	Mandatory:	Information Key	=	["UPT" Number: "UPT" number, PIN: pin],
	Optional:	Requested Info Type	=	Information Present[10].
QyR2 = Query Result:				
	Mandatory:	Requested Info.Information Present	=	TRUE / FALSE.

[9] The capability to support the requested UPT service by the originating network is retrieved. This is different from the SIB description where Screen SIBs are used each time the user places a request to optimize information flow efficiency.

[10] The collected digits representing UPT Number and PIN are checked for validity and a result containing TRUE or FALSE is sent back dependent on the outcome of the test.

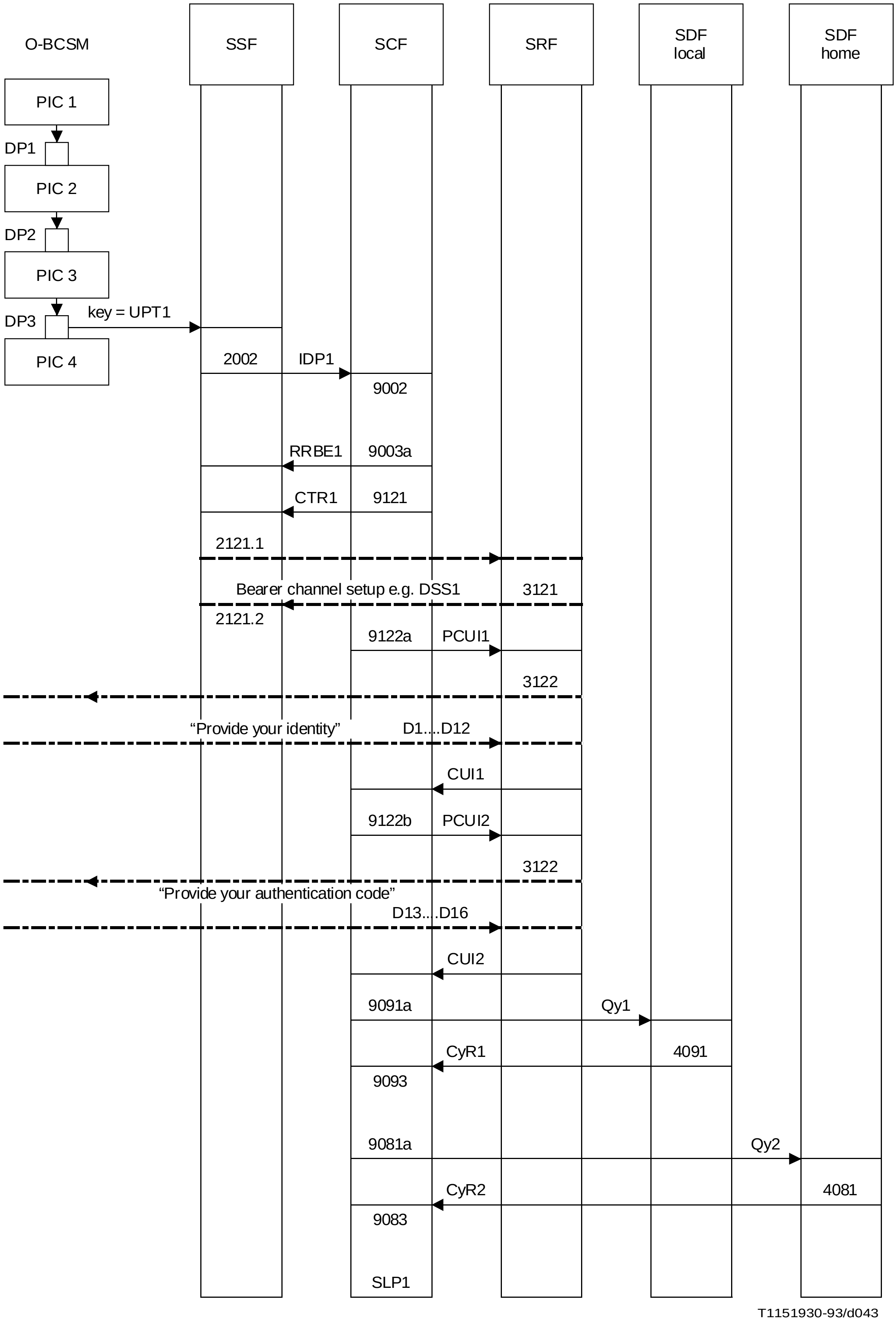


FIGURE A.5-7/Q.1219

IF – Figure 2: Access, Identification, Authentication (using multiple messages)

IF – Figure 4 contains nearly the same flows and information elements as IF – Figure 2. Only the deviating IFs are treated below, the other ones correspond directly to the IF identifiers given above.

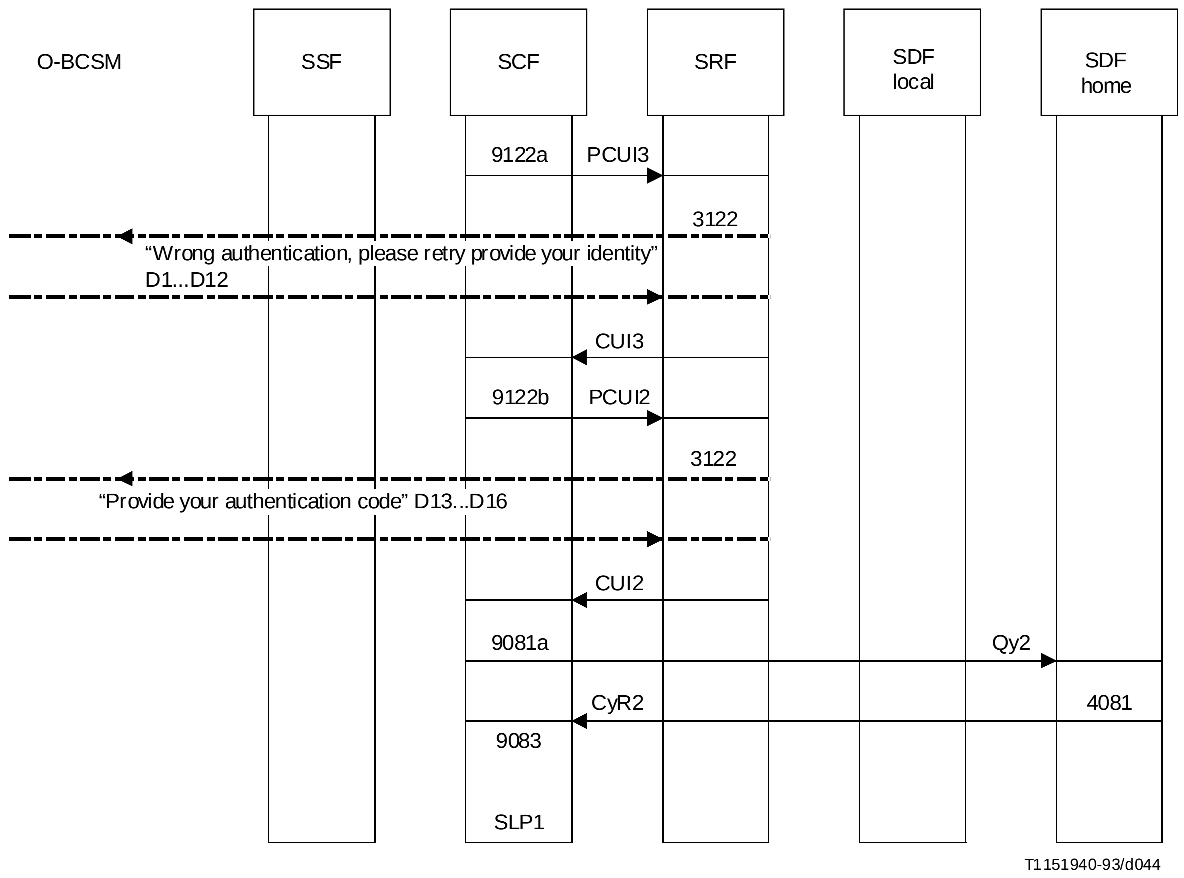


Figure A.5-8/Q.1219

IF – Figure 4: Authentication retry

PCUI3 = Prompt and Collect User Information:

Mandatory:	Collected Info.Maximum Number of Digits	=	12,
	Collected Info.Minimum Number of Digits	=	4,
Optional:	End Of Reply Digit	=	#,
	Collected Info.Error Treatment	=	DEFAULT (i.e. send any collected digits),
	Collected Info.First Digit Timeout	=	10 seconds,
	Collected Info.Inter Digit Timeout	=	5 seconds,
Mandatory:	Disconnect from IP forbidden	=	TRUE.
Optional:	Information to Send.Elementary message ID	=	Message 3 (i.e. “Wrong authentication: Please retry, provide your identity”),
	Interruptable Announcement Indicator	=	TRUE,
Mandatory:	SRF Connect ID	=	SRF 1.

CUI3 = Collected User Information:

Mandatory:	SRF Connect ID	=	SRF 1,
	Received Info	=	"upt" number (i.e. d_1 d_2 d...d_{11} d_{12}).

IF – Figure 5 shows treatment of the error situation that the maximum number of authentication retries is exceeded and the "UPT"-number is blocked. The "UPT"-database is updated to effectuate the blocking of the "UPT"-number.

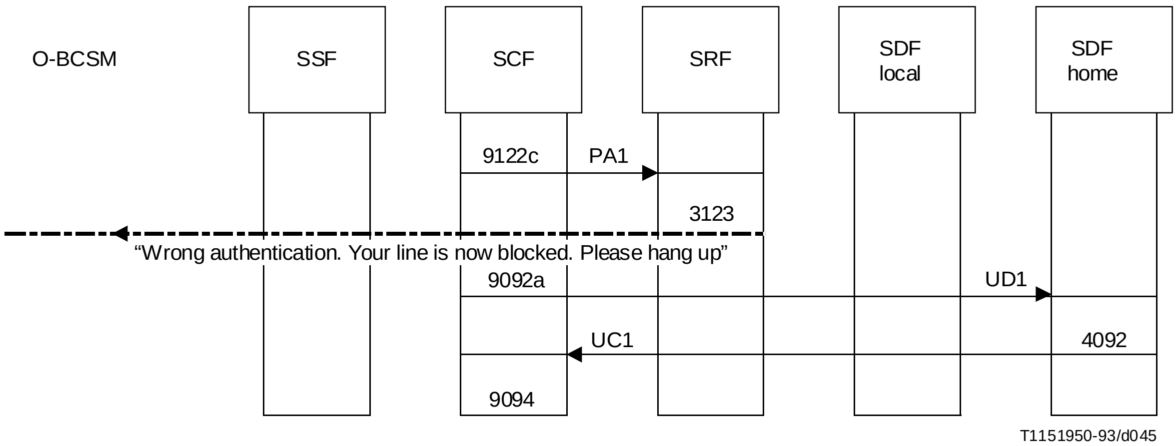


FIGURE A.5-9/Q.1219

IF – Figure 5: Maximum authentication retries exceeded

PA1 = Play Announcement:

Mandatory:	Disconnect from IP forbidden	=	FALSE,
	Information To Send.Elementary Message ID	=	Message 4 (i.e. "Wrong authentication. Your number is now blocked. Please hang up."),
	Request Announcement Completed	=	TRUE,
	SRF Connect ID	=	SRF 1.

UD1 = Update Data:

Optional:	Database ID	=	"UPT" Home SDF 1,
	Function Type	=	Replace,
Mandatory:	Information Key	=	"UPT" Number: "UPT" number,
	Updated Info	=	Line Status: Blocked.

UC1 = Update Confirmation:

Mandatory:	Outcome	=	Success[11].

[11] Failure situations of equipment are not shown (i.e. the outcome "Failure" due to database hardware/software failures is not treated here).

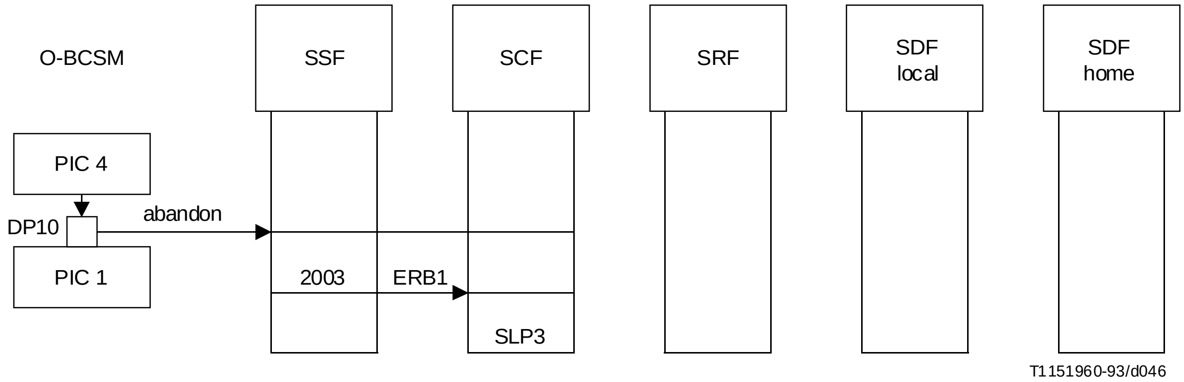


Figure A.5-10/Q.1219

IF – Figure 6: Release of calling user (caller initiated)

ERB1 = Event Report BCSM:

Mandatory:	Call ID	=	A' call,
	Event Type BCSM	=	DP 10,
Optional:	Leg ID	=	A' party.

IF – Figure 7 shows the situation that a prompt and collect sequence fails to provide the information necessary to process the call, due to a timeout. This situation is recorded in the "UPT"-database.

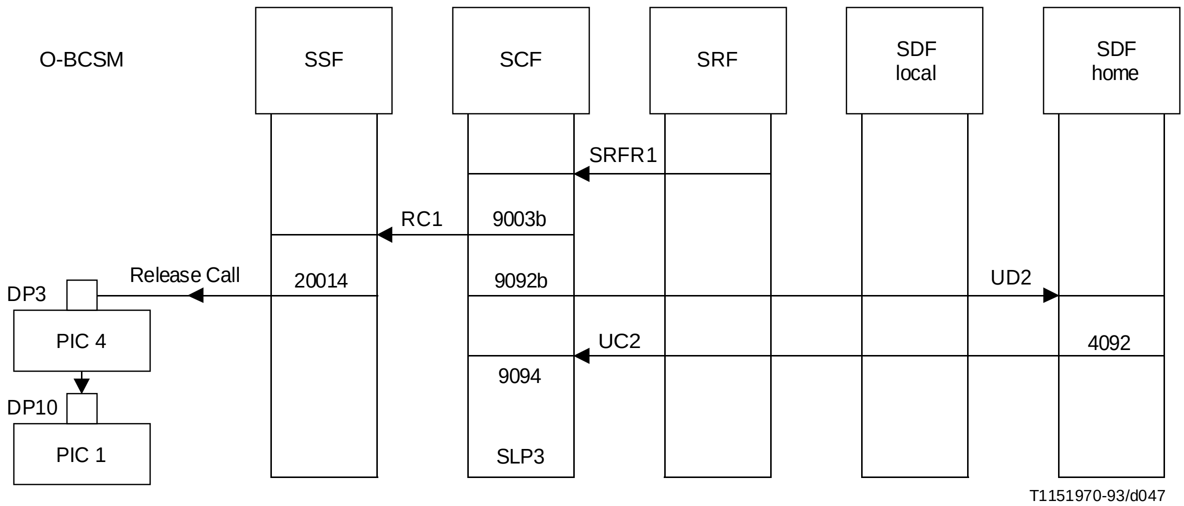


Figure A.5-11/Q.1219

IF – Figure 7: Release of calling user (network initiated)

SRFR1 = Specialized Resource Function Report[12]:
Mandatory: SRF Connect ID = SRF 1.

RC1 = Release Call[13]:
Mandatory: Call ID = A' call,
Optional: Cause = Timeout.

UD2 = Update Data:
Optional: Database ID = "UPT" Home SDF 1,
Function Type = Increment,
Mandatory: Information Key = "UPT" Number: "UPT" number,
Updated Info = Timeout Count[14]: 1.

UC2 = Update Confirmation:
Mandatory: Outcome = Success[15].

IF – Figure 8 shows how a "UPT"-user can identify the required procedure (registration, deregistration, outgoing call). The query is intended to check whether the "UPT"-user is allowed by his own network to use the requested procedure.

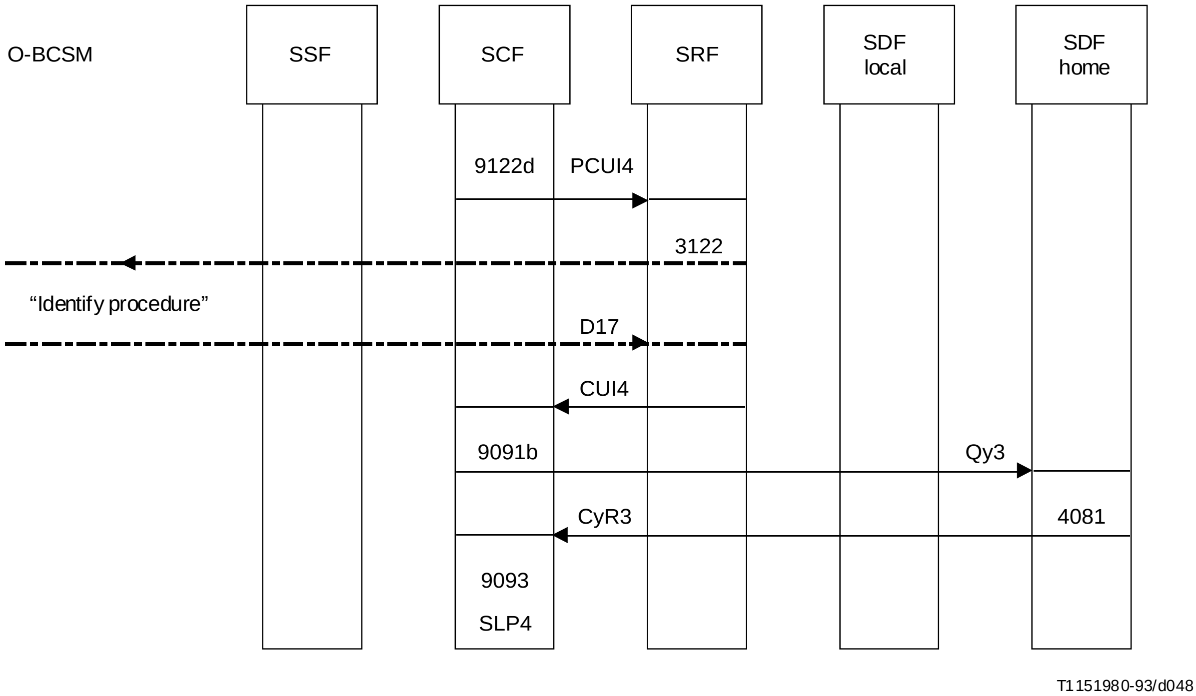


FIGURE A.5-12/Q.1219

IF – Figure 8: Procedure identification

---

12) This report is the response to the 'Request Announcement Completed' Information Element in the PA1 IF.

13) The RC1 IF will automatically release the SRF connection, so no separate Disconnect Forward Connection is needed.

14) Timeout event is recorded in UPT database.

15) Only successful situations are treated here.

PCUI4 = Prompt and Collect User Information:

Mandatory:	Collected Info.Maximum Number of Digits	=	1,
	Collected Info.Minimum Number of Digits	=	1,
Optional:	Collected Info.Error Treatment	=	DEFAULT (i.e. send any collected digits),
	Collected Info.First Digit Timeout	=	10 seconds,
Mandatory:	Disconnect from IP forbidden	=	TRUE.
Optional:	Information to Send.Elementary message ID	=	Message 5 (i.e. "Identify procedure"),
	Interruptable Announcement Indicator	=	TRUE,
Mandatory:	SRF Connect ID	=	SRF 1.

CUI4 = Collected User Information:

Mandatory:	SRF Connect ID	=	SRF 1,
	Received Info	=	selection digit (i.e. d_{17}).

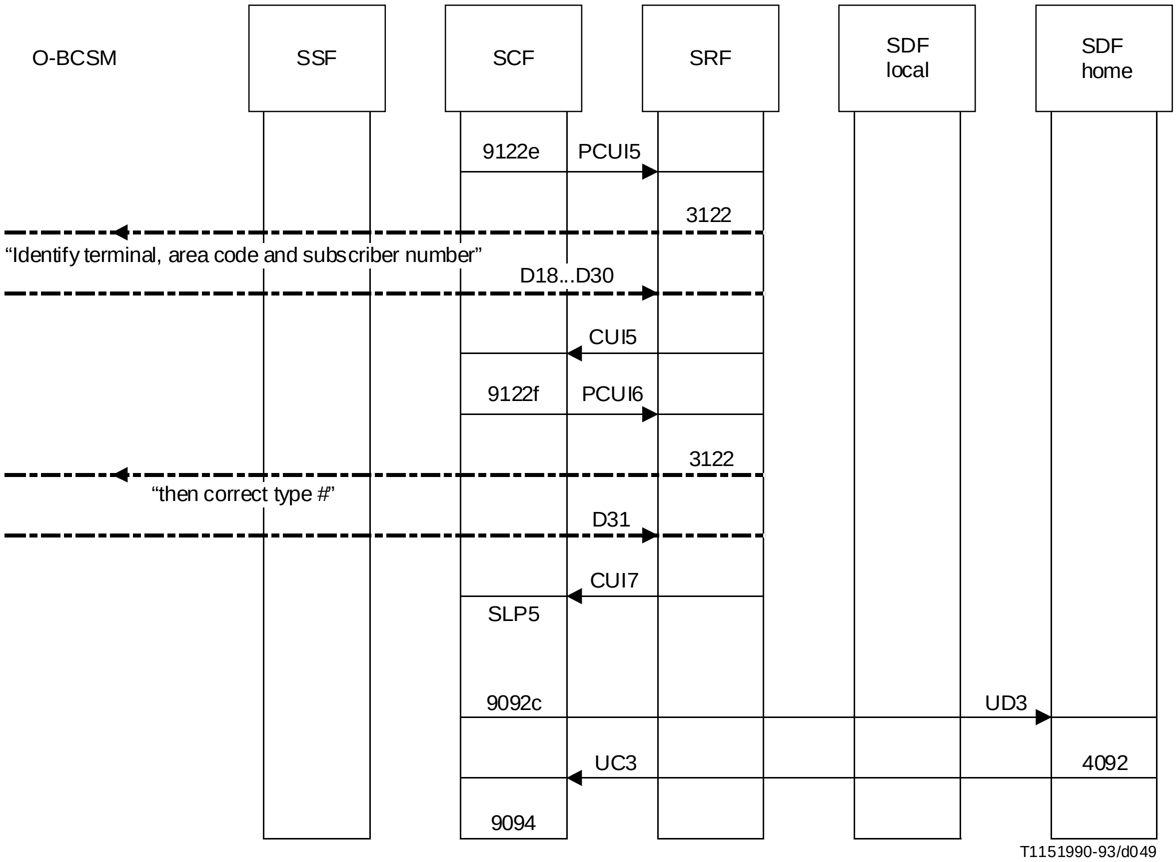


FIGURE A.5-13/Q.1219

IF – Figure 9a: Incall registration

Qy3 =	Query:		
Optional:	Database ID	=	"UPT" Home SDF 1,
Mandatory:	Information Key	=	"UPT" Number: "UPT" number,
Optional:	Requested Info Type	=	Authorized Services.
QyR3 =	Query Result:		
Mandatory:	Requested Info	=	Authorized Services. [Registration: TRUE/FALSE, Outcall: TRUE/FALSE, Incall: TRUE/FALSE].
PCUI5 =	Prompt and Collect User Information:		
Mandatory:	Collected Info.Maximum Number of Digits	=	12,
	Collected Info.Minimum Number of Digits	=	5,
Optional:	End Of Reply Digit	=	#,
	Collected Info.Error Treatment	=	DEFAULT (i.e. send any collected digits),
	Collected Info.First Digit Timeout	=	10 seconds,
	Collected Info.Inter Digit Timeout	=	5 seconds,
Mandatory:	Disconnect from IP forbidden	=	TRUE.
Optional:	Information to Send.Elementary message ID	=	Message 6 (i.e. "Identify Terminal"),
	Interruptable Announcement Indicator	=	TRUE,
Mandatory:	SRF Connect ID	=	SRF 1,
Optional:	Voiceback	=	TRUE.
CUI5 =	Collected User Information:		
Mandatory:	SRF Connect ID	=	SRF 1,
	Received Info	=	terminal identity (i.e. $d_{18} \ldots d_{30}$).
PCUI6 =	Prompt and Collect User Information:		
Mandatory:	Collected Info.Maximum Number of Digits	=	1,
	Collected Info.Minimum Number of Digits	=	1,
Optional:	Collected Info.Error Treatment	=	DEFAULT (i.e. send any collected digits),
	Collected Info.First Digit Timeout	=	10 seconds,
Mandatory:	Disconnect from IP forbidden	=	TRUE.
Optional:	Information to Send.Elementary message ID	=	Message 7 (i.e. "When correct type #"),
	Interruptable Announcement Indicator	=	TRUE,
Mandatory:	SRF Connect ID	=	SRF 1.
CUI6 =	Collected User Information:		
Mandatory:	SRF Connect ID	=	SRF 1,
	Received Info	=	acknowledge digit (i.e. d_{31}).
UD3 =	Update Data:		
Optional:	Database ID	=	"UPT" Home SDF 1,
	Function Type	=	Replace,
Mandatory:	Information Key	=	"UPT" Number: "UPT" number,
	Updated Info	=	[Current Location: terminal identity[16)], Registered: TRUE] .
UC3 =	Update Confirmation:		
Mandatory:	Outcome	=	Success[17)].

[16)] New location of UPT user is registered.

[17)] Only successful situations are treated here.

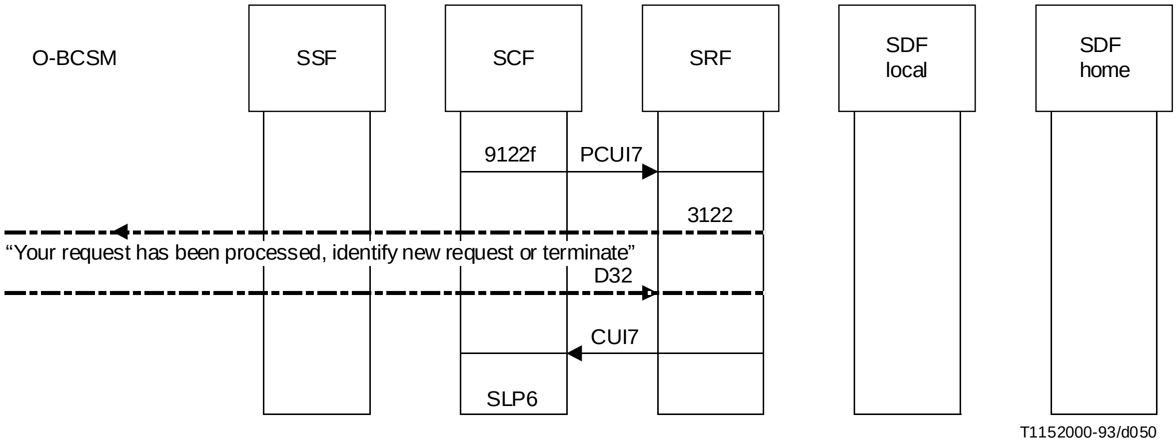


FIGURE A.5-14/Q.1219

IF – Figure 9b: Processing request accepted

PCUI7 = Prompt and Collect User Information:

Mandatory:	Collected Info.Maximum Number of Digits	=	1,
	Collected Info.Minimum Number of Digits	=	1,
Optional:	Collected Info.Error Treatment	=	DEFAULT (i.e. send any collected digits),
	Collected Info.First Digit Timeout	=	10 seconds,
Mandatory:	Disconnect from IP forbidden	=	TRUE.
Optional:	Information to Send.Elementary message ID	=	Message 8 (i.e. “Your request has been processed, identify new request or terminate”),
	Interruptable Announcement Indicator	=	TRUE,
Mandatory:	SRF Connect ID	=	SRF 1.

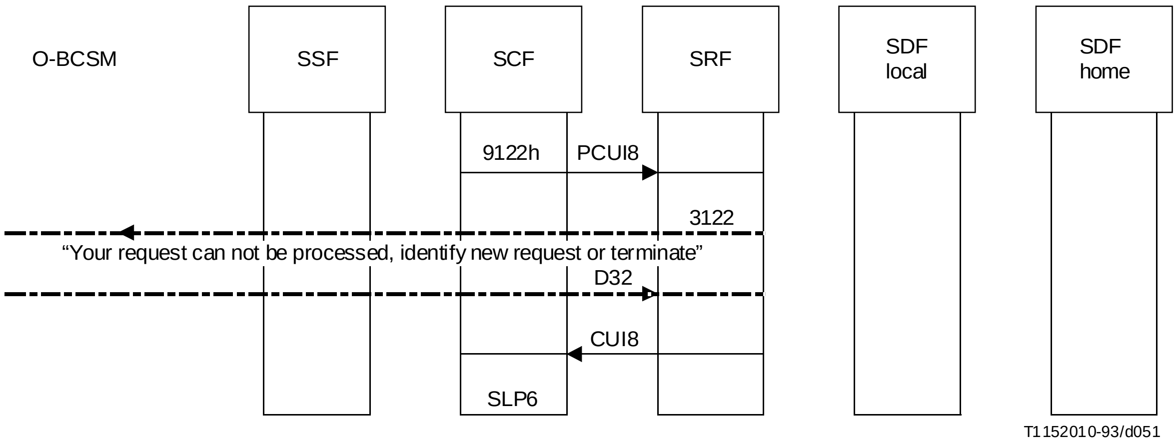


FIGURE A.5-15/Q.1219

IF – Figure 9c: Processing request denied

CUI7 = Collected User Information:

Mandatory:	SRF Connect ID	=	SRF 1,
	Received Info	=	selection digit (i.e. d_{32}).

PCUI8 = Prompt and Collect User Information:

Mandatory:	Collected Info.Maximum Number of Digits	=	1,
	Collected Info.Minimum Number of Digits	=	1,
Optional:	Collected Info.Error Treatment	=	DEFAULT (i.e. send any collected digits),
	Collected Info.First Digit Timeout	=	10 seconds,
Mandatory:	Disconnect from IP forbidden	=	TRUE.
Optional:	Information to Send.Elementary message ID	=	Message 9 (i.e. "Your request can not be processed, identify new request or terminate."),
	Interruptable Announcement Indicator	=	TRUE,
Mandatory:	SRF Connect ID	=	SRF 1.

CUI8 = Collected User Information:

Mandatory:	SRF Connect ID	=	SRF 1,
	Received Info	=	selection digit (i.e. d_{32}).

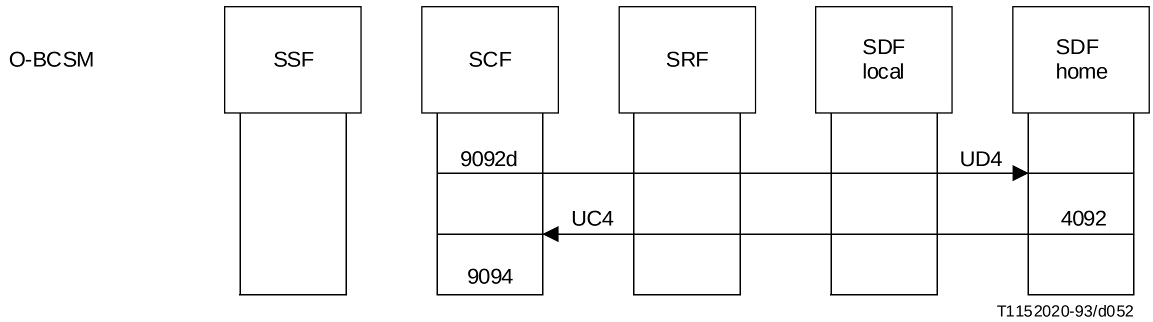


FIGURE A.5-16/Q.1219

IF – Figure 9d: Incall deregistration

UD4 = Update Data:

Optional:	Database ID	=	"UPT" Home SDF 1,
	Function Type	=	Replace,
Mandatory:	Information Key	=	"UPT" Number: "UPT" number,
	Updated Info	=	Registered: FALSE.

UC4 = Update Confirmation:

Mandatory:	Outcome	=	Success[18].

The Outcall shown in figure IF – Figure 10 only allows the "UPT"-user to place a single outcall. Other situations (e.g. follow-on calls) are not considered in this example. The charging of the "UPT"-user is done off-line using the Call Detail Record of the original call (A' call). Since charging is highly operator specific, this example is not elaborated any further with respect to charging.

[18] Only successful situations are treated here.

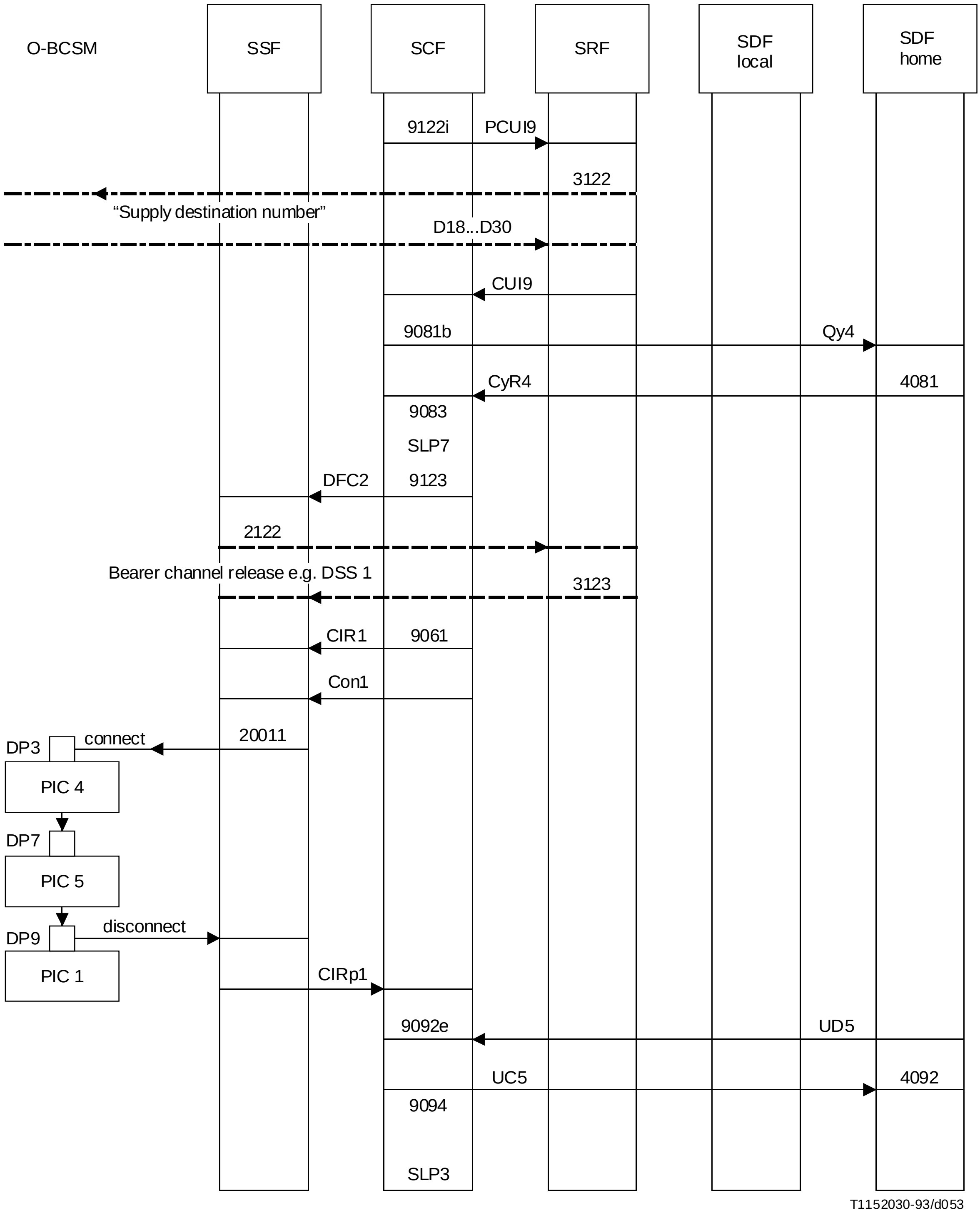


FIGURE A.5-17/Q.1219

IF – Figure 10: Single outcall

PCUI9 =	Prompt and Collect User Information:			
Mandatory:	Collected Info.Maximum Number of Digits	=	12,
	Collected Info.Minimum Number of Digits	=	5,
Optional:	End Of Reply Digit	=	#,
	Collected Info.Error Treatment	=	DEFAULT (i.e. send any collected digits),
	Collected Info.First Digit Timeout	=	10 seconds,
	Collected Info.Inter Digit Timeout	=	5 seconds,
Mandatory:	Disconnect from IP forbidden	=	TRUE,
Optional:	Information to Send.Elementary message ID	=	Message 10 (i.e. "Supply destination number."),
	Interruptable Announcement Indicator	=	TRUE,
Mandatory:	SRF Connect ID	=	SRF 1.

CUI9 =	Collected User Information:		
Mandatory:	SRF Connect ID	=	SRF 1,
	Received Info	=	destination number (i.e. $d_{18} \ldots d_{30}$).

Qy4 =	Query:		
Optional:	Database ID	=	"UPT" Home SDF 1,
Mandatory:	Information Key	=	"UPT" Number: "UPT" number. Authorized Services. Destination Number: destination number[19],
Optional:	Requested Info Type	=	Expected Screening result.

QyR4 =	Query Result:		
Mandatory:	Requested Info	=	TRUE.

DFC2 =	Disconnect Forward Connection:		
Mandatory:	Call ID	=	A' call.

Con1 =	Connect:		
Mandatory:	Call ID	=	A' call,
	Destination Routing Address	=	destination number.

CIR1 =	Call Information Request:		
Mandatory:	Requested Information	=	[Call connected elapsed time, Call stop time, Called address, Calling address].

CIRp1 =	Call Information Report:		
Mandatory:	Requested Information	=	Call Record[20].

UD5 =	Update Data:		
Optional:	Database ID	=	"UPT" Home SDF 1,
	Function Type	=	Increment[21],
Mandatory:	Information Key	=	"UPT" Number: "UPT" number,
	Updated Info	=	Billing Record. Call Record: call record.

UC5 =	Update Confirmation:		
Mandatory:	Outcome	=	Success[22].

Two queries are possible in IF – Figure 11; the first is to find out in the local database where the "UPT"-users home database is situated. The second asks the home database for the current location of the "UPT"-user. Only the successful situations are shown. The charging is treated similar to the situation described in the "UPT" Outcall using the existing call record.

---

[19] The destination number used to identify possible outcall limitiations (e.g. premium rate numbers).

[20] The "Call Record" is a structure containing the in IF CIR1 indicated elements.

[21] The semantics of the Increment value in this IE is as follows: Take the next free list entry and store data given in that location.

[22] Only successful situations are treated here.

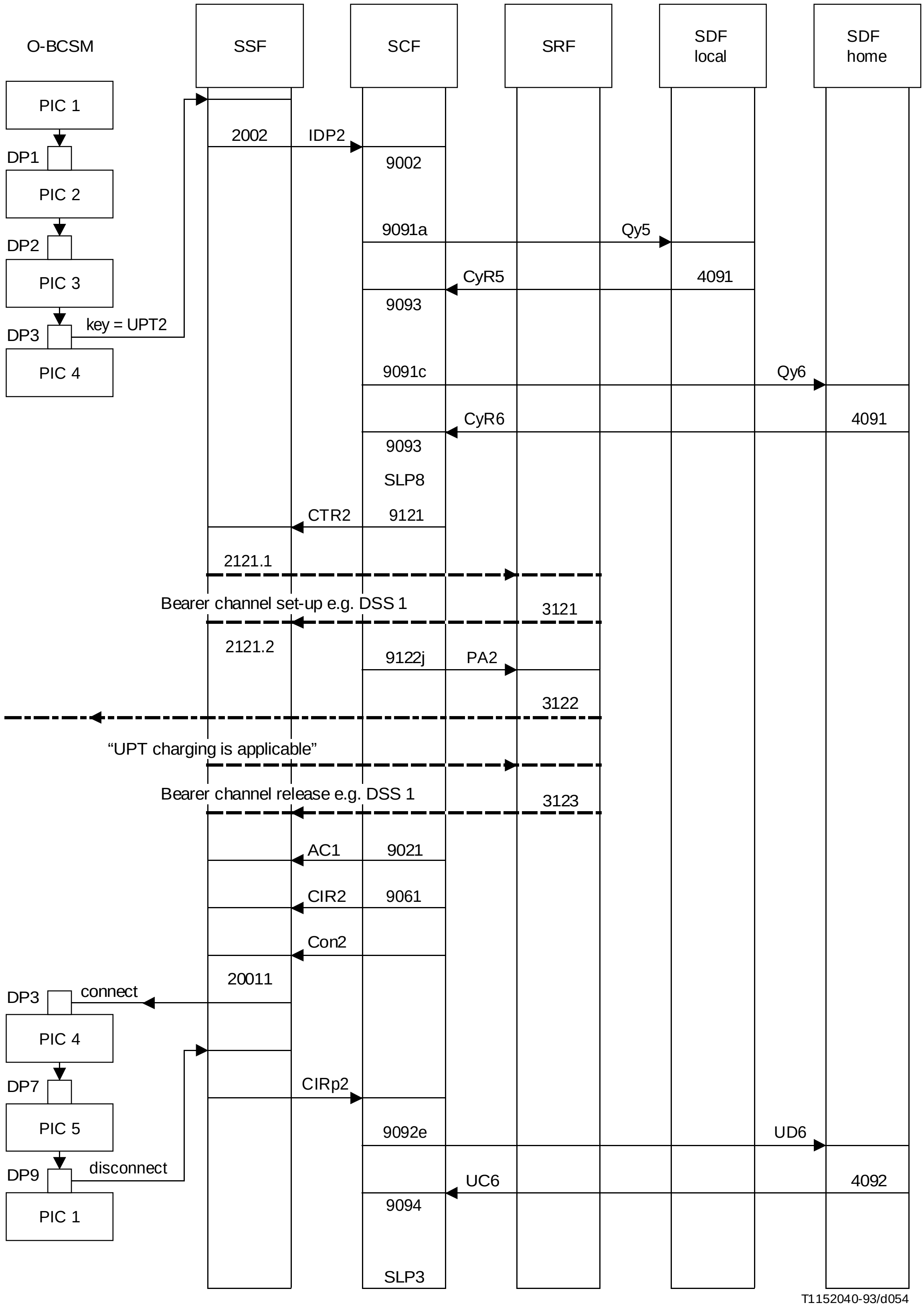


FIGURE A.5-18/Q.1219

IF – Figure 11: Incoming Call to a "UPT" User

IDP2 = Initial DP:				
Mandatory:	Call ID	=	A call,
	Service key	=	UPT2 (version Q.1219),
Optional:	Terminal Type[23]	=	DTMF phone,
	Dialled Digits	=	"UPT" number (i.e. d_1 d_2 . . . d_{12}).
Qy5 = Query:			
Optional:	Database ID	=	"UPT" Local SDF 2,
Mandatory:	Information Key	=	"UPT" Number: "UPT" number,
Optional:	Requested Info Type	=	Home Location.
QyR5 = Query Result:			
Mandatory:	Requested Info	=	"UPT" home sdf 2.
Qy6 = Query:			
Optional:	Database ID	=	"UPT" home sdf 2,
Mandatory:	Information Key	=	"UPT" Number: "UPT" number,
Optional:	Requested Info Type	=	[Current Location, Registered].
QyR6 = Query Result:			
Mandatory:	Requested Info	=	[b' location, TRUE].
CTR2 = Connect To Resource[24]:			
Mandatory:	Call ID	=	A call,
Optional:	IP Routing Address	=	IP Line 2.
PA2 = Play Announcement:			
Mandatory:	Disconnect from IP forbidden	=	FALSE,
	Information To Send.Elementary Message ID	=	Message 11 (i.e. " "UPT" special charging is applicable."),
	SRF Connect ID	=	SRF 2.
AC1 = Apply Charging:			
Mandatory:	Call ID	=	A call,
	Billing Charging Characteristics	=	SDF usage surcharge[25],
	Send Calculation to SCF Indicator	=	FALSE.
Con2 = Connect:			
Mandatory:	Call ID	=	A call,
	Destination Routing Address	=	b' location.
CIR2 = Call Information Request:			
Mandatory:	Requested Information	=	[Call connected elapsed time, Call stop time, Called address, Calling address].
CIRp2 = Call Information Report:			
Mandatory:	Requested Information	=	call record[26].

23) In this service example the SCF is assumed to have knowledge of SSF/ SRF capabilities. An SRF is default present and co-located with the SSF.

24) This is only one of the ways to handle user interaction. The other is to use an Establish Temporary Connection IF and an Assist Request Instructions IF (see 3.1.3.5.1/Q.1218).

25) The charging indicated in this example is an example how charging may be done. For UPT charging the caller is charged for the queries and the portion of the call to the home location of the UPT user, while the UPT user is charged for the portion from his home location to his actual location.

26) The "Call Record" is a structure containing the in IF CIR1 indicated elements.

UD6 =	Update Data:			
Optional:	Database ID		=	"UPT" home sdf 2,
	Function Type		=	Increment[27],
Mandatory:	Information Key		=	"UPT" Number: "UPT" number,
	Updated Info		=	Billing Record. Call record: call record.

UC6 =	Update Confirmation:			
Mandatory:	Outcome		=	Success[28].

D) FEAction descriptions

In IF – Figures 1 to 11 the following Functional entity actions have been identified (see also clause 5/Q.1214):

1) SSF FEAs

2002: Start of IN switch control, stop processing basic call and prepare IDP.

2121.1: Set-up a bearer channel between CCF/SSF and SRF. Signalling used can be any of: switch internal, DTMF or DSS 1.

2121.2: Establish through connection from calling user to SRF.

2003: Notify SCF of user abandon and release user connection to SRF.

20014: Release user connection to SRF.

2122: Release bearer channel between CCF/SSF and SRF. Signalling used can be any of: switch internal, DTMF or DSS 1.

20011: Connect users and start call recording. Data recorded will be: elapsed call time, call stop time, calling user number and called user number.

2) SCF FEAs

9002: Startup of Service Logic Program (SLP).

9003a: Request notification of user abandon.

9121: Initiate connection between SSF en SRF.

9122a: Collect Calling users "UPT" Number and PIN.

9122b: Get calling user PIN.

9091a: Get access rights, local network support capability and home database reference.

9093: Receive Query Response and pass retrieved data to SLP.

9081a: Verify "UPT" Number and PIN.

