TR 101 616 V1.1.1 (1998-12)

Technical Report

Network Aspects (NA); Intelligent Network (IN); Feasibility of standardization of aspects of Service Creation Environment (SCE) output to network elements via the Service Management Function (SMF)

Reference

DTR/NA-060606 (eyo00ics.PDF)

Keywords

IN, service

ETSI

Postal address

F-06921 Sophia Antipolis Cedex - FRANCE

Office address

650 Route des Lucioles - Sophia Antipolis
Valbonne - FRANCE
Tel.: +33 4 92 94 42 00 Fax: +33 4 93 65 47 16
Siret N° 348 623 562 00017 - NAF 742 C
Association à but non lucratif enregistrée à la
Sous-Préfecture de Grasse (06) N° 7803/88

Internet

secretariat@etsi.fr
Individual copies of this ETSI deliverable
can be downloaded from
http://www.etsi.org

Contents

Intellectual Property Rights

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

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

Foreword

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

Introduction

As the Intelligent Network concept becomes implemented more and more into today's telecommunication networks, the extent to which its promises are fulfilled is becoming clearer. One of the important promises was vendor independence; operators should no longer be bound to a single vendor and vendors would still be capable of selling equipment to parties which have also chosen other vendors.

Essential for this vendor independence is the standardization of concepts and protocols. The first interfaces that have been standardized are network interfaces, resulting in the possibility to connect SSPs, SCPs, SDPs, and IPs from different vendors.

The present document will discuss what will be needed to extend the merits of vendor independence up to the choice of the SCE products. The aim is to stimulate the discussion on how to achieve efficient and rapid service creation in a multi-vendor IN.

1 Scope

The present document analyses the service aspects and management aspects which are needed at the SCEF-SMF interface and identifies which of them would be feasibile to standardize in order to enable service creation for a multi-vendor environment.

The scope of this feasibility study is to:

a) identify and select requirements on the standardization of SCE interfaces (technical issues and commercial/business issues). Considering both functional architecture and business role models to clarify and prioritize the requirements;

b) identify possible technical solutions;

c) identify feasibility and acceptability of the possible technical solutions in the time frame of IN CS-3.

The scope of the work includes the SCEF-SMF interfaces. The scope of the work does not include the SMF-FE interfaces. However, the logical relationship between SCEF and the Functional Entities (FEs) can impact the SCEF-SMF interface.

The basic rationale for this work is to be able to create services using all involved (and possibly multivendor) IN Functional Entities - not only using the SCF, but extending also to other IN Functional entities, such as SDF, SRF or SSF.

This work is actually carrying out recommendation 5 of the PAC EG2 report [3].

2 References

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

- References are either specific (identified by date of publication, edition number, version number, etc.) or non-specific.
- For a specific reference, subsequent revisions do not apply.
- For a non-specific reference, the latest version applies.
- A non-specific reference to an ETS shall also be taken to refer to later versions published as an EN with the same number.

[1] ETR 322: "Intelligent Network (IN); Vocabulary of terms and abbreviations for CS-1 and CS-2".

[2] ITU-T Recommendation Q.1224: "Distributed functional plane for intelligent network Capability Set 2".

[3] PAC EG2 report on standardizing service creation environments.

3 Definitions and abbreviations

3.1 Definitions

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

execution platform: all IN Physical Entities that can be involved in real time execution of a service (e.g. SSP, SCP, SDP, SRP).

management definition (M-def): a definition containing all functions needed to be able to manage new functionality (e.g. new charging procedures, additional customer management functionality service instance data structures). It is related to the SCEF-SMF interface.

service creation logical interface: a logical interface exists between SCEF and all other functional entities; it describes the logical relationship between the SCEF and the execution platform in the service creation process. The actual information flow will always be via the SMF. Example: a programming interface on the SCF for the SCEF.

service definition (S-def): a definition containing all functions needed to be able to manage direct service execution (e.g. service logic, service data, announcements, trigger tables). It is related to the logical interfaces between SCEF and execution platform.

service instance data structure: a data structure that is sent from SCEF to SMF in order to allow the SMF to configure a service and/or customize it for individual subscribers. The service instance data structure can be considered as a sort of template.

