

ETSI ES 204 915-10 V1.1.1 (2008-05)

ETSI Standard

Open Service Access (OSA); Application Programming Interface (API); Part 10: Connectivity Manager SCF (Parlay 6)



ITeH STANDARD PREVIEW
(standards.iteh.ai)

Full standard:
<https://standards.iteh.ai/catalog/standards/sist/7823a211-289e-4bbf-aa0a-a289eb67257/etsi-es-204-915-10-v1.1.1-2008-05>



Reference

DES/TISPAN-01032-10-OSA

Keywords

API, IDL, OSA, 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

Important notice

Individual copies of the present document can be downloaded from:

<http://www.etsi.org>

The present document may be made available in more than one electronic version or in print. In any case of existing or perceived difference in contents between such versions, the reference version is the Portable Document Format (PDF). In case of dispute, the reference shall be the printing on ETSI printers of the PDF version kept on a specific network drive within ETSI Secretariat.

Users of the present document should be aware that the document may be subject to revision or change of status. Information on the current status of this and other ETSI documents is available at

<http://portal.etsi.org/tb/status/status.asp>

If you find errors in the present document, please send your comment to one of the following services:

http://portal.etsi.org/chaicor/ETSI_support.asp

Copyright Notification

No part may be reproduced except as authorized by written permission.
The copyright and the foregoing restriction extend to reproduction in all media.

© European Telecommunications Standards Institute 2008.

© The Parlay Group 2008.

All rights reserved.

DECT™, **PLUGTESTS™**, **UMTS™**, **TIPHON™**, the TIPHON logo and the ETSI logo are Trade Marks of ETSI registered for the benefit of its Members.

3GPP™ is a Trade Mark of ETSI registered for the benefit of its Members and of the 3GPP Organizational Partners.

Contents

Intellectual Property Rights	6
Foreword.....	6
1 Scope	7
2 References	7
3 Definitions and abbreviations.....	7
3.1 Definitions	7
3.2 Abbreviations	8
4 Connectivity Manager SCF	9
5 Sequence Diagrams	12
5.1 Operator Selects Service Components and creates a new VPrP.....	12
5.2 Operator Browses Virtual Provisioned Pipe.....	13
5.3 Operator Browses SAPs and Sites.....	14
6 Class Diagrams.....	15
7 The Service Interface Specifications	21
7.1 Interface Specification Format	21
7.1.1 Interface Class	21
7.1.2 Method descriptions.....	21
7.1.3 Parameter descriptions.....	22
7.1.4 State Model.....	22
7.2 Base Interface.....	22
7.2.1 Interface Class IpInterface	22
7.3 Service Interfaces	22
7.3.1 Overview	22
7.4 Generic Service Interface	22
7.4.1 Interface Class IpService	22
7.4.1.1 Method setCallback().....	23
7.4.1.2 Method setCallbackWithSessionID().....	23
8 Connectivity Manager Interface Classes.....	23
8.1 Interface Class IpConnectivityManager	23
8.1.1 Method getQoSMenu()	24
8.1.2 Method getEnterpriseNetwork()	24
8.2 Interface Class IpEnterpriseNetwork.....	24
8.2.1 Method getSiteList().....	25
8.2.2 Method getVPrN().....	25
8.2.3 Method getSite()	26
8.3 Interface Class IpEnterpriseNetworkSite	26
8.3.1 Method getSAPList().....	26
8.3.2 Method getSiteID().....	27
8.3.3 Method getSiteLocation().....	27
8.3.4 Method getSiteDescription().....	27
8.3.5 Method getIPSubnet().....	28
8.3.6 Method getSAIPSubnet()	28
8.4 Interface Class IpQoSMenu	28
8.4.1 Method getTemplate()	29
8.4.2 Method getTemplateList()	29
8.5 Interface Class IpQoSTemplate.....	30
8.5.1 Method getTemplateType()	30
8.5.2 Method getDescription().....	31
8.5.3 Method setSlaID()	31
8.5.4 Method getPipeQoSInfo().....	31
8.5.5 Method setPipeQoSInfo()	32

8.5.6	Method getValidityInfo()	32
8.5.7	Method setValidityInfo()	33
8.5.8	Method setProvisionedQoSInfo()	33
8.5.9	Method getProvisionedQoSInfo()	34
8.5.10	Method getDsCodepoint()	34
8.6	Interface Class IpVPrN	35
8.6.1	Method getVPrPList()	35
8.6.2	Method getVPrP()	36
8.6.3	Method createVPrP()	36
8.6.4	Method deleteVPrP()	37
8.7	Interface Class IpVPrP	37
8.7.1	Method getVPrPID()	38
8.7.2	Method getSlaID()	38
8.7.3	Method getStatus()	39
8.7.4	Method getProvisionedQoSInfo()	39
8.7.5	Method getValidityInfo()	39
8.7.6	Method getPipeQoSInfo()	40
8.7.7	Method getDsCodepoint()	40
9	State Transition Diagrams	40
10	Data Definitions	41
10.1	Connectivity Manager Data Types	41
10.1.1	TpIPSubnet	41
10.1.2	TpIPv4AddType	42
10.1.3	TpIPVersion	42
10.1.4	TpVprpStatus	42
10.1.5	TpDsCodepoint	42
10.1.6	TpProvisionedQoSInfo	42
10.1.7	TpDelayDescriptor	43
10.1.8	TpLossDescriptor	43
10.1.9	TpJitterDescriptor	43
10.1.10	TpNameDescrpTagInt	43
10.1.11	TpNameDescrpTagString	44
10.1.12	TpTagValue	44
10.1.13	TpNameDescrpTagExcessLoadAction	44
10.1.14	TpAction	44
10.1.15	TpPipeQoSInfo	45
10.1.16	TpNameDescrpTagDir	45
10.1.17	TpTrafficDirection	45
10.1.18	TpEndpoint	45
10.1.19	TpSiteOrSap	45
10.1.20	TpLoadDescriptor	45
10.1.21	TpValidityInfo	46
10.1.22	TpNameDescrpTagDateTime	46
10.1.23	TpNameDescrpTagTimePeriod	46
10.1.24	TpNameDescrpTagTimeOfDay	46
10.1.25	TpNameDescrpTagDayOfWeek	47
10.1.26	TpNameDescrpTagMonth	47
11	Exception Classes	48
Annex A (normative):	OMG IDL Description of Connectivity Manager SCF	50
Annex B (informative):	W3C WSDL Description of Connectivity Manager SCF	51
Annex C (informative):	Java™ API Description of the Connectivity Management SCF	52
Annex D (informative):	Record of changes	53
D.1	Interfaces	53
D.1.1	New	53
D.1.2	Deprecated	53
D.1.3	Removed	53