9083: Receive Query Response and pass screening result to SLP.

9122c: Notify user of service termination.

9092a: Update retries exceeded. Register fraud attempt on "UPT" Number.

9094: Receive Update Confirmation and pass response to service logic.

9003b: Release call forced by SCF.

9092b: Response timeout expired. Register timeout event on fraud list of "UPT" Number.

---

[27] The semantics of the Increment value in this IE is as follows: Take the next free recold list location and store the data given in that location.

[28] Only successful situations are treated here.

9122d: Request user to identify desired operation.

9091b: Query Home Database for access rights of "UPT" user.

9122e: Request user to identify new Incall registration address.

9122f: Request user to for confirmation.

9092c: If user has acknowledged the new Incall registration address, store this address.

9122g: Request new action to be performed.

9122h: Notify user of failure to process his request.

9092d: User has successfully requested deregistration, update home database registration status.

9122i: Request user to supply the desired destination number.

9081b: Check requested Outcall number.

9123: Disconnect forward connection.

9061: Request Call Detail Record from SSF for billing of "UPT" user.

9092e: Store Call Record for billing purposes.

9021: Set charging to special "UPT" charging (e.g. for database usage).

9091c: Query home database for allowance of "UPT" Incall facilities and if allowed, retrieve the "UPT" users home location.

9122j: Notify caller that "UPT" charging will be applied.

3) SRF FEAs

3121: Setup connection to CCF/SSF.

3122: Play announcement (and collect digits).

3123: Release connection to CCF/SSF.

4) SDF FEAs

4081: Process verification request.

4091: Extract requested data.

4092: Modify requested data.

5) SCF Service Logic Program

SLP1: Test whether the PIN and PTN match.

If 'match' is reported, goto IF – Figure 8 FEA 9122d.

If 'no match' is returned, test if the maximum number of retries is exceeded.

If the number of retries is exceeded, goto IF – Figure 5 FEA 9122c.

If the number of retries is not exceeded, goto IF – Figure 4 FEA 9122a.

SLP2: Initiate forced release of through timeout by IF – Figure 7 SRFR1 or user abandoned by IF – Figure 6.

SLP3: Terminate Service Logic Program.

SLP4: Test the access rights of "UPT" user.

If "UPT" action is not allowed, goto IF – Figure 9c FEA 9122h.

If "UPT" action is allowed, select the appropriate procedure:

For Terminate, goto IF – Figure 7 RC1.

For Incall registration, goto IF – Figure 9a FEA 9122e.

For Incall deregistration, goto IF – Figure 7 FEA 9092d.

For Outcall, goto IF – Figure 10 FEA 9122i.

SLP5: If user has acknowledged the new Incall registration address, store this address by performing FEA 9092c in IF – Figure 9a.

If the user typed something else, test the allowed number of retries.

If the retries are exceeded, goto IF – Figure 7 RC1.

If the retries are not exceeded, goto FEA 9122e in IF – Figure 9a.

SLP6: If response is Terminate, goto IF – Figure 7 RC1.

If response is new request, goto IF – Figure 8 FEA 9122d.

SLP7: Check requested Outcall number.

If check is ok, proceed.

If check is not ok, goto IF – Figure 9c FEA 9122h.

SLP8: If the local network support "UPT" Incall facilities and these are allowed for this "UPT" users, continue. If not, goto FEA 9003b in IF – Figure 7.

5) Physical view

The physical view of this "UPT" example consists of three parts:

1) *Physical Entity (PE) diagrams*, consisting of:
   - Physical Entities;
   - location of the Functional entities;
   - interface protocols.
2) *Time Sequence Diagrams*, consisting of:
   - operations;
   - parameters;
   - instances of TC_Begin;
   - instances of TC_End.
3) *Application Entity procedures* showing states and state transitions for each FE.

A) Physical Entity Diagram

In Figure A.5-19 the following mapping of Functional Entities (FEs) to Physical Entities (PEs) is made:

- Service Switching Point (SSP): SSF/CCF and SRF,
- Service Control Point (SCP): SCF and Local SDF,
- Service Data Point (SDP): Home SDF.

All links are implemented using Signalling System No. 7 and TCAP.

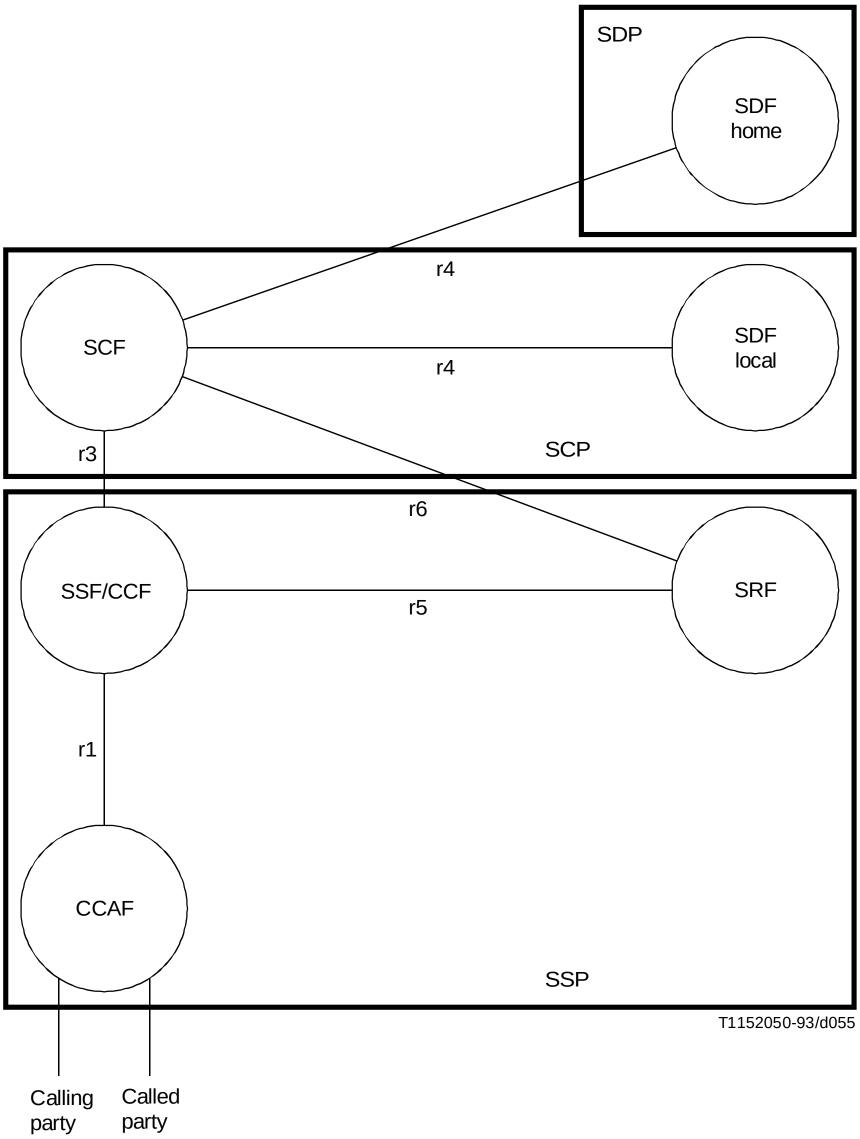


FIGURE A.5-19/Q.1219
Physical entity diagram

B) Time Sequence Diagrams

In the following the time sequence diagrams of the "UPT" service are given. The numbers of the Time Sequence Diagrams figures (TSD – Figure i) correspond to the Information Flow figures [IF – Figure i, see **4) c)**].

The operations under each message arrow are grouped as they would be grouped into TCAP messages.

In some parameters logical names are given rather than physical codes for ease of reading. Where physical codes are listed, the use of this code is indicated in between comment brackets (< >).

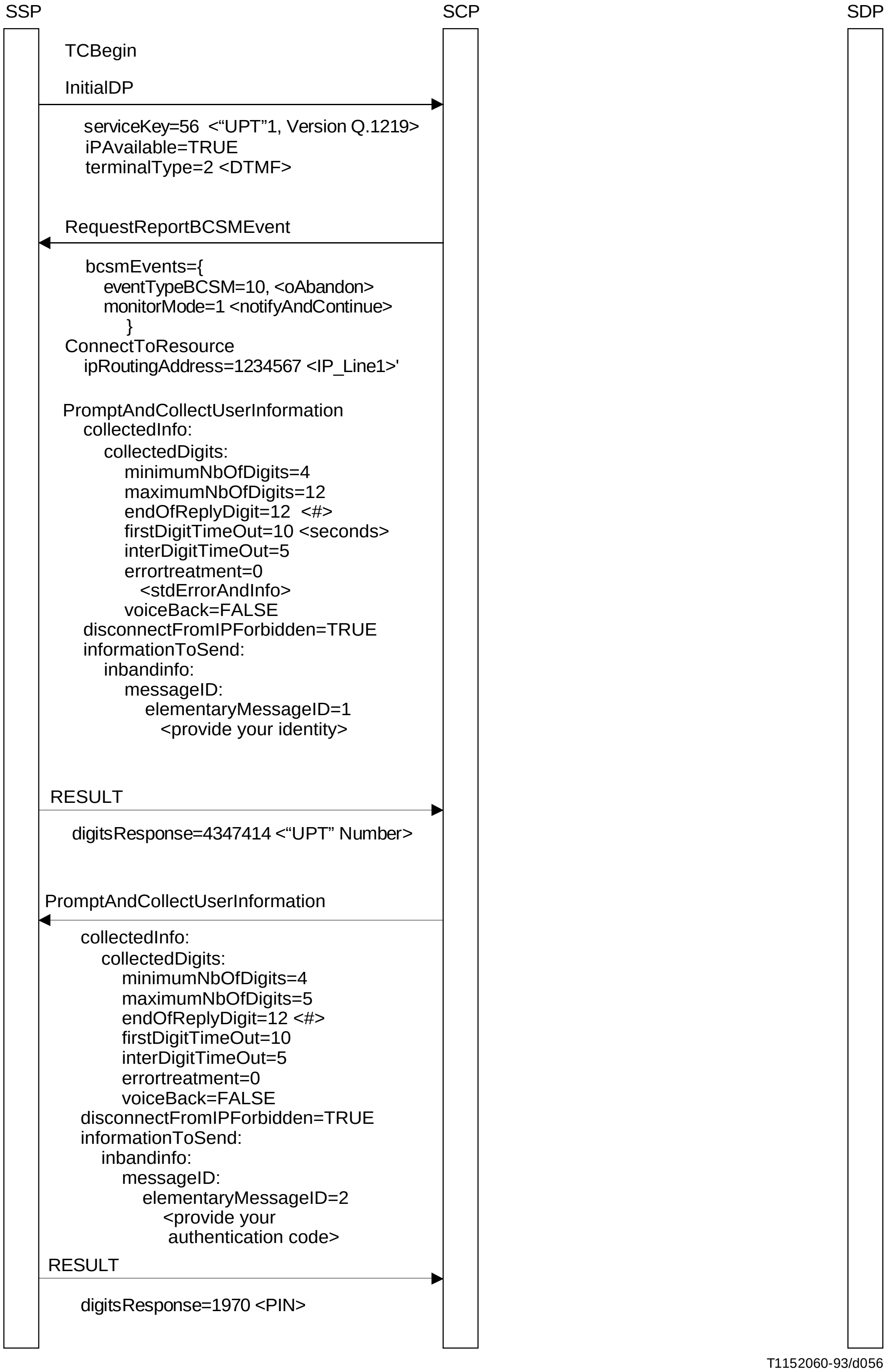


FIGURE A.5-20/Q.1219

TSD – Figure 2a: Access, Identification and Authentication of the "UPT" user (multiple message situation) part a

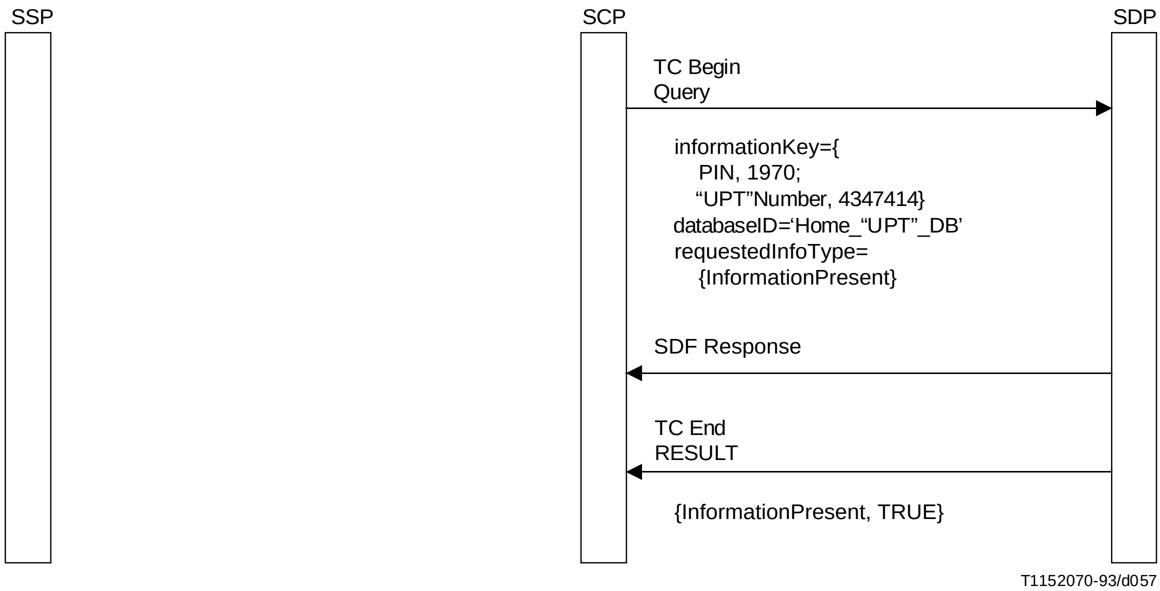


FIGURE A.5-21/Q.1219

TSD – Figure 2b: Access, Identification and Authentication of the “UPT” user (multiple message situation) part b

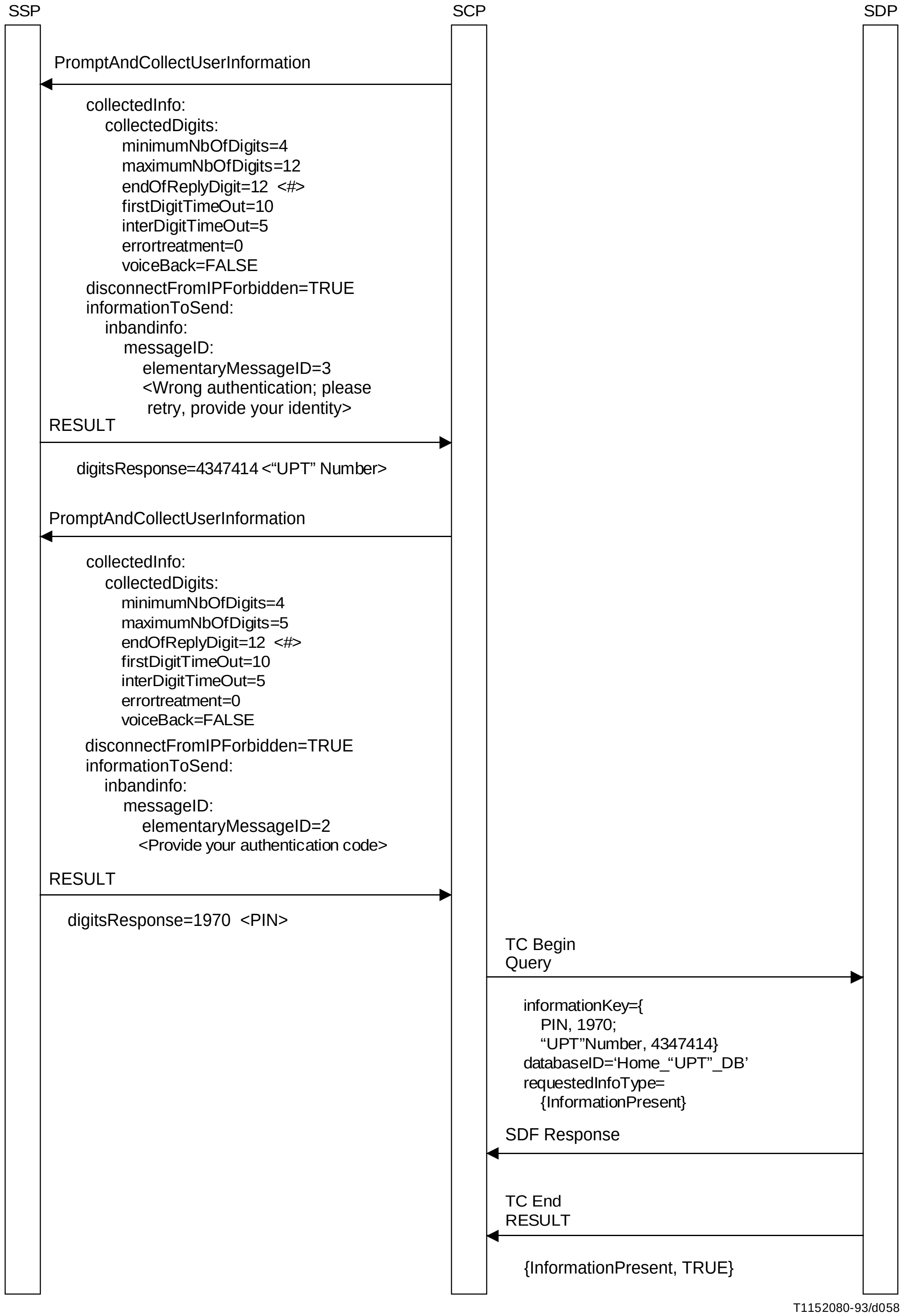


FIGURE A.5-22/Q.1219

TSD – Figure 4: Authentication retry

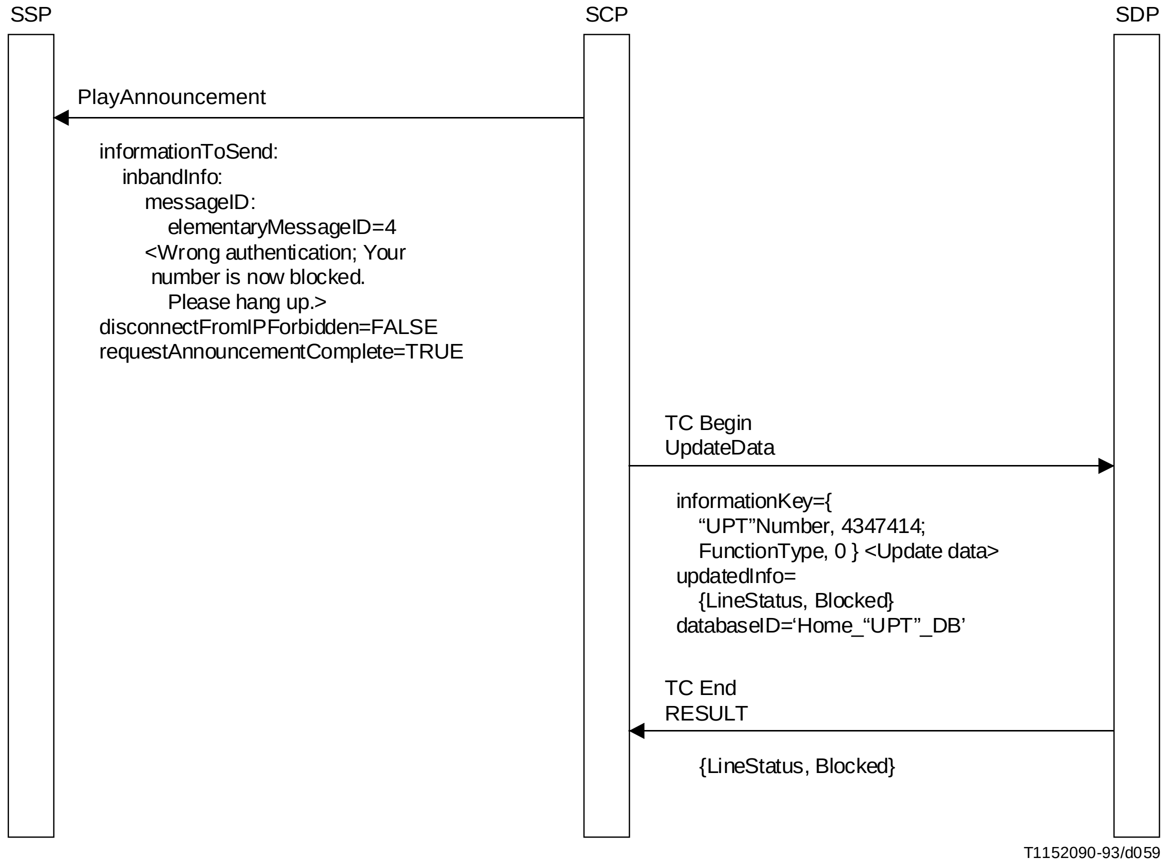


FIGURE A.5-23/Q.1219

TSD – Figure 5: Maximum number of retries exceeded

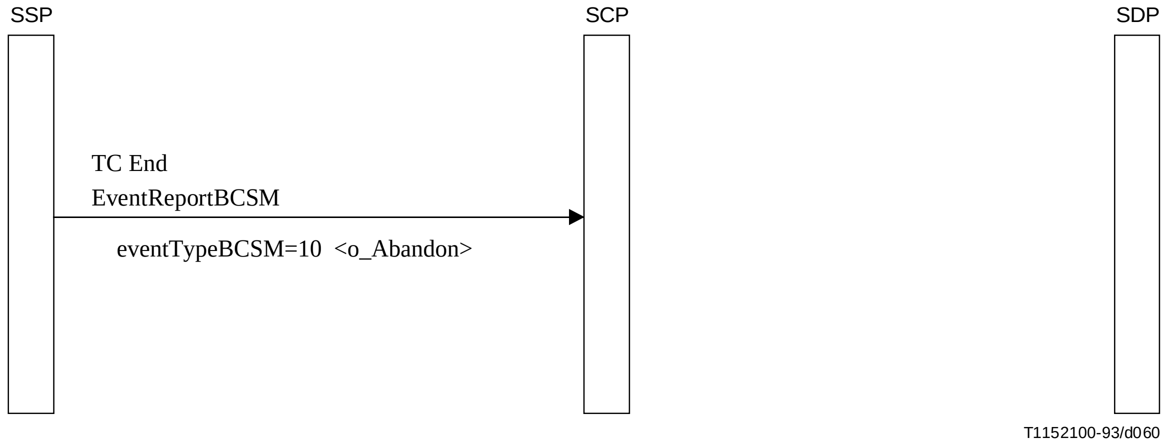


FIGURE A.5-24/Q.1219

TSD – Figure 6: Release of calling user (user initiated)

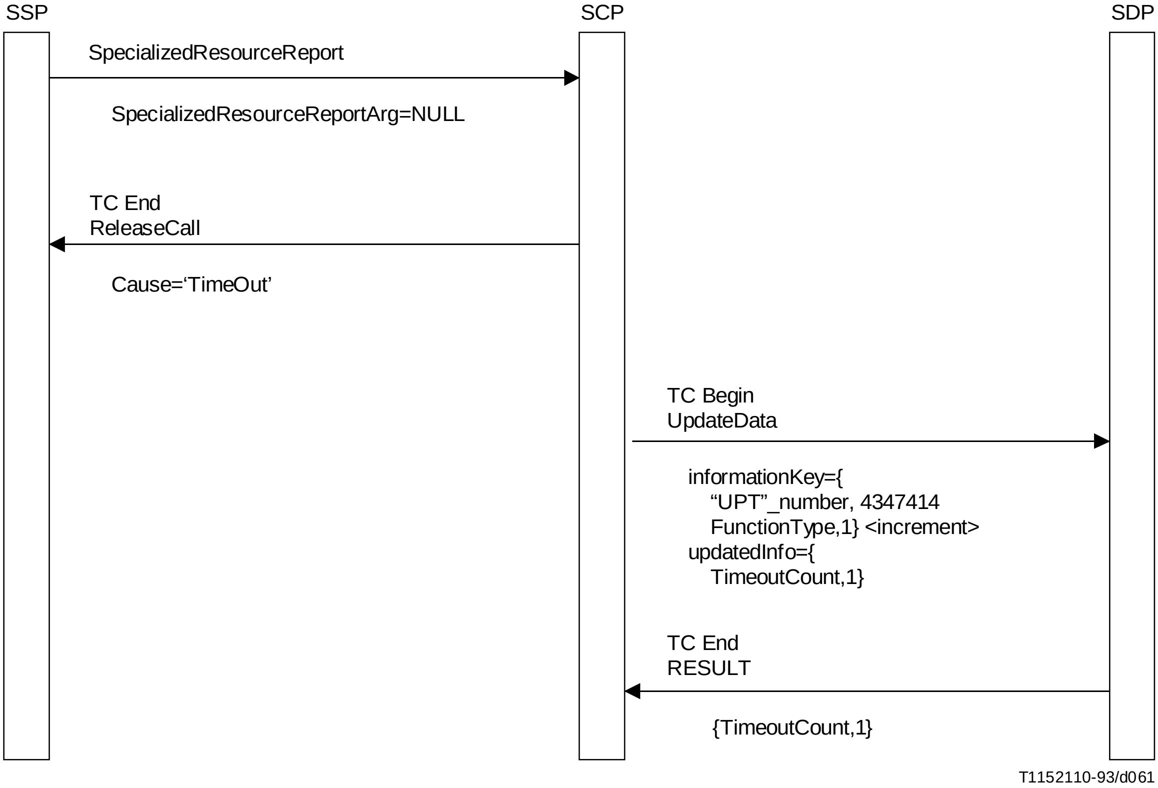


FIGURE A.5-25/Q.1219

TSD – Figure 7: Release of calling user (network initiated)