3.2 Abbreviations

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

AOPL	Application Oriented Parsing Language
API	Application Programming Interface
ASN	Abstract Syntax Notation
BSS	Business Support Systems
CID	Call Instance Data
CORBA	Common Object Request Broker Architecture
CPE	Customer Premises Equipment
DPE	Distributed Processing Environment
IN	Intelligent Network
INAP	Intelligent Network Application Part
ISDN	Integrated Services for Digital Network
IT	Information Technology
NEM	Network Element Management system
ORB	Object Request Broker
PBX	Private Branche eXchange
PSTN	Public Switched Telephone Network
SCE	Service Creation Environment
SCEF	Service Creation Environment Function
SCF	Service Control Function
SCP	Service Control Point
SDF	Service Data Function
SDL	Specification and Description Language
SDP	Service Data Point
SIB	Service Independent Building Block
SID	Service Instance Data
SLC	Service Life Cycle
SMF	Service Management Function
SMS	Service Management System
SPIN	Service Provisioning for IN
SRF	Specialized Resource Function

SRP	Specialized Resource Point
SSD	Service Script Data
SSF	Service Switching Function
SSP	Service Switching Point
QoS	Quality of Service

4 Current situation for service creation interfaces

In the current situation, service creation interfaces are proprietary. The SCE, SMS and SCP are a single vendor product combination. Management for the other Functional Elements is usually not integrated. See figure 1.

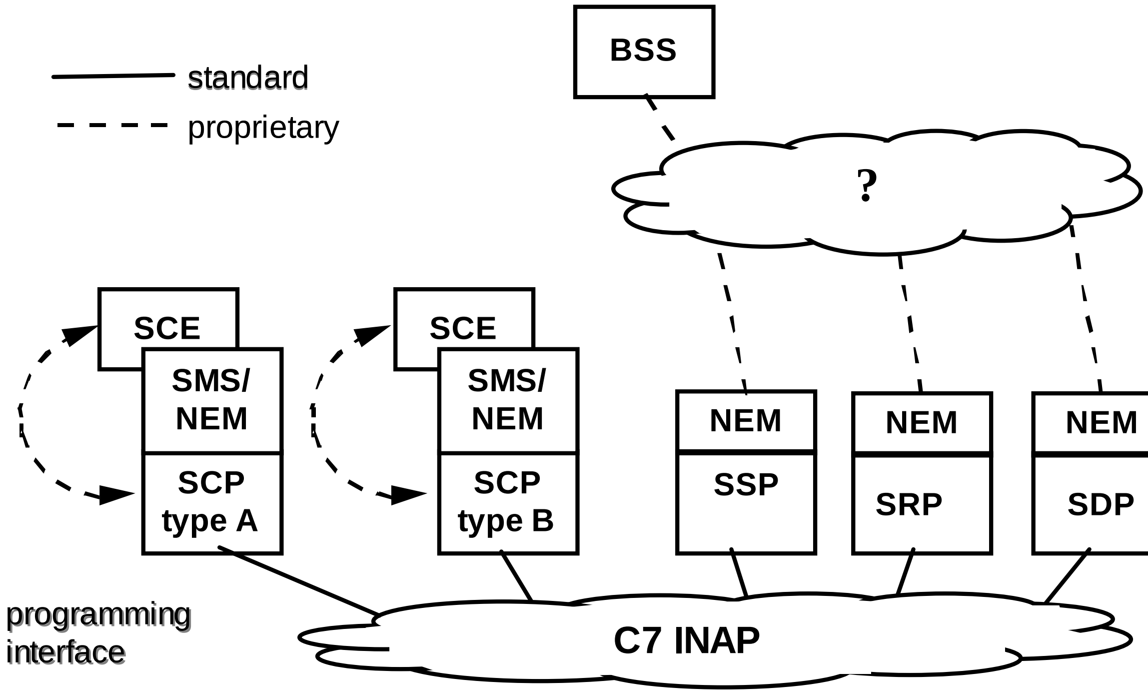

Figure 1: Current situation for service creation interfaces

The programming interface in the SCE is dedicated only to the SCP of the same vendor. The service logic, service data and data entry forms (e.g. for subscription) are only applicable to the SCP and the SMS of that particular vendor.

Other aspects of service creation (defining trigger tables, announcements, etc.) are at present usually not defined in the SCE and are entered into the SSP, SRP, SDP with separate Network Element Management (NEM) systems. Coordination of all these activities is not supported by standard systems.

