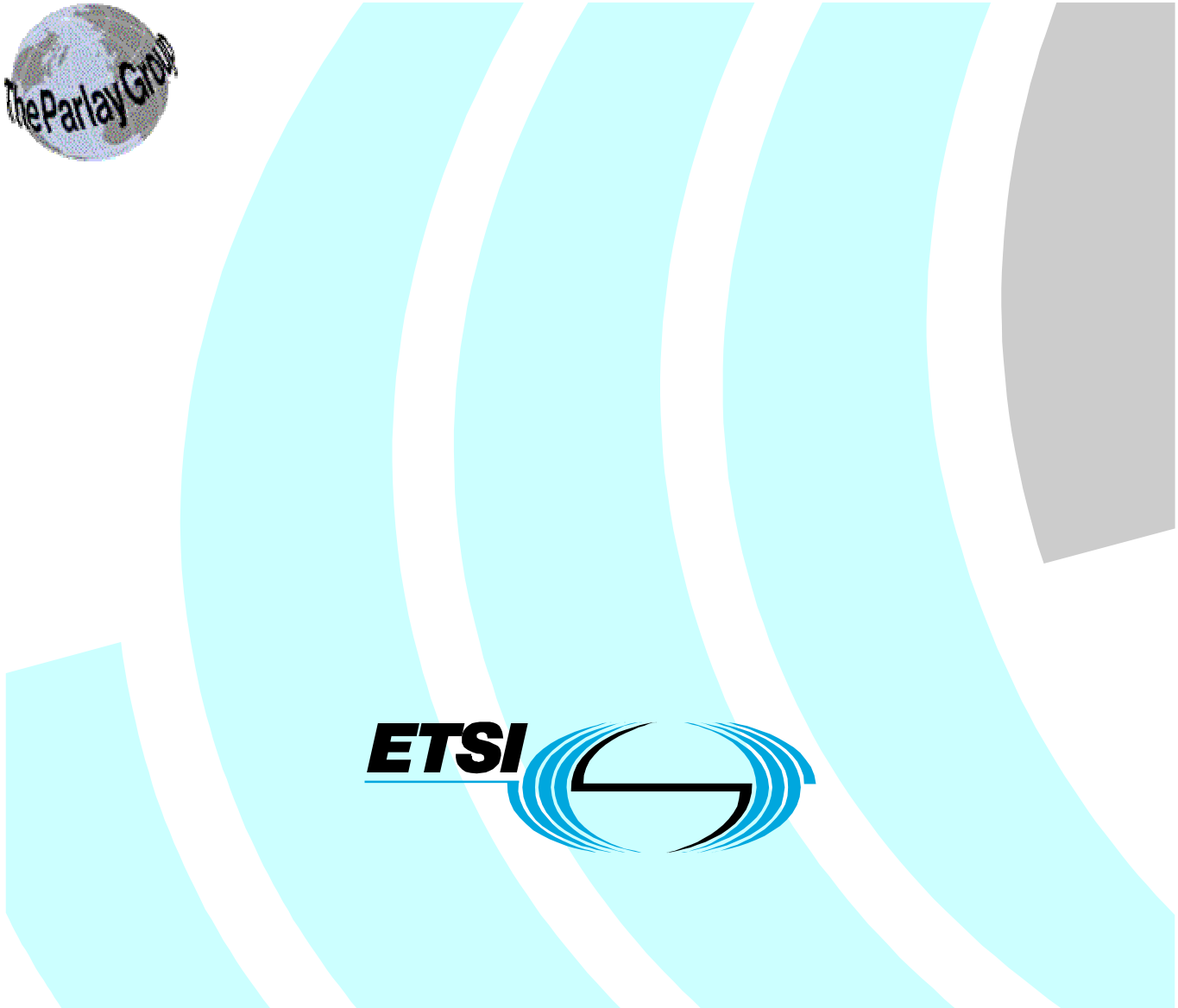
ETSI ES 201 915-1 V1.1.1 (2002-02)

ETSI Standard

Open Service Access (OSA);
Application Programming Interface (API);
Part 1: Overview