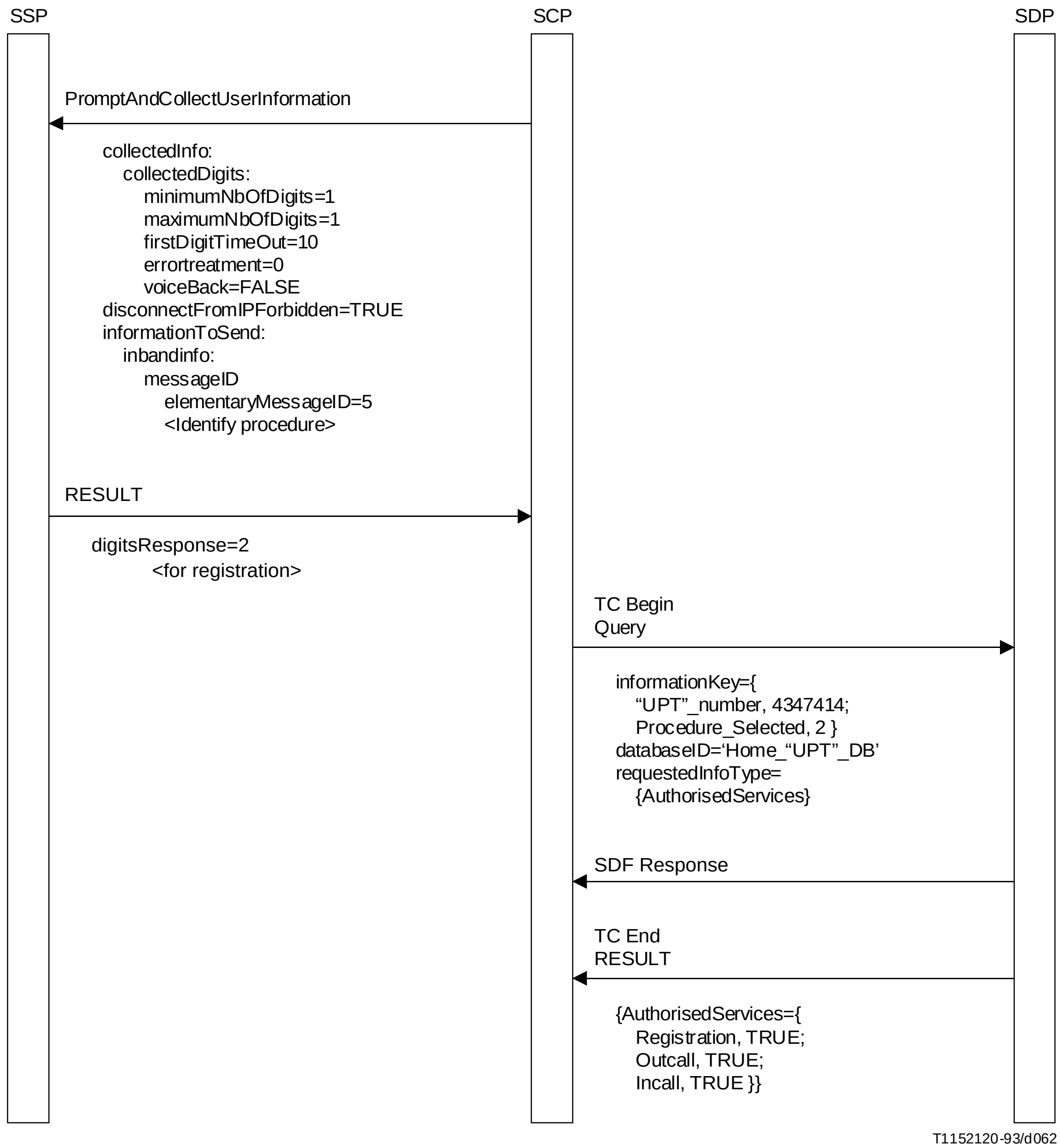


FIGURE A.5-26/Q.1219

TSD – Figure 8: Procedure identification

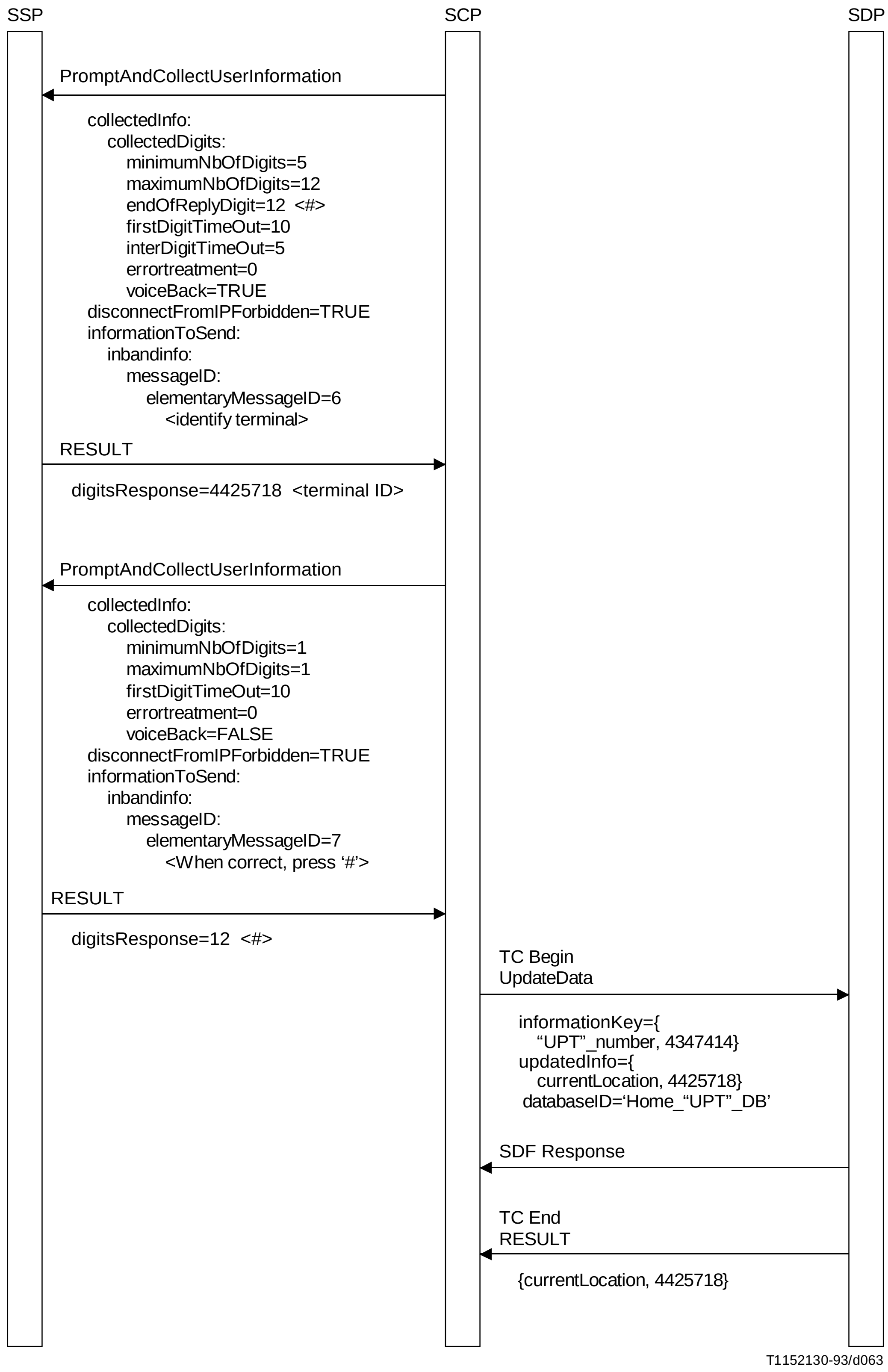


FIGURE A.5-27/Q.1219

TSD – Figure 9a: Incall registration

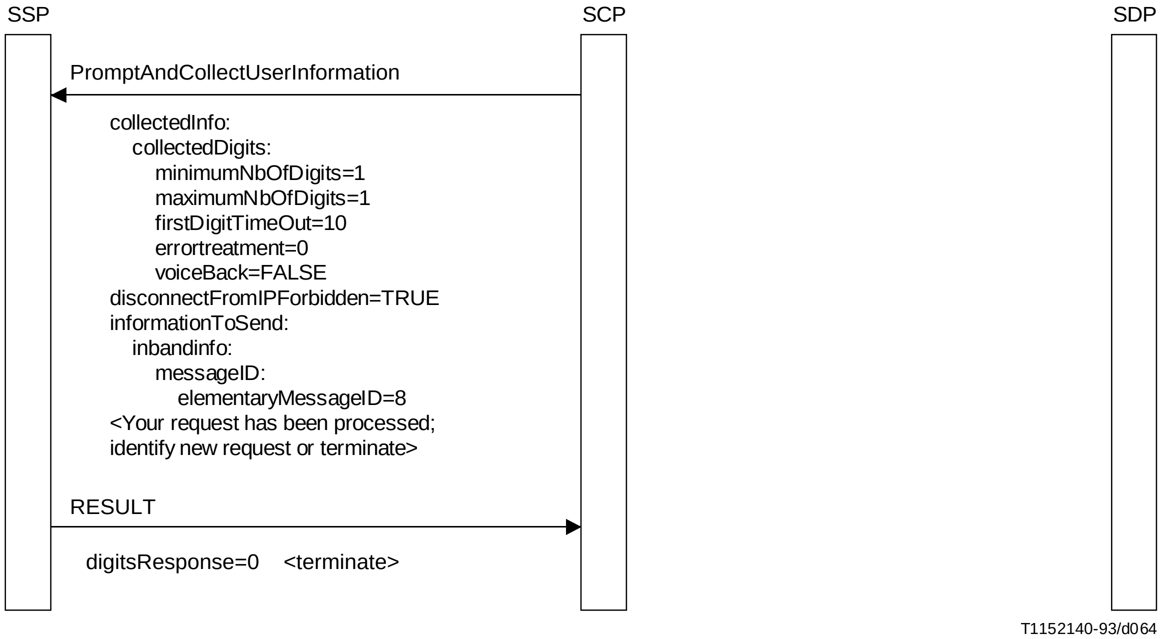


FIGURE A.5-28/Q.1219

TSD – Figure 9b: Incall registration successfully executed

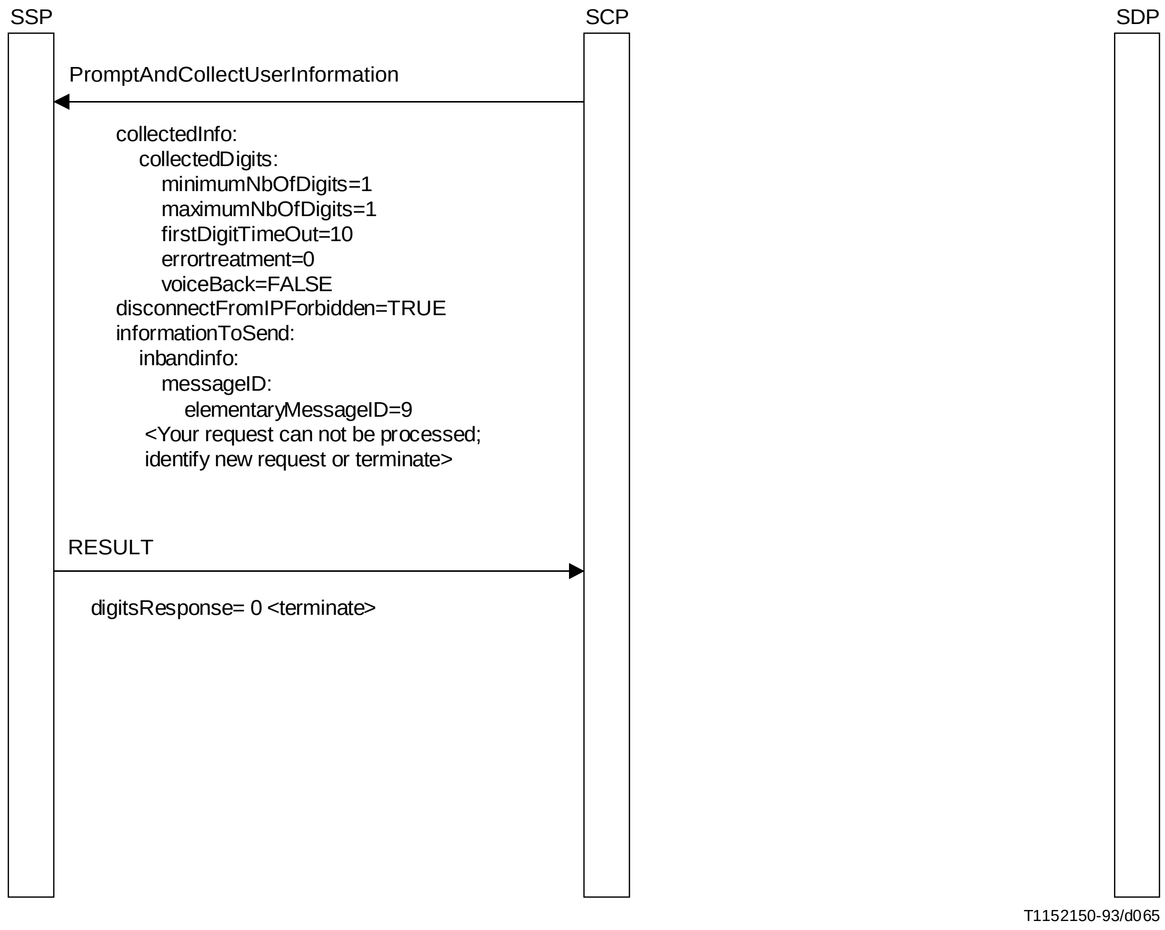


FIGURE A.5-29/Q.1219

TSD – Figure 9c: Incall registration failed

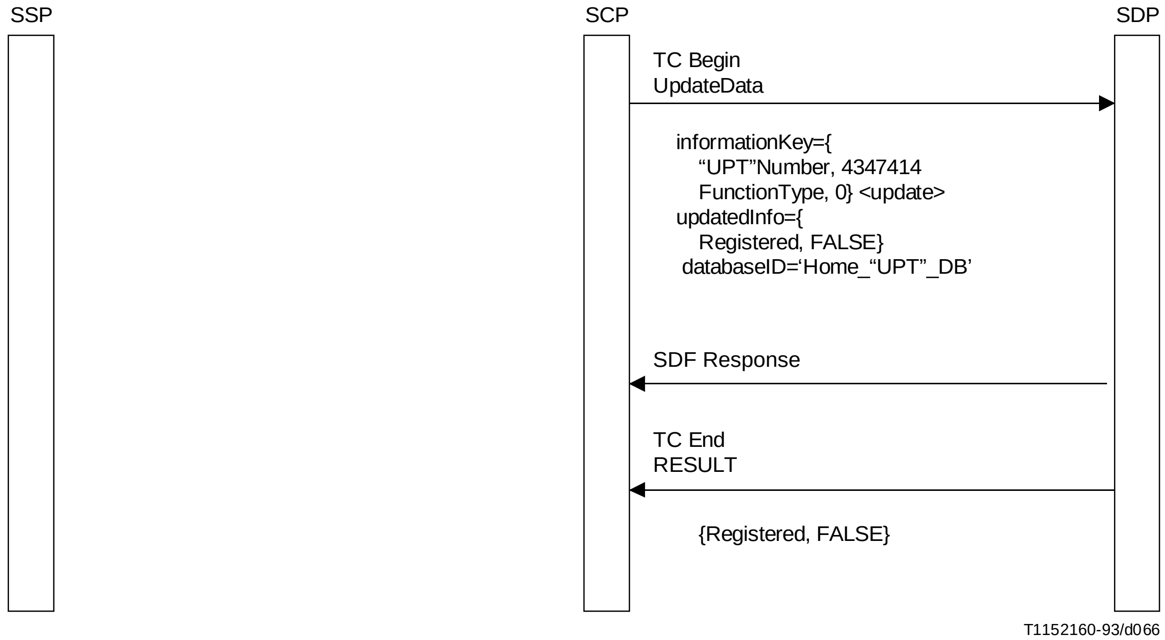


FIGURE A.5-30/Q.1219

TSD – Figure 9d: Incall deregistration

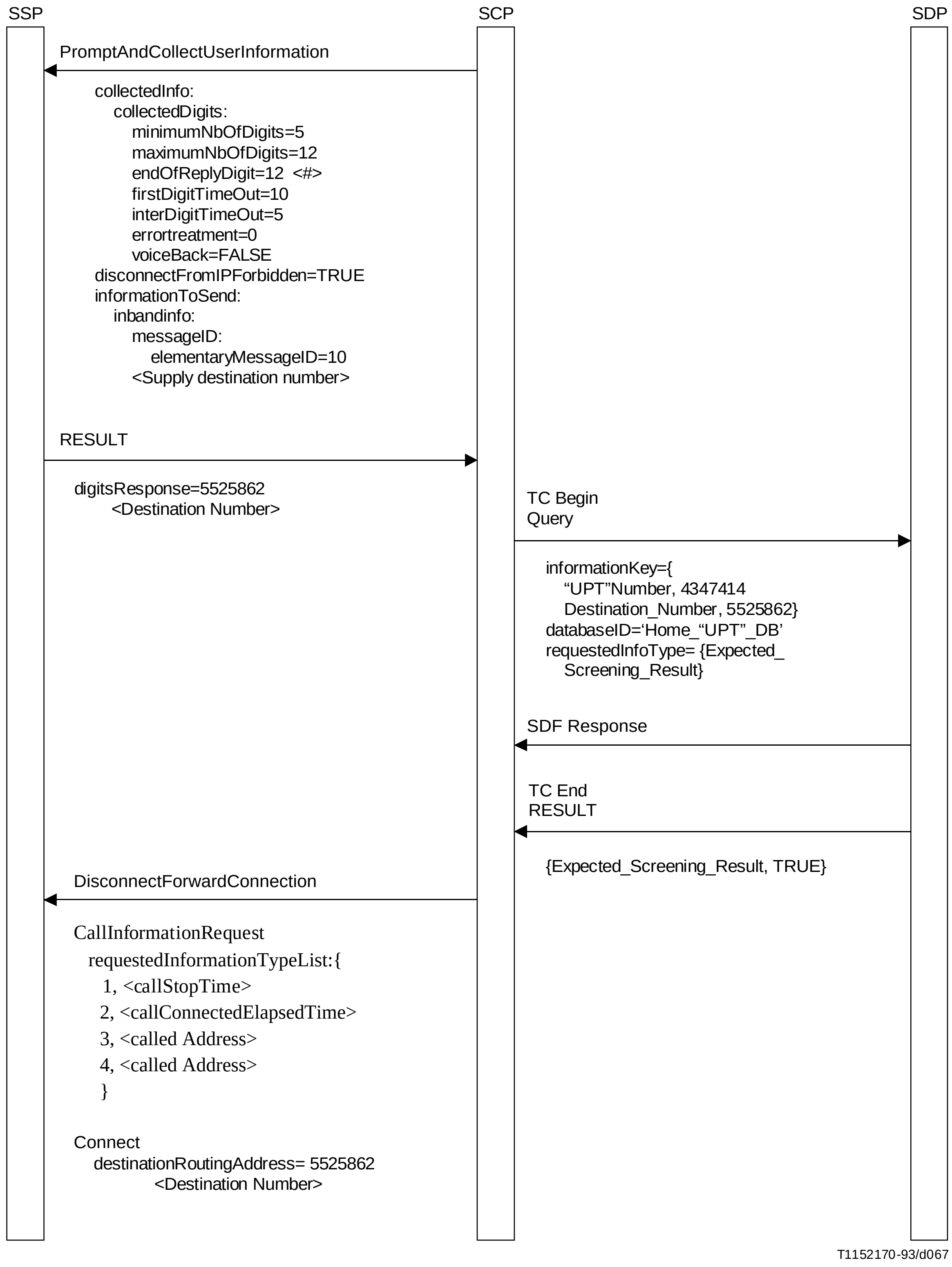


FIGURE A.5-31/Q.1219

TSD – Figure 10a: Outcall series part a

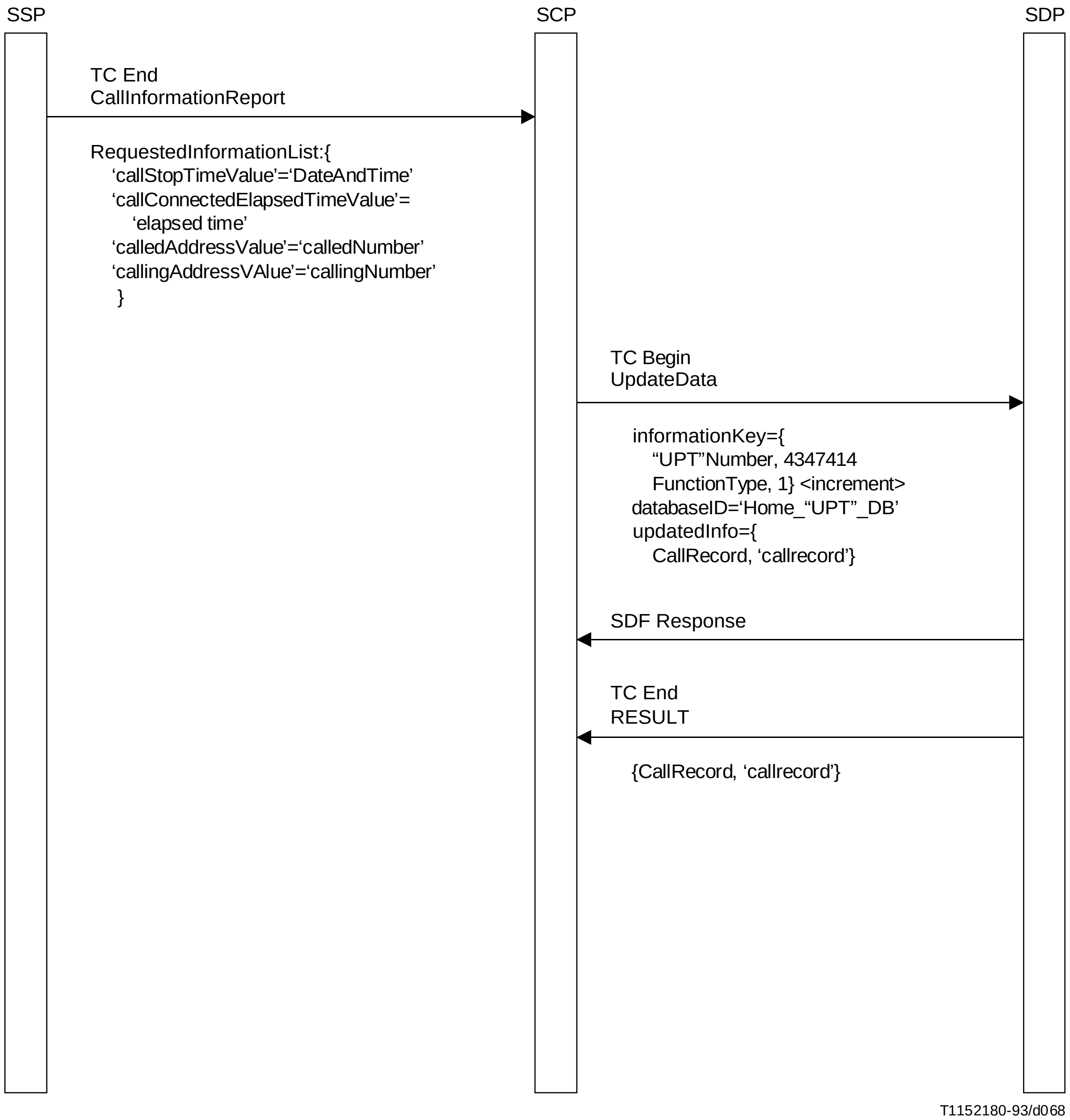


FIGURE A.5-32/Q.1219

TSD – Figure 10b: Outcall series part b

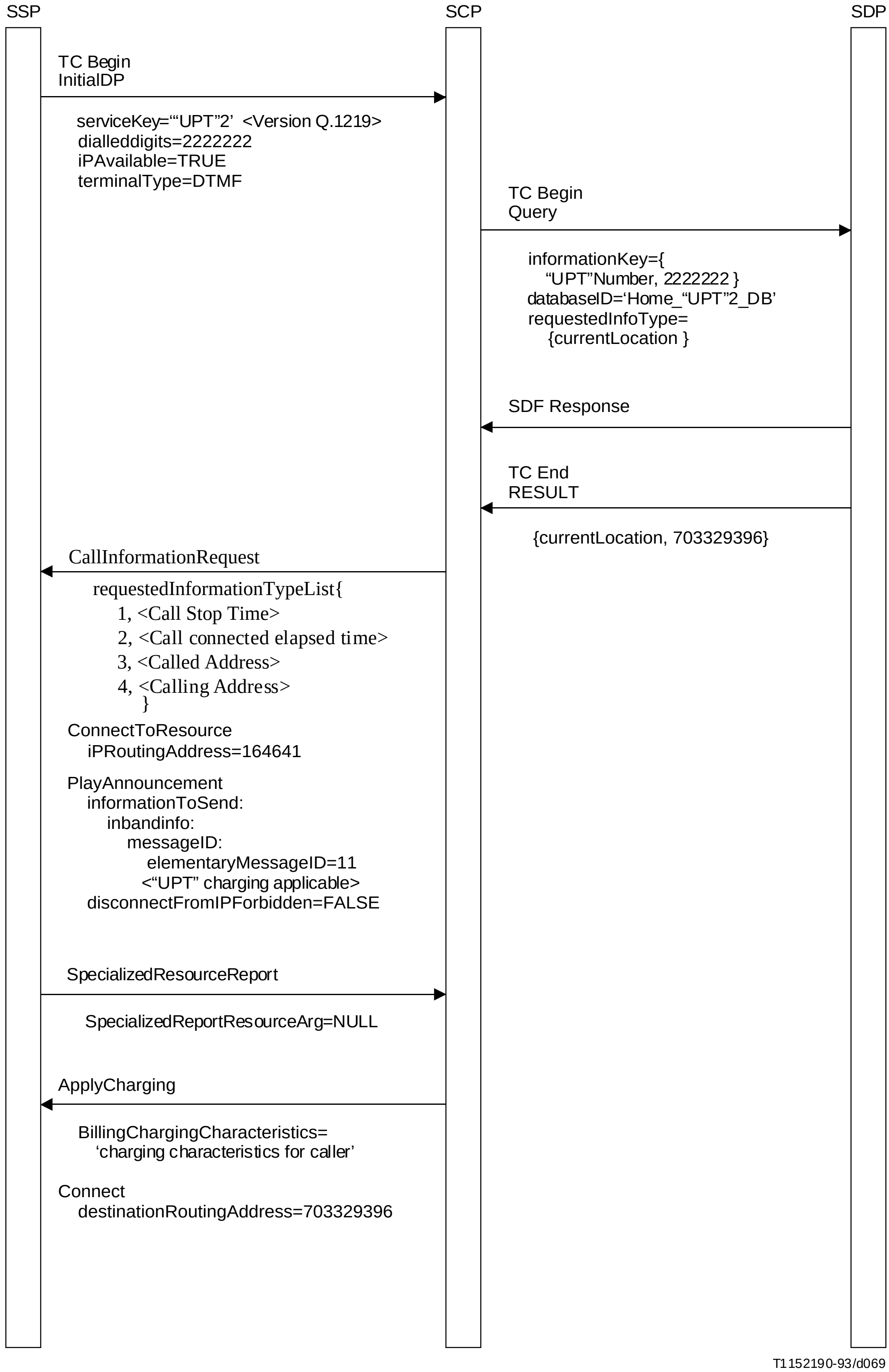


FIGURE A.5-32/Q.1219

TSD – Figure 11a: Incall series part a

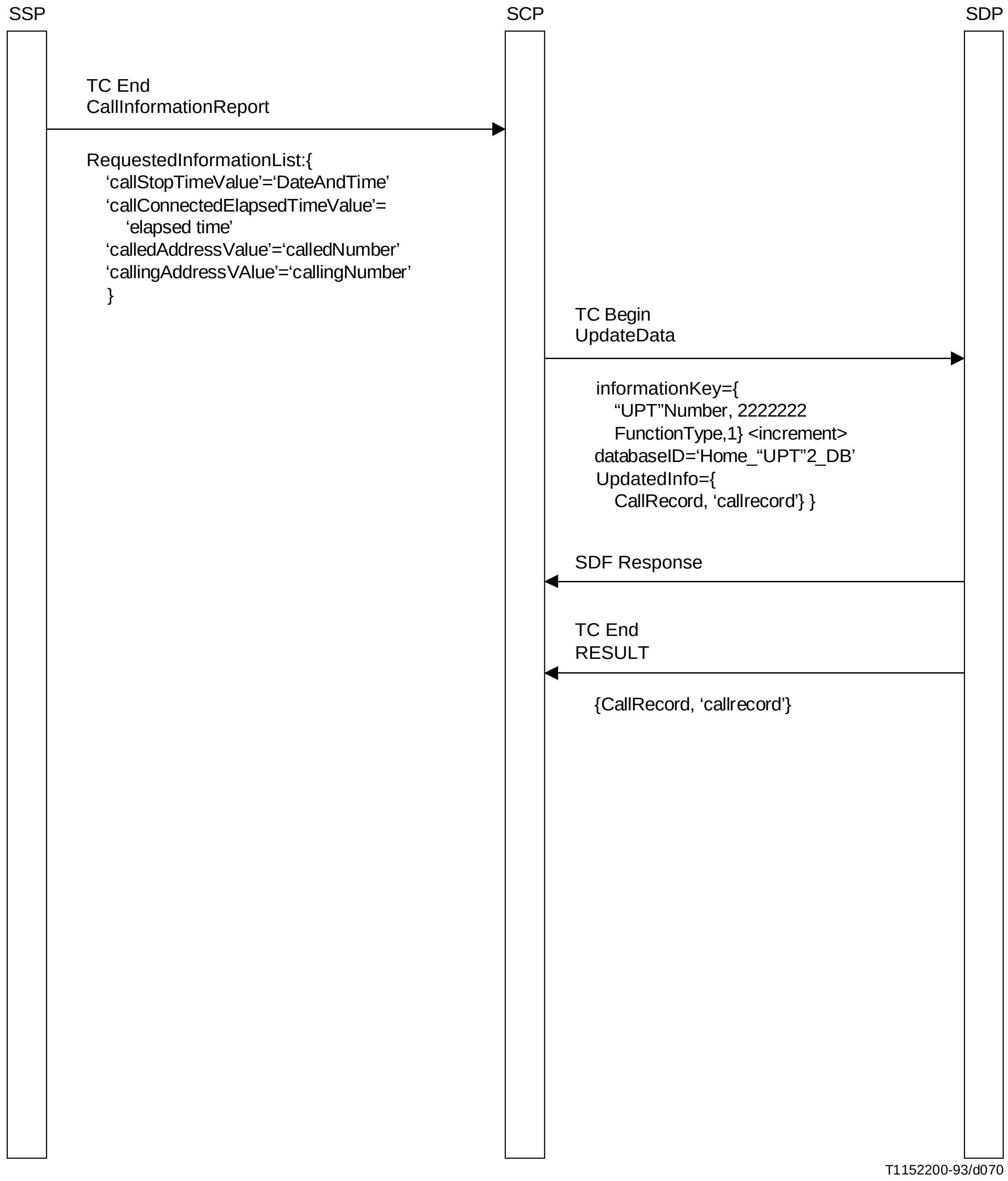


FIGURE A.5-34/Q.1219

TSD – Figure 11b: Incall series part b

C) Physical Entity State Transition Diagrams

In Figures A.5-35 to A.5-39 the events and state transitions of the FEs are given.

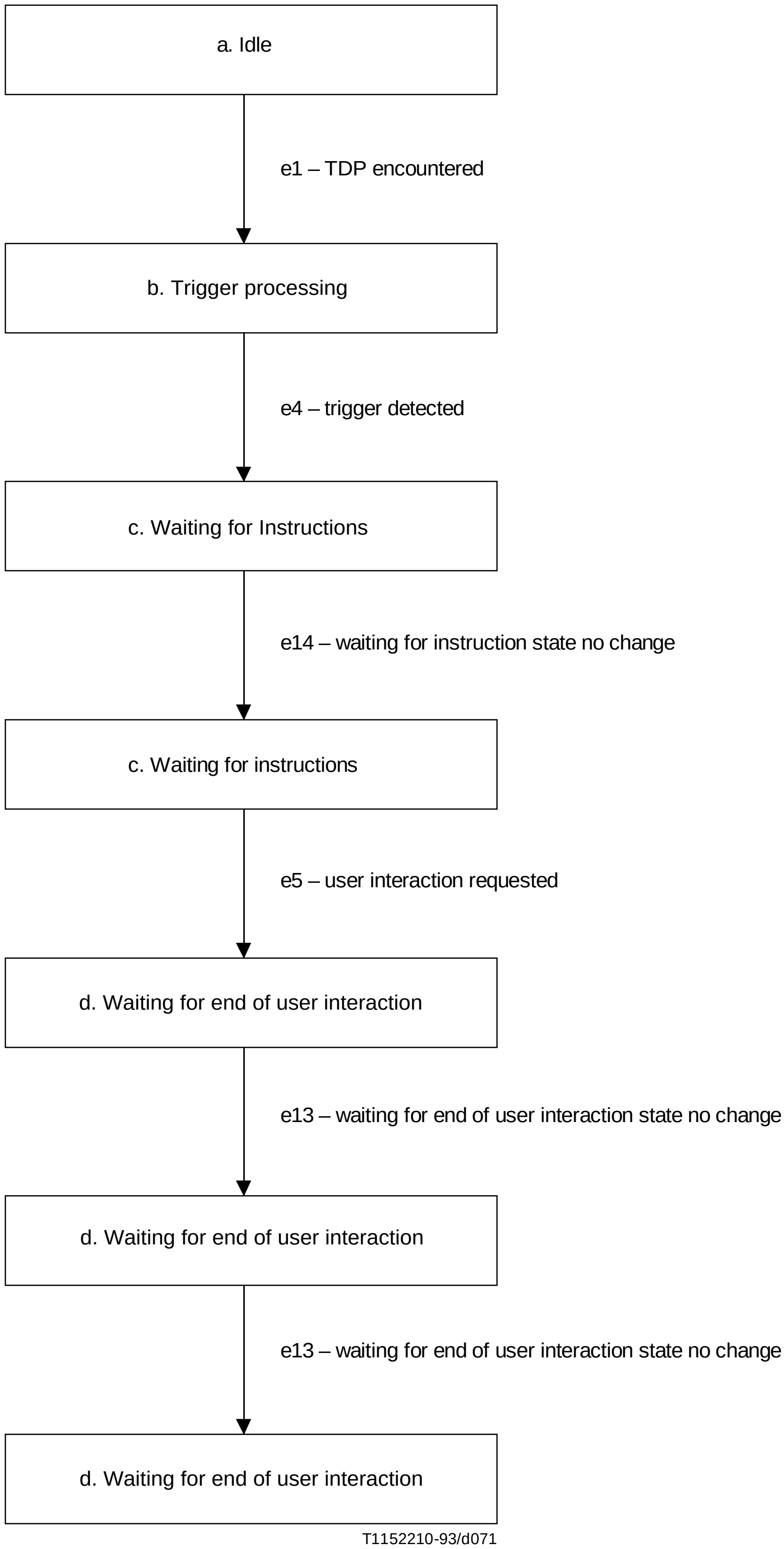


FIGURE A.5-35/Q.1219

Physical entity State Diagram: Service Switching Point

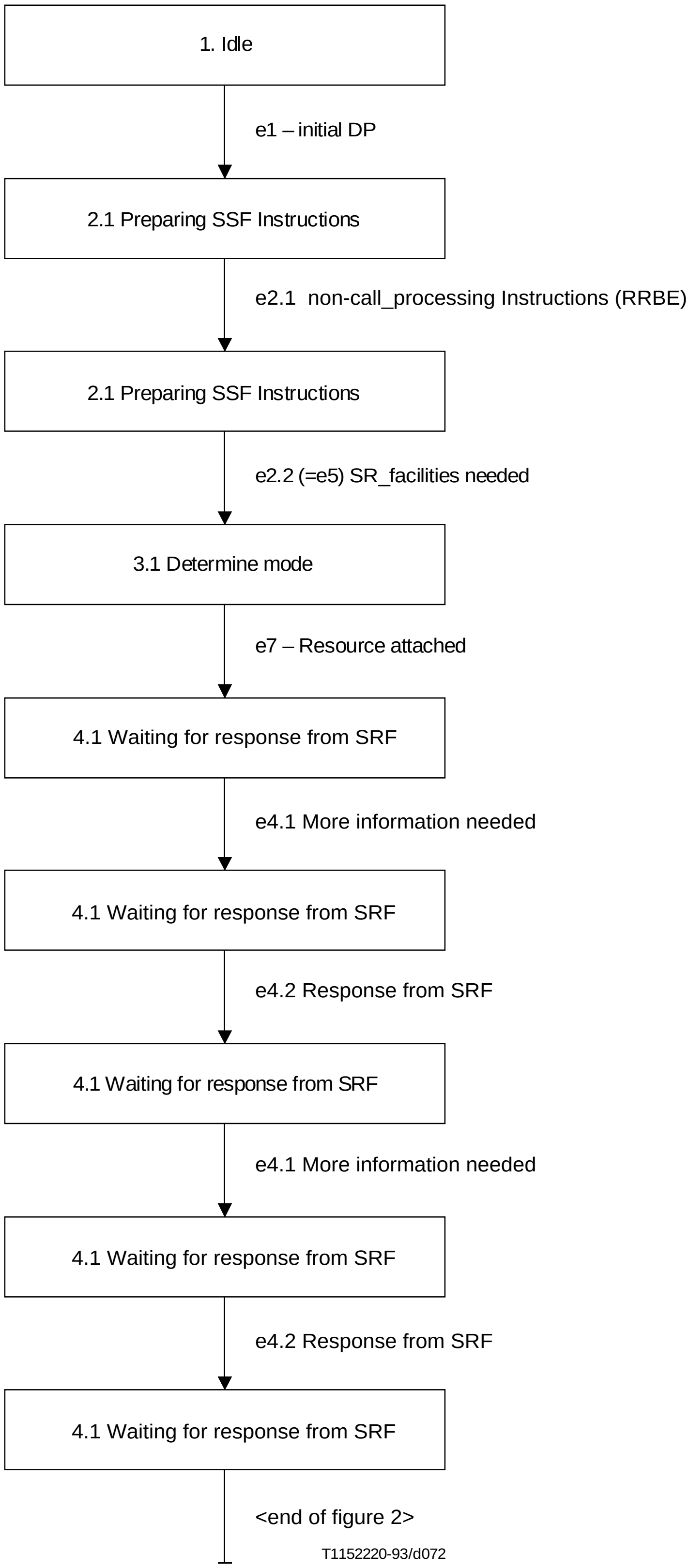


FIGURE A.5-36/Q.1219

Physical entity State Diagram: Service Control Point (Service Switching Point related states)

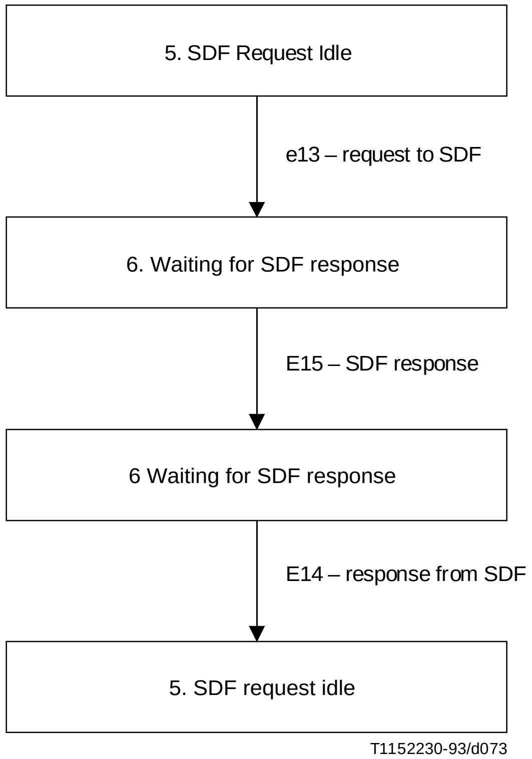


FIGURE A.5-37/Q.1219

Physical entity State Diagram: Service Control Point (Service Data Point related states)

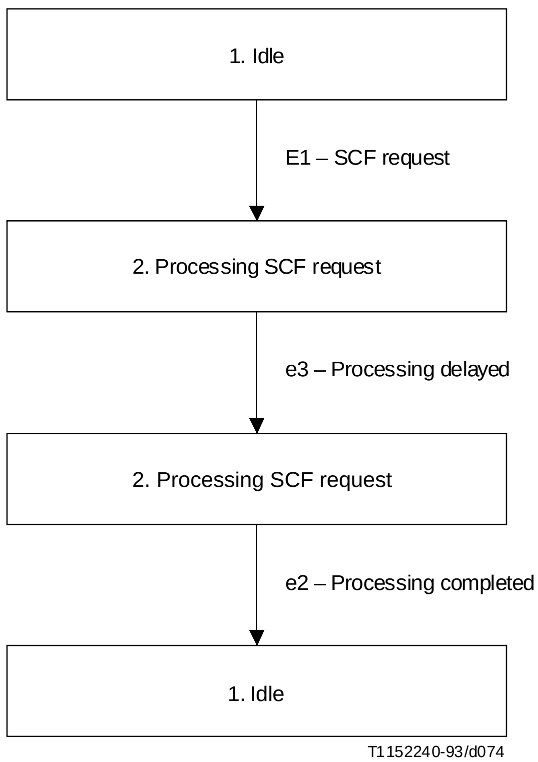


FIGURE A.5-38Q.1219

Physical entity State Diagram: Service Data Point

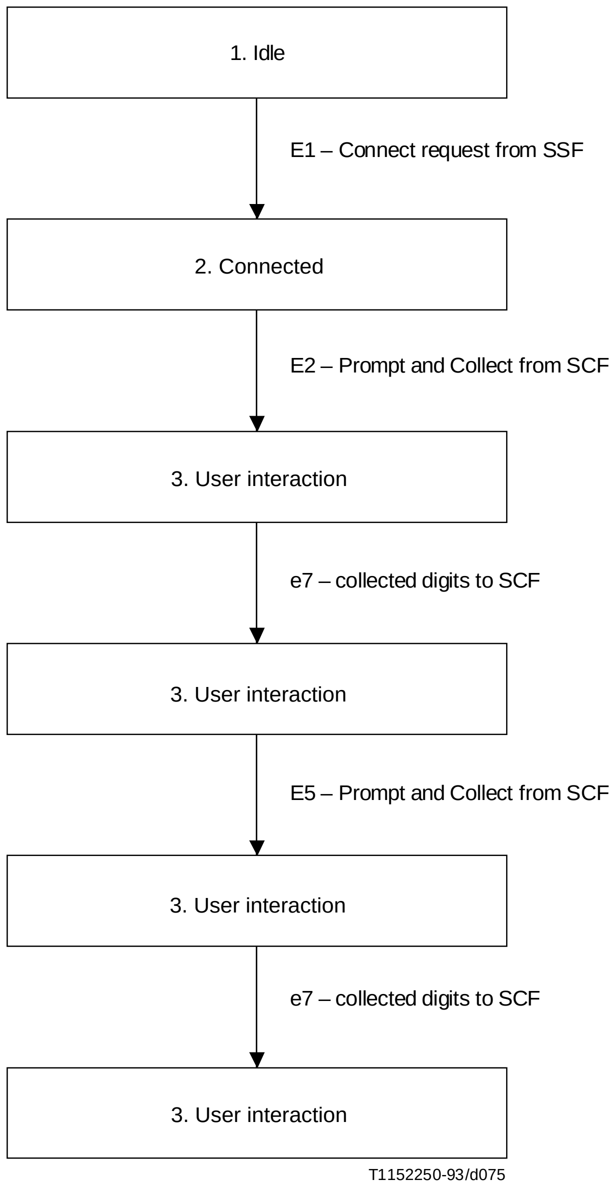


FIGURE A.5-39/Q.1219

Physical entity State Diagram: Specialized Resource Point

A.6 Call Forwarding Unconditional with Announcement service scenario example

1) CS-1 IN Capability illustrated by CFU-Ann

This service scenario illustrates how a call during the set-up phase, can be re-directed to another directory number, using capabilities entirely within the scope of CS-1.

Call re-direction is described through use of the CS-1 call segment association mechanism.

Also, this scenario examines the use of an SRF external to the switch, in an Intelligent Peripheral (IP), but directly connected to the switch, as used to play announcements to a Calling Party, utilizing SS7.

2) Textual description of service scenario

2.1) General considerations

Call forwarding Unconditional (CFU) allows the service subscriber, who is the Called Party (i.e. Forwarding Party), to forward calls to another telephone number. Forwarding will occur whenever the service is activated, regardless of the present status of the subscriber’s line (i.e. busy, idle). With this service, all calls destined to the subscriber’s number are redirected to the new telephone number. The service can be located in an originating exchange or terminating exchange.

The service is assumed to be under the control of the subscriber and can be activated/deactivated by the subscriber. This service scenario does not describe the procedures and required actions to activate/deactivate the service.

Switch based charging is turned off for the forwarded portion of the call. IN determines bill for forwarded portion of the call.

2.2) Call Parties’ perspective

Step 1: Calling Party dials the directory number of the service subscriber (i.e. Forwarding Party).

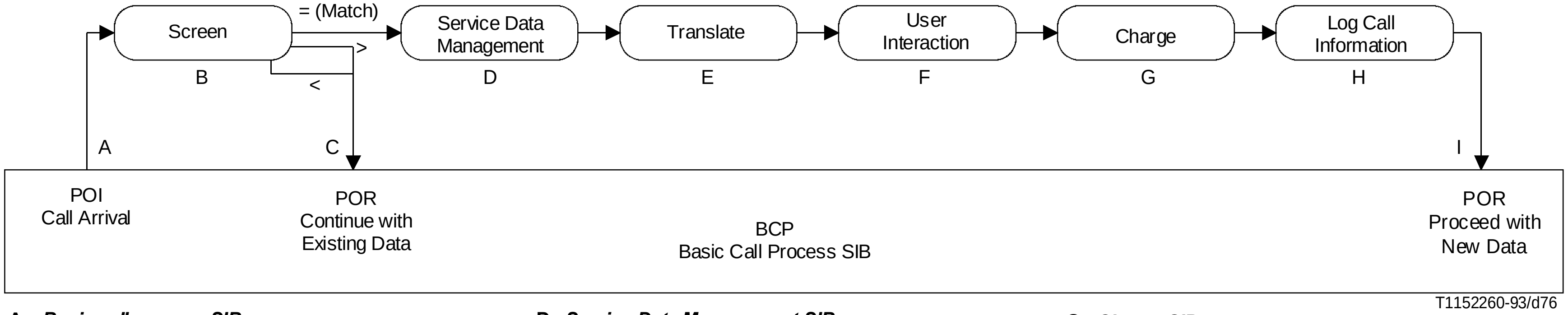
Step 2: Calling Party hears announcement, informing the party of the redirection of the call, but that the party is not being charged for the forwarded portion of the call. Forwarding Party is not given any indication that the CFU-Ann service is being implemented.

Step 3: Forwarded-To Party is alerted of incoming call. Calling Party is given ringback.

Step 4: Forwarded-To Party and Calling Party participate in an active call.

Step 5: Calling Party goes on-hook, thereby initiating the sequence of actions to terminate the service. (The procedures described in this scenario would remain the same if the Forwarded-To Party was the first to go on-hook.)

3) Global view (refer to Recommendation Q.1213)



A ***Basic call process SIB***

The switch determines that IN actions are required.
POI – Call Arrival
CID – Calling Line Identity (1)
– Dialled Number (2)

B ***Screen SIB***

Service logic determines that the Call Forwarding service should be applied.
Input :
SSD – Screen List Indicator = Forwarding Conditions
CID – Identifier = Dialled Number (2)
Output :
Logical End – Match
– No Match

C ***Basic call process SIB***

Call Forwarding should not be applied, the switch proceeds as normal.
POR – Continue with Existing Data

D ***Service Data Management SIB***

The Forwarded-To Number is determined.
Input:
SSD – File Indicator = Dialled Number (2)
– Action = Retrieve
CID – Element Indicator
Output :
Logical End – Success
CID – Data Retrieved = Forwarded-To Number (3)

E ***Translate SIB***

The Dialled number is replaced by the Forwarded-To number.
Input:
SSD – Type = one number to one number
CID – Information = Dialled Number (2)
Output:
Logical End – Success
CID – translated data = Forwarded-To number (3)

F ***User Interaction SIB***

The Calling Party is informed that the call will be forwarded.
Input:
SSD – Announcement Parameters = Announcement ID
CID – Call Party Identifier = Calling Line Identity (1)
Output:
Logicial End – Success

G ***Charge SIB***

Switch based charging is turned off for the forwarded portion of the call. IN billing for the call is initiated. (Time for the Charge SIB extends beyond end of the
Input:
SSD – Account Number = CIDFP-Line
– Account % = 100
– Resource Type = Bearer Connection
– Service Identifier = Call Forwarding
CID – Line = Calling Line Identity (2)
Output :
Logical End – Success

H ***Log call information SIB***

The start and stop times of the call are stored, to be used to determine IN billing of the call. (Time for this SIB extends beyond end of the call.)
Input:
SSD – CIDFP Log = Call stop time
= Call connect time
= Calling Line ID
CID – Indentifier = Call stop time
= Call connect time
= Dialled Number (2)
Output:
Logical End – Success

I ***Basic call process SIB***

The call is routed towards the Forwarded-To party.
POR – Proceed with New Data

4) Distributed view

This service is applied to a terminating call segment (T_CS), on behalf of the Forwarding Party. This T_CS is modelled by Term BCSM (1), in this scenario.

The service requires a connection segment association between the T_CS and a new originating call segment (O_CS), in order to begin the process of completing the call to the Forwarded-To Party. This new CS is modelled by Orig BCSM (2).

This scenario does not imply that there are limitations on the CFU-Ann service being applied to an originating call segment, on behalf of the Forwarding User.

The Call Forwarding Unconditional with Announcement service can be activated and deactivated by the Subscriber. The procedures for Subscriber activation and deactivation are not detailed in this scenario. For successful completion of this scenario, it is assumed that the Subscriber has previously activated the service.

Step 1: The SSF recognizes an attempt to terminate a call to the Subscriber (Forwarding Party), queries the SCF for call handling instructions, providing information on the current call termination attempt.

Step 2: The SCF starts CFU-Ann service logic and queries the SDF twice. First, to check if call forwarding should currently be invoked and the second time for the translation of the Called Party's number to the Forwarded-To number.

Step 3: The SDF translates the Called Party number into the Forwarded-To Party's number and responds to the SCF.

Step 4: The SCF initiates an SSF to SRF connection over the Bearer channel.

NOTE – This scenario does not model the connections between the SRF and Call Party.

Step 5: The SCF initiates the playing of an announcement to the Calling Party, using the SRF.

Step 6: The SCF initiates the SSF actions to terminate the connection to the SRF.

Step 7: The Network prepares to forward the call. First, the SCF initiates the SSF arming of Event Detection Points (EDP), for subsequent SCF notification. Then, the SCF directs the SSF to disable switch based billing for the portion of the call to be forwarded. In this scenario, the SCF will track the duration of the call, and provide IN billing independent of switch based billing.

Step 8: The SSF redirection of the call is initiated by the SCF, to the directory number of the Forwarded-To Party. This causes the SSF to generate a new originating Call Segment and originating BCSM on behalf of the Forwarding Party, which is associated with the current terminating Call Segment on behalf of the Forwarding Party.

Step 9: The SSF generates a new terminating call segment, modelled by Term BCSM (2), in order to complete the call to the Forwarded-To Party.

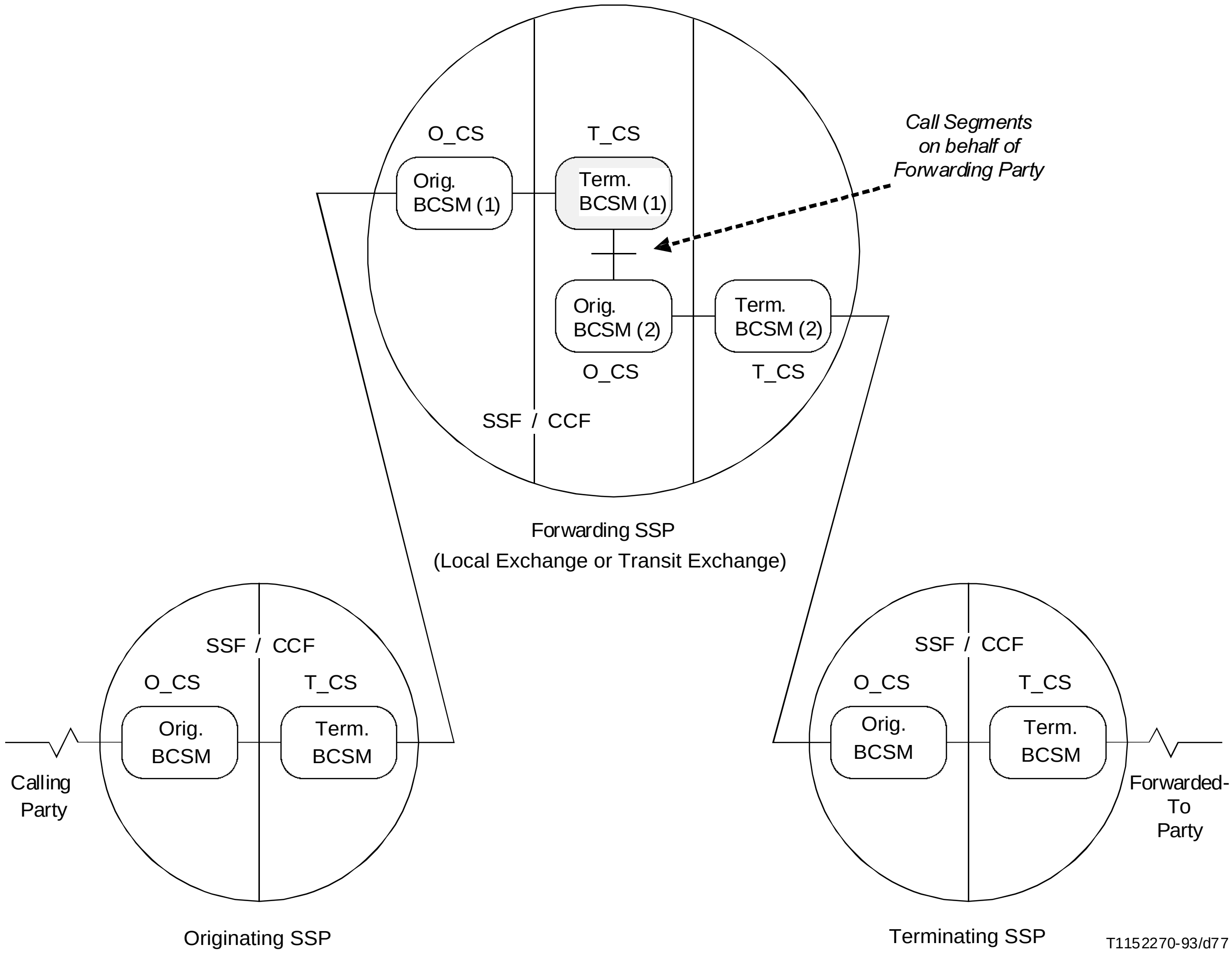
Step 10: The SSF notifies the SCF just prior to the call becoming active.

Step 11: The SSF notifies the SCF just as the call terminates.

Step 12: The SCF calculates billing for the call and updates the SDF database.

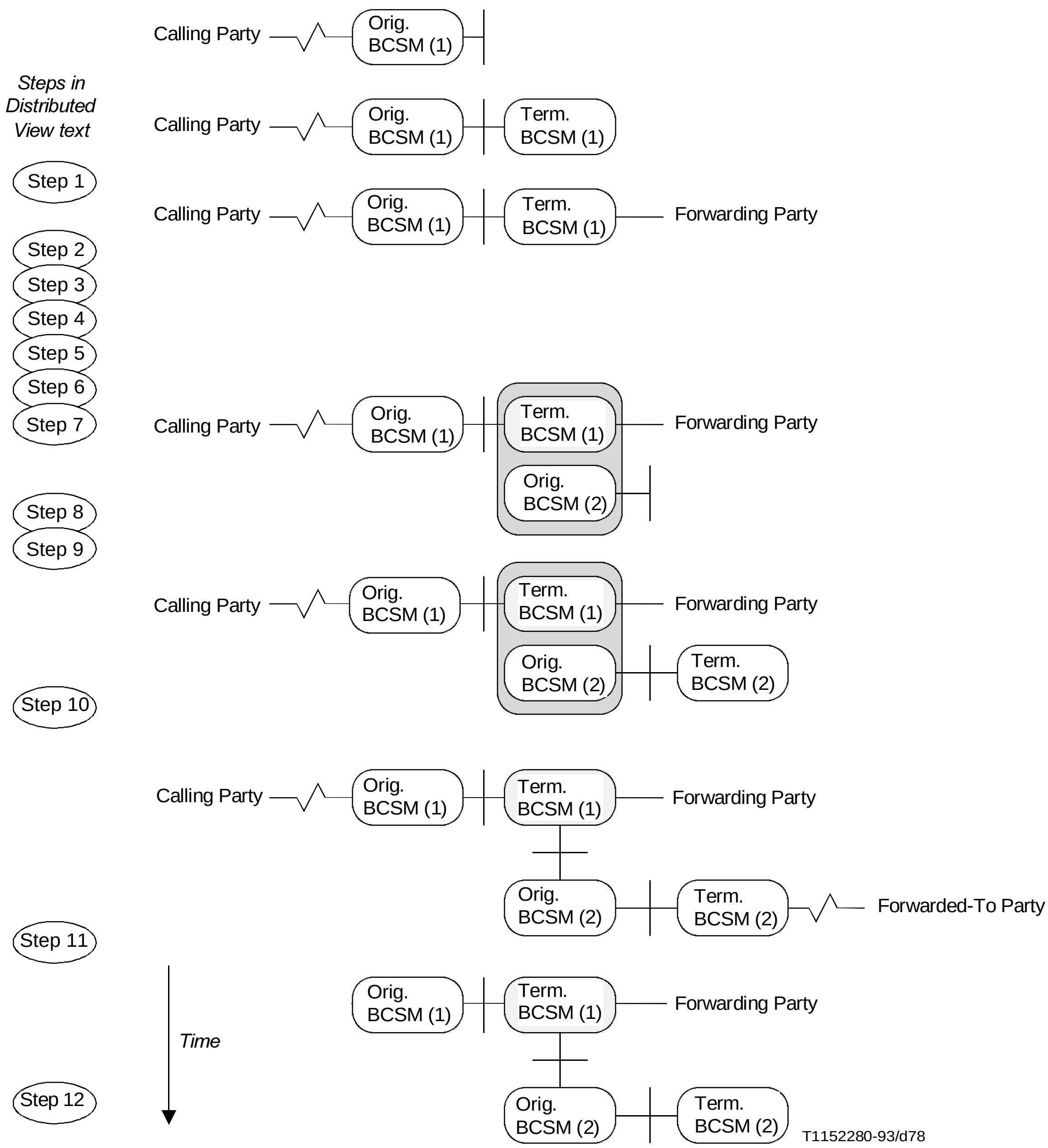
A) Call Segment Diagram (refer to Recommendation Q.1214)

1) Network Overview of Call Segments Diagram



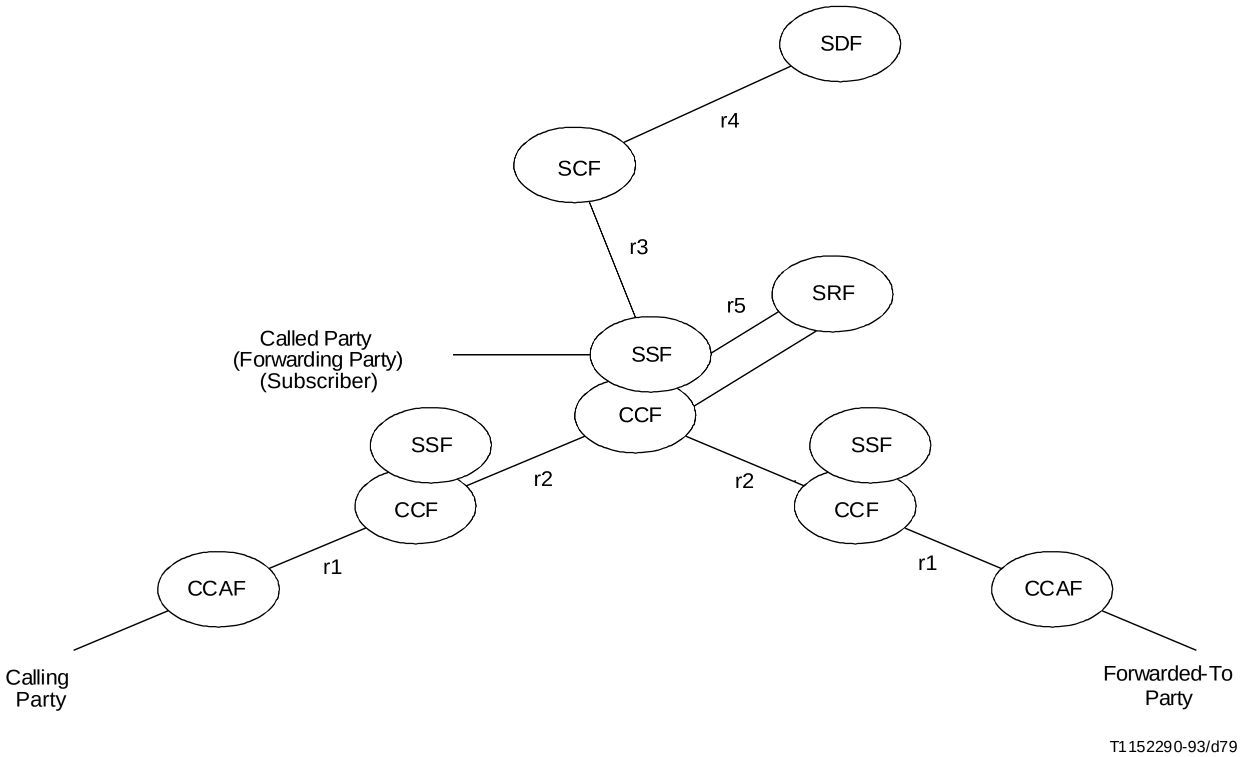
NOTE – Service logic is invoked on behalf of shaded terminating CS (T-CS), modeled by Term BCSM (1).

2) Call Segments in Forwarding SSP Diagram (refer to Recommendation Q.1214)

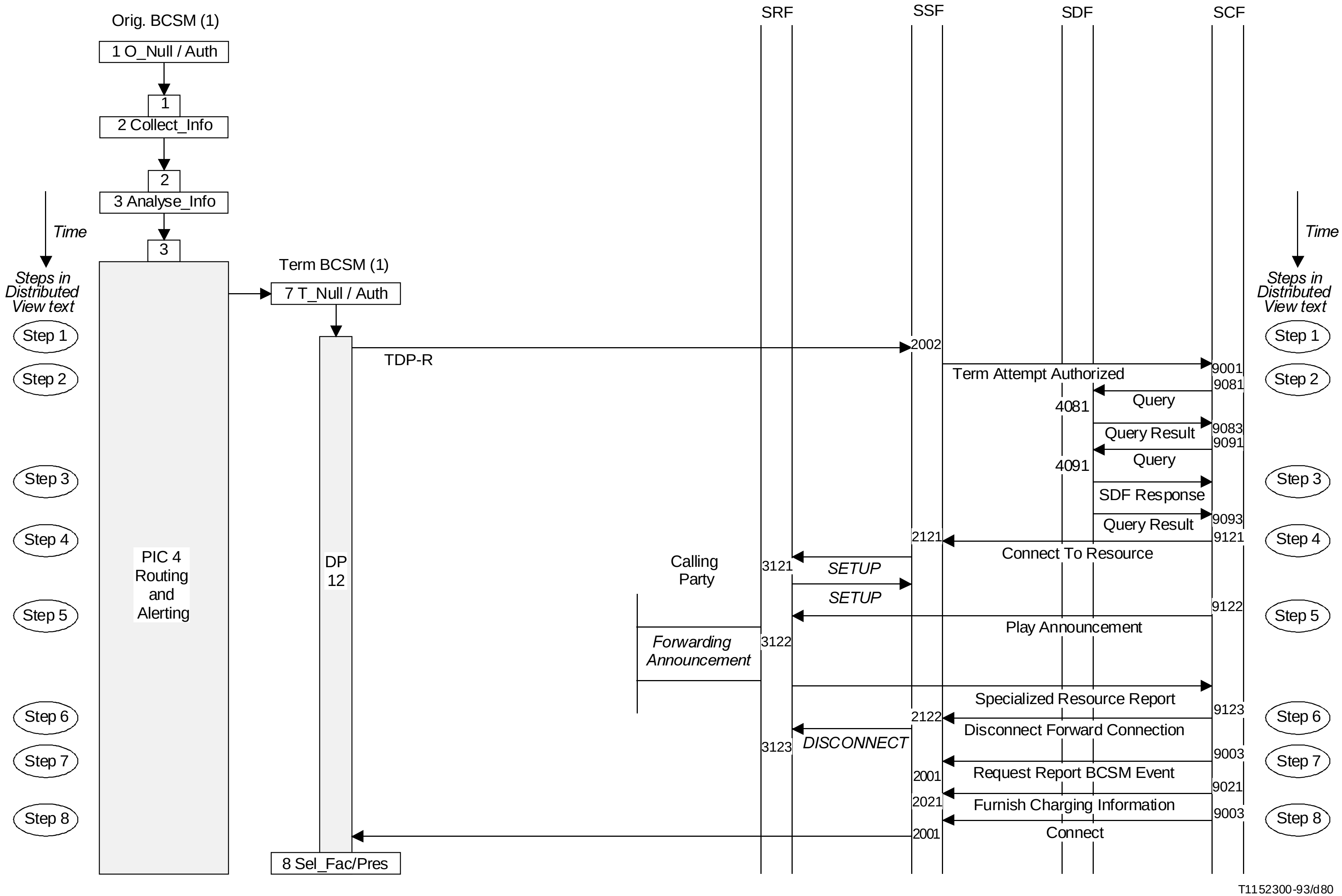


NOTE – Service logic is invoked on behalf of shaded terminating CS (T_CS), modeled by Term BCSM (1).