Also the relation with operator specific Business Support Systems (BSS), such as billing systems, customer care systems, operator screens, entering user data, statistics generation, etc. is not described by standard specifications.

As a result, in a multi-vendor IN environment the current situation may give operators the following problems:

- for the creation of a new service, the operator is dependent on the vendors of his network equipment;
- multiple SCEs are required to be able to run the service in the whole network (in case of multi-vendor SCPs);
- no defined management support for distribution, coordination and maintenance of the new service elements.

As a result, the operator suffers from:

- high vendor dependency;
- cost inefficiency;
- slow time-to-market.

5 Rationale

In this clause the reasons for this study are summarized.

- **Service creation for all IN functional entities.** Instead of only taking into account the development of the service logic for the SCPs, also the configuration of the other IN-elements (SSPs, IPs, SDPs) is considered.

- **Service creation for different network types.** Service creation impacts more than just the SCP; support of services in more network types (data, voice, Internet, X.25, mobile, leased lines, etc.).

- **Service creation for execution environments of different vendors.** Compare for instance the PBX world, where general applications can run on top of a middleware platform interfacing with each of the proprietary PBX control interfaces.

- **Ability to use the best suited SCE for the task.** Considering the rapid software changes in the computing revolution, it is very likely that operators will wish to use a variety of differing service creation environments simultaneously, as some may be better for developing different types of services: e.g. a traditional telecoms supplier for line based services, like monitoring and call completion; and a computer vendor for management services, like special charging.

- **SCE as mature product.** Open interfaces are not only a basic requirement from the operators perspective, but also from a vendors point of view. Service Creation and Management applications are mature selling products and not something which is just sold together with SCPs. Therefore a vendor wants the opportunity to sell his SCE and service creation methodology and support tools to more operators than just the ones that have purchased its SCPs. The field of a specialized service creation environment would be a more attractive option for a vendor.

- **One service creation organization.** An operator only wants one service creation organization, that deals with all the aspects of service creation - not only the service logic design. His management organization will be able to take care of all deployment activities. This leads to higher cost efficiency and lower vendor dependency. As a result, an operator will be able to launch new services much quicker.

- **3rd party service creation.** An open service creation interface will allow 3rd parties to develop services using the network resources offered by the network provider. For example if network A provides operator services for other administrations thus if another administration wants operator support for a service they have created they will have to negotiate with network A. The other administration cannot just deploy into network A. The implication is that service creation in the real world network will require creation and deployment by many organizations. Having open interfaces for service deployment is therefore desirable because it is not always possible for one administration to do all the deployment themselves.

6 Service creation interfaces

In this clause it is discussed which interfaces between the IN functional entities are involved in this feasibility study.

A **logical interface** exists between SCEF and all other functional entities; it describes the logical relationship between the SCEF and the execution platform in the service creation process. The actual information flow will always be via the SMF. Example: a programming interface on the SCF for the SCEF.

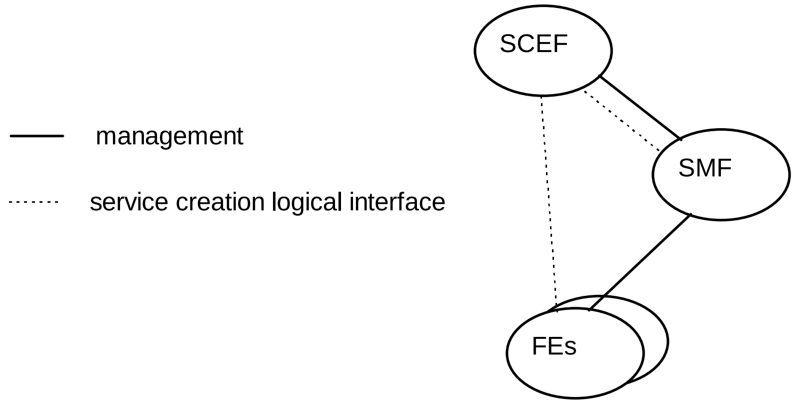

Figure 2: Logical interfaces for the service creation process

7 Working method

In this clause the working method for this feasibility study is outlined. A three step approach is performed and described in the next clauses. Clauses 8 to 10 correspond to each of these steps.