Reference
DES/SPAN-120070-1

Keywords
API, OSA, IDL, UML

ETSI

650 Route des Lucioles
F-06921 Sophia Antipolis Cedex - 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

Contents

Intellectual Property Rights 4
Foreword 4
1 Scope 5
2 References 5
3 Definitions and abbreviations 6
3.1 Definitions 6
3.2 Abbreviations 7
4 Open Service Access API's 8
5 Document structure 9
6 Methodology 10
6.1 Tools and Languages 10
6.2 Packaging 10
6.3 Colours 10
6.4 Naming scheme 10
6.5 State Transition Diagram text and text symbols 11
6.6 Exception handling and passing results 11
6.7 References 11
6.8 Strings and Collections 11
6.9 Prefixes 11
Annex A (normative): OMG IDL 12
A.1 Tools and Languages 12
A.2 Strings and Collections 12
A.3 Naming space across CORBA modules 12
Annex B (informative): Bibliography 13
History 14

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

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 ETSI 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 ETSI Standard (ES) has been produced by ETSI Technical Committee Services and Protocols for Advanced Networks (SPAN).

The present document is part 1 of a multi-part deliverable covering Open Service Access (OSA); Application Programming Interface (API), as identified below. The API specification (ES 201 915) is structured in the following parts:

Part 1: "Overview";

Part 2: "Common Data Definitions";

Part 3: "Framework";

Part 4: "Call Control SCF";

Part 5: "User Interaction SCF";

Part 6: "Mobility SCF";

Part 7: "Terminal Capabilities SCF";

Part 8: "Data Session Control SCF";

Part 9: "Generic Messaging SCF";

Part 10: "Connectivity Manager SCF";

Part 11: "Account Management SCF";

Part 12: "Charging SCF".

The present document has been defined jointly between ETSI, The Parlay Group [24] and the 3GPP, in co-operation with a number of JAIN™ Community [25] member companies.

The present document forms part of the Parlay 3.0 set of specifications.

1 Scope

The present document is the part 1 of the Stage 3 specification for an Application Programming Interface for Open Service Access (OSA), and provides an overview of the content and structure of the various parts of this specification, and of the relation to other standards documents.

The OSA specifications define an architecture that enables service application developers to make use of network functionality through an open standardised interface, i.e. the OSA API's.

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 and/or edition number or version number) or non-specific.
- For a specific reference, subsequent revisions do not apply.
- For a non-specific reference, the latest version applies.

[1] ETSI TR 121 905: "Universal Mobile Telecommunications System (UMTS); Vocabulary for 3GPP Specifications (3GPP TR 21.905)".

[2] ETSI TS 122 024: "Digital cellular telecommunications system (Phase 2+) (GSM); Universal Mobile Telecommunications System (UMTS); Description of Charge Advice Information (CAI) (3GPP TS 22.024)".

[3] ITU-T Recommendation Q.850: "Usage of cause and location in the Digital Subscriber Signalling System No. 1 (DSS1) and the Signalling System No. 7 (SS No. 7) ISDN User Part (ISUP)".

[4] ITU-T Recommendation Q.2931: "Digital Subscriber Signalling System No. 2 - User-Network Interface (UNI) layer 3 specification for basic call/connection control".

[5] ETSI TS 122 101: "Universal Mobile Telecommunications System (UMTS); Service aspects; Service principles (3GPP TS 22.101)".

[6] World Wide Web Consortium "Composite Capability/Preference Profiles (CC/PP): A user side framework for content negotiation" (http://www.w3.org/TR/NOTE-CCPP/).

[7] ETSI TS 129 002: "Digital cellular telecommunications system (Phase 2+) (GSM); Universal Mobile Telecommunications System (UMTS); Mobile Application Part (MAP) specification (3GPP TS 29.002)".

[8] ETSI TS 129 078: "Universal Mobile Telecommunications System (UMTS); Digital cellular telecommunications system (Phase 2+); Customised Applications for Mobile network Enhanced Logic (CAMEL) Phase 3; CAMEL Application Part (CAP) specification (3GPP TS 29.078)".

[9] Wireless Application Protocol (WAP), Version 1.2: "User Agent Profiling Specification" (WAP-174) (http://www.wapforum.org/what/technical_1_2.htm).