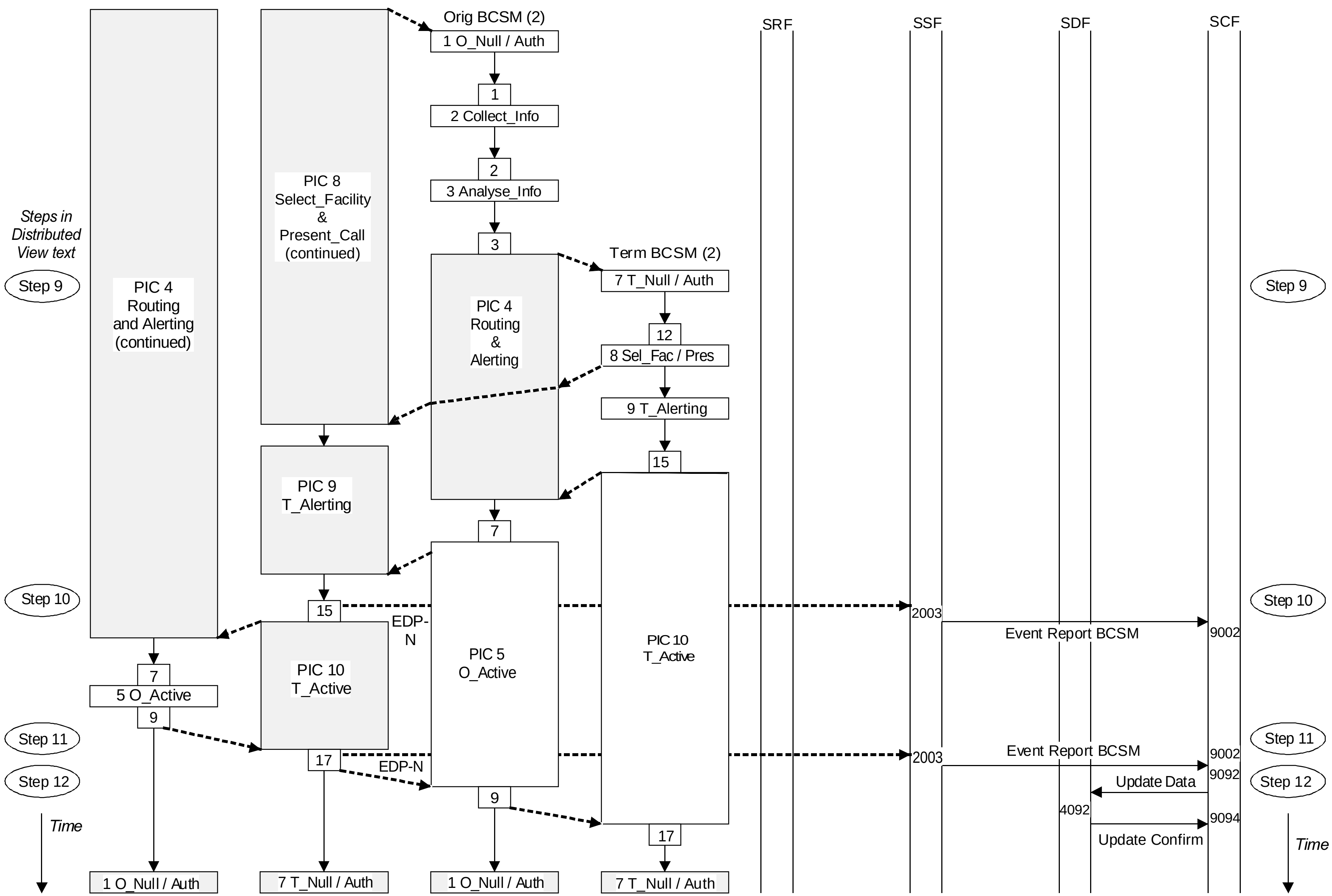
B) Functional Entity Interfaces Diagram (refer to Recommendation Q.1214)



C) Information Flow Sequence Diagram (refer to Recommendation Q.1214)



(Sheet 1 of 2)



T1152310-93/d081

NOTES

1 Shaded PICs or DPs denote a pause in transitions for a given BCSM white other IN actions occur.

2 All BCSM transitions are internal to Forwarding SSP.

(Sheet 2 of 2)

D) Functional Entity Actions (FEA) Descriptions

The FEAs are listed in the order in which they appear in the scenario. Along with each FEA reference number is the Functional Entity to which the FEA applies and the source SIB from Recommendation Q.1214.

2002 SSF Basic Call Process SIB

Detect Trigger Detection Point – Request (TDP-R)

- Send Initial DP or DP-specific Initial req. ind. and
- Suspend call processing

9001 SCF Basic Call Process SIB

Process Request

- Process initial information flow (e.g. Initial DP or DP-specific)

9081 SCF Screen SIB

- Process request for service logic
- Generate and send a Query req. ind.

4081 SDF Screen SIB

- Receive and analyse Query req. ind
- Retrieve data in the base
- Generate and send a Query Result resp. conf.

9083 SCF Screen SIB

- Receive Query Result resp. conf.
- Return response (match/no match) to service logic

9091 SCF Service Data Management SIB

- Process request for service logic
- Generate and send a Query req. ind.

4091 SDF Service Data Management SIB

- Receive and analyse Query req. ind
- Retrieve data in the base
- Generate and send a Query Result resp. conf.

9093 SCF Service Data Management SIB

- Receive Query Result resp. conf.
- Return response to service logic

9121 SCF User Interaction SIB

Initiate Request

- Initiate a Connect_To_Resource req. ind.

2121 SSF User Interaction SIB

Process Request

- Receive Connect_To_Resource req. ind from the SCF
- Analyse information (call involved, announcement address, routing requirements, etc.)
- Formulate and send a SETUP req. ind. to the SRF (if required)

3121 SRF User Interaction SIB

Process Request

- Receive and analyse SETUP req. ind. from CCF/SSF
- Select appropriate announcement resource

9122 SCF User Interaction SIB

Request Prompt/Collect Information or Announcement

– Receive a Connect_To_Resource resp. conf.

– Initiate a Prompt and Collect or Play Announcement req. ind. and send to the SRF

3122 SRF User Interaction SIB

Prompt/Play Announcement

– Receive and analyse Prompt and Collect or Play Announcement req. ind. from SCF

– Apply prompt/announcement on resource toward user

– Return SRF.RPT req. ind at conclusion of announcement if requested in Play Announcement req. ind.

9123 SCF User Interaction SIB

Initiate Disconnect

– Initiate a Disconnect_Forward_Connection req. ind and send to the CCF/SSF

2122 SSF User Interaction SIB

Disconnect Forward

– Receive DISC.FWD.CONN req. ind. from the SCF

– Formulate and send DISCONNECT req. ind. to SRF

3123 SRF User Interaction SIB

Process Request

– Receive and analyse DISCONNECT req. ind. from CCF/SSF

– Continue disconnect process per Recommendation Q.71

9003 SCF Basic Call Process SIB

Initiate Request

– Send one or more BCP information flows

20013 SSF Basic Call Process SIB

Process Request Report BCSM Event req. ind.

– Arm EDP(s)

9021 SCF Charge SIB (type 1)

Initiate Request

– Initiate a Furnish Charging Information req. ind. or

– Initiate an Apply Charging req. ind. and await a response or

– Initiate a Request Notification Charging Event req. ind. and await a response

2021 SSF Charge SIB (type 1)

Process Furnish Charging Information req. ind.

– Receive and analyse Furnish Charging Information req. ind.

– Apply specified Furnish Charging Information procedures.

9003 SCF Basic Call Process SIB

Initiate Request

– Send one or more BCP information flows

20011 SSF Basic Call Process SIB

Process Connect or Proceed with Call Processing req. ind.

2003 SSF Basic Call Process SIB

Detect Event Detection Point – Notification (EDP-N)

– Send Event Report BCSM or DP-specific report req. ind.

9002 SCF Basic Call Process SIB

Process Request and Send Immediate Instruction

– Process initial or report information flow (e.g. Initial DP, DP specific, or Event Report BCSM) and send one or more BCP information flows in response.

9092 SCF Service Data Management SIB

– Process request from service logic

– Generate and send an Update Data req. ind.

4092 SDF Service Data Management SIB

– Receive and analyse Update Data req. ind.

– Execute specified action in the base

– Process and return result

– Generate and send an Update Confirmation resp. conf.

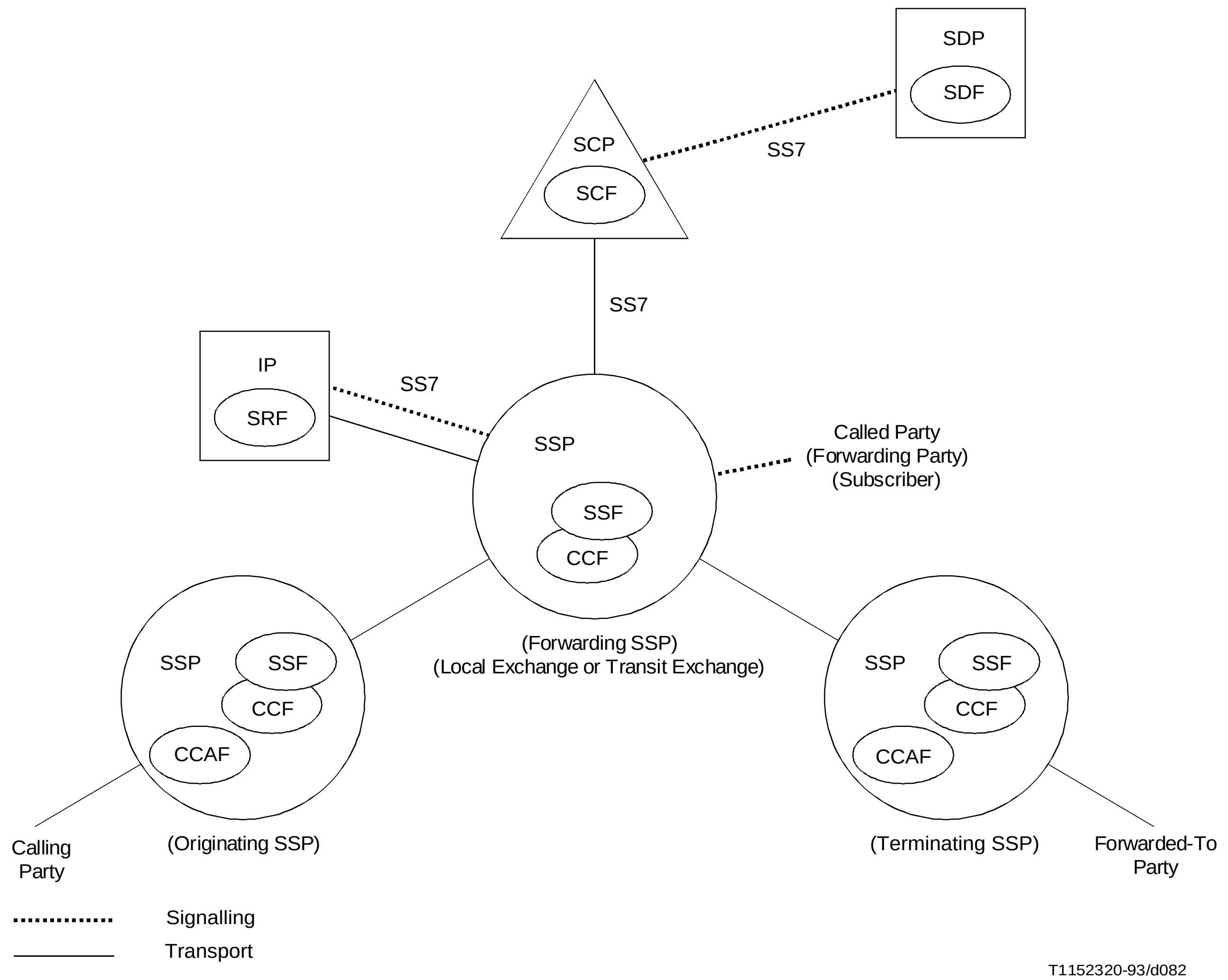
9094 SCF Service Data Management SIB

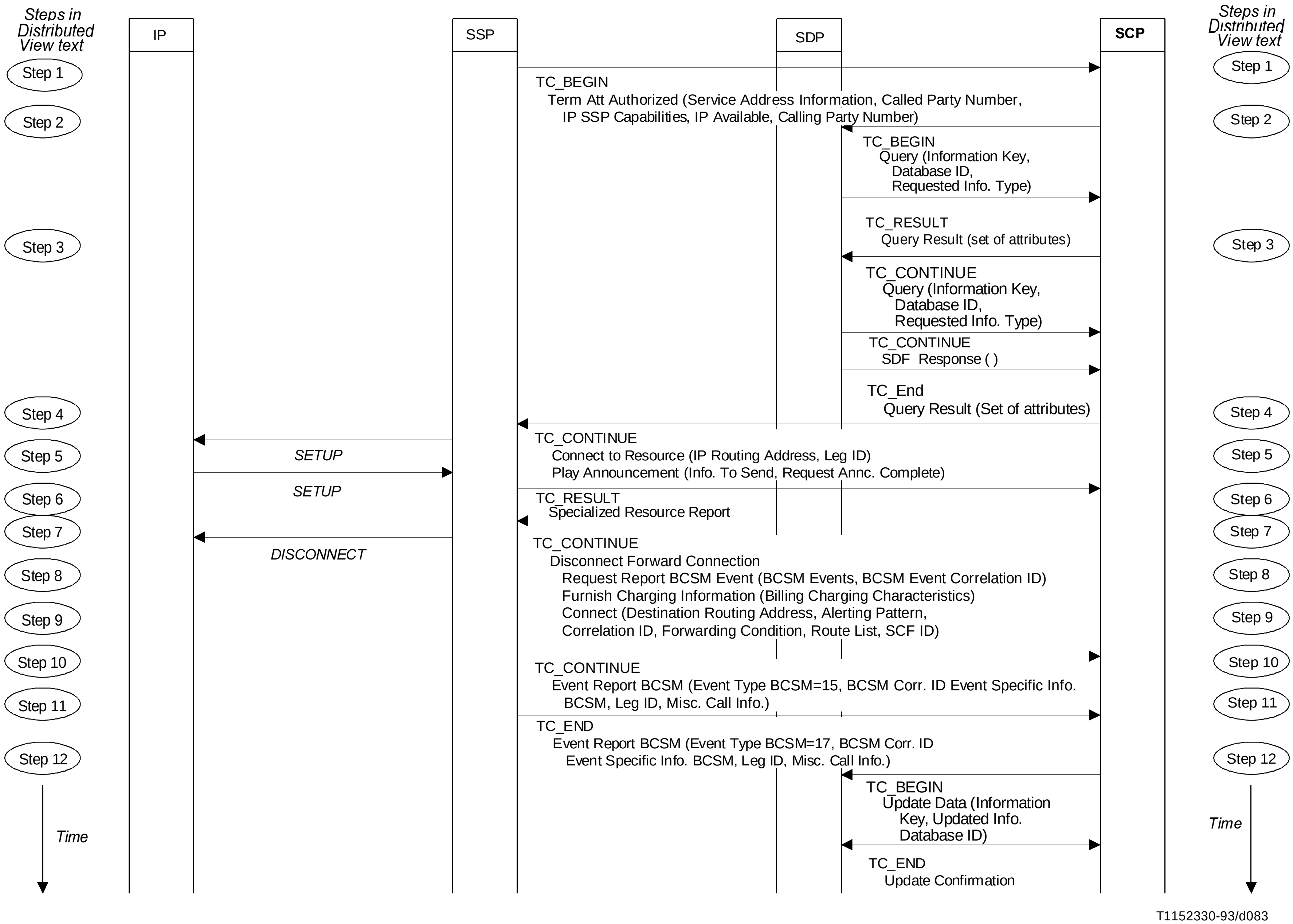
– Receive Update Confirmation resp. conf.

– Return response to service logic

5) Physical View

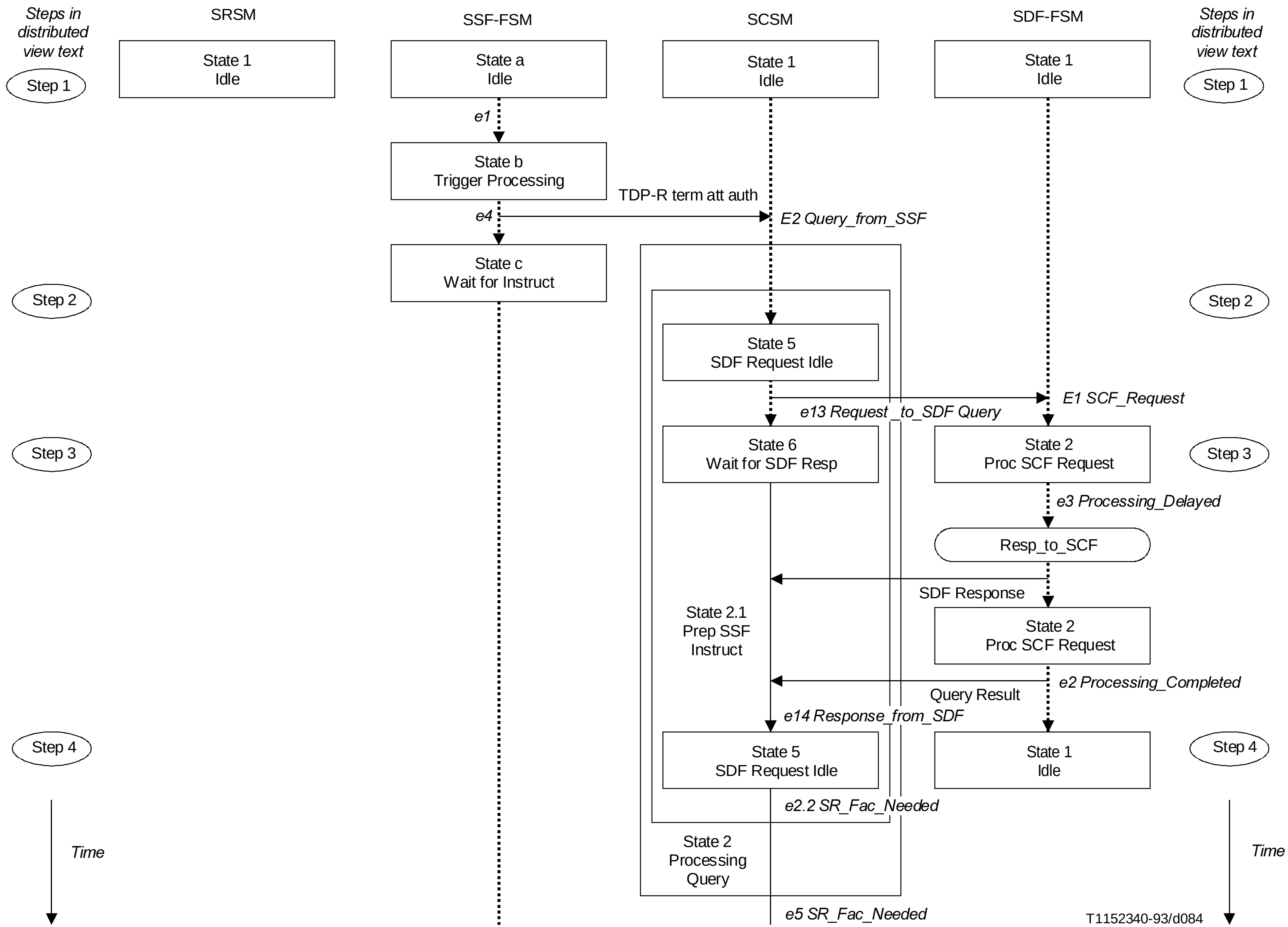
A) Physical Entity Interface Diagram (refer to Recommendation Q.1215)



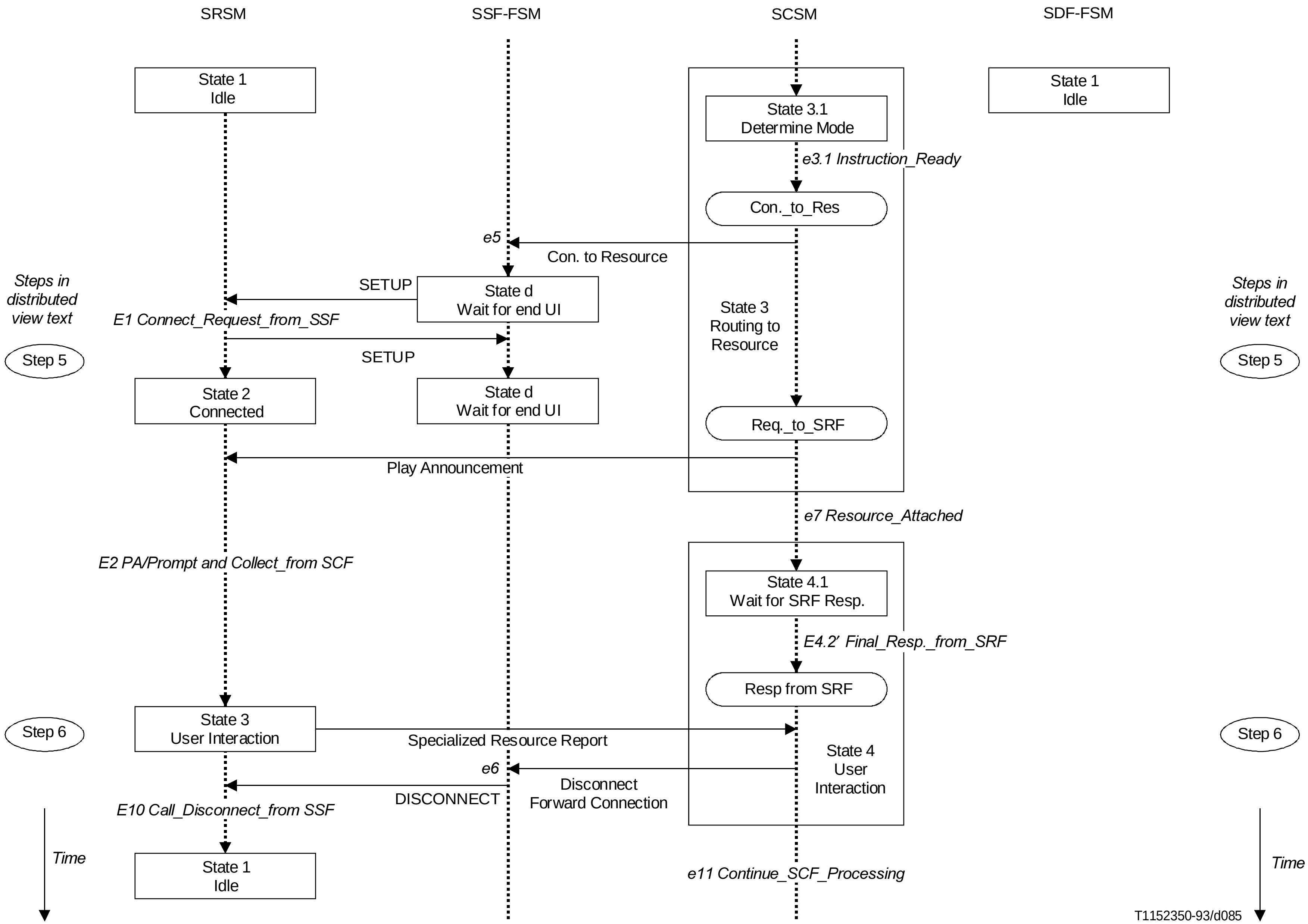


B) Time Sequence Diagram (refer to Recommendation Q.1218)

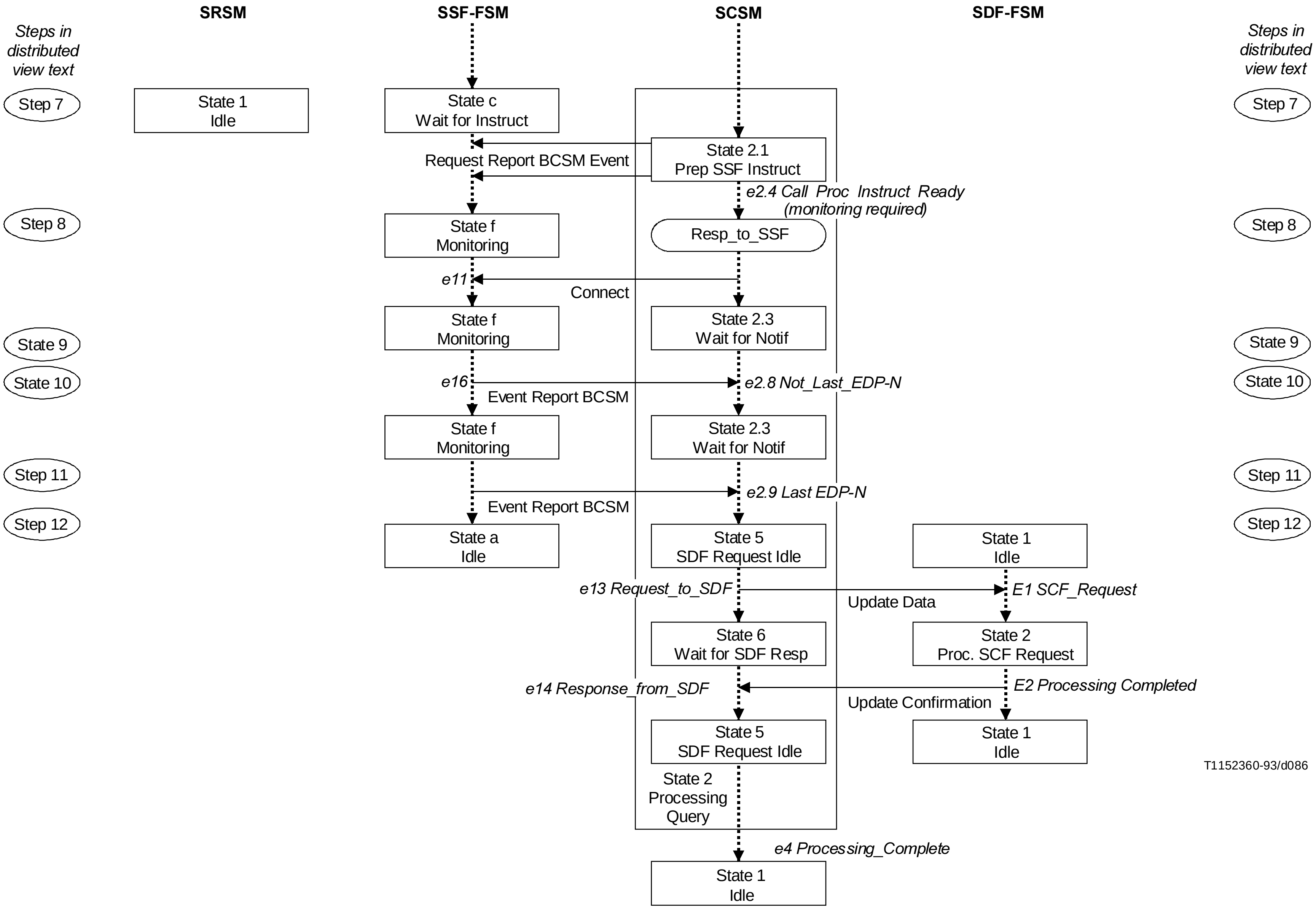
C) Application Entity (AE) Procedures (refer to Recommendation Q.1218)



(Sheet 1 of 3)


SRSM
SSF-FSM
SCSM
SDF-FSM
State 1
Idle
State 3.1
Determine Mode
State 1
Idle
e3.1 Instruction_Ready
Con._to_Res
e5
Con. to Resource
Steps in distributed view text
SETUP
State d
Wait for end UI
State 3
Routing to
Resource
Steps in distributed view text
E1 Connect_Request_from_SSF
SETUP
Step 5
Step 5
State 2
Connected
State d
Wait for end UI
Req._to_SRF
Play Announcement
e7 Resource_Attached
E2 PA/Prompt and Collect_from SCF
State 4.1
Wait for SRF Resp.
E4.2' Final_Resp._from_SRF
Resp from SRF
Step 6
State 3
User Interaction
Step 6
Specialized Resource Report
e6
State 4
User
Interaction
Disconnect
Forward Connection
DISCONNECT
E10 Call_Disconnect_from SSF
Time
State 1
Idle
e11 Continue_SCF_Processing
Time
T1152350-93/d085

(Sheet 2 of 3)


SRSM
SSF-FSM
SCSM
SDF-FSM
Steps in distributed view text
Steps in distributed view text
Step 7
State 1
Idle
State c
Wait for Instruct
State 2.1
Prep SSF Instruct
Step 7
Request Report BCSM Event
e2.4 Call Proc Instruct Ready
(monitoring required)
Step 8
State f
Monitoring
Resp_to_SSF
Step 8
e11
Connect
State f
Monitoring
State 2.3
Wait for Notif
State 9
State 9
State 10
e16
e2.8 Not_Last_EDP-N
State 10
Event Report BCSM
State f
Monitoring
State 2.3
Wait for Notif
Step 11
Step 11
e2.9 Last EDP-N
Event Report BCSM
Step 12
Step 12
State a
Idle
State 5
SDF Request Idle
State 1
Idle
e13 Request_to_SDF
E1 SCF_Request
Update Data
State 6
Wait for SDF Resp
State 2
Proc. SCF Request
e14 Response_from_SDF
E2 Processing Completed
Update Confirmation
State 5
SDF Request Idle
State 1
Idle
T1152360-93/d086
State 2
Processing
Query
e4 Processing_Complete
State 1
Idle

(Sheet 3 of 3)

A.7 Malicious Call Identification (MCID) Service Scenario

1) IN CS-1 Capabilities illustrated by MCID

This service scenario illustrates IN CS-1 Midcall capabilities, the ability to connect an SRF to a Called Party who is connected to an SSP either by lines or by trunks, the ability of an IN Network to initiate calls on the behalf of a Calling Party and the ability of an IN Network to end IN calls on the behalf of a Called Party or Calling Party. This scenario uses capabilities entirely within the scope of IN CS-1.

The Midcall capabilities require that an SSP notify an SCP of an event during the active phase of a call, on behalf of the Subscriber (the Called Party). A control relationship can then be established between the SSF and the SCF.

An SRF is used to play the announcement or tone to the Subscriber who, for the purposes of this scenario, is directly connected to the switch by a subscriber line.

An SRF is also used to play an announcement to the malicious caller who, for the purposes of this scenario, is connected to the SSP over a trunk. It is also possible for the malicious caller to be connected to the SSP over a line. Also for this scenario, the SRF is external to the switch and inside an Intelligent Peripheral (IP). The IP is directly connected to the switch, utilizing SS7 for signalling with voice lines for transport or utilizing ISDN.

It is possible for either the SRF or the SCF to terminate the connection between the SRF and the User. Both forms are illustrated in this scenario.

The scenario also illustrates a mechanism whereby IN service logic can cause an IN network to initiate IN calls on behalf of a Subscriber. Note that the subscriber could already be participating in a call. The calls will be separate and there will be no association between the calls.

It is also possible for IN service logic to end IN calls during any phase of the call. This scenario illustrates IN service logic ending a call during both the setup and active phases.

2) Textual Description of Service Scenario

2.1) General Considerations

Malicious Call Identification (MCID) allows the service subscriber, who is the Called Party, to request that the Calling Party Number of an incoming call be identified and registered in the network. The Called Party Number, time, date and requested features are also registered.

In this scenario, MCID is assumed to be the only feature, that is active, which responds to the hookflash.

The service has two options.

Option 1 – Network Returns Call

Option 1 is initiated by the Midcall actions (i.e. hookflash, feature buttons) of the subscriber. In this scenario, it is assumed that the subscriber hookflashes to initiate the Midcall event while the malicious call is taking place. The call information is registered in the network.

The subscriber is then presented with an announcement or tone, confirming that the malicious call is being registered in the network. The network then ends the call.

Finally, the network places a return call to the malicious caller and play an announcement to discourage future malicious calls.

Option 2 – Network Authority Notification

Option 2 is also initiated by the Midcall actions of the subscriber. The call information is registered with the network and the subscriber is given a confirmation tone. In this option, the call remains active.

The network then initiates a call to the Network Authority and play an announcement to report the malicious call that is currently taking place.

2.2) Call Party Perspective

Steps Common to Both Options

Step 1: Calling Party (malicious caller) dials the directory number of the service subscriber.

Step 2: An active two party call is established between the malicious caller and the subscriber.

Step 3: The subscriber hookflashes after determining the call to be malicious.

Option 1 – Network Returns Call

Step 4: The subscriber hears an announcement which confirms that the Malicious Call Identification service is being invoked. The malicious caller hears nothing and the malicious caller's phone goes idle.

Step 5: The subscriber's phone goes idle.

Step 6: The malicious caller's phone rings.

Step 7: The malicious caller answers the phone.

Step 8: An announcement is played to the malicious caller to discourage future malicious calls.

Option 2 – Network Authority Notification

Step 4: A short tone is played to the subscriber, indicating that the Malicious Call Identification service is being invoked. The malicious caller hears nothing.

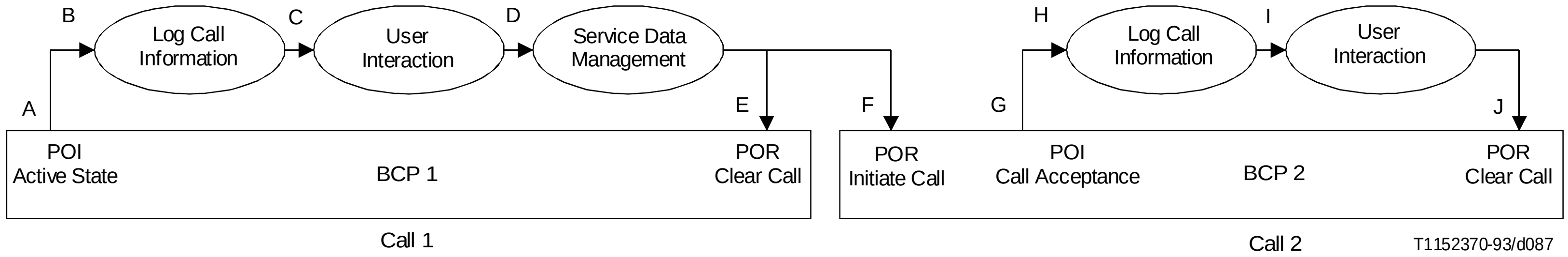
Step 5: The malicious caller and subscriber continue to participate in a two party call.

Step 6: The Network Authority's phone rings.

Step 7: The Network Authority answers.

Step 8: An announcement is played to the Network Authority, alerting the authority to the ongoing malicious call.

Option 1: Network Returns Call (refer to Recommendation Q.1213)



A *Basic Call Process SIB*

The switch determines that IN actions are required.

POI – Active State
CID – Calling Line Identity (1)
– Called Party ID (2)

B *Log Call Information SIB*

The start time, stop time, Calling Line ID, Called Party ID are stored.

Input:
SSD – CIDFP Log = Call start time
= Call connect time
= Calling Line ID
CID – Identifier = Call start time
= Call connect time
= Called Party ID (2)
Output:
Logical End – Success

C *User Interaction SIB*

Confirmation tone/annc. played to Called Party (Subscriber).

Input:
SSD – Announcement Parameters
= Announcement ID
CID – Call Party Identifier
= Called Party ID (2)
Output:
Logical End – Success

D *Service Data Management SIB*

MCID Option 1 is determined.

Input:
SSD – File Indicator = Called Party ID (2)
– Action = Retrieve
CID – Element Indicator
Output:
Logical End – Success
CID – Data Retrieved = Option 1 (3)
= Called Party (4)

E *Basic Call Process SIB*

Call between malicious caller and subscriber is terminated.

POR – Clear Call

F *Basic Call Process SIB*

A new call instance is created, to malicious caller, by network on behalf of subscriber.

POR – Initiate Call
CID – Called Party ID (4)

G *Basic Call Process SIB*

Switch determines that IN actions are required.

POI – Call Acceptance
CID – Called Party ID (4)

H *Log Call Information SIB*

The occurrence of a successful call attempt to malicious caller, is stored.

Input:
SSD – CIDFP Log = Call start time
= Call connect time
= Calling Line ID
CID – Identifier = Call start time
= Call connect time
= Called Party ID (4)
Output:
Logical End – Success

I *User Interaction SIB*

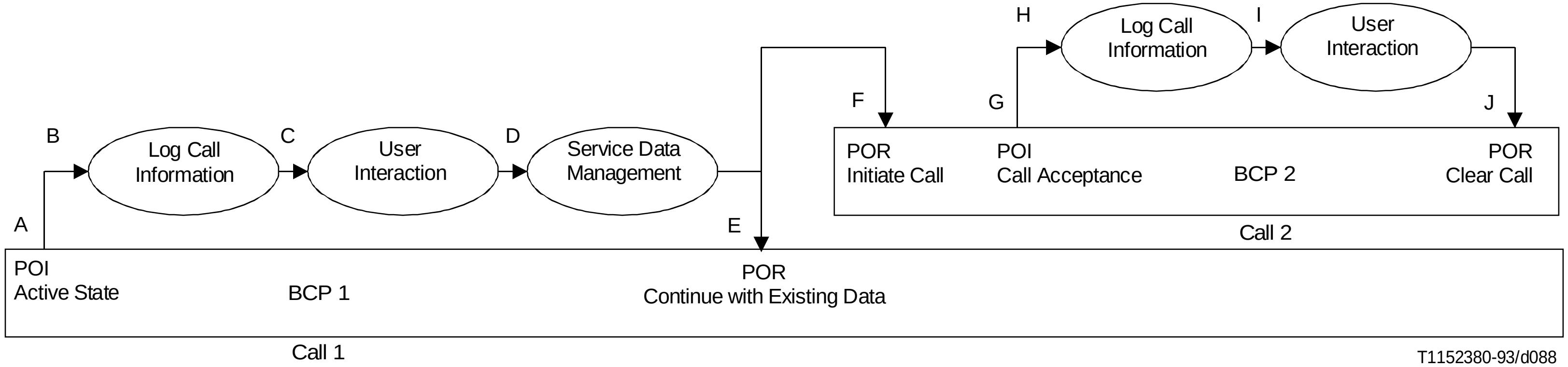
Announcement is played to called party (malicious caller).

Input:
SSD – Announcement Parameters
= Announcement ID
CID – Call Party Identifier
= Called Party ID (4)
Output:
Logical End – Success

J *Basic Call Process SIB*

Call to malicious caller is terminated.

POR – Clear Call



A ***Basic Call Process SIB***

The switch determines that IN actions are required.

POI – Active State

CID – Calling Line Identity (1)
– Called Party ID (2)

B ***Log Call Information SIB***

The start time, stop time, Calling Line ID, Called Party ID are stored.

Input:
SSD – CIDFP Log = Call start time
= Call connect time
= Calling Line ID
CID – Identifier = Call start time
= Call connect time
= Called Party ID (2)
Output:
Logical End – Success

C ***User Interaction SIB***

Confirmation tone/annc. played to Called Party (Subscriber).

Input:
SSD – Announcement Parameters
= Announcement ID
CID – Call Party Identifier
= Called Party ID (2)
Output:
Logical End – Success

D ***Service Data Management SIB***

MCID Option 2 is determined.

Input:
SSD – File Indicator = Called Party ID (2)
– Action = Retrieve
CID – Element Indicator
Output:
Logical End – Success
CID – Data Retrieved = Option 2 (3)
= Called Party (4)

E ***Basic Call Process SIB***

Call between malicious caller and subscriber continues.

POR – Continue with Existing Data

F ***Basic Call Process SIB***

A new call instance is created to network authority, on behalf of subscriber to inform of ongoing malicious call.

POR – Initiate Call
CID – Called Party ID (4)

G ***Basic Call Process SIB***

Switch determines that IN actions are required.

POI – Call Acceptance
CID – Called Party ID (4)

H ***Log Call Information SIB***

The occurrence of a successful call attempt, to network authority, is stored.

Input:
SSD – CIDFP Log = Call start time
= Call connect time
= Calling Line ID
CID – Identifier = Call start time
= Call connect time
= Called Party ID (4)
Output:
Logical End – Success

I ***User Interaction SIB***

Announcement is played to called party (network authority).

Input:
SSD – Announcement Parameters
= Announcement ID
CID – Call Party Identifier
= Called Party ID (4)
Output:
Logical End – Success

J ***Basic Call Process SIB***

Call to network authority is terminated.

POR – Clear Call

4) Distributed View

Network Perspective Description – Steps Common to Both Options

Step 1: The call is initiated by the malicious caller.

Step 2: The call is routed through the network, from a switch that is local to the caller to a switch that is local to the subscriber.

Step 3: The subscriber is alerted that there is an incoming call.

Step 4: An active call is established between the malicious caller and the subscriber.

Step 5: The midcall trigger is recognized by the SSF.

Option 1 – Network Returns Call

Step 6: The database item, associated with the subscriber, is updated by the SCF to contain the parameters of the current call.

Step 7: The SCF requests that the SSF connect the subscriber to an SRF.

Step 8: The SCF requests that the SRF play an announcement to the subscriber.

Step 9: The SRF plays an announcement to the subscriber.

Step 10: The SRF causes the termination of the connection between the SRF and the subscriber.

Step 11: The SCF causes the call to be terminated.

Step 12: The network initiates a call attempt to the malicious caller on behalf of the subscriber.

Step 13: The call is routed through the network from a switch that is local to the subscriber to a switch that is local to the malicious caller.

Step 14: The malicious caller is alerted that there is an incoming call.

Step 15: The SSF notifies the SCF as the malicious caller goes off hook.

Step 16: The SCF updates the subscriber’s record that the return call was successfully placed to the malicious caller.

Step 17: The SCF requests that the SSF connect the malicious caller to an SRF.

Step 18: The SCF requests that the SRF play an announcement to the malicious caller.

Step 19: The SRF plays an announcement to the malicious caller.

Step 20: The SRF causes the termination of the connection between the SRF and the malicious caller.

Step 21: The SCF terminates the call to the malicious caller.

Option 2 – Network Authority Notification

Step 6: The database item, associated with the subscriber, is updated by the SCF to contain the parameters of the current call.

Step 7: The SCF requests that the SSF connect the subscriber to an SRF.

Step 8: The SCF requests that the SRF play a tone to the subscriber.

Step 9: The SRF plays a tone to the subscriber.

Step 10: The SCF causes the tone to terminate and the connection between the SRF and the subscriber to terminate.

Step 11: The malicious caller and the subscriber resume a normal two party call.

Step 12: The network initiates a call attempt to the Network Authority on behalf of the subscriber.

Step 13: The call is routed through the network from a switch that is local to the subscriber to a switch that is local to the Network Authority.

Step 14: The Network Authority is alerted that there is an incoming call.

Step 15: The SSF notifies the SCF as the Network Authority goes off hook.

Step 16: The SCF updates the subscriber's record that the call was successfully placed to the Network Authority.

Step 17: The SCF requests that the SSF connect the Network Authority to an SRF.

Step 18: The SCF requests that the SRF play an announcement to the Network Authority.

Step 19: The SRF plays an announcement to the Network Authority.

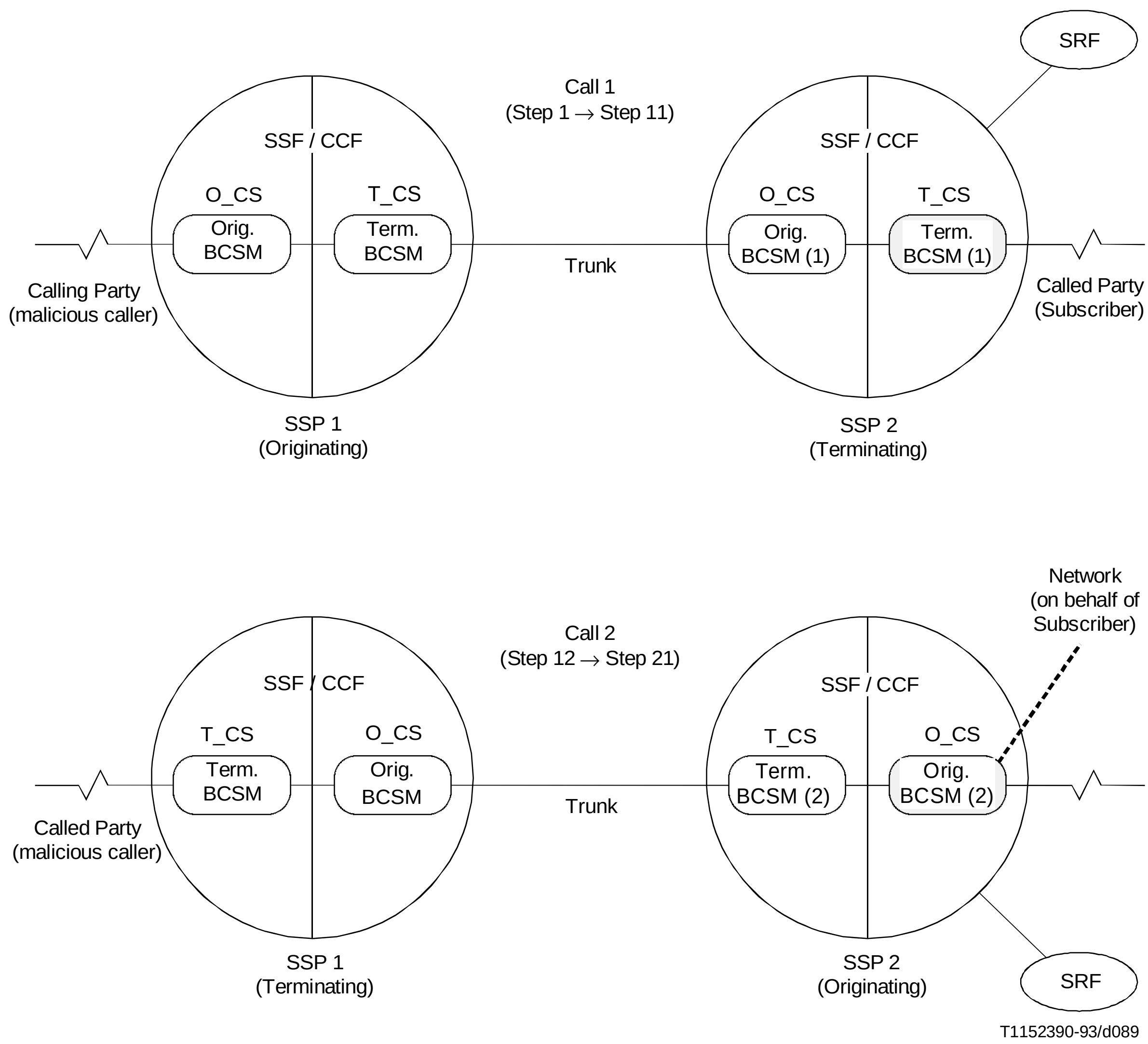
Step 20: The SRF causes the termination of the connection between the SRF and the Network Authority.

Step 21: The SCF terminates the call to the Network Authority.