Step 1: requirements on SCE interfaces

In this feasibility study the ETSI Service Life Cycle (SLC) [1] will be used to identify requirements imposed on the SCEF-SMF interface, see figure 3.

Especially the following steps are important for this feasibility study:

- Service creation; consisting of the activities service specification, service validation, service type development and building block development.
- Service acceptance testing; the service acceptance test and/or a service pilot.
- Service deployment; downloading the service logic and service data into the network elements, and installation of management functions and management data associated with the service.

Needs analysis

⇓

Service Creation

⇓

Service Acceptance Testing

⇓

Service deployment

⇓

Service Operation and provisioning

⇓

Service removal

Figure 3: ETSI Service Life Cycle

Re-development of services involving regression handling and version handling resulting in feedback loops in the SLC may have an impact on the SCE interfaces.

Step 2: Possible technical solutions

In this second step technical solution for the two aspects of the work: the SCEF-SMF interface and the service creation logical interfaces are identified.

Step 3: Feasibility and acceptability of the standardizing the SCE output

The third step is to identify the acceptability and feasibility of standardizing the SCE output.

8 Requirements on SCE interfaces

8.1 SCE functions

The following requirements are identified for the SCE functions.

The SCE should provide:

- service logic programs [2];
- service logic scripts (e.g. SRF mini scripts) [2];
- service data scripts (e.g. SDF mini script) [2];
- support for changes resulting from re-development of the service;
- service instant data structures (e.g. billing support data structures);

NOTE: The management of the parameters in the service instant data structure may be controlled by different roles, e.g. network manager, service provider, service subscriber, service user.

- feature interaction requirements;
- trigger tables for service specific events;
- service configuration data, e.g. trading / QoS data (time outs, alarms) / SRF address;
- user data scripts;
- naming technologies, trading;
- charging data scripts.

8.2 Service creation logical interfaces

The following requirements are identified for the service creation logical interfaces:

- It should be possible to define service logic in a SCEF and execute it on other vendor's execution platforms.
- It should be possible that service logic is portable across execution platforms.

NOTE: For some scenarios, for example operators who choose not to invest in more than one SCE hardware platform, it may be a requirement to site one or more SCEF entities on a single SCE hardware platform. Different methods may exist: use of common APIs, use of standard or open operating systems, etc.

- Interworking between naming technologies is required to resolve SSD/SID/CID addressing (reading/writing).
- Considering the need to be able to capitalize on already developed software, when the underlying platform is being replaced (software portability), or when a variant of the application is introduced. This may relate to the SCP, SDP and SCE. As these entities may become technologically obsolete; i.e. in need of upgrade (software platform or hardware platform). In future this may happen fairly regularly. This is a desirable requirement for an operator that could be difficult to implement.
- 3rd party service creation has to be possible.

9 Possible technical solutions

In this clause a possible technical solution to meet the requirements for the SCE is described. Open interfaces for service creation are required to overcome most of these problems.

9.1 SCEF modelling

The service creation could provide a standardized output, which will consist of two types:

- "Service Definitions" (S-def);
- "Management Definitions" (M-def).

The S-def contains all functions which are needed to manage direct service execution (e.g. service logic, service support data, announcements, trigger tables). It is related to the logical interfaces between SCEF and execution platform.

The M-def contains all functions needed to be able to manage new functionality (e.g. new charging procedures, additional customer management functionality, service instant data structures). It is related to the SCEF-SMF interface.

When the format of S-def and M-def is fixed, the deployment of the Service-definition over the involved network elements (identified in the SCE) can be done automatically by the Service Creation Manager (SCM), (see figure 4).

The SCM takes care of:

- distribution to all of the involved Network Elements (not only the SCP);
- coordination (e.g. announcementId in the SCP with an announcement in the SRP);
- translation (generating actual code for the execution platforms).