[10] Wireless Application Protocol (WAP), Version 1.2: "WAP Service Indication Specification" (WAP-167) (http://www.wapforum.org/what/technical_1_2.htm).

[11] Wireless Application Protocol (WAP), Version 1.2: "Push Architectural Overview" (WAP-165) (http://www.wapforum.org/what/technical_1_2.htm).

[12] Wireless Application Protocol (WAP), Version 1.2: "Wireless Application Protocol Architecture Specification" (WAP-100) (http://www.wapforum.org/what/technical_1_2.htm).

[13] "SUN IDL Compiler" (http://www.javasoft.com/products/jdk/idl/index.html).

[14] "UML Unified Modeling Language" (http://www.rational.com/uml).

[15] "Object Management Group" (http://www.omg.org/).

[16] ETSI TS 122 002: "Digital cellular telecommunications system (Phase 2+) (GSM); Universal Mobile Telecommunications System (UMTS); Circuit Bearer Services (BS) supported by a Public Land Mobile Network (PLMN) (3GPP TS 22.002)".

[17] ETSI TS 122 003: "Digital cellular telecommunications system (Phase 2+) (GSM); Universal Mobile Telecommunications System (UMTS); Circuit Teleservices supported by a Public Land Mobile Network (PLMN) (3GPP TS 22.003)".

[18] ETSI TS 124 002: "Digital cellular telecommunications system (Phase 2+) (GSM); Universal Mobile Telecommunications System (UMTS); GSM - UMTS Public Land Mobile Network (PLMN) access reference configuration (3GPP TS 24.002)".

[19] ITU-T Recommendation Q.763: "Signalling System No. 7 - ISDN User Part formats and codes".

[20] ITU-T Recommendation Q.931: "ISDN user-network interface layer 3 specification for basic call control".

[21] ISO 8601: "Data elements and interchange formats - Information interchange - Representation of dates and times".

[22] ISO 4217: "Codes for the representation of currencies and funds".

[23] ISO 639: "Code for the representation of names of languages".

[24] http://www.parlay.org

[25] http://www.java.sun.com/products/jain

[26] IETF RFC 822: "Standard for the format of ARPA Internet text messages".

[27] IETF RFC 1738: "Uniform Resource Locators (URL)".

[28] ETSI TS 129 198 (V3.2.0): "Universal Mobile Telecommunications System (UMTS); Open Service Architecture Application Programming Interface - Part 1 (3GPP TS 29.198 version 3.2.0 Release 1999)".

[29] ETSI TS 123 107: "Universal Mobile Telecommunications System (UMTS); Quality of Service (QoS) concept and architecture" (3GPP TS 23.107).

3 Definitions and abbreviations

3.1 Definitions

For the purposes of the present document, the terms and definitions given in TS 122 101 [5] and the following apply:

applications: services, which are designed using service capability features

gateway: synonym for Service Capability Server

NOTE: From the viewpoint of applications, a Service Capability Server can be seen as a gateway to the core network.

HE-VASP: Home Environment Value Added Service Provider

NOTE: This is a VASP that has an agreement with the Home Environment to provide services.

Home Environment: responsible for overall provision of services to users

Local Service: service which can be exclusively provided in the current serving network by a Value Added Service Provider

OSA Interface: standardised Interface used by application to access service capability features

Personal Service Environment: contains personalised information defining how subscribed services are provided and presented towards the user

NOTE: The Personal Service Environment is defined in terms of one or more User Profiles.

Service Capabilities: bearers defined by parameters, and/or mechanisms needed to realise services

NOTE: These are within networks and under network control.

Service Capability Feature: functionality offered by service capabilities that are accessible via the standardised OSA interface

Service Capability Server: Functional Entity providing OSA interfaces towards an application

Service: alternative for Service Capability Feature (in this specification)

User Interface Profile: contains information to present the personalised user interface within the capabilities of the terminal and serving network

User Profile: label identifying a combination of one user interface profile, and one user services profile

User Services Profile: contains identification of subscriber services, their status and reference to service preferences

Value Added Service Provider: provides services other than basic telecommunications service for which additional charges may be incurred

Virtual Home Environment: concept for personal service environment portability across network boundaries and between terminals

3.2 Abbreviations

For the purposes of the present document, the abbreviations given in TR 121 905 [1] and the following apply:

API	Application Programming Interface
CAMEL	Customised Application for Mobile Network Enhanced Logic
CSE	Camel Service Environment
HE	Home Environment
HE-VASP	Home Environment Value Added Service Provider
HLR	Home Location Register
IDL	Interface Description Language
INAP	Intelligent Networks Application Part
MAP	Mobile Application Part
ME	Mobile Equipment
MExE	Mobile Station (Application) Execution Environment
MS	Mobile Station
MSC	Mobile Switching Centre
OSA	Open Service Access
PLMN	Public Land Mobile Network
PSE	Personal Service Environment
SAT	SIM Application Tool-Kit
SCF	Service Capability Feature
SCP	Service Control Point
SIM	Subscriber Identity Module
SMS	Short Message Service
SMTP	Simple Mail Transfer Protocol
USIM	User Service Identity Module
VASP	Value Added Service Provider
VHE	Virtual Home Environment
VLR	Visited Location Register

WAP	Wireless Application Protocol
WGP	Wireless Gateway Proxy
WPP	Wireless Push Proxy

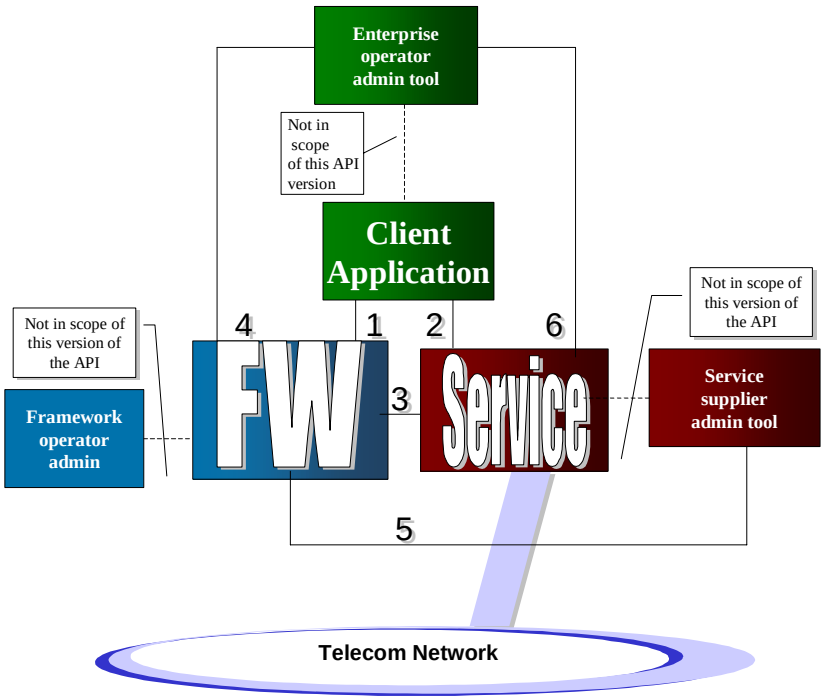
4 Open Service Access API's

The OSA specifications define an architecture that enables service application developers to make use of network functionality through an open standardised interface, i.e. the OSA API's. The network functionality is describes as Service Capability Features or Services (see note). The OSA Framework is a general component in support of Services (Service Capabilities) and Applications.

The OSA API is split into three types of interface classes, Service and Framework.

- Interface classes between the Applications and the Framework, that provide applications with basic mechanisms (e.g. Authentication) that enable them to make use of the service capabilities in the network.
- Interface classes between Applications and Service Capability Features (SCF), which are individual services that may be required by the client to enable the running of third party applications over the interface e.g. Messaging type service.
- Interface classes between the Framework and the Service Capability Features, that provide the mechanisms necessary for multi-vendorship.

These interfaces represent interfaces 1, 2 and 3 of the Figure below. The other interfaces are not yet part of the scope of the work.



Within the OSA concept a set of Service Capability Features has been specified. The OSA documentation is structured in parts. The first Part (the present document) contains an overview, the second Part contains common Data Definitions, the third Part the Framework interfaces. The rest of the Parts contain the description of the SCFs.