A) Call Segment Diagram (refer to Recommendation Q.1214)

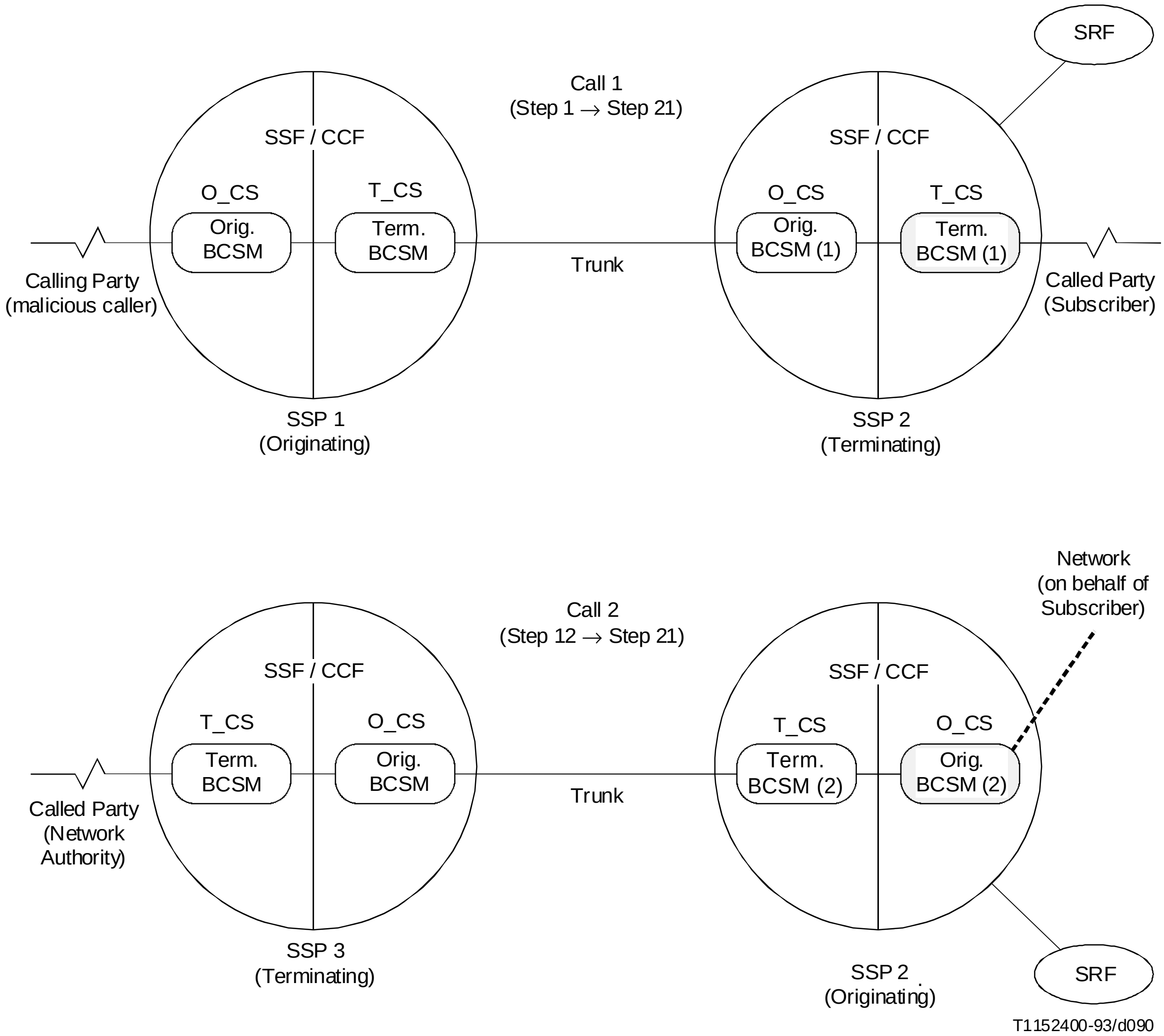
1) Network Overview of Call Segments

Option 1 – Network Returns Call



NOTE – Service logic is for shaded CSs.

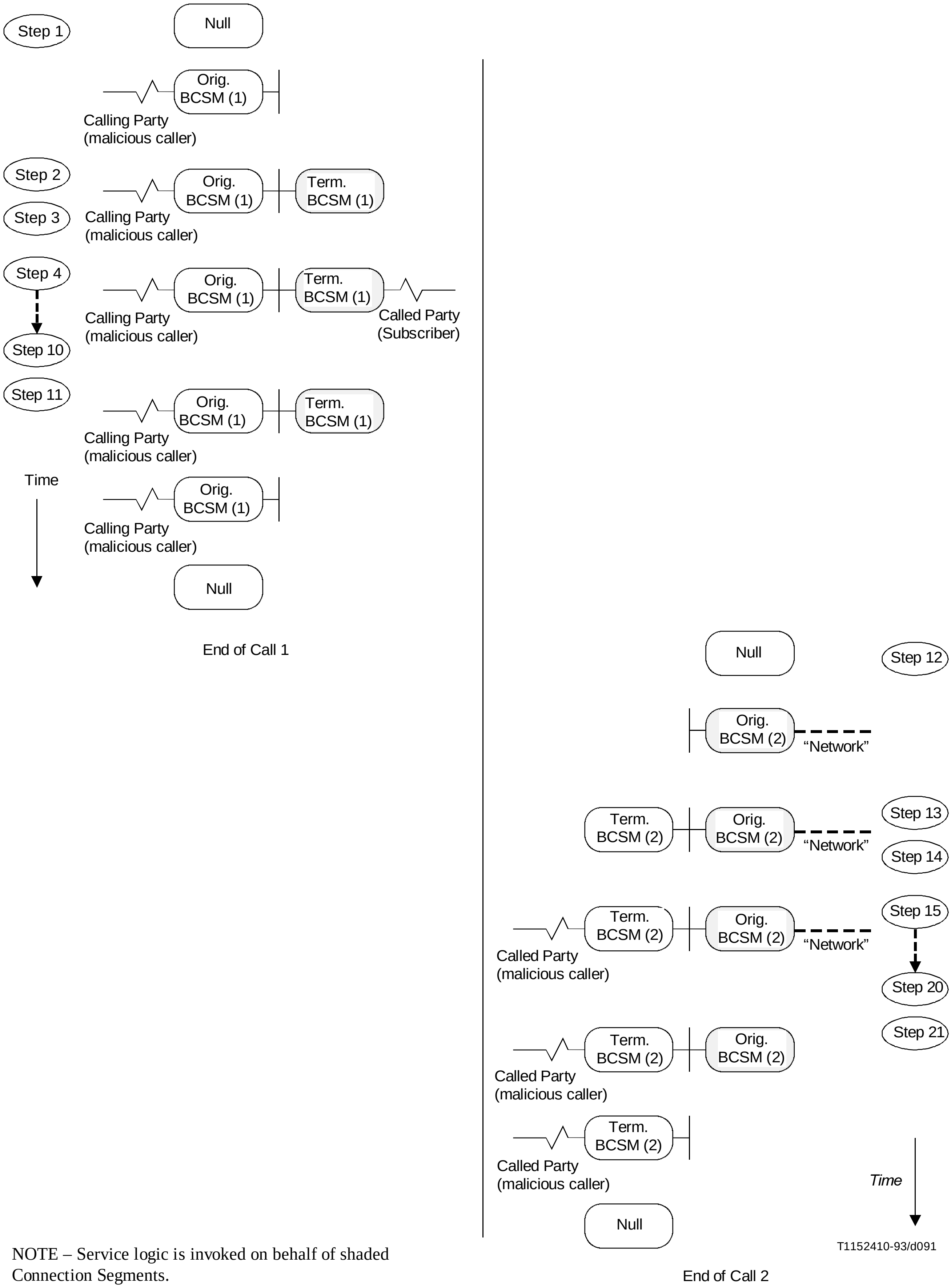
Option 2 – Network Authority Notification



NOTE – Service logic is for shaded CSs.

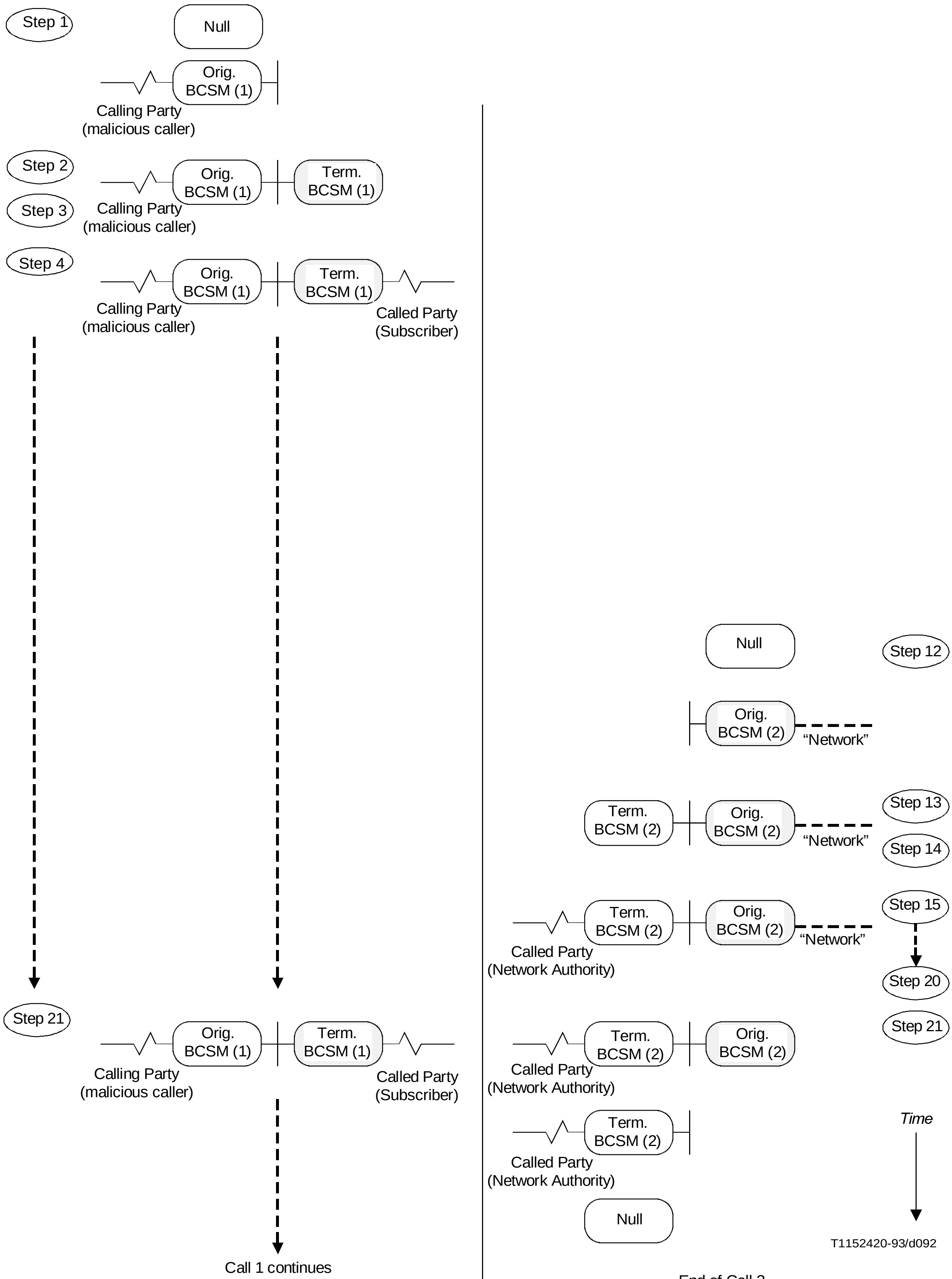
2) Call Segments in SSP2 (refer to Recommendation Q.1214)

Option 1 – Network Returns Call



NOTE – Service logic is invoked on behalf of shaded Connection Segments.

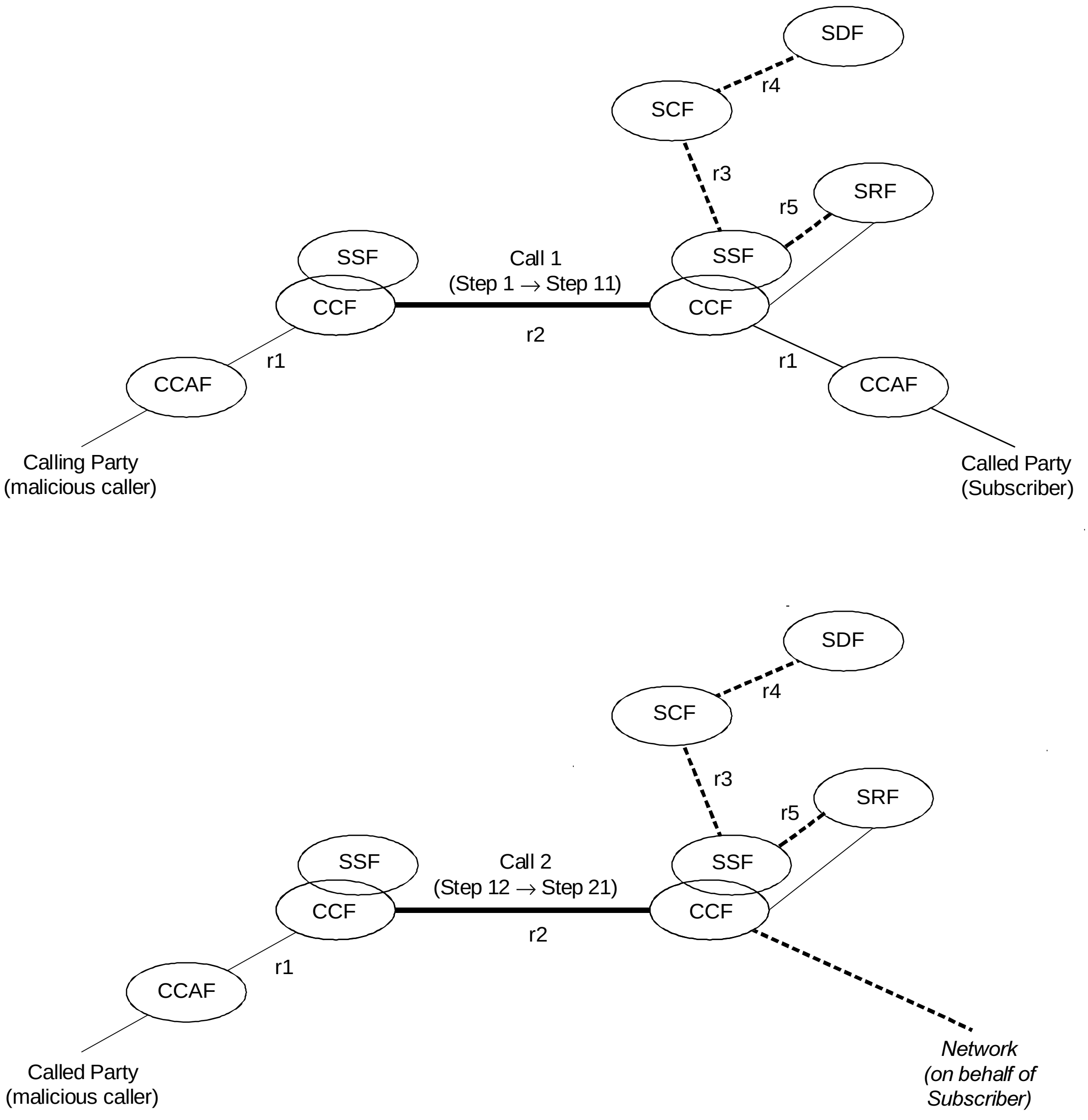
Option 2 – Network Authority Notification



NOTE – Service logic is invoked on behalf of shaded Connection Segments.

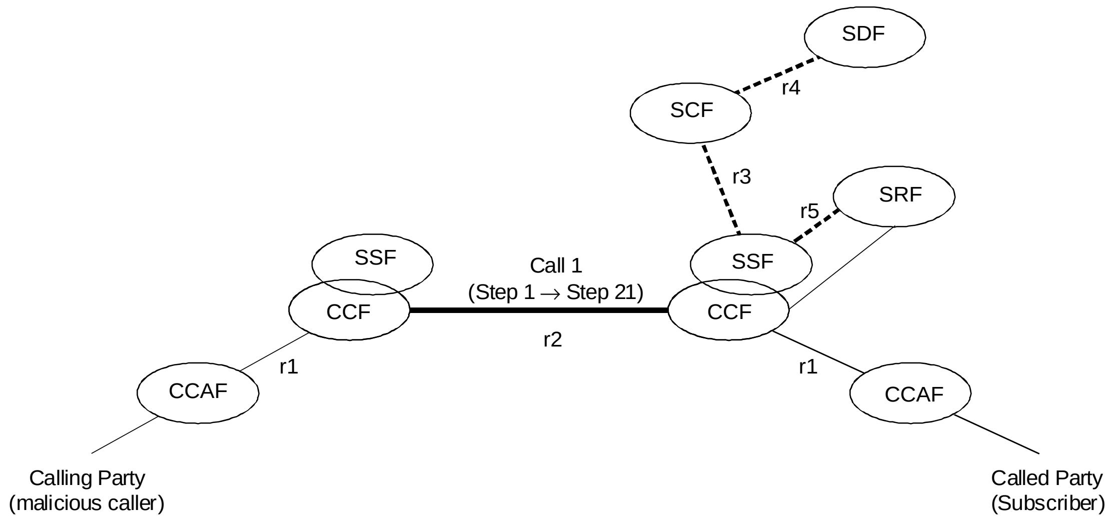
B) Functional Entity Interface Diagram (refer to Recommendation Q.1214)

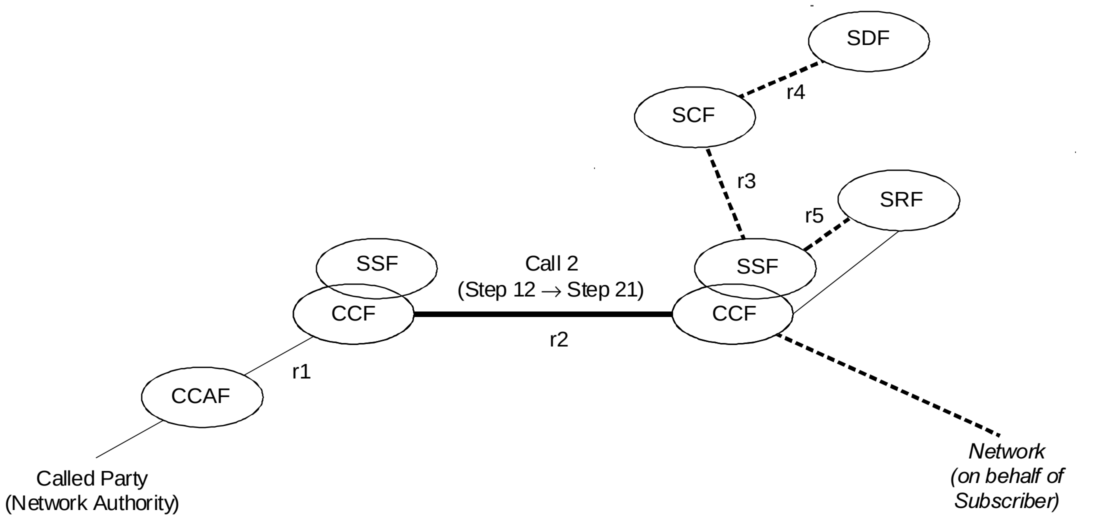
Option 1 – Network Returns Call



T1152430-93/d093

Option 2 – Network Authority Notification

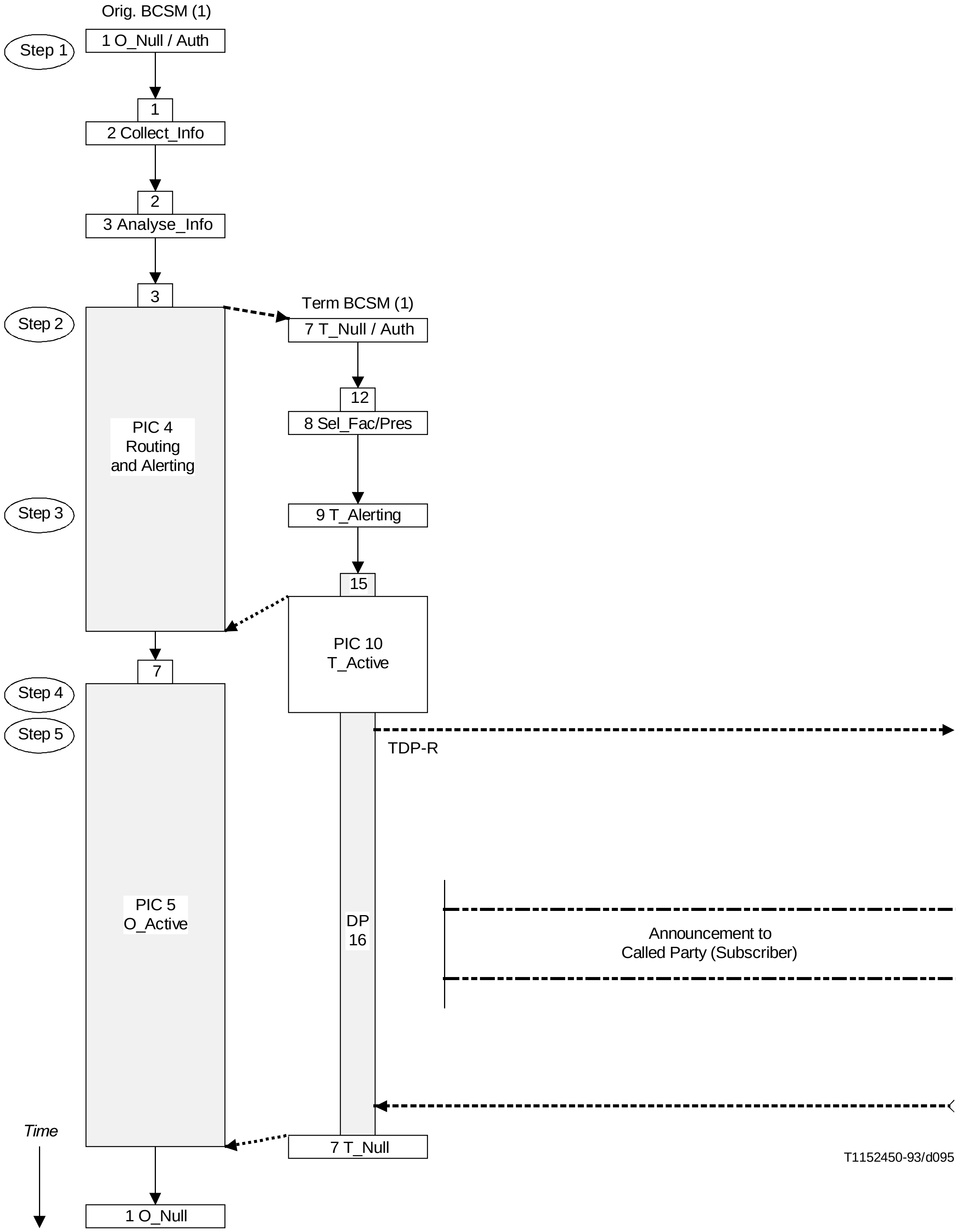




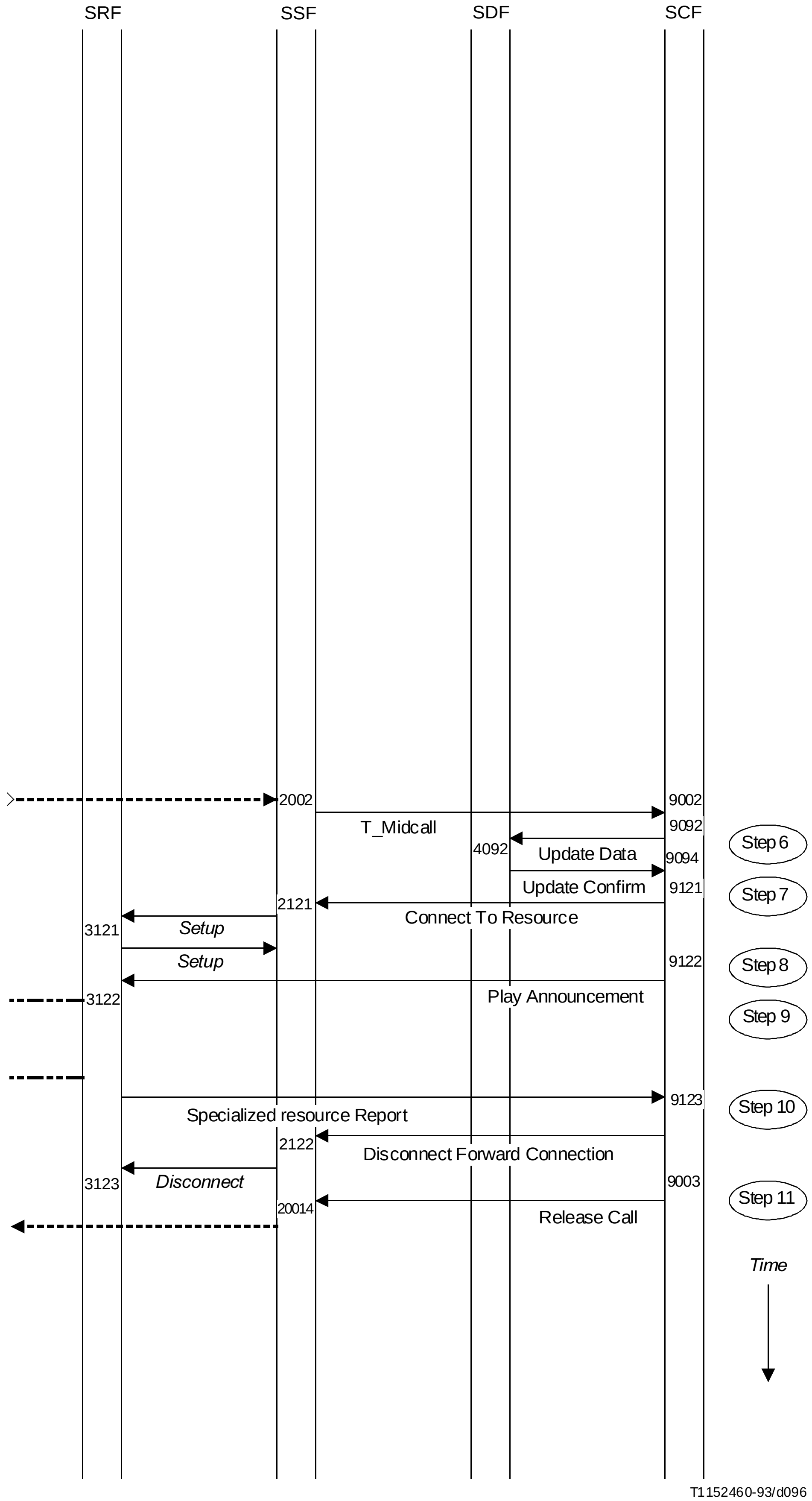
T1152440-93/d094

C) Information Flow Sequence Diagram (refer to Recommendation Q.1214)

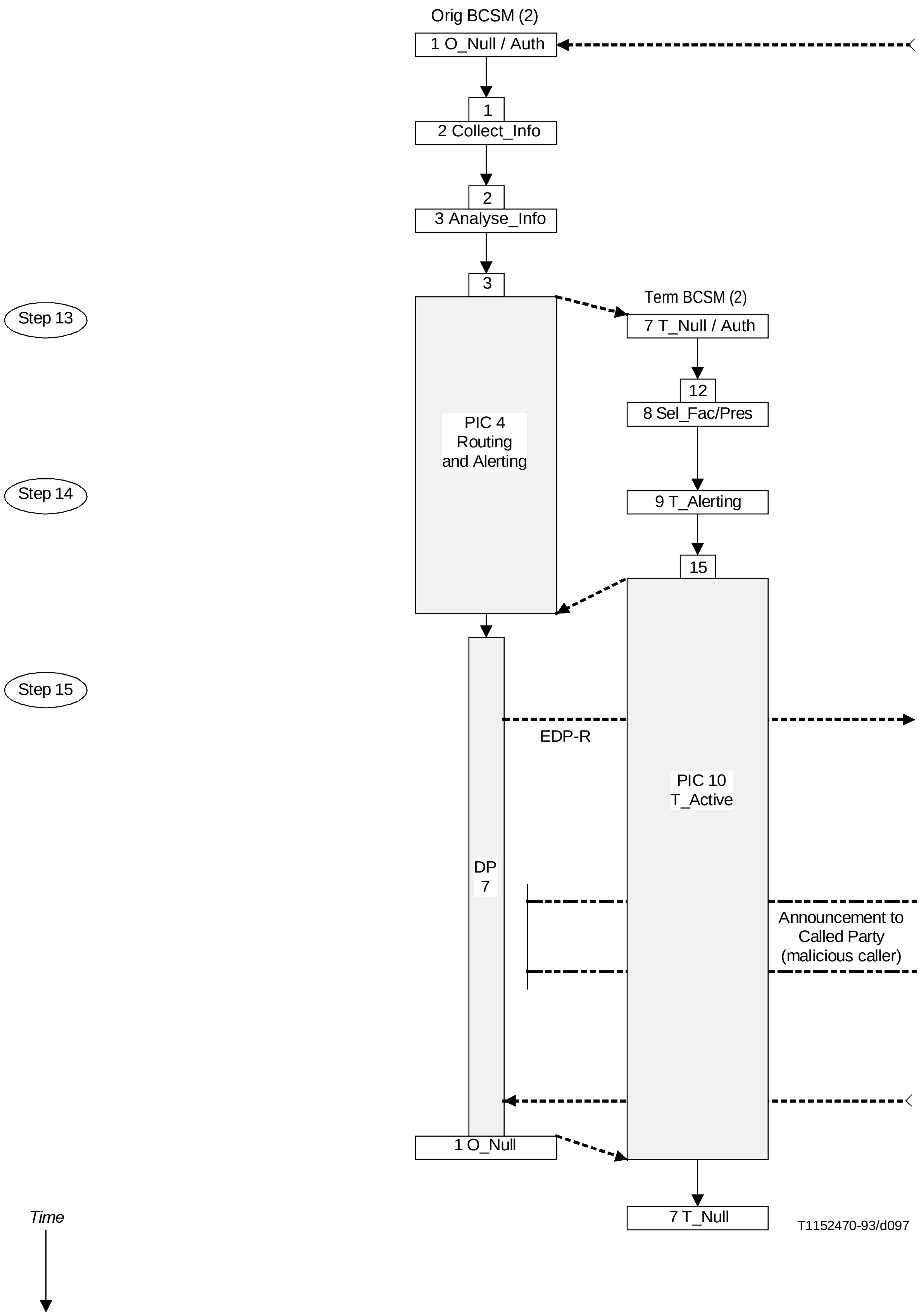
Option 1 – Network Returns Call



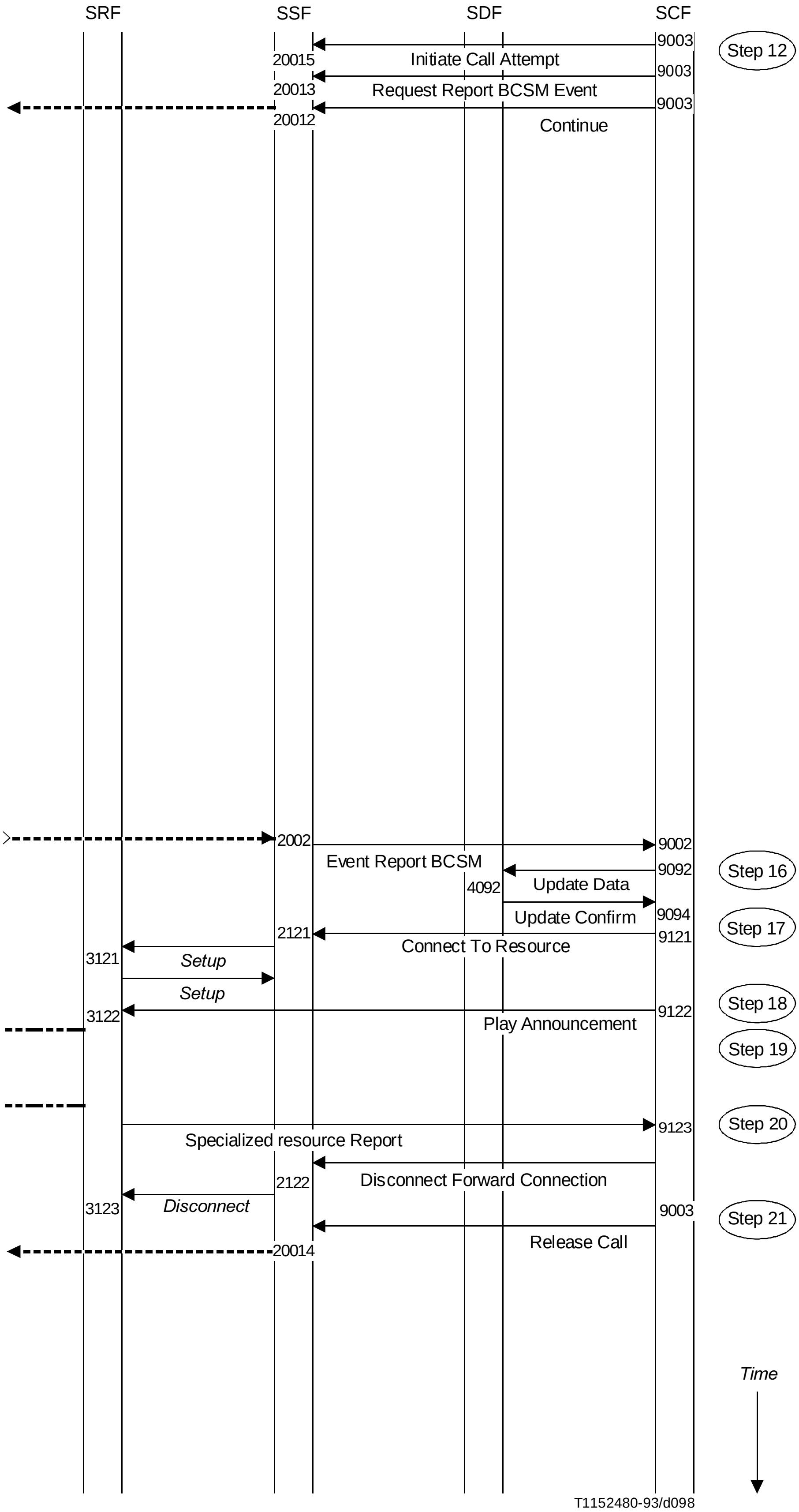
(Sheet 1 of 4)


SRF
SSF
SDF
SCF
2002
9002
T_Midcall
9092
4092
Update Data
9094
Update Confirm
9121
2121
Connect To Resource
3121
Setup
Setup
9122
3122
Play Announcement
9123
Specialized resource Report
2122
Disconnect Forward Connection
3123
Disconnect
9003
20014
Release Call
Step 6
Step 7
Step 8
Step 9
Step 10
Step 11
Time
T1152460-93/d096

(Sheet 2 of 4)


Orig BCSM (2)
1 O_Null / Auth
1
2 Collect_Info
2
3 Analyse_Info
3
Step 13
Term BCSM (2)
7 T_Null / Auth
12
8 Sel_Fac/Pres
PIC 4
Routing
and Alerting
Step 14
9 T_Alerting
15
Step 15
EDP-R
PIC 10
T_Active
DP
7
Announcement to
Called Party
(malicious caller)
1 O_Null
Time
7 T_Null
T1152470-93/d097

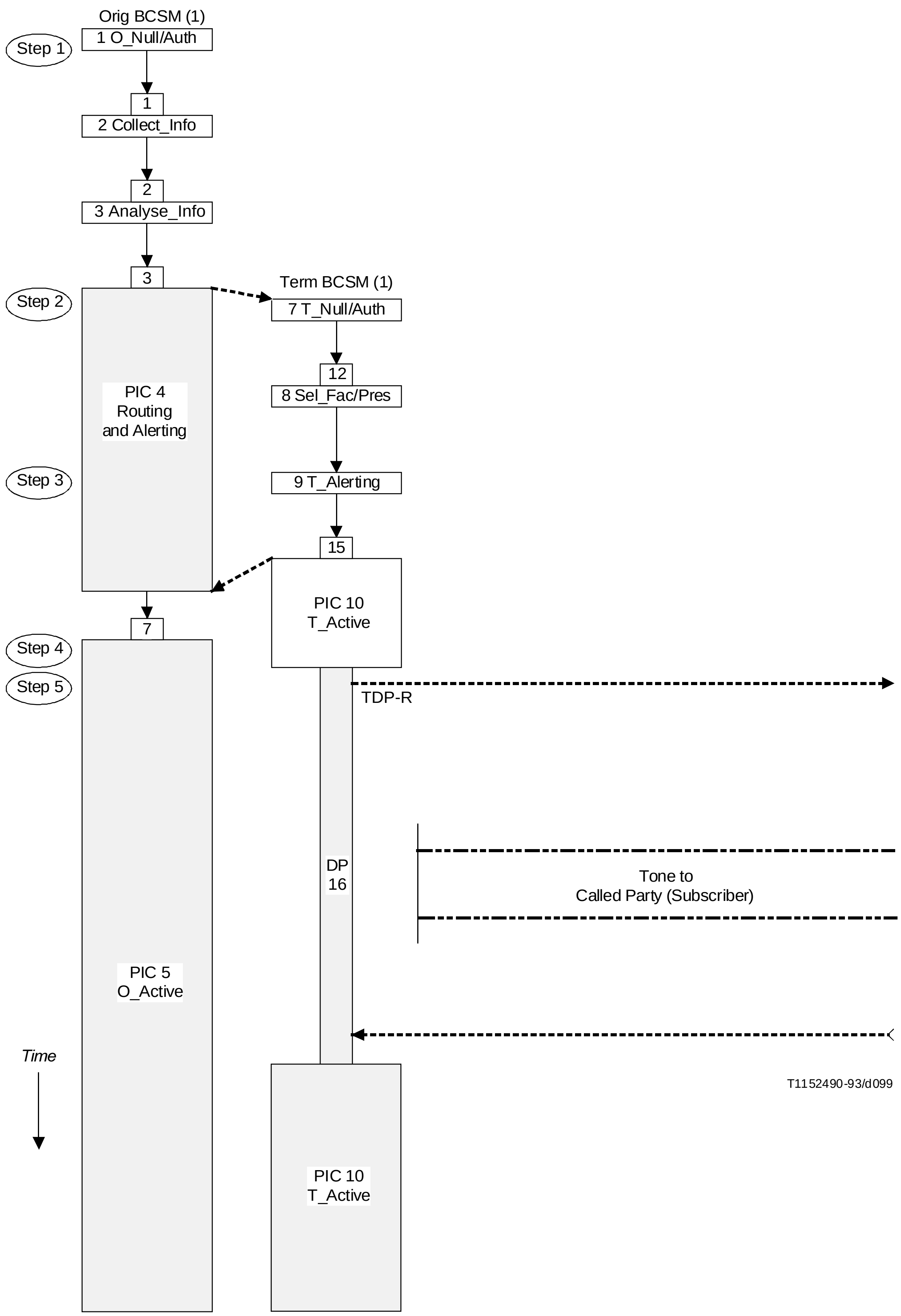
(Sheet 3 of 4)



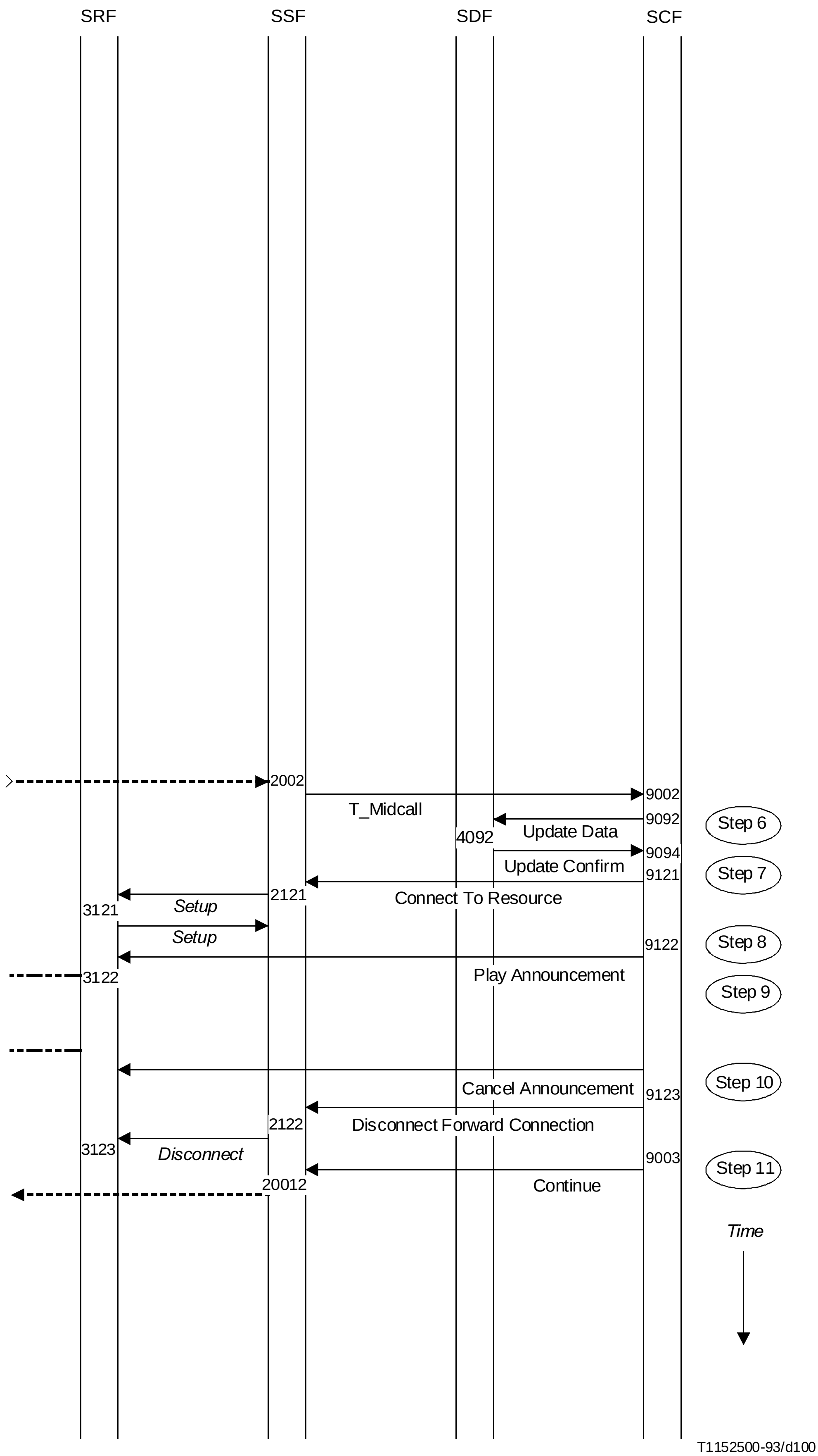
NOTE – Shaded PICs or DPs denote a pause in transitions for a given BCSM while other IN actions occur.

(Sheet 4 of 4)

Option 2 – Network Authority Notification

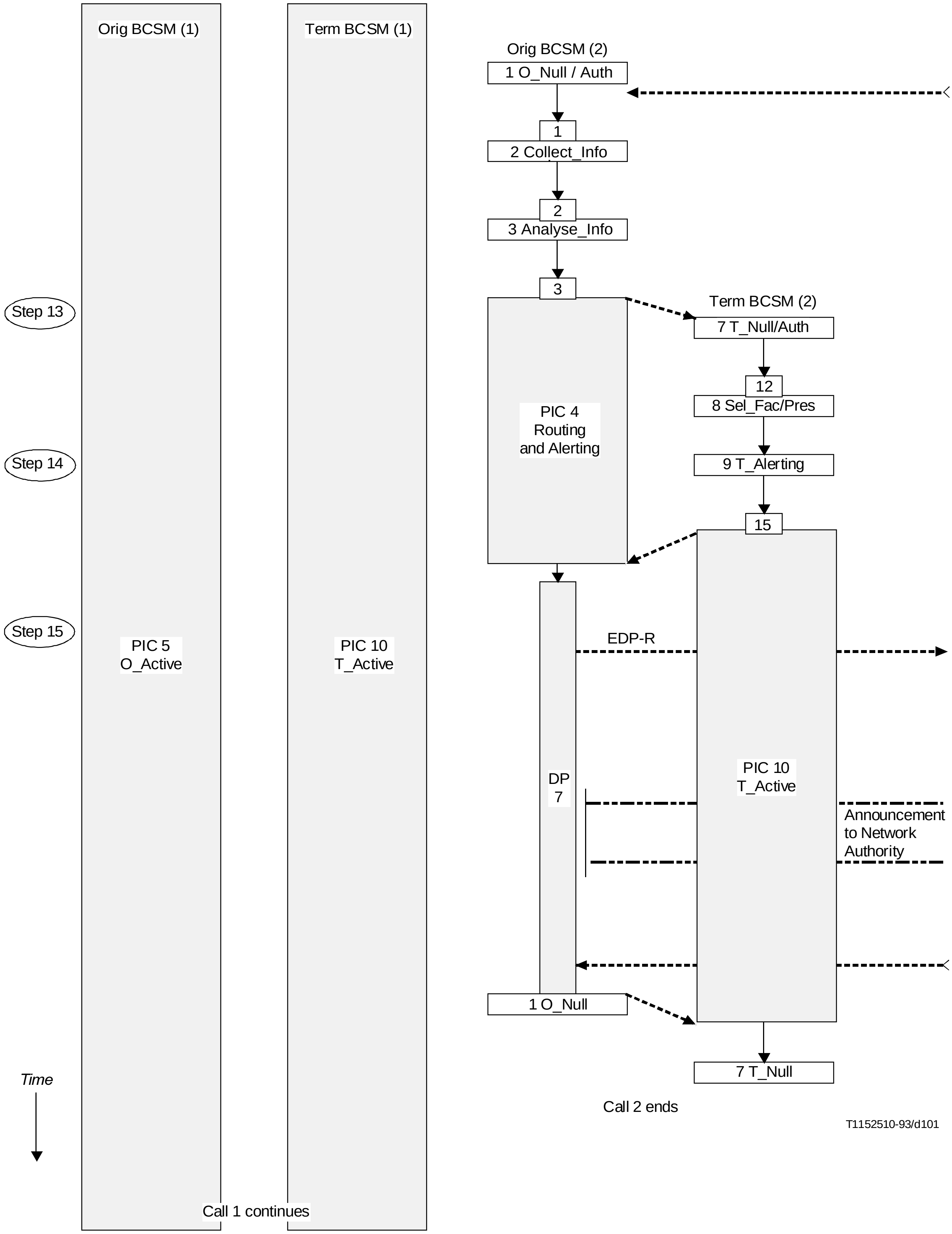


(Sheet 1 of 4)

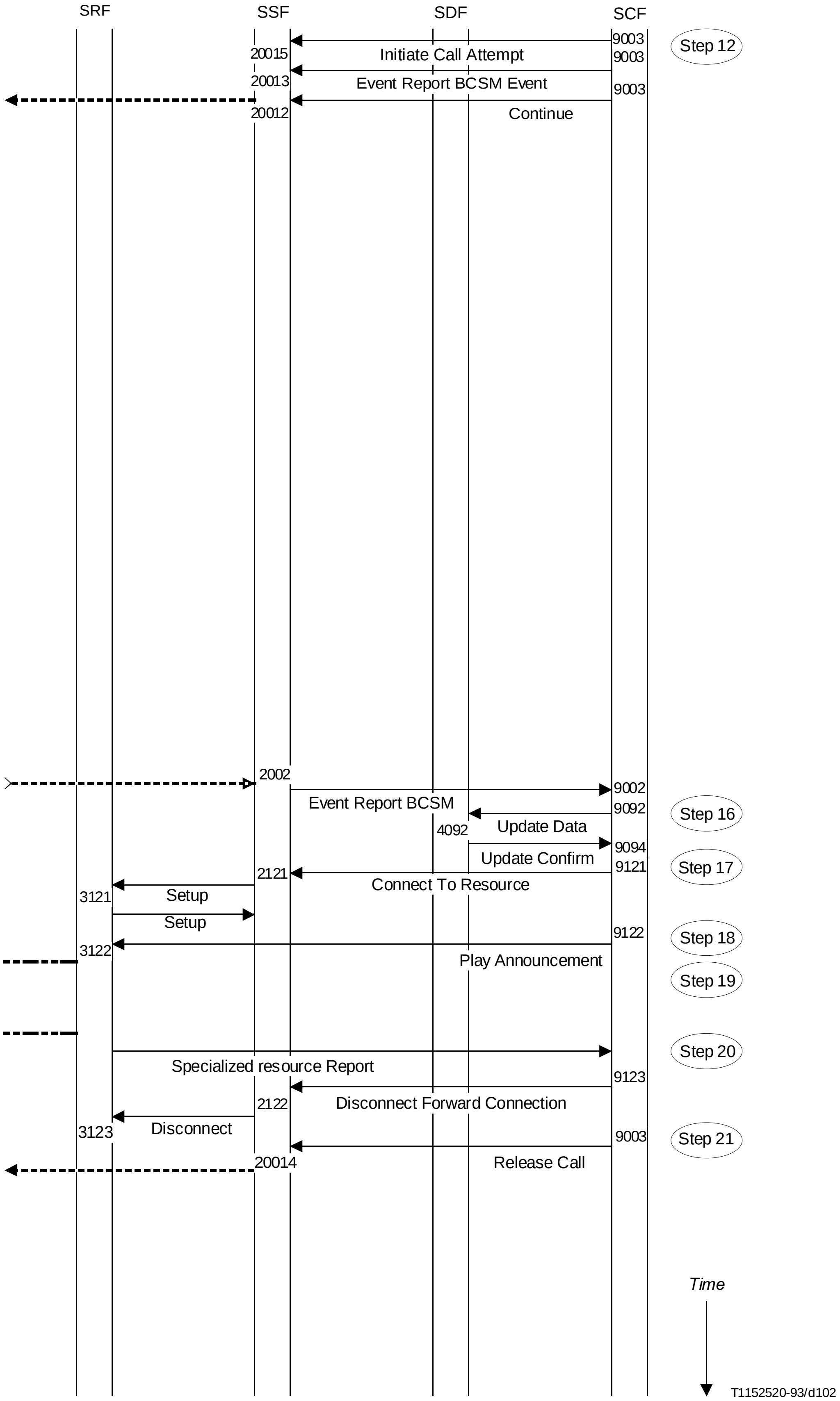


SRF
SSF
SDF
SCF
2002
9002
T_Midcall
9092
Step 6
4092
Update Data
9094
Update Confirm
9121
Step 7
2121
Connect To Resource
3121
Setup
Setup
9122
Step 8
3122
Play Announcement
Step 9
Cancel Announcement
9123
Step 10
2122
Disconnect Forward Connection
3123
Disconnect
9003
Step 11
20012
Continue
Time
T1152500-93/d100

(Sheet 2 of 4)


Orig BCSM (1)
Term BCSM (1)
Orig BCSM (2)
1 O_Null / Auth
1
2 Collect_Info
2
3 Analyse_Info
3
Step 13
Term BCSM (2)
7 T_Null/Auth
12
8 Sel_Fac/Pres
PIC 4
Routing
and Alerting
Step 14
9 T_Alerting
15
Step 15
PIC 5
O_Active
PIC 10
T_Active
EDP-R
DP
7
PIC 10
T_Active
Announcement
to Network
Authority
1 O_Null
7 T_Null
Time
Call 2 ends
T1152510-93/d101
Call 1 continues

(Sheet 3 of 4)



NOTE – Shaded PICs or DPs denote a pause in transitions, for a given BCSM, while other IN actions occur.