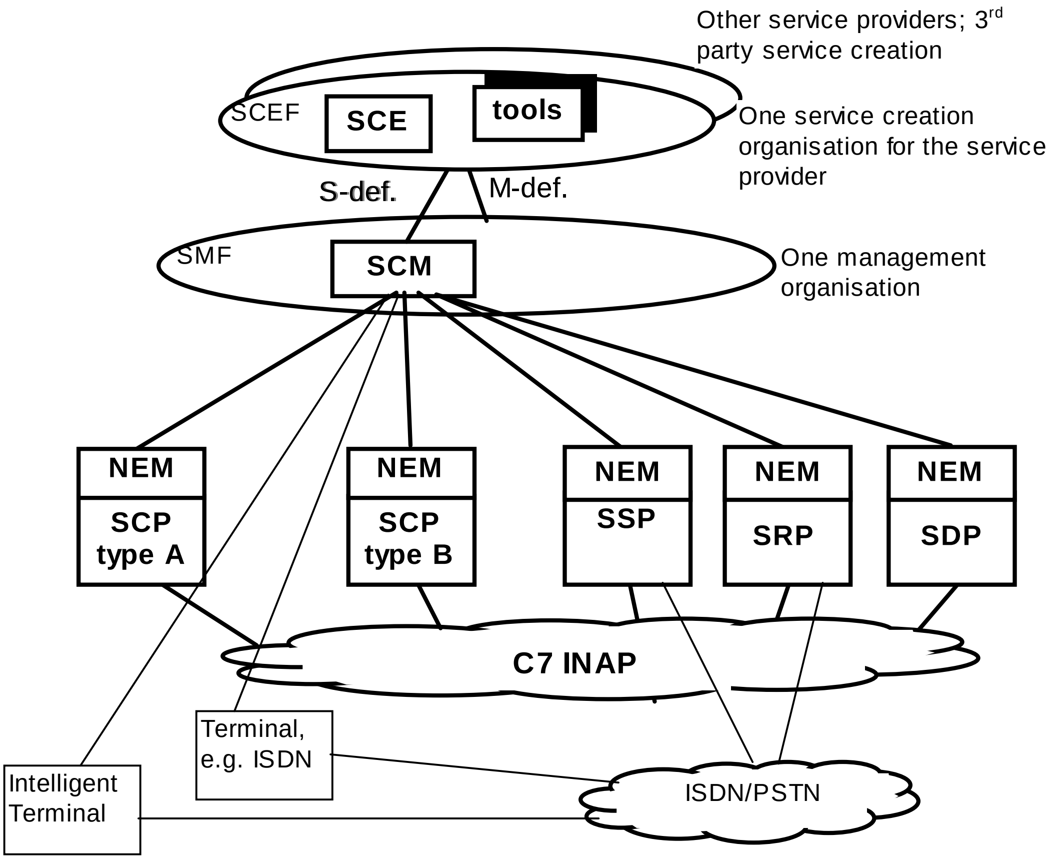

Figure 4: Target situation for service creation interfaces

In IN CS-2 the terminal may include a computer interface which uses e.g. dial up or remote operations. Intelligent terminals may receive service logic via DPE services, which means that the SCE to terminal logic interface possibly will imply a service logic / data repository outside the network.

9.2 Service creation logical interfaces

Several ways are identified to realize a service creation logical interface:

- Object Oriented techniques:
    - Object Request Broker (ORB) client on every FE;
    - use of CORBA;
- a vendor specific API on every FE, middleware layer on top of that offering an API for application designers;
- standard or open operating system;
- a virtual machine in every execution environment (Java like, "develop once, run anywhere");
- Application Oriented Parsing Language (AOPL);
- service definition in SDL or UML.

10 Feasibility and acceptability

In the time frame of CS-3 the standardization of the SCE output to network elements via the SMF is not feasible. The following reasons are identified:

- The benefits for parties within the business model (e.g. vendor, network operator, service provider) are not clear. This slows down the work performed on the subject.
- The fact that the functionality of standardized SCE interfaces could be limited compared to the functionality already offered with vendor specific SCEs.
- Since the SCEF functional model is not standardized, it is difficult to standardize the procedures.
- The fact that the logical interface between the SCEF and the execution platform is not yet standardized.

It is concluded that standardization of the SCE output to network elements via the SMF is not feasible in the short term (within the time frame of CS-3). The subject is for further study.

In the medium term the balance between the need for competition in the SCE market and the need to differentiate in SCE functionality will determine the feasibility and acceptability of the standardization of the SCE interfaces.

On the long term DPE technologies and standardized APIs could allow software portability and drive the need for standardizing the SCE interfaces.

History

Document history		
V1.1.1	December 1998	Publication

ISBN 2-7437-2708-X
Dépôt légal : Décembre 1998