NOTE: The terms "Service" and "Service Capability Feature" are used as alternatives for the same concept in this specification. In the OSA API itself the Service Capability Features as identified in the 3GPP requirements and architecture are reflected as 'service', in terms like serviceFactory, serviceDiscovery.

5 Document structure

The OSA API documentation contains two document sets:

The API specification (ES 201 915)
The Parts of the present document ES 201 915 (apart from 1 (the present document) and 2) define the interfaces, parameters and state models that form part of the API specification. UML is used to specify the interface classes. As such it provides a UML interface class description of the methods (API calls) supported by that interface and the relevant parameters and types. The interfaces are specified in IDL.

The Mapping specification of the OSA API's and network protocols (TR 101 917)
The Parts of TR 101 917 contain a possible mapping from the API's defined in ES 201 915 to various network protocols (i.e. MAP [7], CAP [8],..). It is an informative document, since this mapping is considered as implementation/vendor dependent. On the other hand this mapping will provide potential service designers with a better understanding of the relationship of the OSA API interface classes and the behaviour of the network associated to these interface classes.

The purpose of the OSA API is to shield the complexity of the network, its protocols and specific implementation from the applications. This means that applications do not have to be aware of the network nodes a Service Capability Server interacts with in order to provide the Service Capability Features to the application. The specific underlying network and its protocols are transparent to the application.

The API specification ES 201 915 is structured in the following parts:

Part 1: "Overview";

Part 2: "Common Data Definitions";

Part 3: "Framework";

Part 4: "Call Control SCF";

Part 5: "User Interaction SCF";

Part 6: "Mobility SCF";

Part 7: "Terminal Capabilities SCF";

Part 8: "Data Session Control SCF";

Part 9: "Generic Messaging SCF";

Part 10: "Connectivity Manager SCF";

Part 11: "Account Management SCF";

Part 12: "Charging SCF".

The Mapping document TR 101 917 is also structured according to the same parts. A mapping to network protocols is however not applicable for all Parts, but the numbering of Parts is kept. Also in case a Part is not supported in a Release, the numbering of the parts is maintained.

Structure of the Parts of ES 201 915:

The Parts with API specification themselves are structured as follows:

- The Sequence diagrams give the reader a practical idea of how each of the service capability feature is implemented.
- The Class relationships clause show how each of the interfaces applicable to the SCF, relate to one another.
- The Interface specification clause describes in detail each of the interfaces shown within the Class diagram part.
- The State Transition Diagrams (STD) show the progression of internal processes either in the application, or Gateway.

- The Data Definitions clause show a detailed expansion of each of the data types associated with the methods within the classes. Note that some data types are used in other methods and classes and are therefore defined within the Common Data types part of this specification.
- IDL description of the interface (normative Annex).

6 Methodology

Following is a description of the methodology used for the establishment of API specification for OSA.

6.1 Tools and Languages

The Unified Modelling Language (UML) [14] is used as the means to specify class and state transition diagrams.

6.2 Packaging

A hierarchical packaging scheme is used to avoid polluting the global name space. The root is defined as:

org.csapi

6.3 Colours

For clarity, class diagrams follows a certain colour scheme. Blue for application interface packages and yellow for all the others.

6.4 Naming scheme

The following naming scheme is used for documentation.

packages:

lowercase.

Using the domain-based naming (For example, org.csapi)

classes, structures and types. Start with T:

TpCapitalizedWithInternalWordsAlsoCapitalized

Exception class:

TpClassNameEndsWithException

Interface. Start with Ip:

IpThisIsAnInterface

constants:

P_UPPER_CASE_WITH_UNDERSCORES_AND_START_WITH_P

methods:

firstWordLowerCaseButInternalWordsCapitalized()

method's parameters:

firstWordLowerCaseButInternalWordsCapitalized

collections (set, array or list types):

TpCollectionEndsWithSet

class/structure members:

FirstWordAndInternalWordsCapitalized

Spaces in between words are not allowed.

6.5 State Transition Diagram text and text symbols

The descriptions of the State Transitions in the State Transition Diagrams follow the convention:

when_this_event_is_received [guard condition is true] /do_this_action ^send_this_message

Furthermore, text underneath a line through the middle of a State indicates an exit or entry event (normally specified which one).