(Sheet 4 of 4)

D) Functional Entity Actions (FEA) Descriptions

This subsection lists the FEA reference number, the Functional Entity to which the FEA applies and the source SIB from Recommendation Q.1214. The FEAs for Option 1 and Option 2 are listed.

2002 SSF Basic Call Process SIB

Detect Trigger Detection Point - Request (TDP-R)

- send Initial DP or DP-specific Initial req. ind. and
- suspend call processing

9121 SCF User Interaction SIB

Initiated Request

- Initiate a Connect_To_Resource req. ind.

2121 SSF User Interaction SIB

Process Request

- Receive Connect_To_Resource req. ind. from the SCF
- Analyse information (call involved, announcement address, routing requirements, etc.)
- Formulate and send a SETUP req. ind. to the SRF (if required)

3121 SRF User Interaction SIB

Process Request

- Receive and analyse SETUP req. ind. from CCF/SSF
- Select appropriate announcement resource

9122 SCF User Interaction SIB

Request Prompt/Collect Information or Announcement

- Receive a Connect_To_Resource resp. conf.
- Initiate a Prompt and Collect or Play Announcement req. ind. and send to the SRF

3122 SRF User Interaction SIB

Prompt/play Announcement

- Receive and analyse Prompt and Collect or Play Announcement req. ind. from SCF
- Apply prompt/announcement on resource toward user
- Return SRF. RPT req. ind. at conclusion of announcement if requested in Play Announcement req. ind.

9123 SCF User Interaction SIB

Initiate Disconnect

- Initiate a Disconnect_Forward_Connection req. ind. and send to the CCF/SSF

2122 SSF User Interaction SIB

Disconnect Forward

- Receive DISC.FWD.CONN req. ind. from the SCF
- Formulate and send DISCONNECT req. ind. to SRF

3123 SRF User Interaction SIB

Process Request

- Receive and analyse DISCONNECT req. ind. from CCF/SSF
- Continue disconnect process per Recommendation Q.71

9003 SCF Basic Call Process SIB

Initiate Request

– send one or more BCP information flows

20013 SSF Basic Call Process SIB

Process Request Report BCSM Event req. ind.

–Arm EDP(s)

9002 SCF Basic Call Process SIB

Process Request and Send Immediate Instruction

–Process initial or report information flow (e.g. Initial DP, DP specific, or Event Report BCSM) and send one or more BCP information flows in response.

9092 SCF Service Data Management SIB

– process request from service logic
–generate and send an Update Data req. ind.

4092 SDF Service Data Management SIB

–receive and analyse Update Data req. ind.
–execute specified action in the base
–process and return result
–generate and send an Update Confirmation resp. conf.

9094 SCF Service Data Management SIB

–receive Update Confirmation resp. conf.
–return response to service logic

20012 SSF Basic Call Process SIB

–Process Continue req. ind.

20014 SSF Basic Call Process SIB

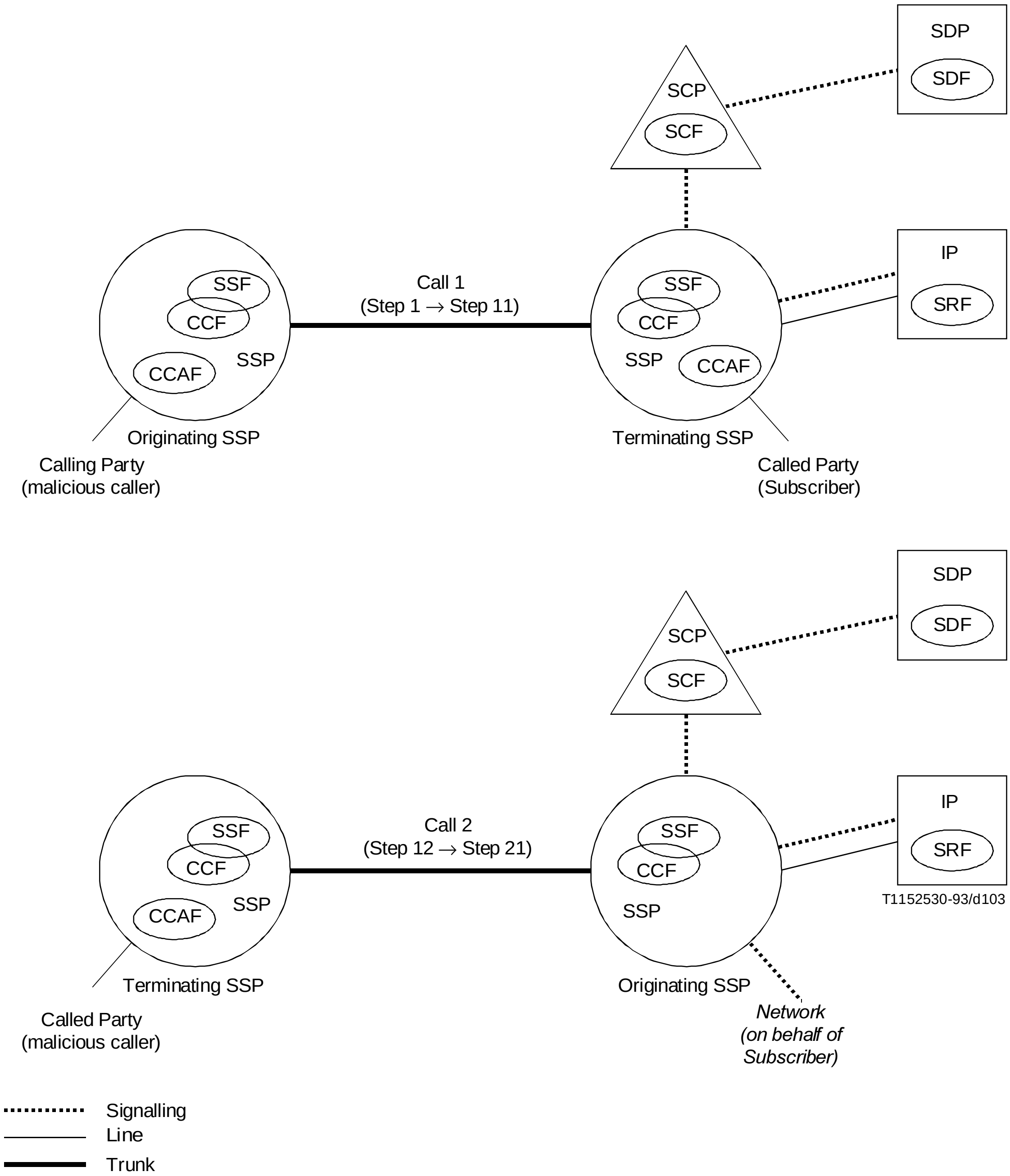
–Process Release Call req. ind.

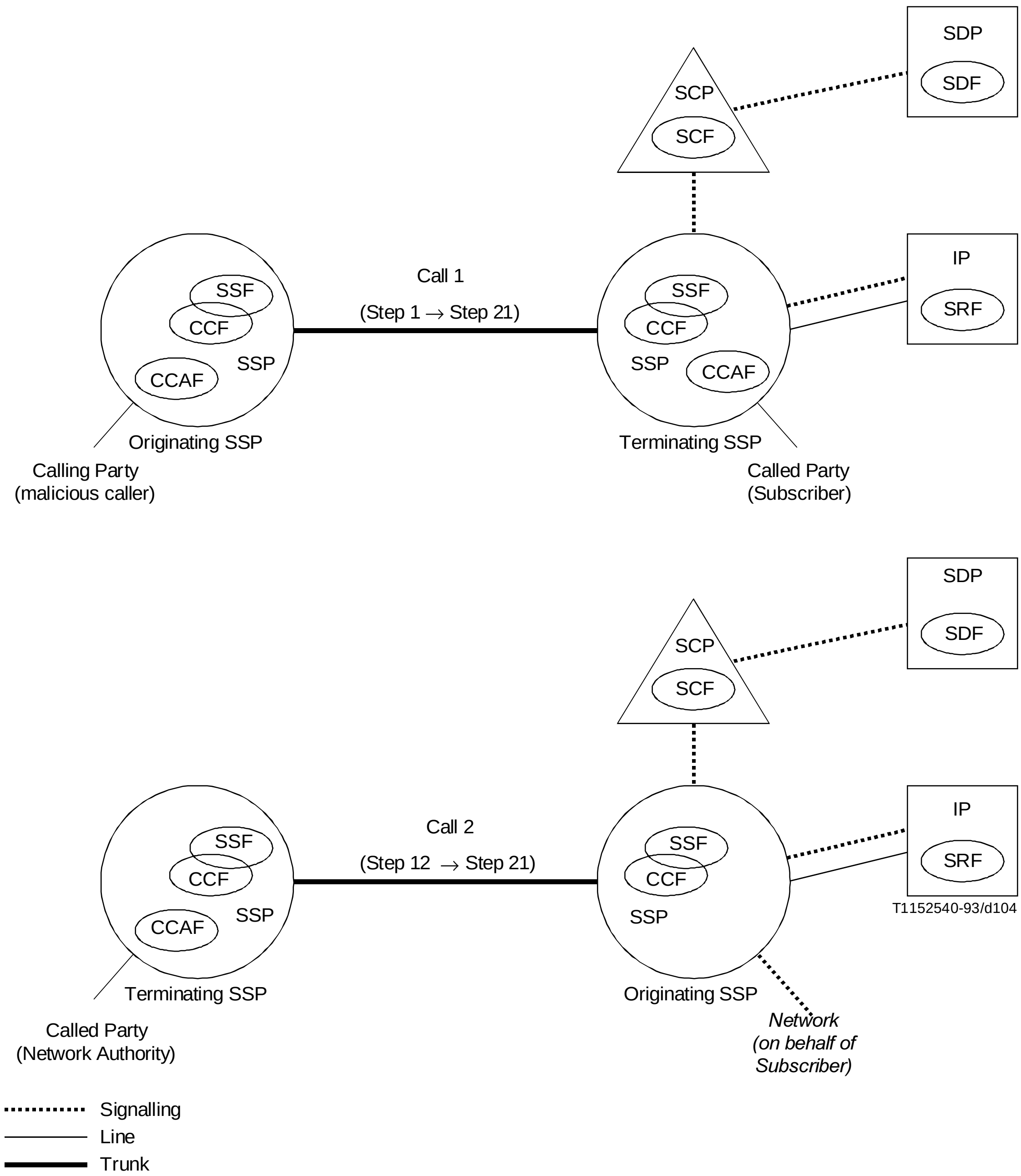
20015 SSF Basic Call Process SIB

–Process Initiate Call req. ind.

5) **Physical View**

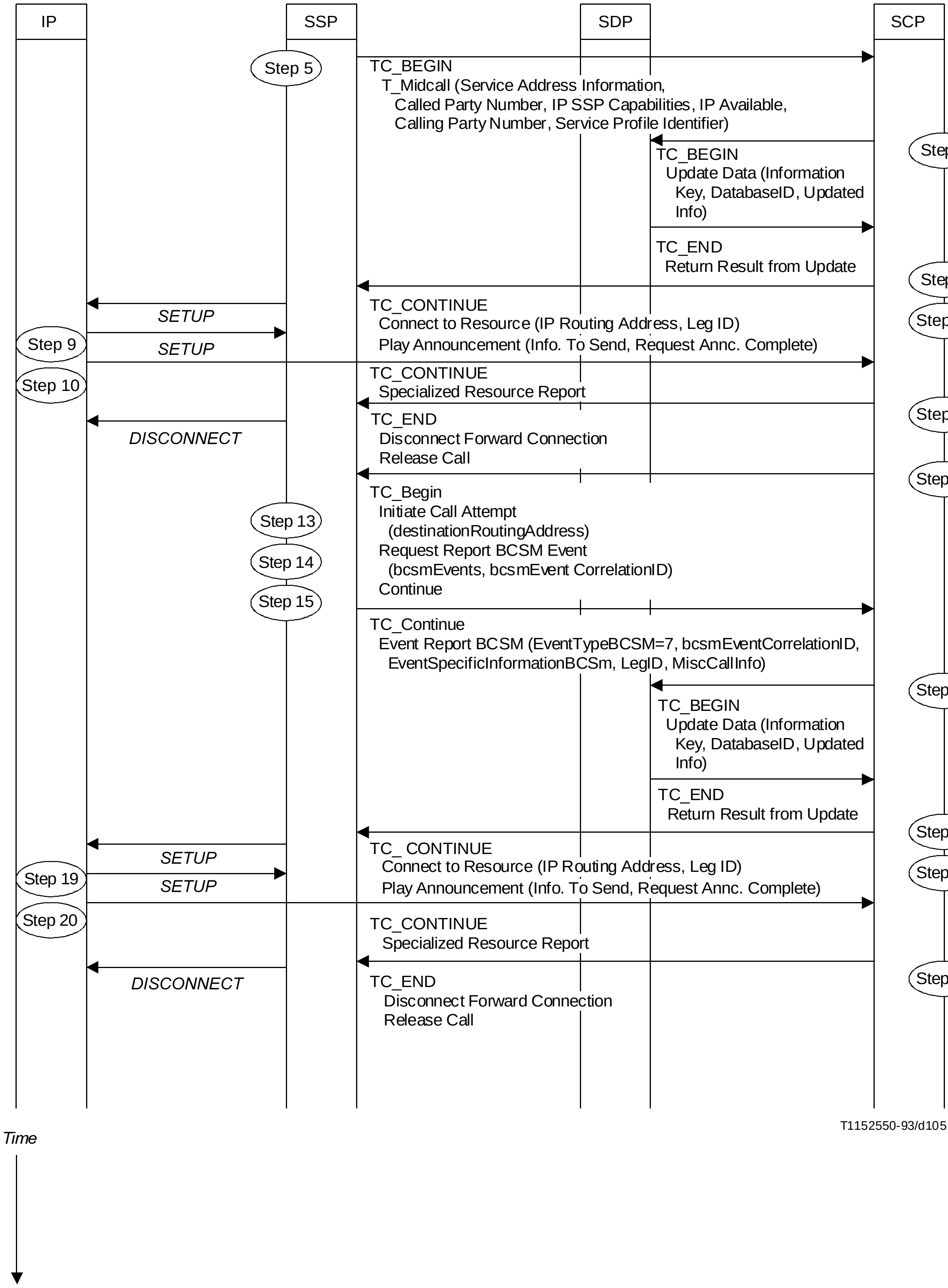
A) Physical Entity Interface Diagram (refer to Recommendation Q.1215)




SDP
SDF
SCP
SCF
IP
SRF
Call 1
(Step 1 → Step 21)
SSF
CCF
CCAF
SSP
Originating SSP
Terminating SSP
Calling Party
(malicious caller)
Called Party
(Subscriber)
Call 2
(Step 12 → Step 21)
Terminating SSP
Originating SSP
Called Party
(Network Authority)
Network
(on behalf of
Subscriber)
T1152540-93/d104
Signalling
Line
Trunk

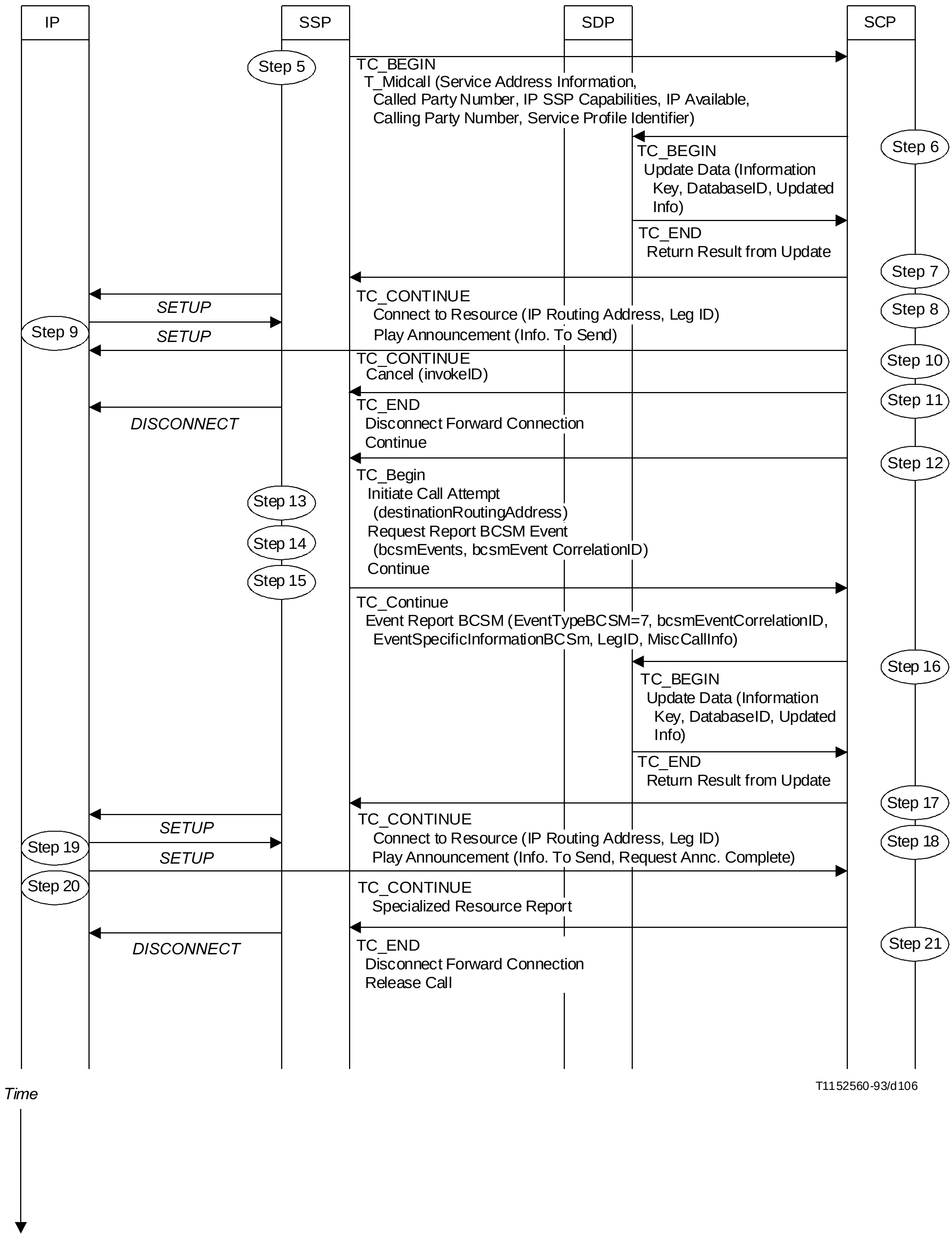
B) Time Sequence Diagram (refer to Recommendation Q.1218)

Option 1 – Network Returns Call



NOTE – Steps are described in 4) "Distributed View".

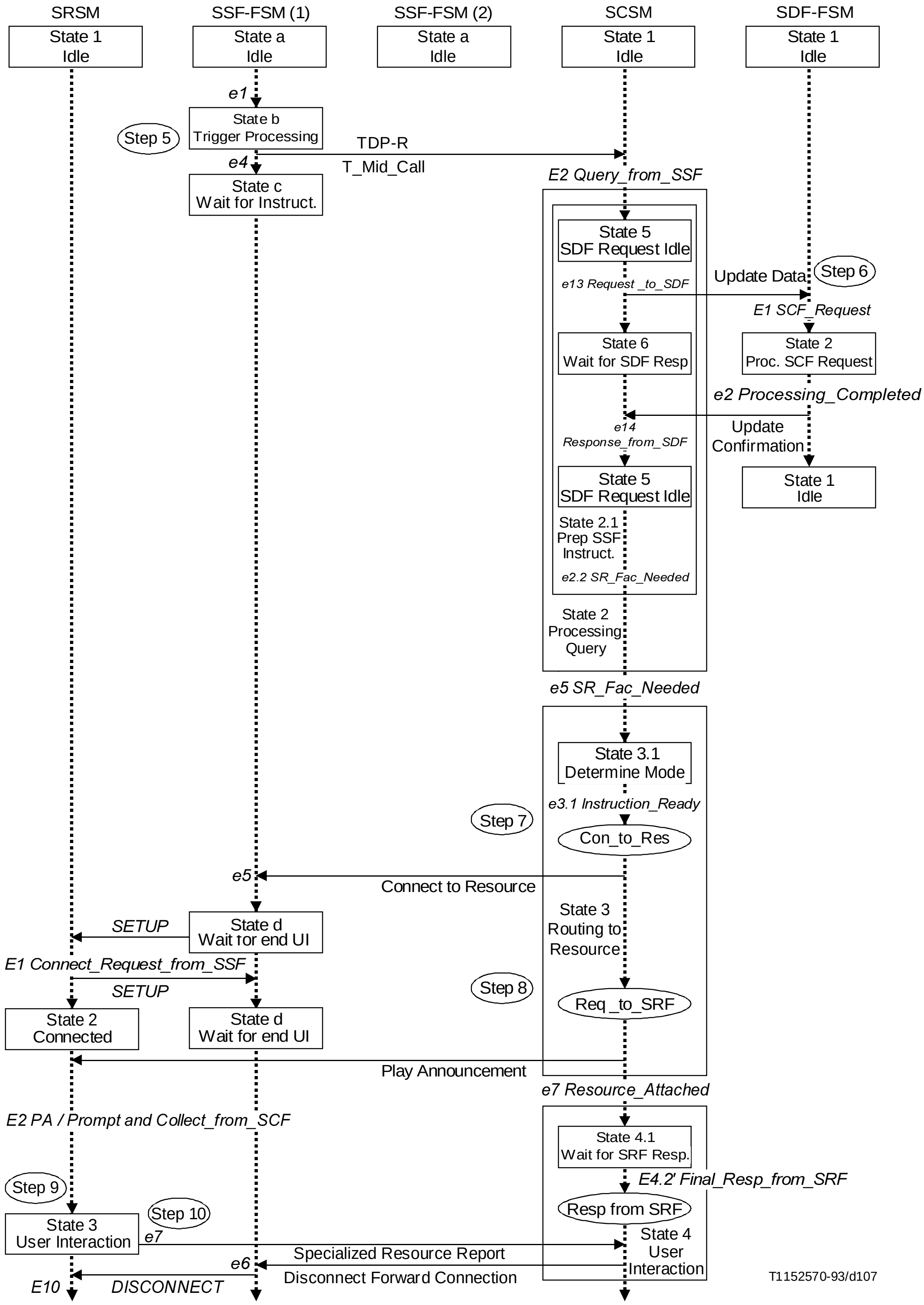
Option 2 – Network authority notification



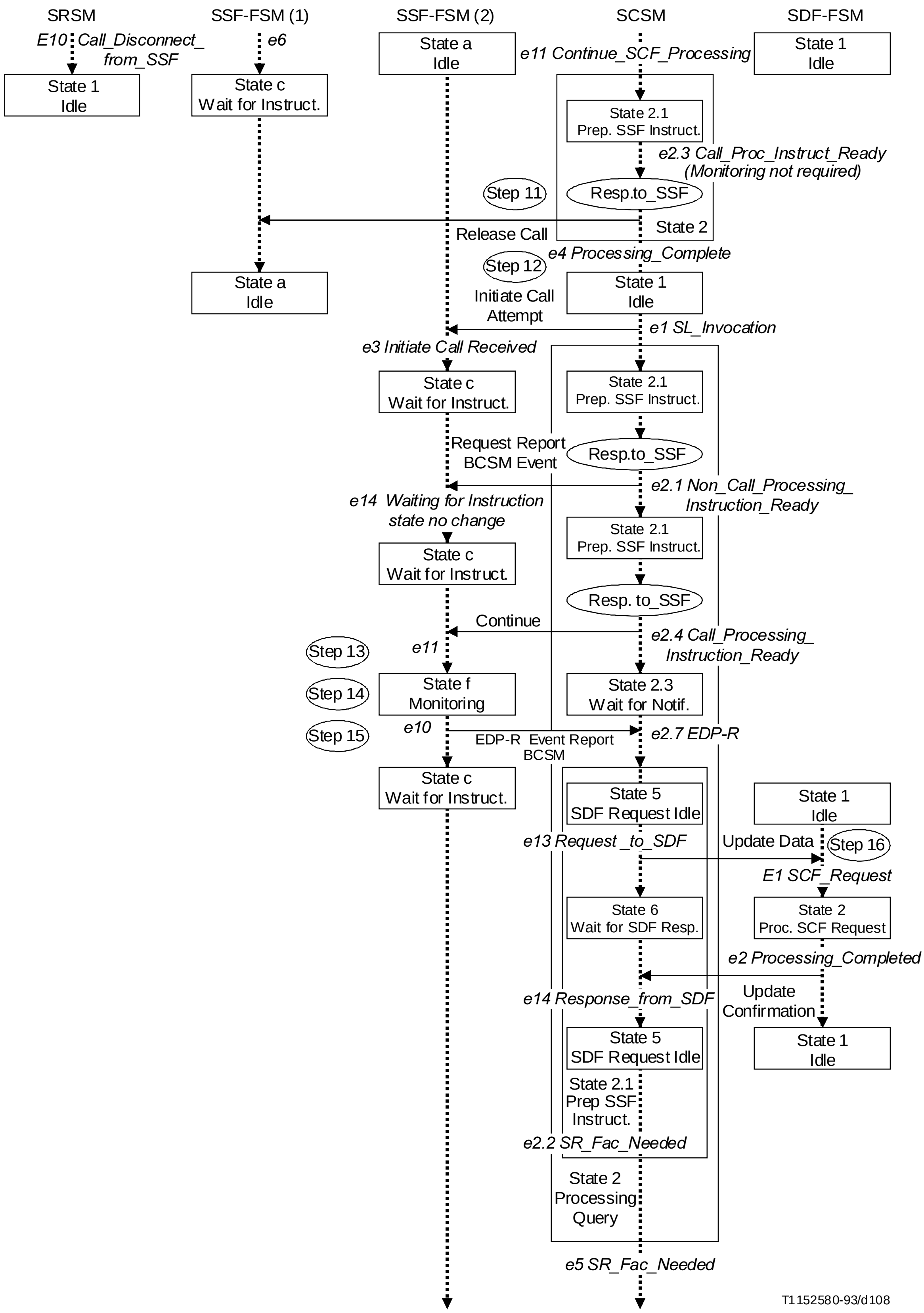
NOTE – Steps are described in 4) "Distributed View".

C) Application Entity (AE) Procedures (refer to Recommendation Q.1218)

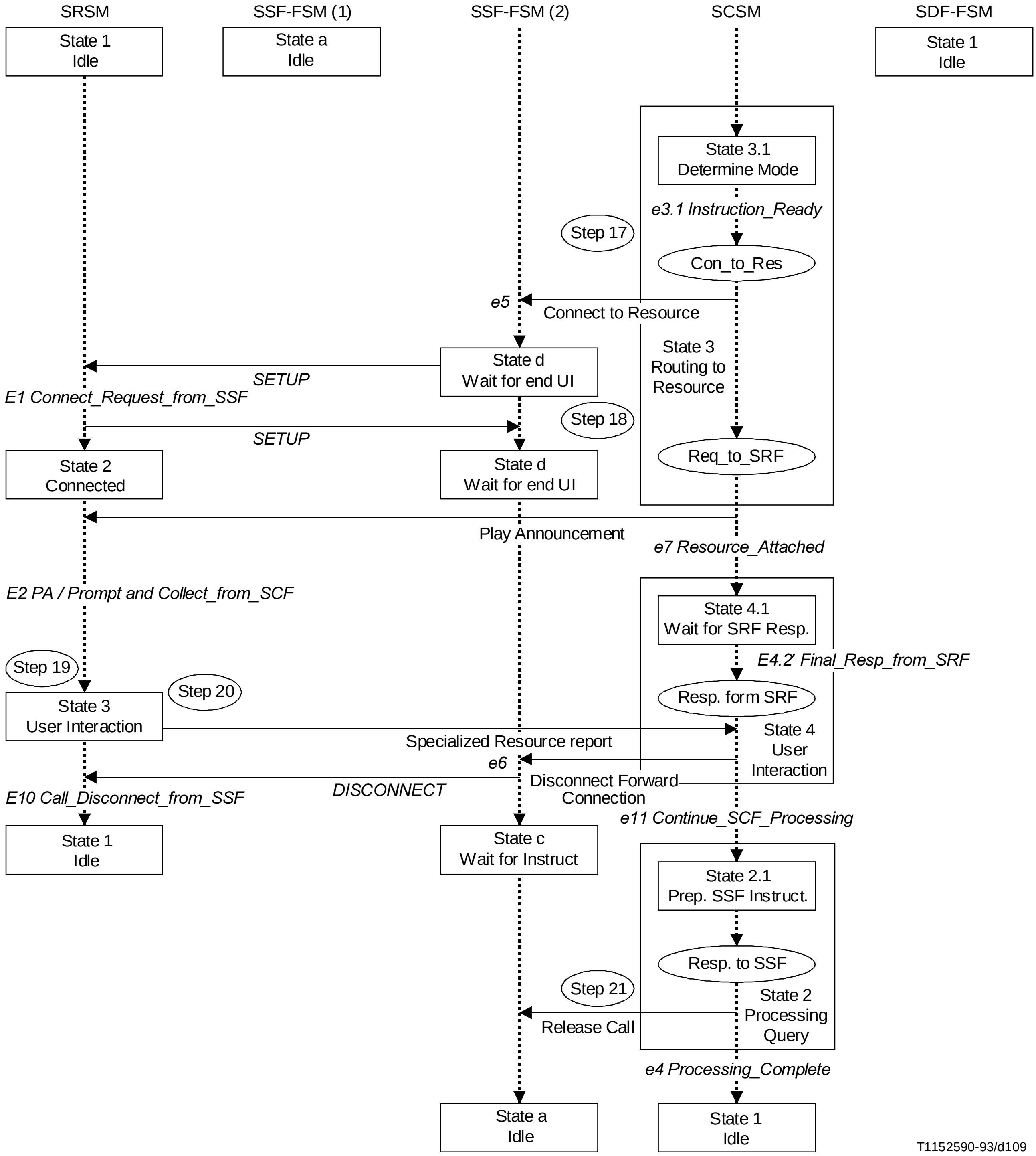
Option 1 – Network Returns Call



(Sheet 1 of 3)


SRSM
SSF-FSM (1)
SSF-FSM (2)
SCSM
SDF-FSM
E10 Call_Disconnect_ from_SSF
e6
State a Idle
e11 Continue_SCF_Processing
State 1 Idle
State 1 Idle
State c Wait for Instruct.
State 2.1 Prep. SSF Instruct.
e2.3 Call_Proc_Instruct_Ready (Monitoring not required)
Step 11
Resp.to_SSF
Release Call
State 2
e4 Processing_Complete
Step 12
State a Idle
Initiate Call Attempt
State 1 Idle
e1 SL_Invocation
e3 Initiate Call Received
State c Wait for Instruct.
State 2.1 Prep. SSF Instruct.
Request Report BCSM Event
Resp.to_SSF
e2.1 Non_Call_Processing_ Instruction_Ready
e14 Waiting for Instruction state no change
State 2.1 Prep. SSF Instruct.
State c Wait for Instruct.
Resp. to_SSF
Continue
e2.4 Call_Processing_ Instruction_Ready
Step 13
e11
Step 14
State f Monitoring
State 2.3 Wait for Notif.
Step 15
e10
EDP-R Event Report BCSM
e2.7 EDP-R
State c Wait for Instruct.
State 5 SDF Request Idle
State 1 Idle
e13 Request _to_SDF
Update Data
Step 16
E1 SCF_Request
State 6 Wait for SDF Resp.
State 2 Proc. SCF Request
e2 Processing_Completed
e14 Response_from_SDF
Update Confirmation
State 5 SDF Request Idle
State 1 Idle
State 2.1 Prep SSF Instruct.
e2.2 SR_Fac_Needed
State 2 Processing Query
e5 SR_Fac_Needed
T1152580-93/d108

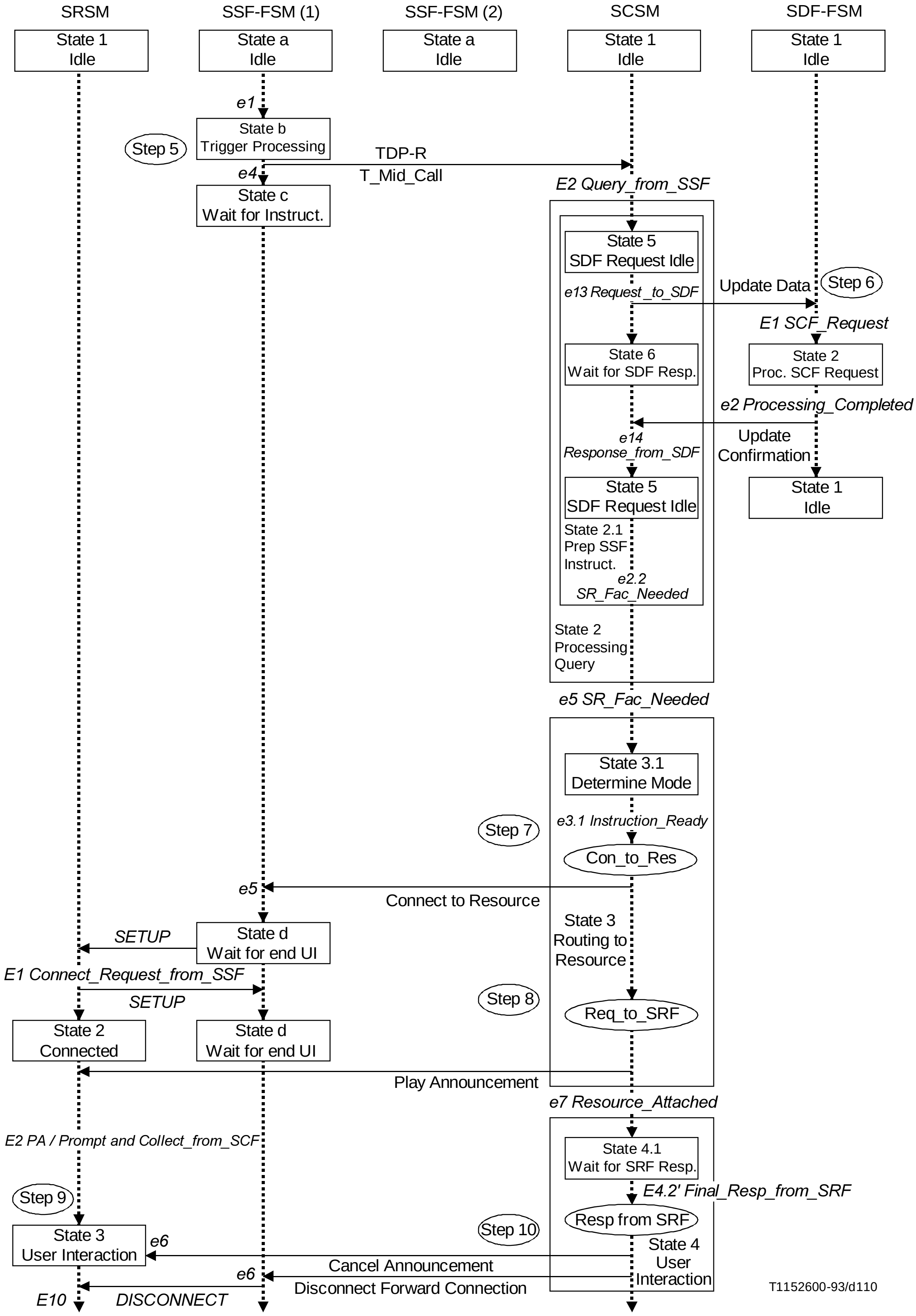
(Sheet 2 of 3)



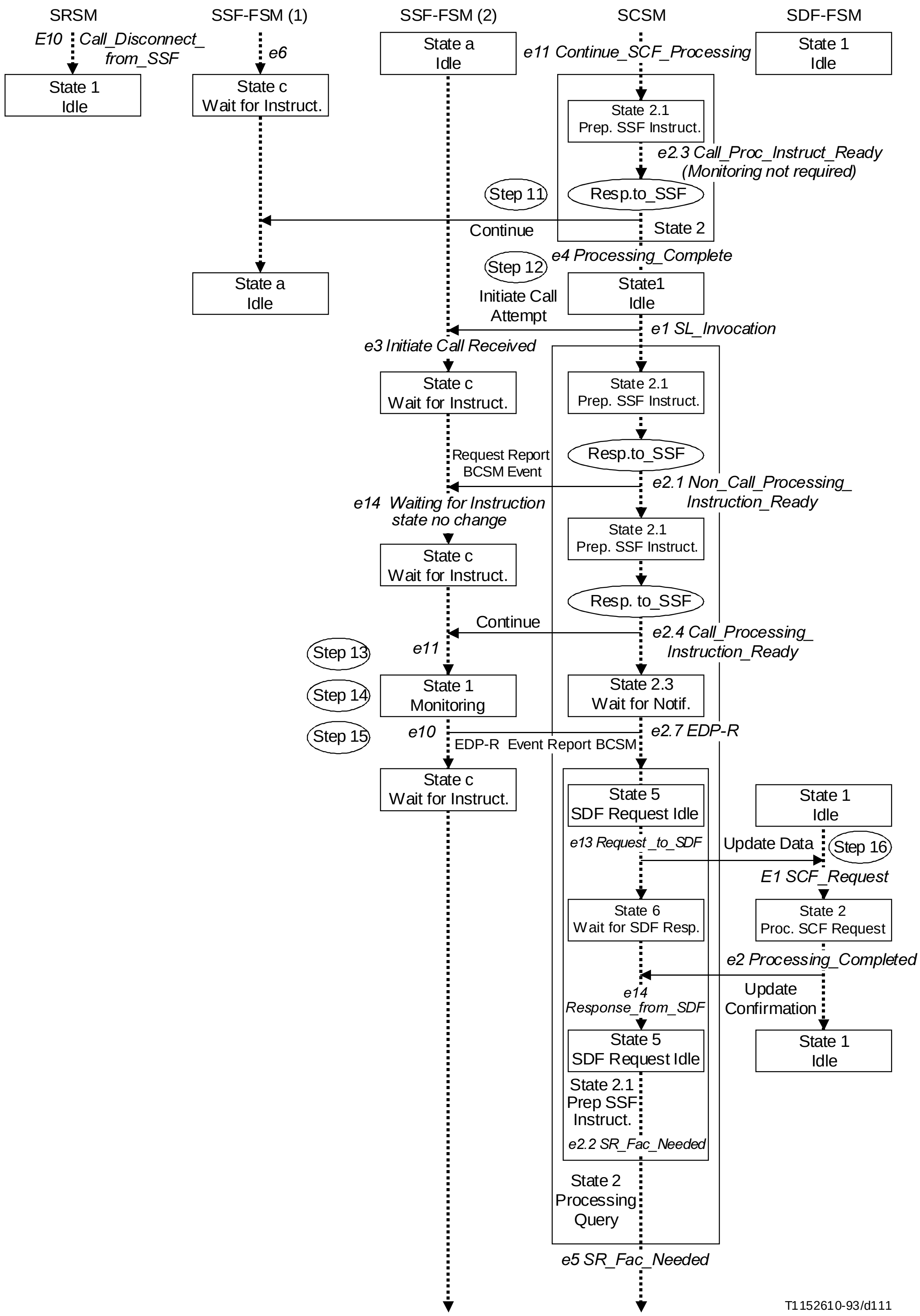
NOTE – Steps are described in 4) "Distributed View".

(Sheet 3 of 3)

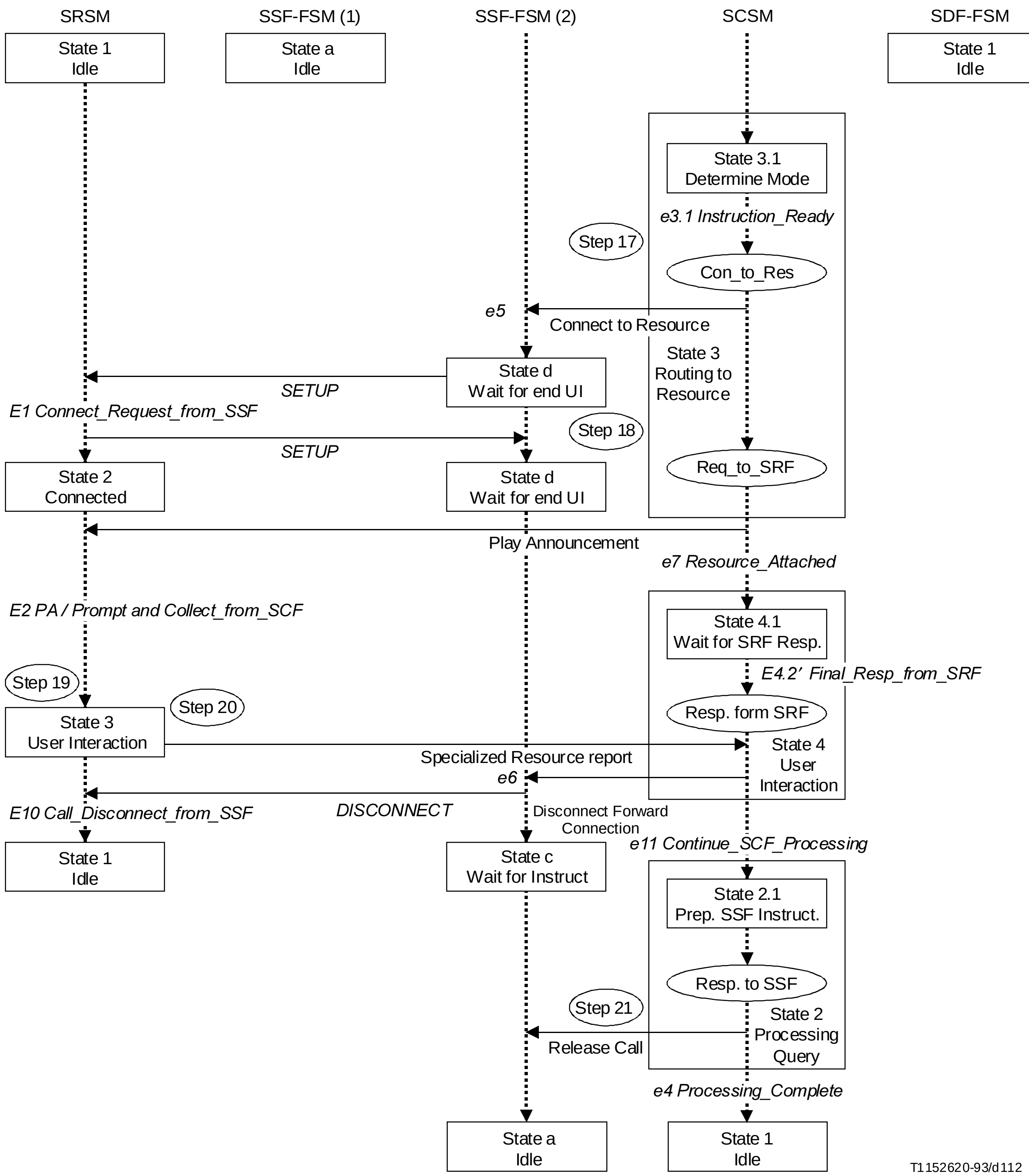
Option 2 – Network authority notification



(Sheet 1 of 3)


SRSM
SSF-FSM (1)
SSF-FSM (2)
SCSM
SDF-FSM
E10 Call_Disconnect_from_SSF
State 1 Idle
e6
State c Wait for Instruct.
State a Idle
e11 Continue_SCF_Processing
State 1 Idle
State 2.1 Prep. SSF Instruct.
e2.3 Call_Proc_Instruct_Ready (Monitoring not required)
Step 11
Resp.to_SSF
Continue
State 2
State a Idle
Step 12
e4 Processing_Complete
Initiate Call Attempt
State1 Idle
e1 SL_Invocation
e3 Initiate Call Received
State c Wait for Instruct.
State 2.1 Prep. SSF Instruct.
Request Report BCSM Event
Resp.to_SSF
e2.1 Non_Call_Processing_Instruction_Ready
e14 Waiting for Instruction state no change
State 2.1 Prep. SSF Instruct.
State c Wait for Instruct.
Resp. to_SSF
Continue
e2.4 Call_Processing_Instruction_Ready
Step 13
e11
Step 14
State 1 Monitoring
State 2.3 Wait for Notif.
Step 15
e10
EDP-R Event Report BCSM
e2.7 EDP-R
State c Wait for Instruct.
State 5 SDF Request Idle
State 1 Idle
e13 Request_to_SDF
Update Data
Step 16
E1 SCF_Request
State 6 Wait for SDF Resp.
State 2 Proc. SCF Request
e2 Processing_Completed
e14 Response_from_SDF
Update Confirmation
State 5 SDF Request Idle
State 1 Idle
State 2.1 Prep SSF Instruct.
e2.2 SR_Fac_Needed
State 2 Processing Query
e5 SR_Fac_Needed
T1152610-93/d111

(Sheet 2 of 3)



NOTE – Steps are described in 4) "Distributed View."

(Sheet 3 of 3)

Annex B

BCSM SDLs

(This annex forms an integral part of this Recommendation)

Subclause 6.3.3.3 of this Recommendation contains the following text:

> "With respect to BCM DP processing described in 4.2.2.5/Q.1214, note that DP processing spans the CCF and SSF, with the CCF detecting events in the BCSM and passing them on to the SSF, where EDP and TDP criteria are checked. This separation of functionality between the CCF and SSF is not defined in Recommendation Q.1214 since the CCF and SSF are considered as a unit in CS-1, but it is useful to understand from the perspective of mapping to Recommendation Q.71, which only addresses CCF processing, and from an evolutionary perspective, in which the CCF and SSF may ultimately be separated. As such, Annex B provides an SDL description of DP processing in the SSF to complement the SDLs on BCSM processing in the CCF. These SDLs are based on the SDL description in Recommendation Q.1214, assuming the separation of functionality between the CCF and SSF described above rather than treating the SSF/CCF as a unit."

The SDL diagrams contained in this annex model the BCSM in the CCF and DP processing in the SSF as separable functions. No other actions are considered. The block diagram of Figure B.1 shows the separation of Functional Entity 2 (FE2) into two blocks, FE2A (CCF) and FE2B (SSF). The processes within these two blocks are shown in Figures B.2 and B.3. These three Figures present a very simple model, with no attempt to show process creation or interfacing to any processes other than those described here.

The BCSM SDL diagrams have been derived from the text of 4.2.2.2/Q.1214, together with descriptions of CCF processing from revised Recommendation Q.71, "Circuit mode switched bearer services". The underlying BCSM model is assumed to be one where instances of both originating and terminating BCSMs are created to handle each call, with the instances terminating on call completion. In the SDL diagrams termination has been modelled by the SDL Stop symbol. This representation differs from that of the BCSM descriptions in Recommendation Q.1214, where the BCSMs return to the original state and process termination is not described. The manner of process creation is not shown in the SDL diagrams.

The DP processing description has been derived from Figures 4-6/Q.1214 to 4-10/Q.1214, modified as necessary to show functional separation between the CCF and the SSF. The model in this case has been shown as persistent, with the DP process returning to the initial state at the end of handling each DP rather than terminating. This model would be compatible with one DP processing instance handling all the DPs for a particular call. The manner of process creation and destruction are not shown.

The SDL diagrams contained in this annex are high-level and informal, and are explanatory in nature only.

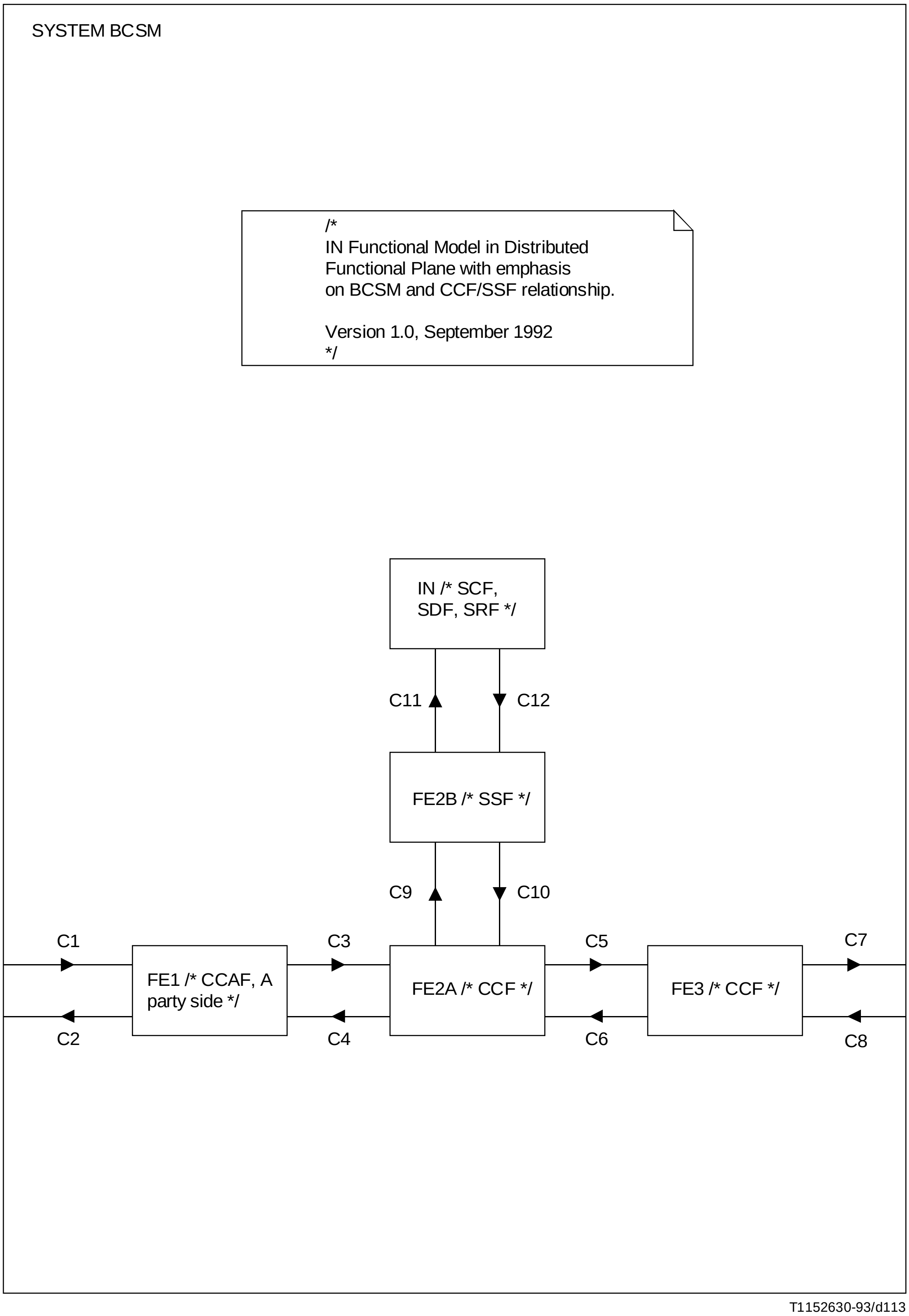


FIGURE B.1/Q.1219

IN functional model in the Distributed Functional Plane, with emphasis on BCSM and SSF/CCF relationship

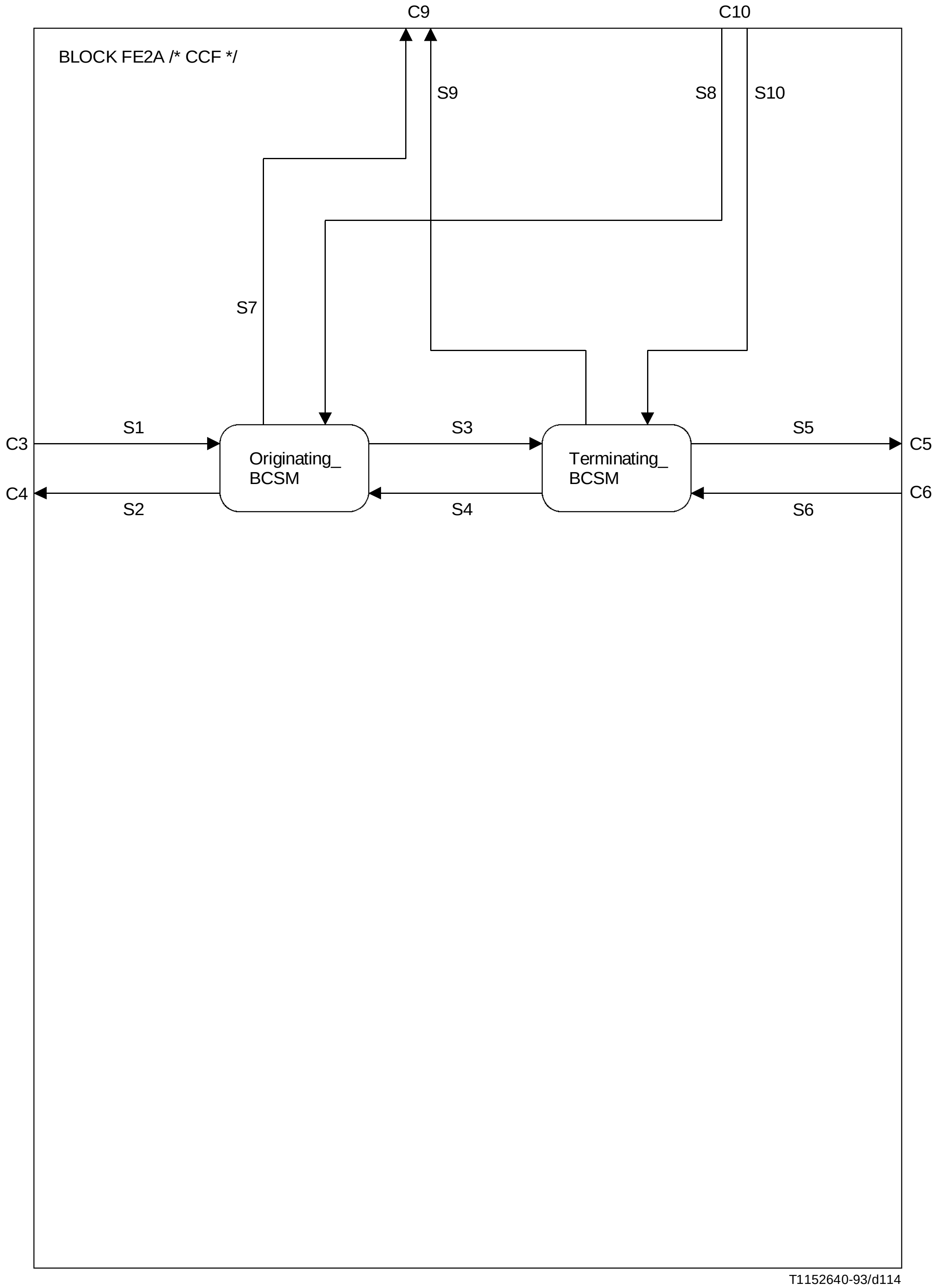


FIGURE B.2/Q.1219

Expansion of CCF block to show origination and terminating BCSMs

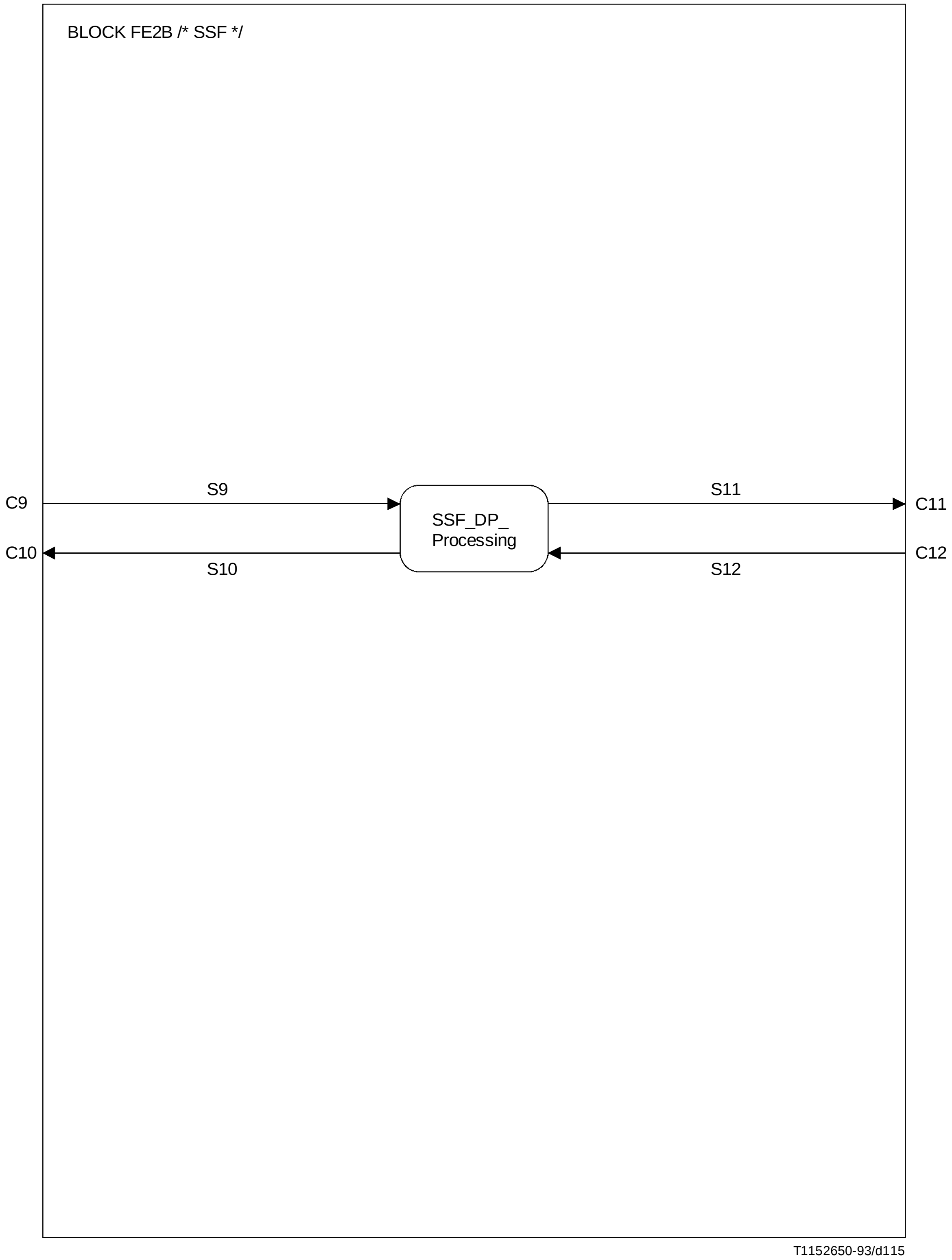


FIGURE B.3/Q.1219

Expansion of SSF block to show Detection Point processing

PROCESS Originating_BCSM;

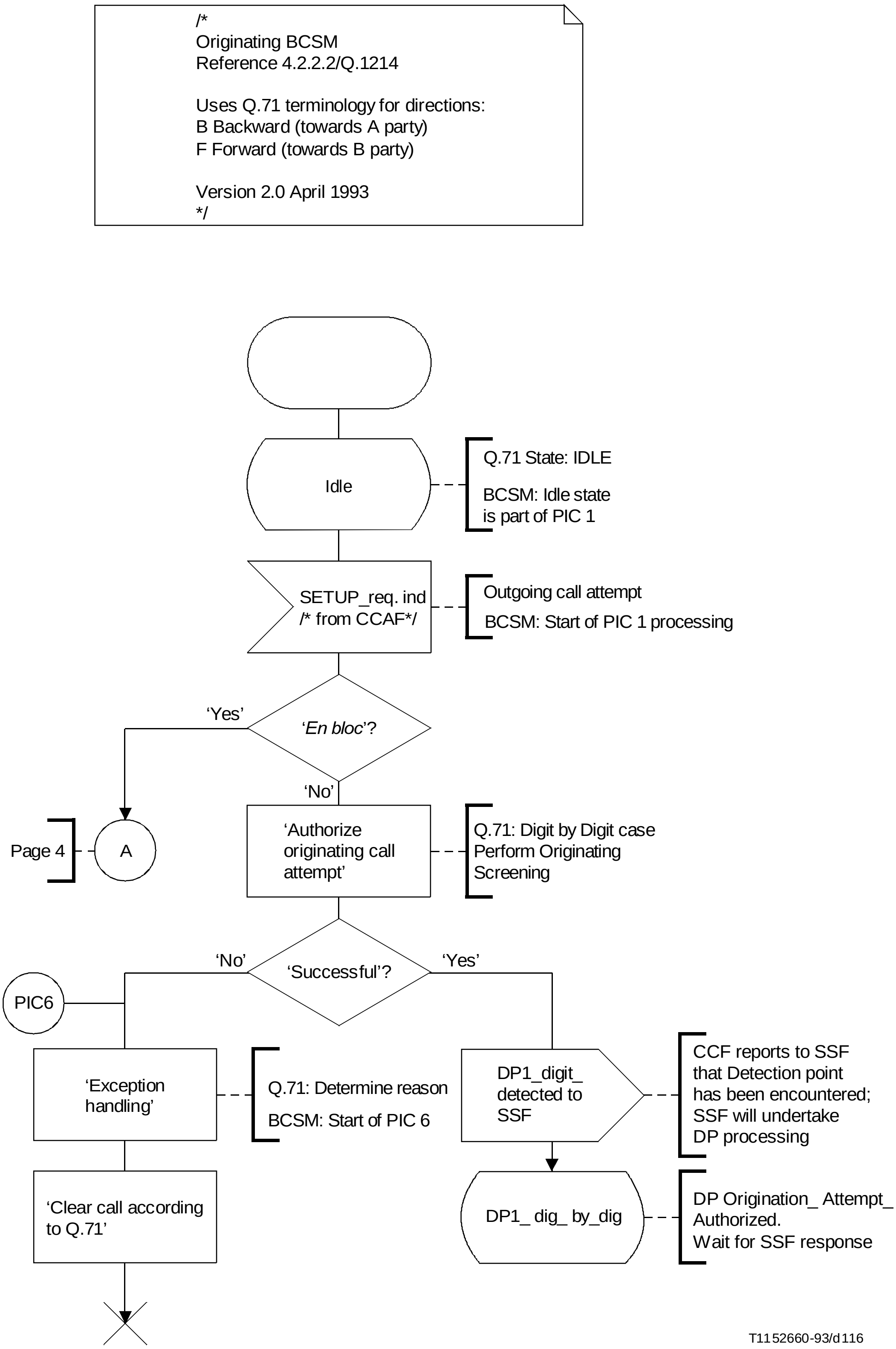


FIGURE B.4/Q.1219 (Sheet 1 of 11)

SDL for originating BCSM

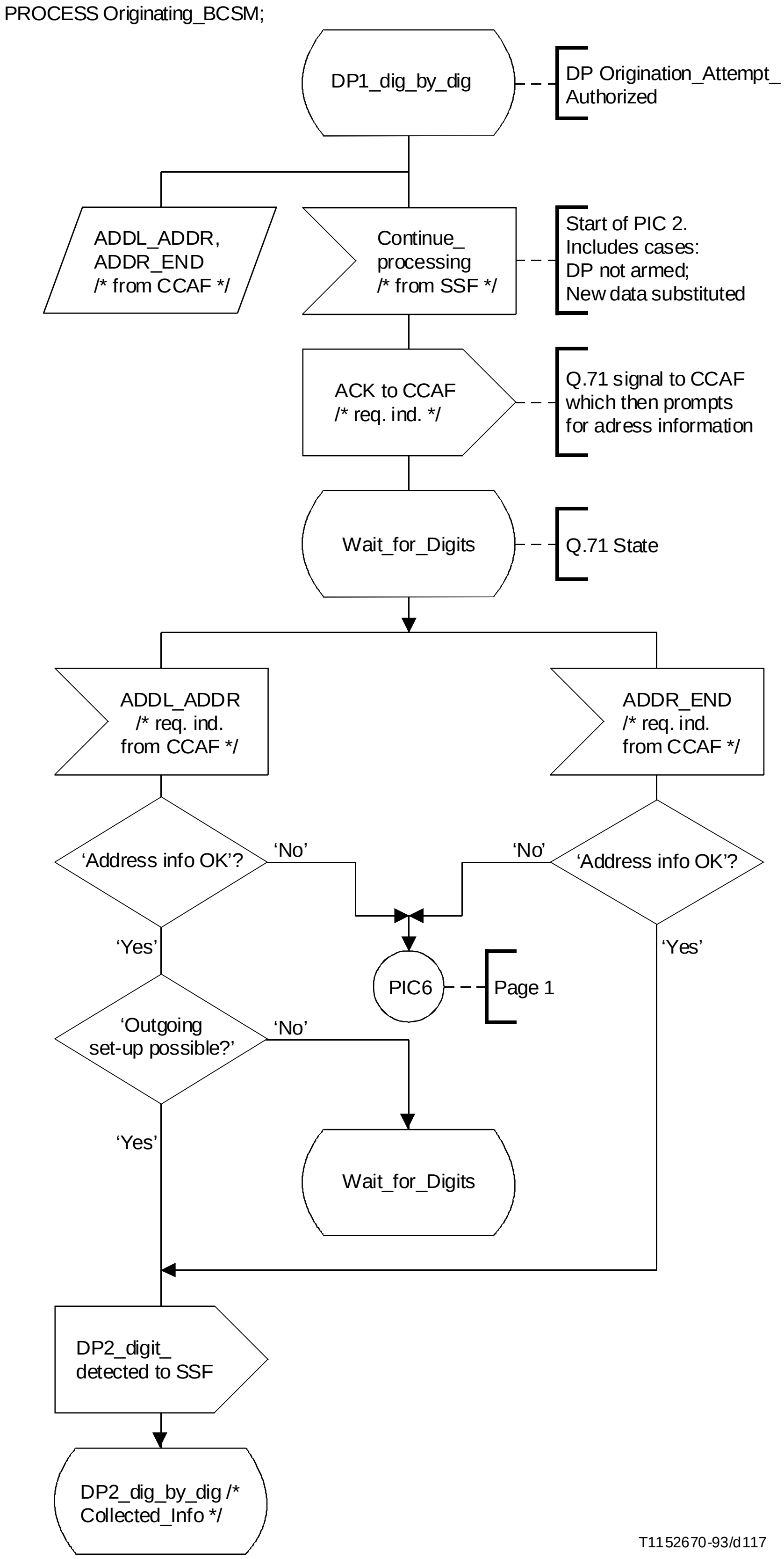


FIGURE B.4/Q.1219 (Sheet 2 of 11)

SDL for originating BCSM

PROCESS Originating_BCSM;

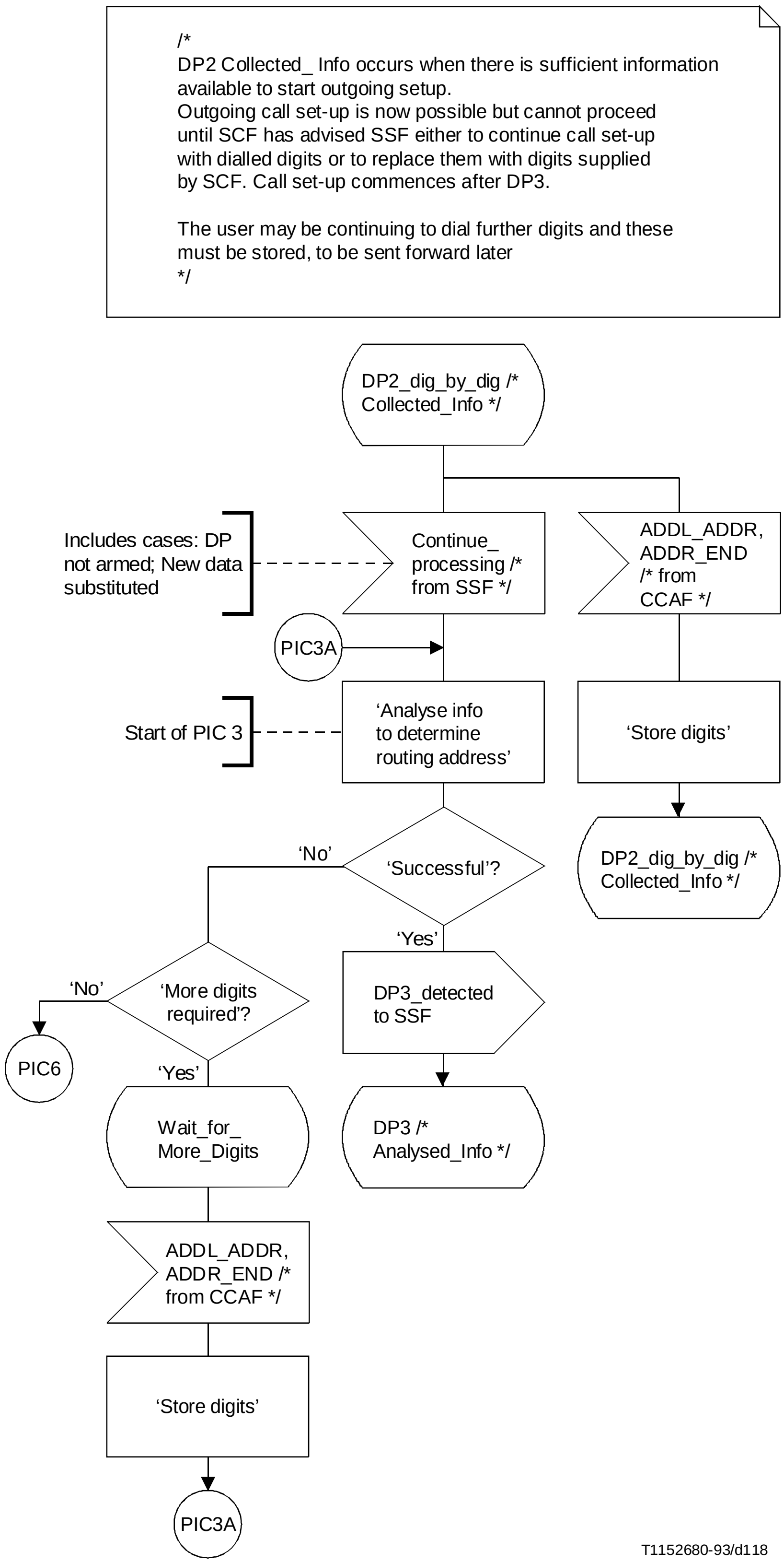


FIGURE B.4/Q.1219 (Sheet 3 of 11)

SDL for originating BCSM

PROCESS Originating_BCSM;

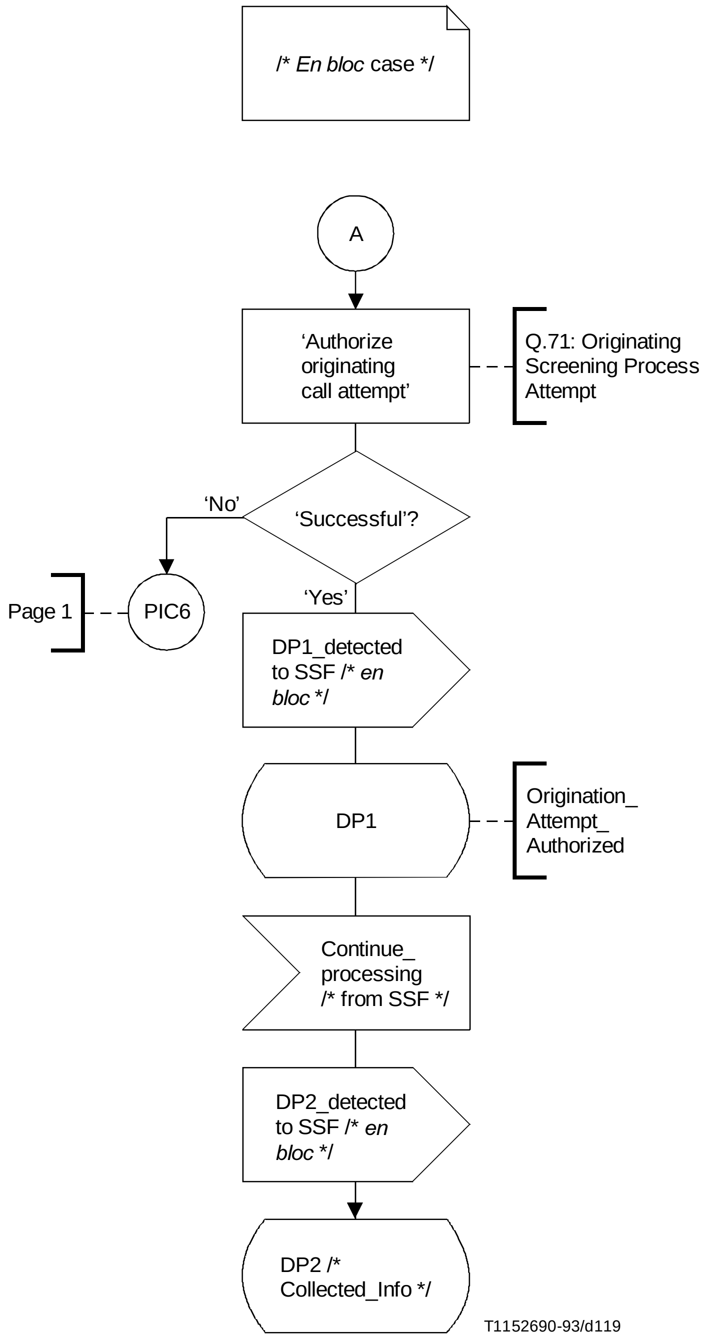


FIGURE B.4/Q.1219 (Sheet 4 of 11)

SDL for originating BCSM

PROCESS Originating_BCSM;

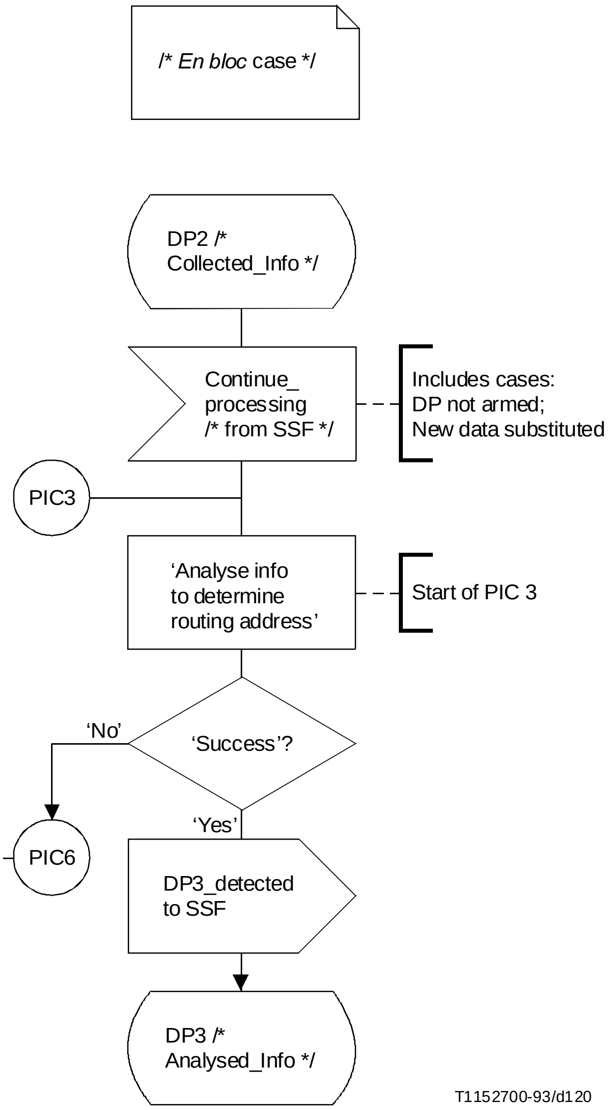


FIGURE B.4/Q.1219 (Sheet 5 of 11)

SDL for originating BCSM

PROCESS Originating_BCSM;

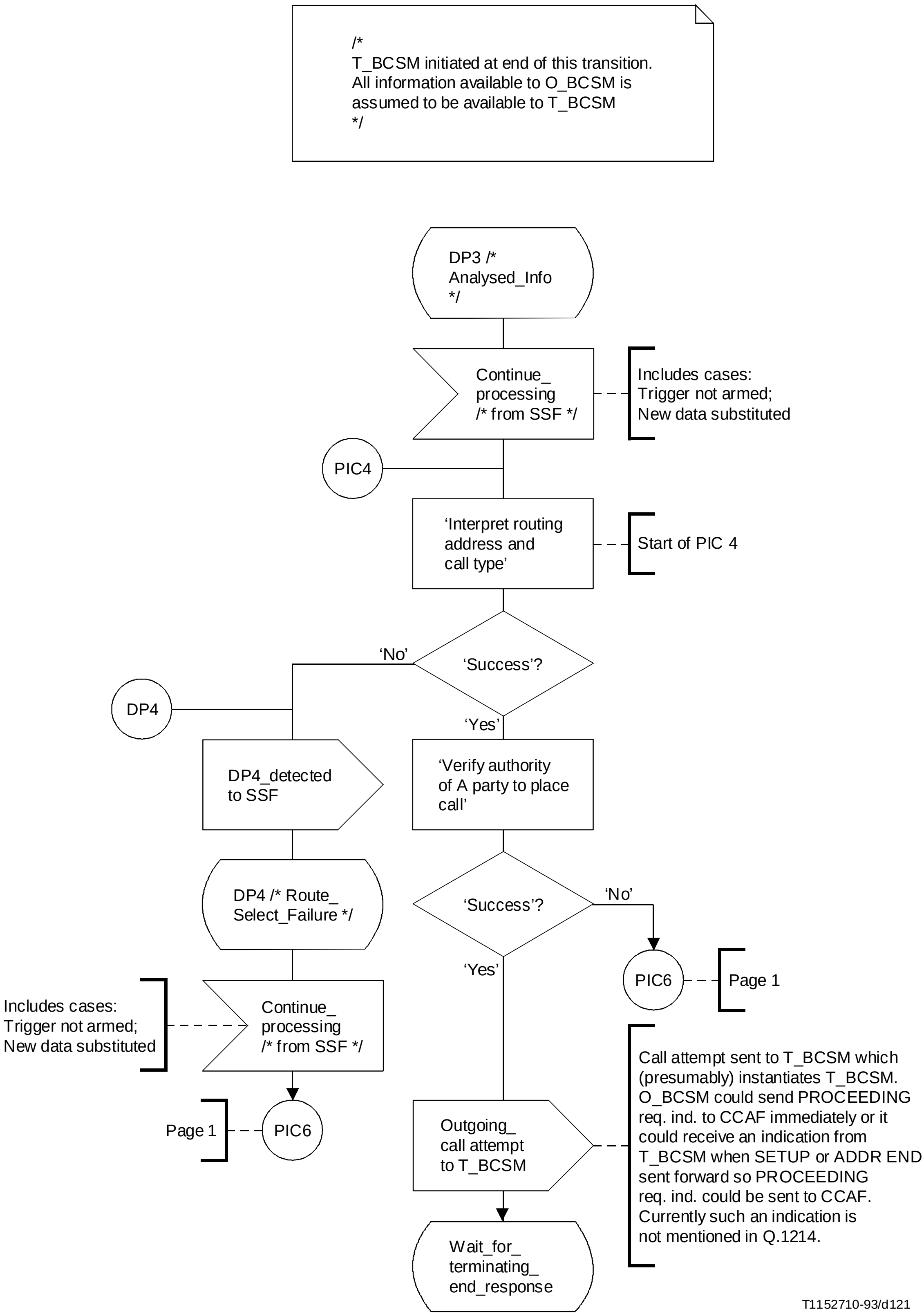


FIGURE B.4/Q.1219 (Sheet 6 of 11)

SDL for originating BCSM

PROCESS Originating_BCSM;

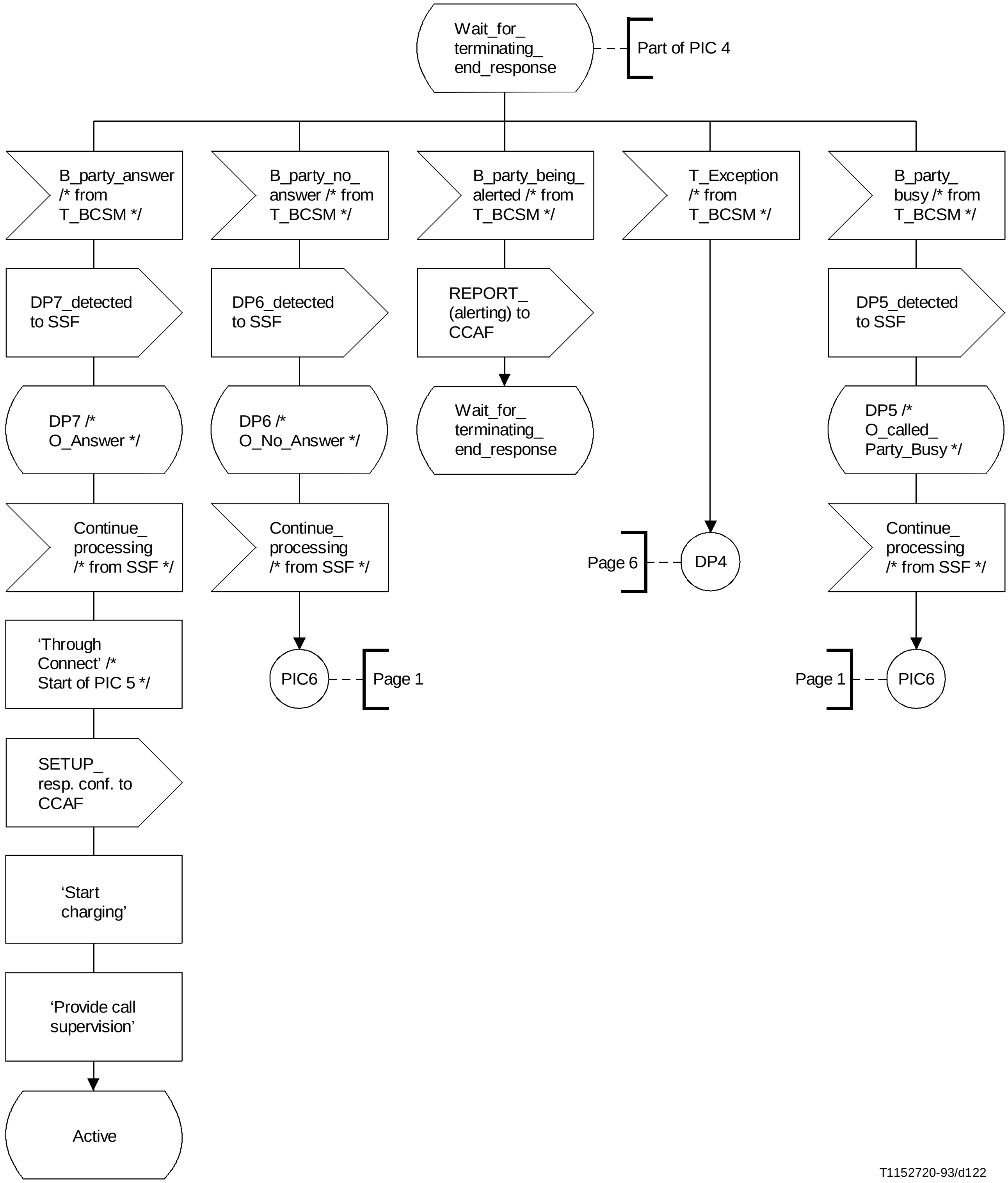


FIGURE B.4/Q.1219 (Sheet 7 of 11)

SDL for originating BCSM

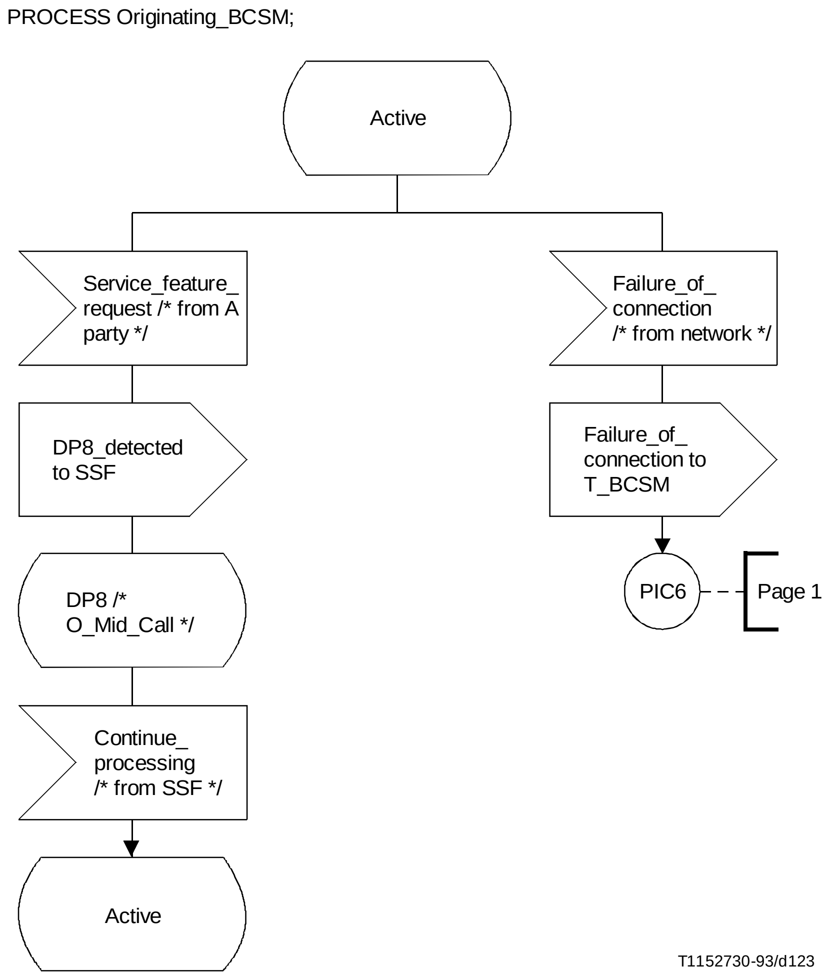


FIGURE B.4/Q.1219 (Sheet 8 of 11)

SDL for originating BCSM

PROCESS Originating_BCSM;

/*
Flow of processing control can be altered
by SCF instructing SSF to restart processing
at a different PIC.
See 5.3.2.2/Q.1214, point 2b, Proceed with Call Processing
family of information flows.