6.6 Exception handling and passing results

OSA methods communicate errors in the form of exceptions. OSA methods themselves always use the return parameter to pass results. If no results are to be returned a void is used instead of the return parameter. In order to support mapping to as many languages as possible, no method *out* parameters are allowed.

6.7 References

In the interface specification whenever Interface parameters are to be passed as an *in* parameter, they are done so by reference, and the "Ref" suffix is appended to their corresponding type (e.g. IpAnInterfaceRef anInterface), a reference can also be viewed as a logical indirection.

Original type	**IN parameter declaration**	
IpInterface	parm : IN IpInterfaceRef	

6.8 Strings and Collections

For character strings, the *String* data type is used without regard to the maximum length of the string. For homogeneous collections of instances of a particular data type the following naming scheme is used: <datatype>Set.

6.9 Prefixes

OSA constants and data types are defined in the global name space: *org.csapi* module.

Annex A (normative): OMG IDL

A.1 Tools and Languages

The Object Management Group's (OMG) [15] Interface Definition Language (IDL) is used as a means to programmatically define the interfaces. IDL files are either generated manually from class diagrams or by using a UML tool. In the case IDLs are manually written and/or being corrected manually, correctness has been verified using a CORBA2 (orbos/97-02-25) compliant IDL compiler, e.g. [13].

A.2 Strings and Collections

In IDL, the data type *String* is typedefed (see Note below) from the CORBA primitive *string*. This CORBA primitive is made up of a length and a variable array of byte.

NOTE: A *typedef* is a type definition declaration in IDL.

In OMG IDL, this maps to a sequence of the data type. A CORBA sequence is implicitly made of a length and a variable array of elements of the same type.

EXAMPLE 1: typedef sequence<TpSessionID> TpSessionIDSet;

Collection types can be implemented (for example, in C++) as a structure containing an integer for the *number* part, and an array for the *data* part.

EXAMPLE 2: The TpAddressSet data type may be defined in C++ as:

```
typedef struct {
    short       number;
    TpAddress   address [];
} TpAddressSet;
```

The array "address" is allocated dynamically with the exact number of required TpAddress elements based on "number".

A.3 Naming space across CORBA modules

The following shows the naming space used in the present document.

```
module org {
   module csapi {
    /* The fully qualified name of the following constant is
    org::csapi::P_THIS_IS_AN_OSA_GLOBAL_CONST */
    const long P_THIS_IS_AN_OSA_GLOBAL_CONST= 1999;
    // Add other OSA global constants and types here
    module fw {
    /* no scoping required to access  P_THIS_IS_AN_OSA_GLOBAL_CONST */
      const long P_FW_CONST= THIS_IS_AN_OSA_GLOBAL_CONST;
    };
    module mm {
    // scoping required to access P_FW_CONST
      const long P_M_CONST= fw::P_FW_CONST;
    };
   };
};
```

Annex B (informative): Bibliography

- ETSI TR 101 917: "Services and Protocols for Advanced Networks (SPAN); API mapping for Open Service Access".
- ETSI TS 123 127: "Universal Mobile Telecommunications System (UMTS); Virtual Home Environment / Open Service Architecture (3GPP TS 23.127 Release 4)".
- ETSI TS 122 127: "Universal Mobile Telecommunications System (UMTS); Service Requirement for the Open Services Access (OSA); Stage 1 (3GPP TS 22.127 Release 4)".
- ETSI TS 123 057: "Digital cellular telecommunications system (Phase 2+) (GSM); Universal Mobile Telecommunications System (UMTS); Mobile Execution Environment (MExE); Functional description; Stage 2 (3GPP TS 23.057)".
- ETSI TS 123 078: "Digital cellular telecommunications system (Phase 2+) (GSM); Universal Mobile Telecommunications System (UMTS); Customised Applications for Mobile network Enhanced Logic (CAMEL) Phase 3 - Stage 2 (3GPP TS 23.078)".
- ETSI TS 129 198: "Universal Mobile Telecommunications System (UMTS); Open Service Access (OSA); Application Programming Interface (API) (3GPP TS 29.198 version 4 Release 4)".

History

Document history		
V1.1.1	December 2001	Membership Approval Procedure MV 20020215: 2001-12-18 to 2002-02-15
V1.1.1	February 2002	Publication