This results in common processing actions for
DP1 and DP1_ dig_ by_ dig – Origination_ Attempt_ Authorized
DP2 and DP2_ dig_ by_ dig – Collected_ Info
DP3 - Analysed Info
DP4 - Route_ Select_ Failure
DP5 - O_ Called_ Party_ Busy
DP6 - O_ No_ Answer
DP9 - O_ Disconnect (called party disconnect only)
*/

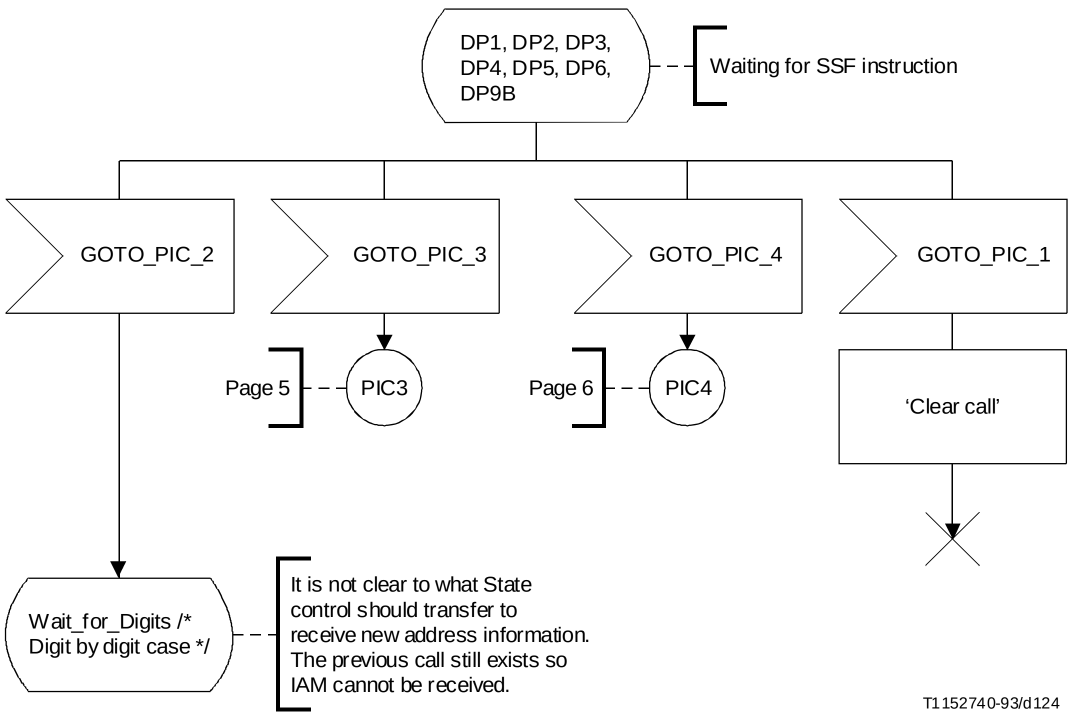


FIGURE B.4/Q.1219 (Sheet 9 of 11)

SDL for originating BCSM

PROCESS Originating_BCSM;

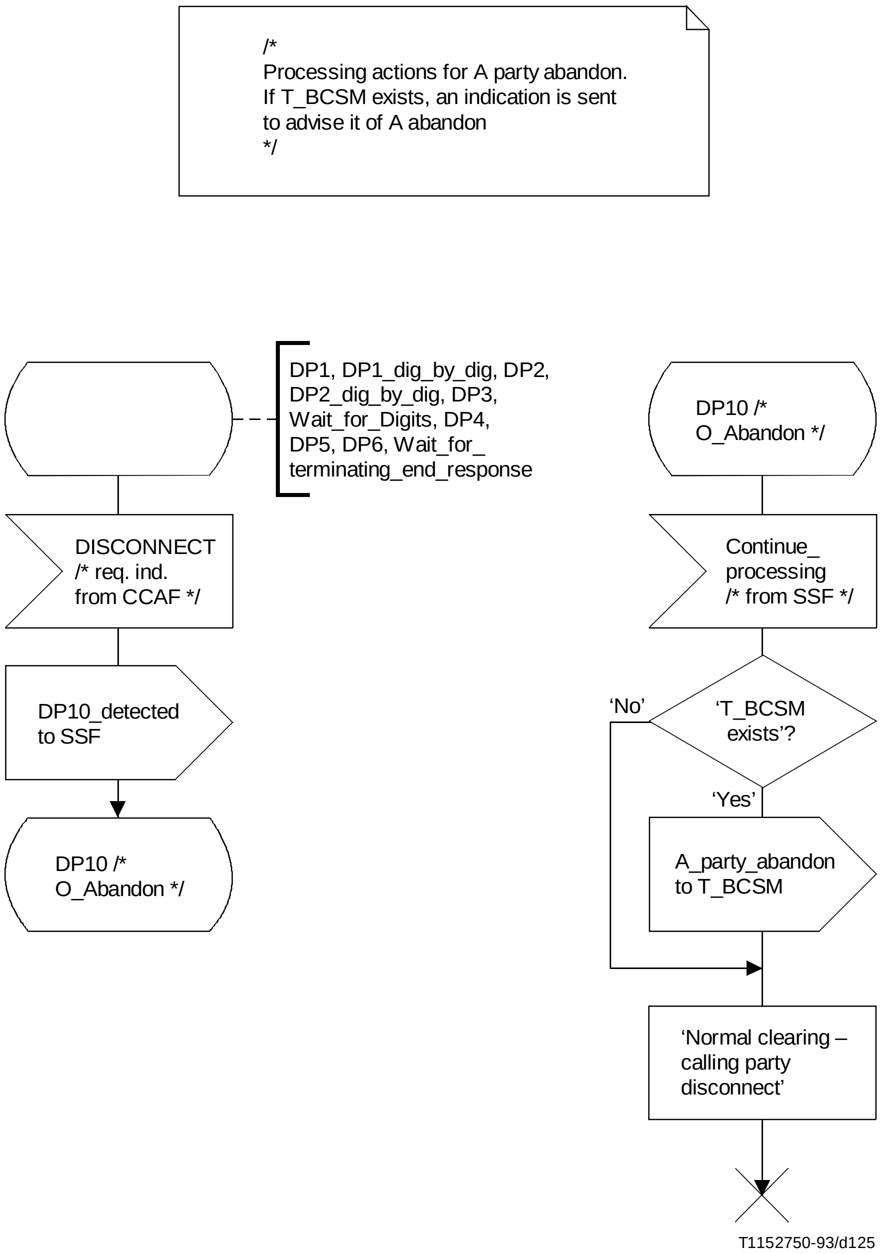


FIGURE B.4/Q.1219 (Sheet 10 of 11)

SDL for originating BCSM

PROCESS Originating_BCSM;

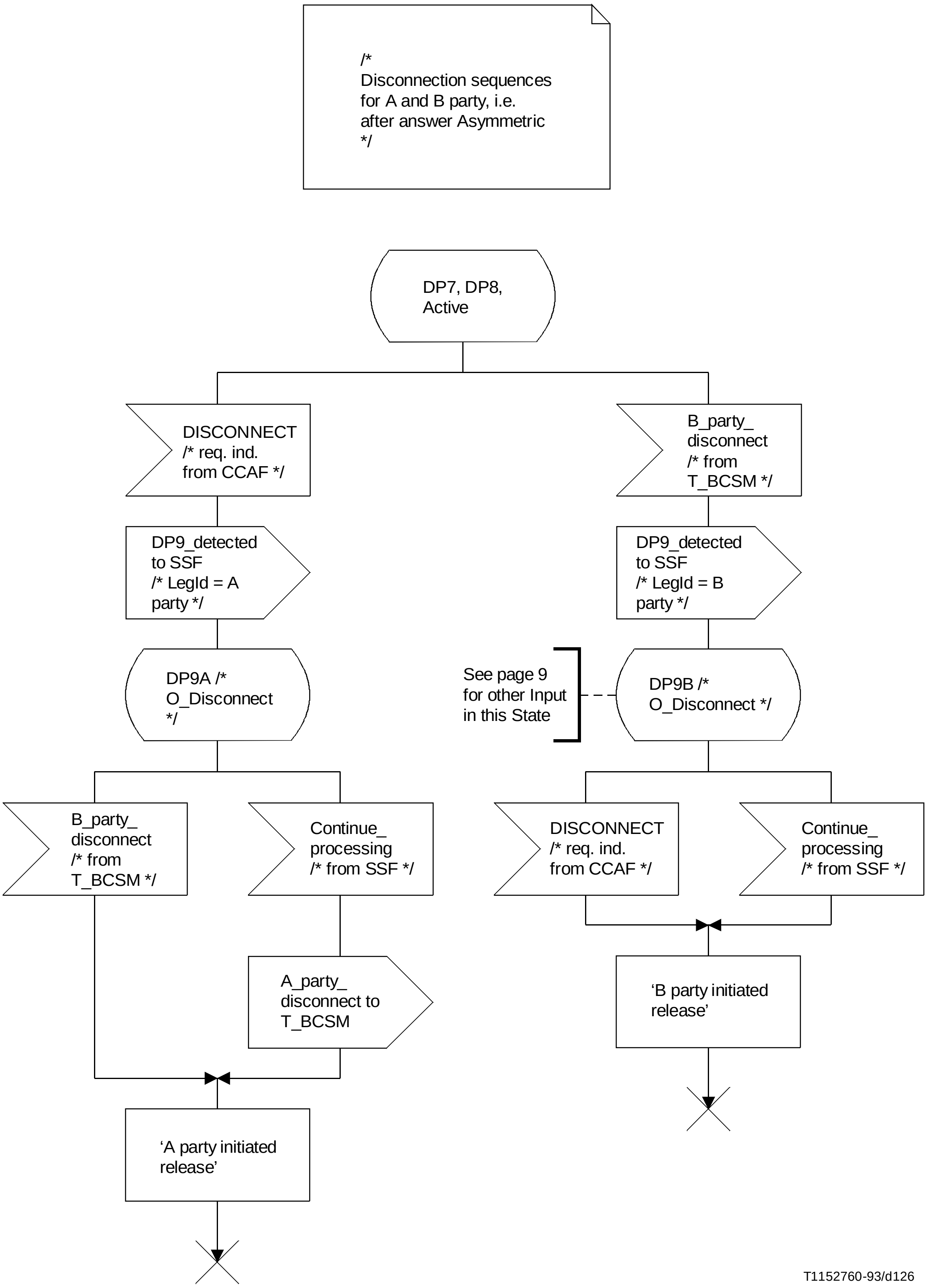


FIGURE B.4/Q.1219 (Sheet 11 of 11)

SDL for originating BCSM

PROCESS Terminating_BCSM;

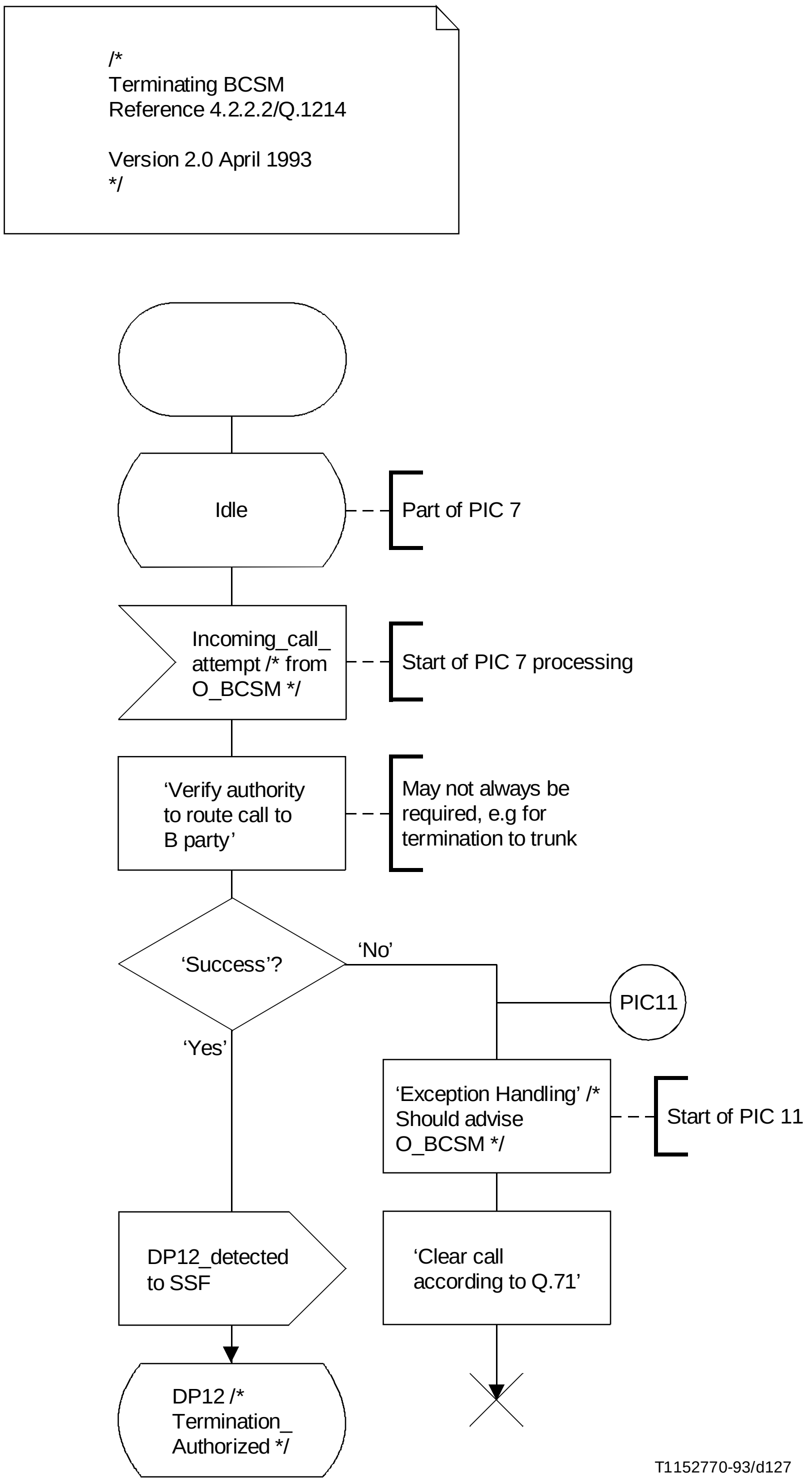


FIGURE B.5/Q.1219 (Sheet 1 of 6)

SDL for terminating BCSM

PROCESS Terminating_BCSM;

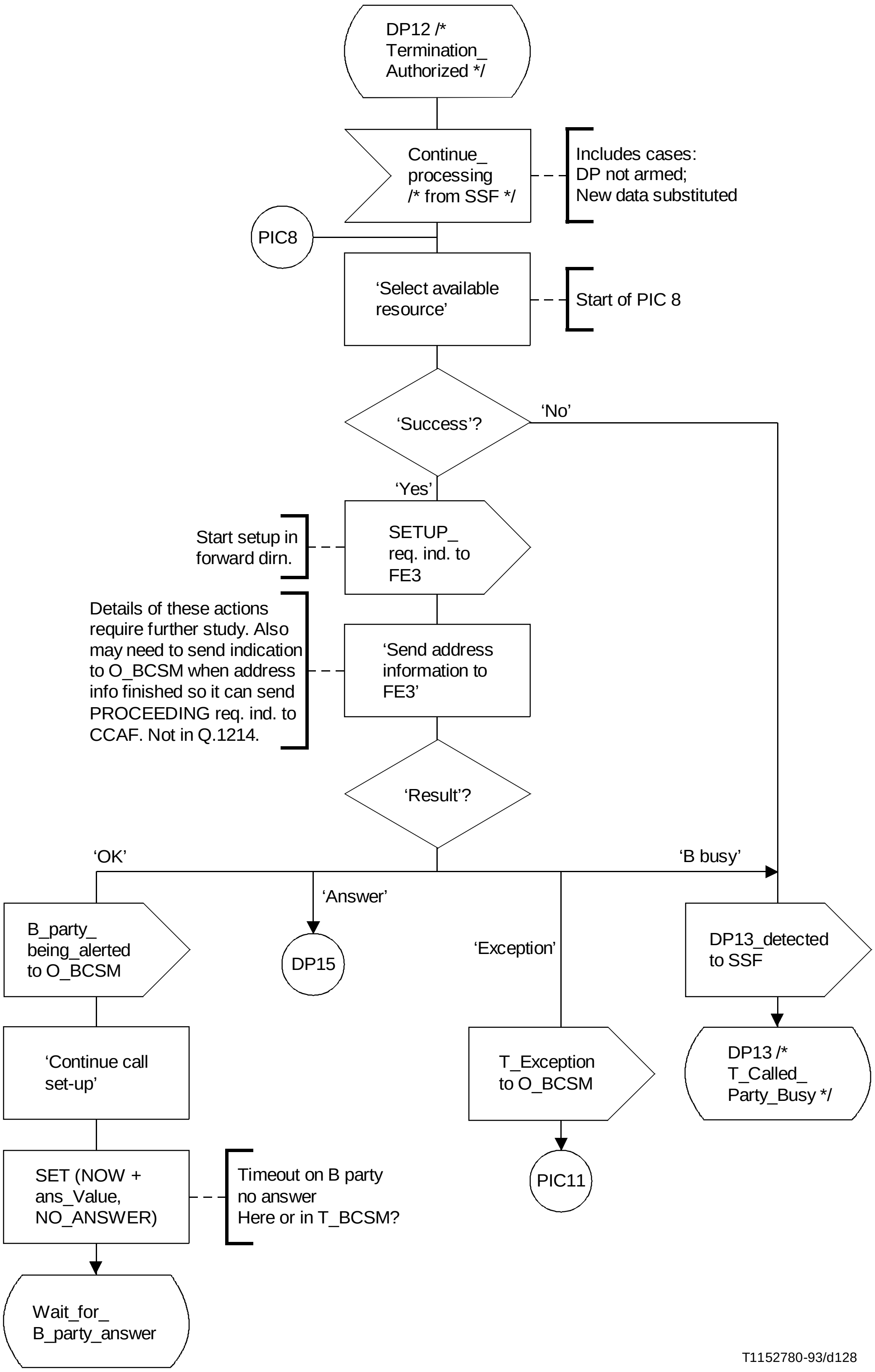


FIGURE B.5/Q.1219 (Sheet 2 of 6)

SDL for terminating BCSM

PROCESS Terminating_BCSM;

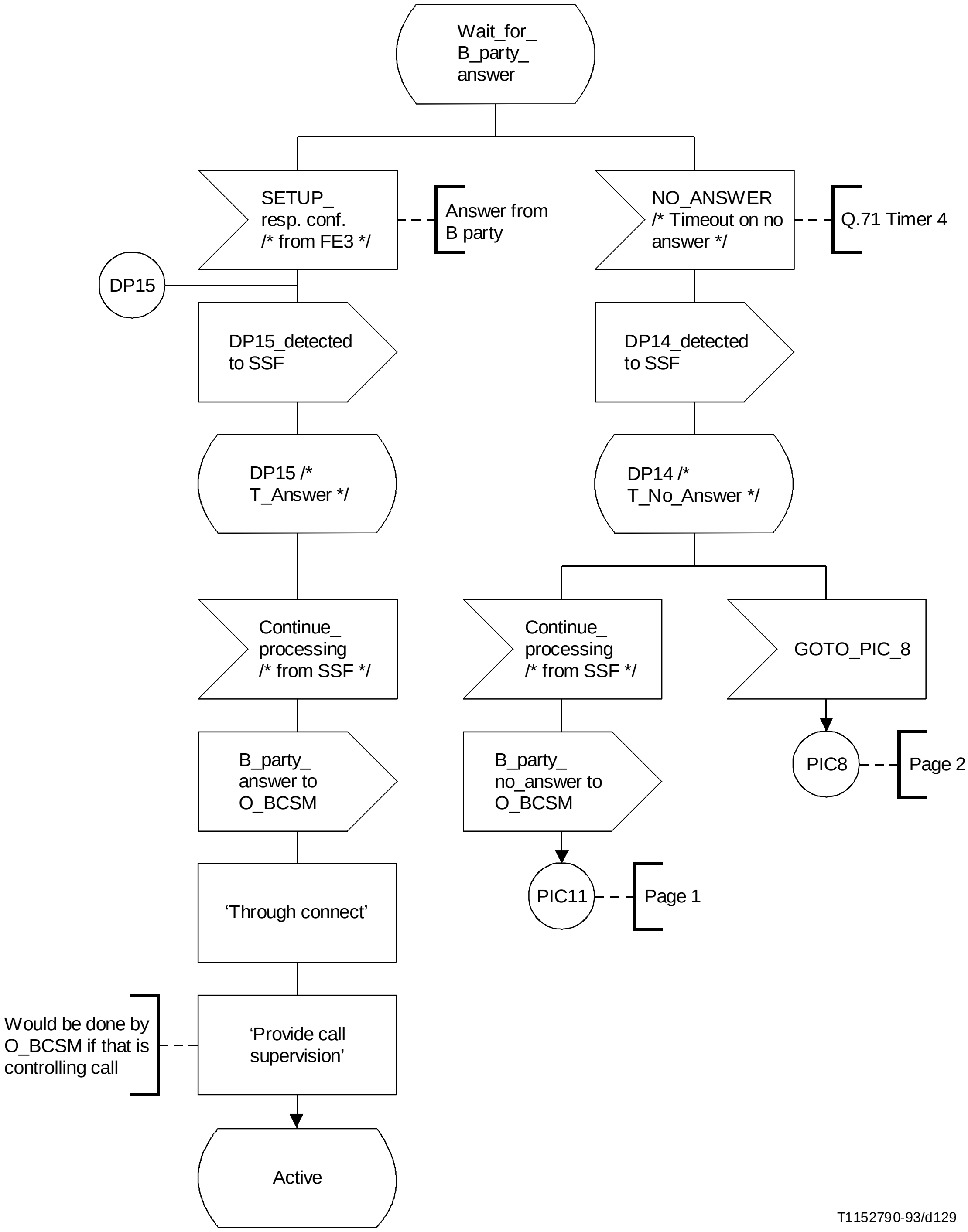


FIGURE B.5/Q.1219 (Sheet 3 of 6)

SDL for terminating BCSM

PROCESS Terminating_BCSM;

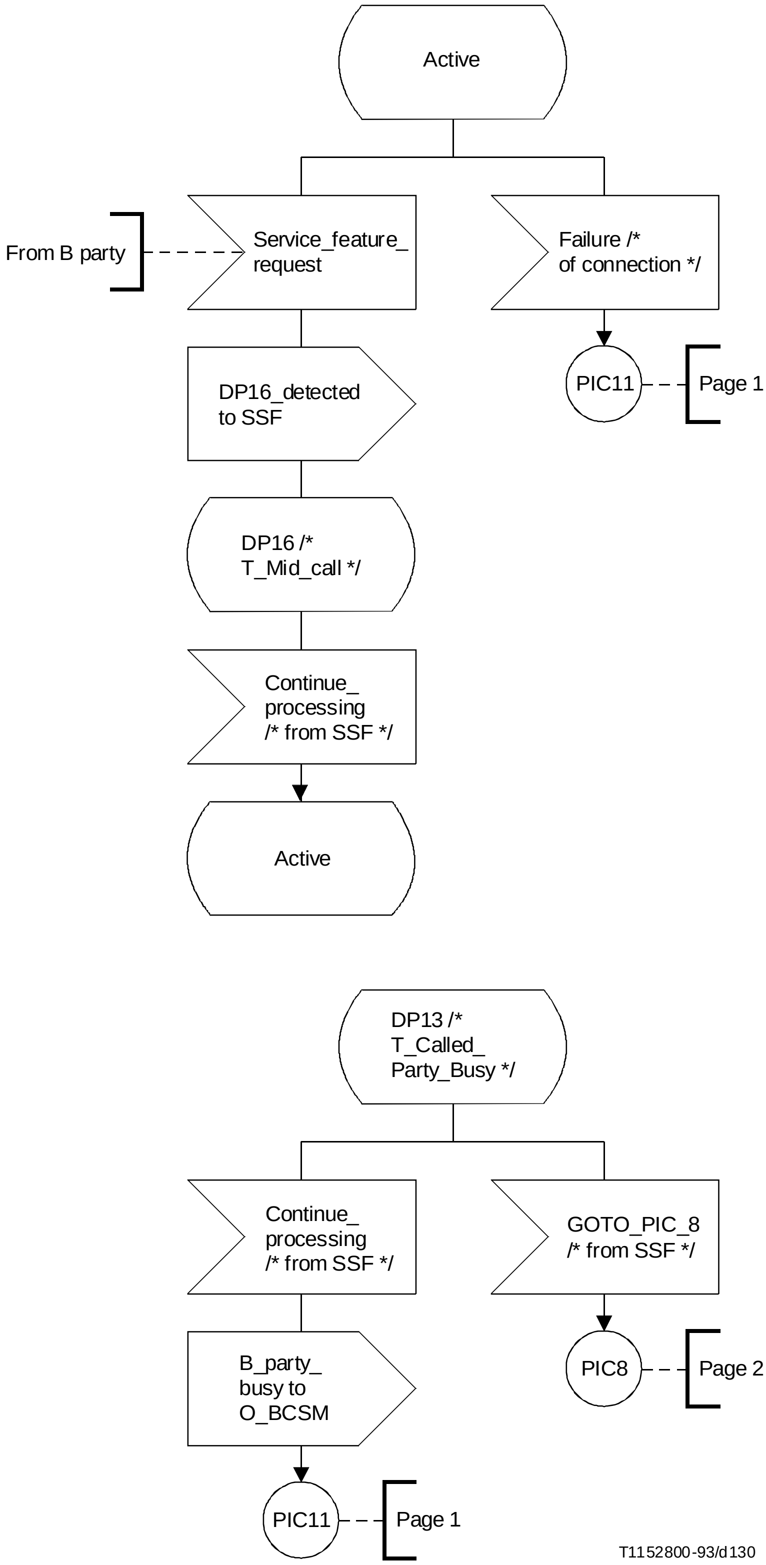


FIGURE B.5/Q.1219 (Sheet 4 of 6)

SDL for terminating BCSM

PROCESS Terminating_BCSM;

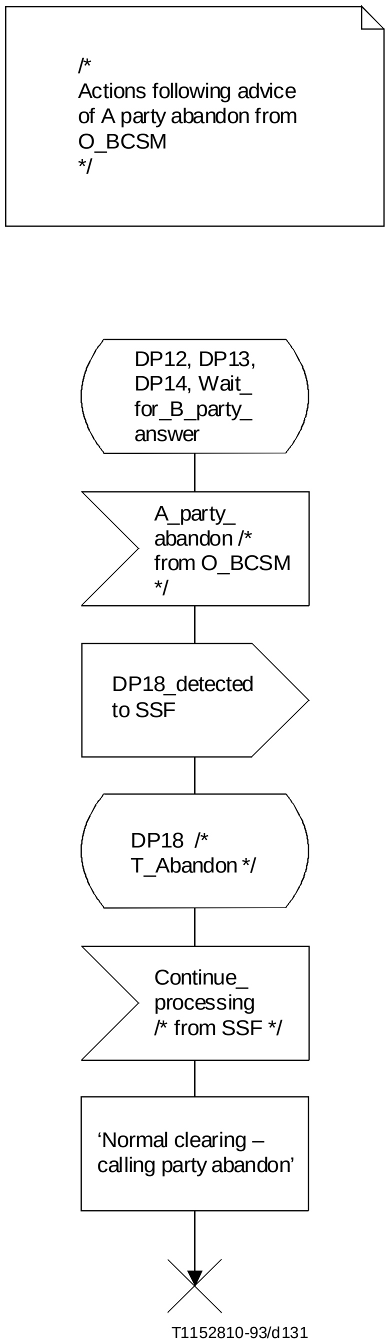


FIGURE B.5/Q.1219 (Sheet 5 of 6)

SDL for terminating BCSM

PROCESS Terminating_BCSM;

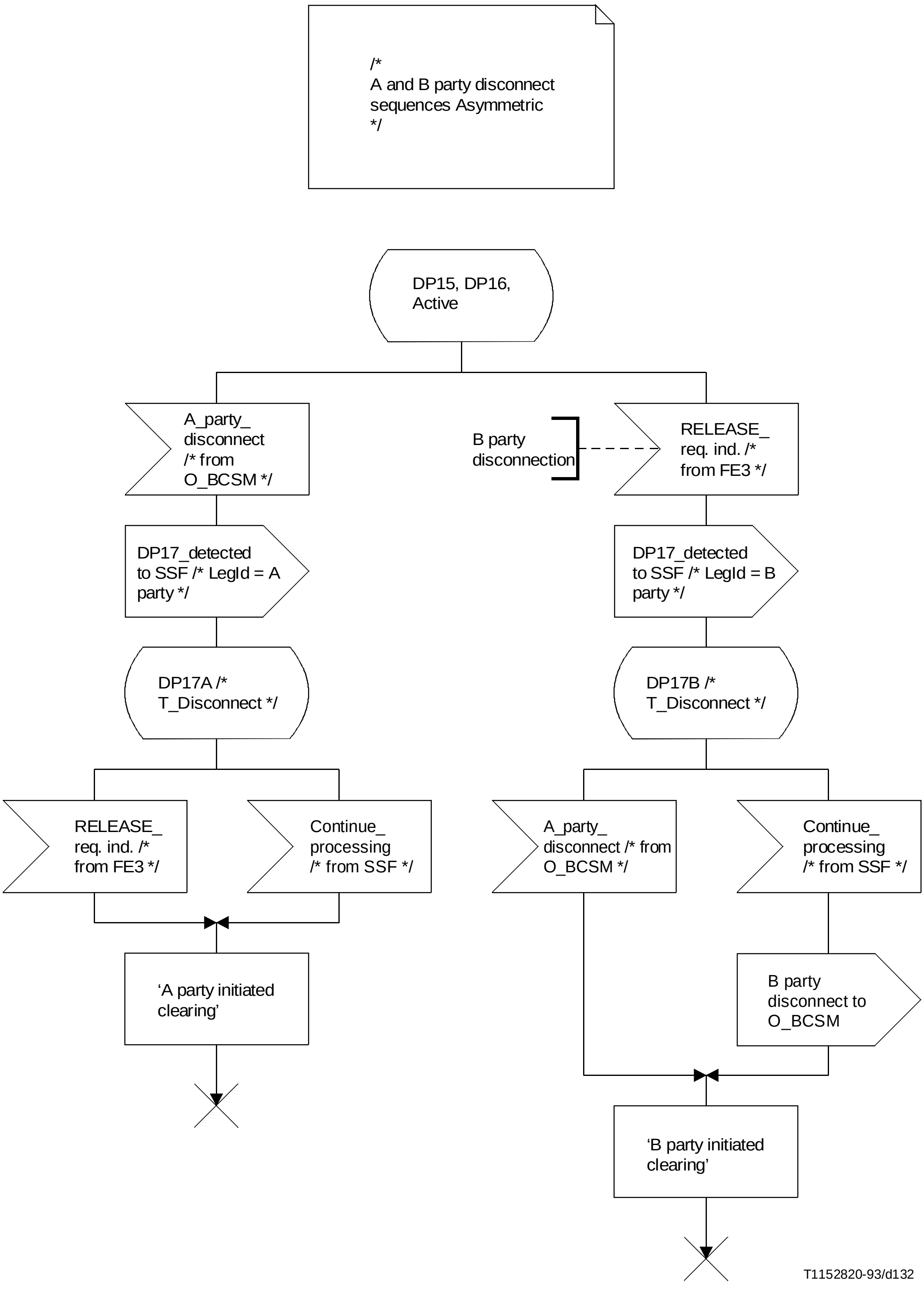


FIGURE B.5/Q.1219 (Sheet 6 of 6)

SDL for terminating BCSM

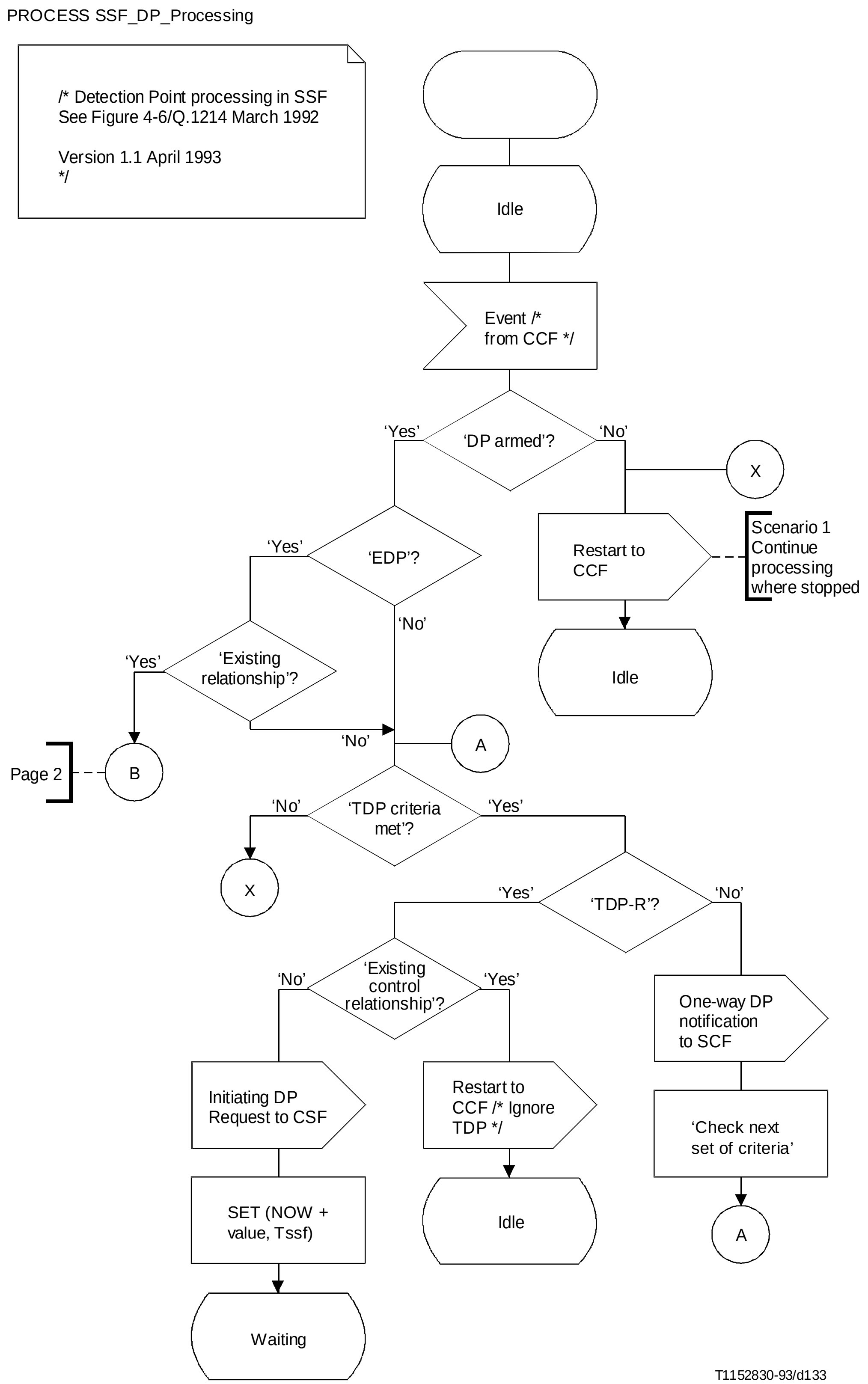


FIGURE B.6/Q.1219 (Sheet 1 of 5)

SDL for Detection Point processing

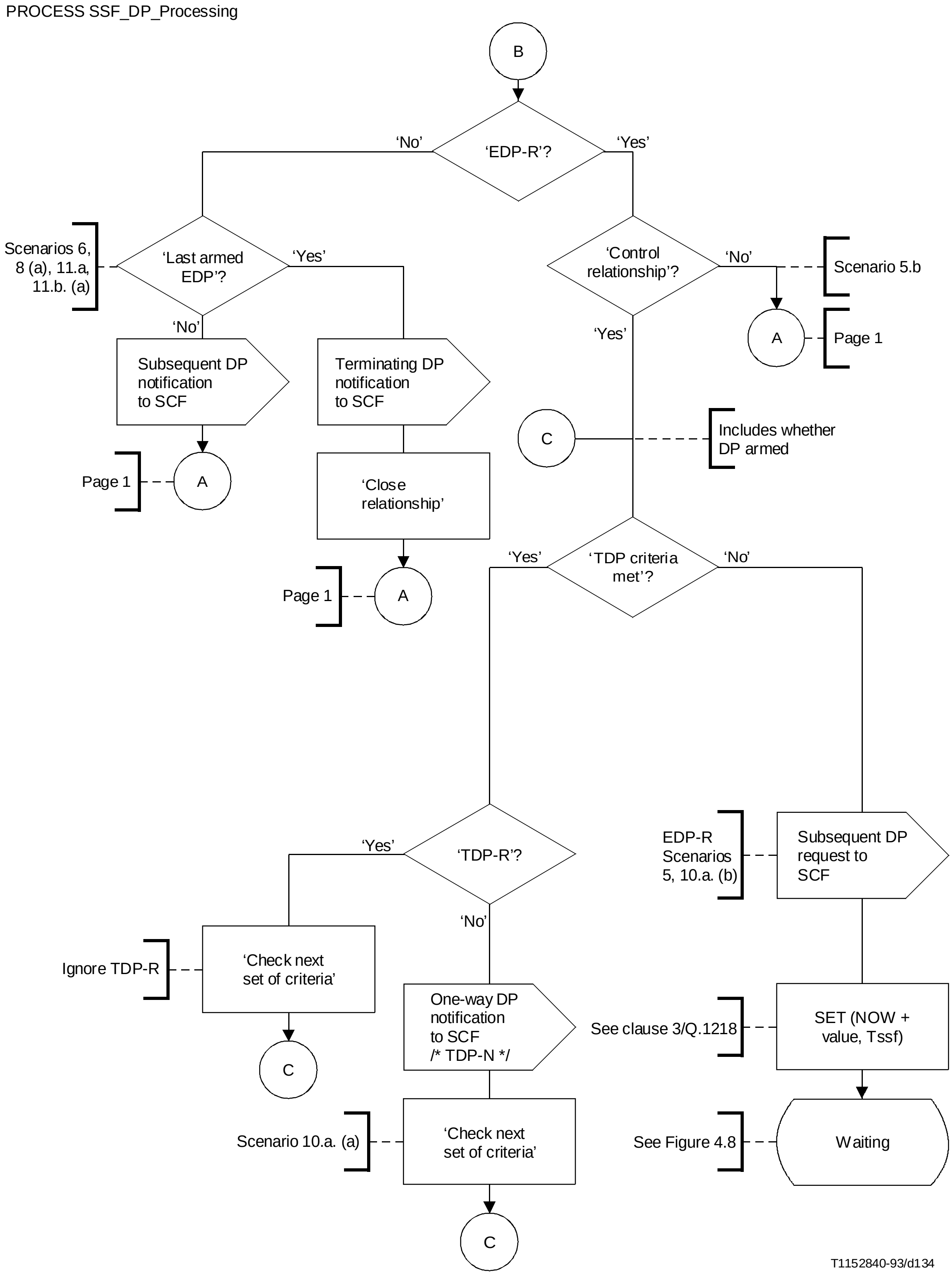


FIGURE B.6/Q.1219 (Sheet 2 of 5)

SDL for Detection Point processing

PROCESS SSF_DP_Processing

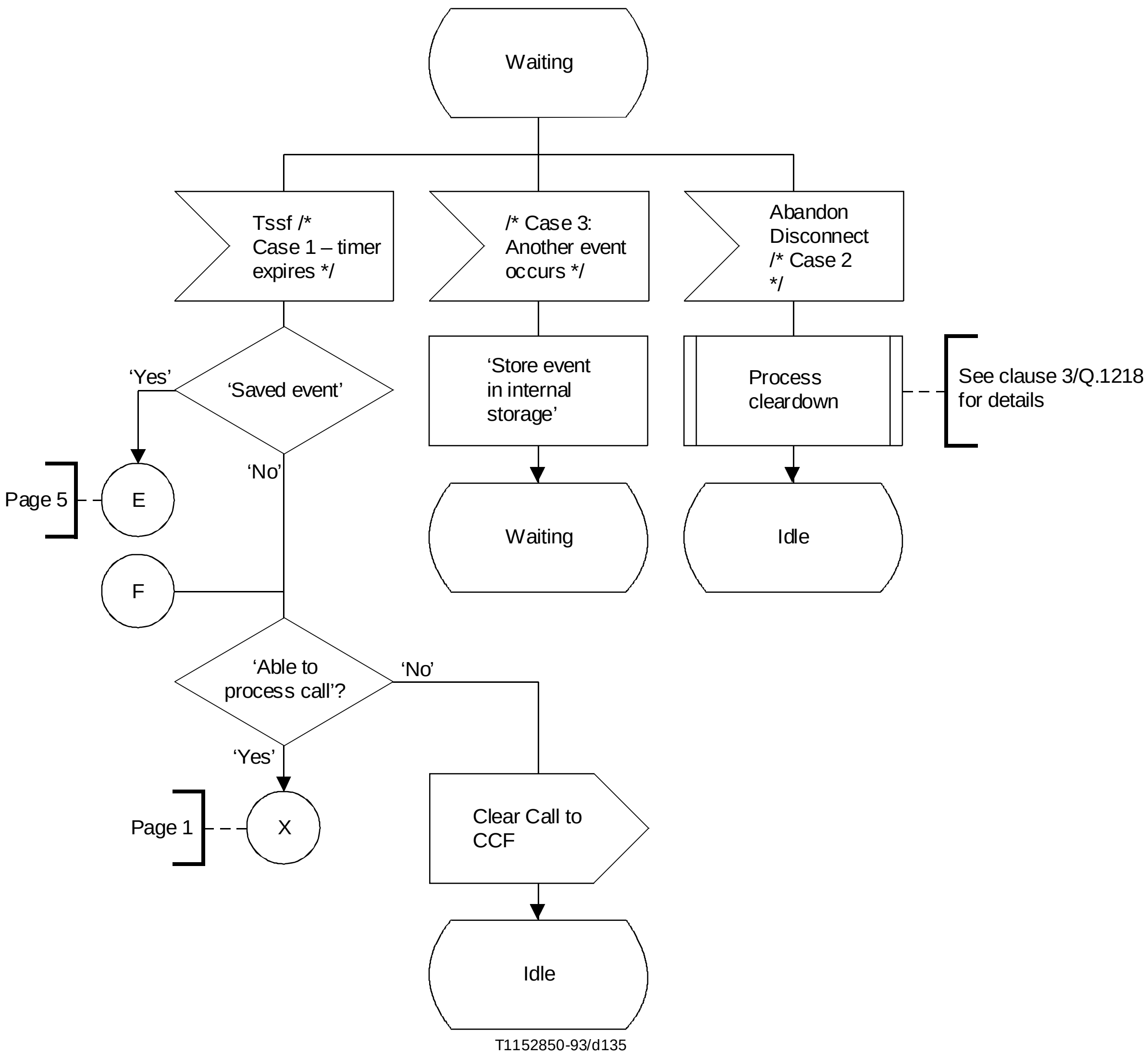


FIGURE B.6/Q.1219 (Sheet 3 of 5)

SDL for Detection Point processing

PROCESS SSF_DP_Processing

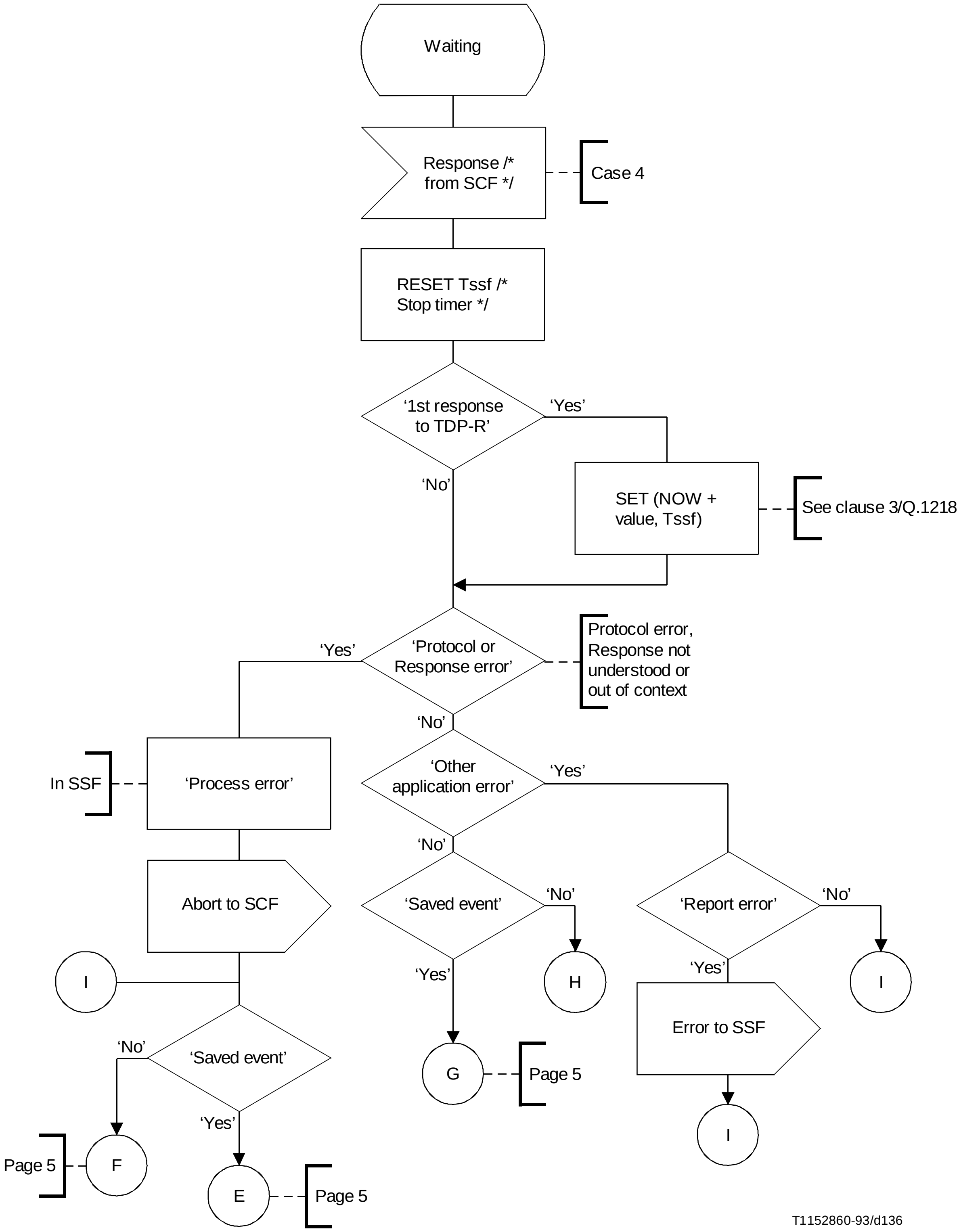


FIGURE B.6/Q.1219 (Sheet 4 of 5)

SDL for Detection Point processing

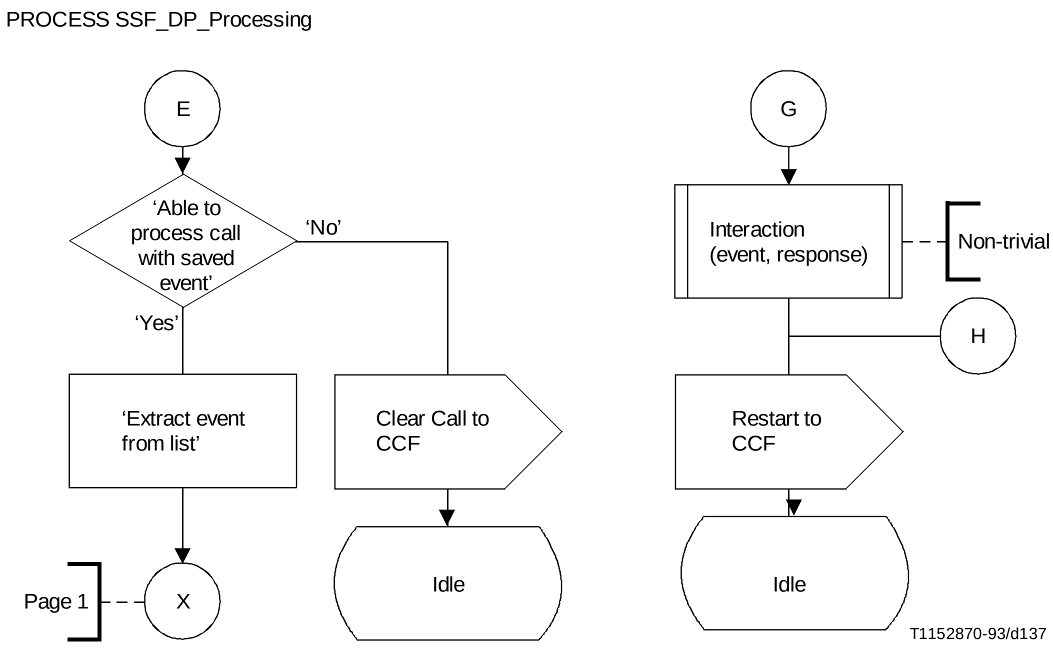


FIGURE B.6/Q.1219 (Sheet 5 of 5)

SDL for Detection Point processing

References

[1] CCITT SG XI Recommendation Q.76, *Service Procedures for Universal Personal Telecommunication Functional Modelling and Information Flows*, version 2, Geneva, May 1993.

[2] CCITT SG I Recommendation F.851, *Universal Personal Telecommunications (UPT) Service Description*, version 9, Geneva, April 1993.