D.2	Methods.....	53
D.2.1	New.....	53
D.2.2	Deprecated.....	53
D.2.3	Modified.....	54
D.2.4	Removed.....	54
D.3	Data Definitions.....	54
D.3.1	New.....	54
D.3.2	Modified.....	54
D.3.3	Removed.....	54
D.4	Service Properties.....	54
D.4.1	New.....	54
D.4.2	Deprecated.....	55
D.4.3	Modified.....	55
D.4.4	Removed.....	55
D.5	Exceptions.....	55
D.5.1	New.....	55
D.5.2	Modified.....	55
D.5.3	Removed.....	55
D.6	Others.....	55
	History.....	56

iTeh STANDARD PREVIEW
 (standards.iteh.ai)
 Full standard:
<https://standards.iteh.ai/catalog/standards/sist/7823a211-289e-4bbf-aa0a-a289eb67257/etsi-es-204-915-10-v1.1.1-2008-05>

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 Telecommunications and Internet converged Services and Protocols for Advanced Networking (TISPAN).

The present document is part 10 of a multi-part deliverable covering Open Service Access (OSA); Application Programming Interface (API), as identified below. The API specification (ES 204 915) is structured in the following parts:

- Part 1: "Overview";
- Part 2: "Common Data Definitions";
- Part 3: "Framework";
- Part 4: "Call Control";
- 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";
- Part 13: "Policy Management SCF";
- Part 14: "Presence and Availability Management SCF";
- Part 15: "Multi-Media Messaging SCF";
- Part 16: "Service Broker SCF".

The present document has been defined jointly between ETSI, The Parlay Group (<http://www.parlay.org>) and the 3GPP, in co-operation with a number of JAIN™ Community (<http://www.java.sun.com/products/jain>) member companies.

The present document forms part of the Parlay 6.0 set of specifications.

1 Scope

The present document is part 10 of the Stage 3 specification for an Application Programming Interface (API) for Open Service Access (OSA).

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

The present document specifies the Connectivity Manager Service Capability Feature (SCF) aspects of the interface. All aspects of the Connectivity Manager SCF are defined here, these being:

- Sequence Diagrams.
- Class Diagrams.
- Interface specification plus detailed method descriptions.
- State Transition diagrams.
- Data Definitions.
- IDL Description of the interfaces.

The process by which this task is accomplished is through the use of object modelling techniques described by the Unified Modelling Language (UML).

2 References

The references listed in clause 2 of ES 204 915-1 contain provisions which, through reference in this text, constitute provisions of the present document.

ETSI ES 204 915-1: "Open Service Access (OSA); Application Programming Interface (API); Part 1: Overview (Parlay 6)".

3 Definitions and abbreviations

3.1 Definitions

For the purposes of the present document, the terms and definitions given in ES 204 915-1 and the following apply:

best effort traffic: traffic that is not carried by a VPrN established for the enterprise network by the provider

differentiated services: effort in the Internet Engineering Task Force (IETF) to provide quality of service in networks employing small well defined building blocks from which variety of service may be built

DS Codepoint: marking associated with a specific VPrP

enterprise operator: administrator of the enterprise network/user of APIs

NOTE: Also referred to as operator.

operator: enterprise operator

provider network: provides a VPN and VPrP service to the enterprise network, and offers APIs for connectivity manager to the enterprise operator

NOTE: Also referred to as network service provider.

provider: entity that offers the VPN and VPrP services, and implements the APIs in the provider network

Quality of Service (QoS): collection of service levels delivered by a provider network to an enterprise network

NOTE: QoS can be characterised by various performance attributes such as: packet loss, packet delay; traffic policing measures such as maximum bandwidth and burst rate for traffic entering the providers network.

Service Access Point (SAP): enterprise network is connected to the provider network through the enterprise network service access points

NOTE: A SAP is typically the egress router from the enterprise network that connects to the provider network.

TOS bits: value held in the TOS field

NOTE: IETF defined the use the TOS field in the IPv4 packet header as a signalling mechanism aimed at providing definitions of aggregation of flows, where each aggregate is supported by the same level of QoS.

Virtual Leased Line (VLL): network that uses a provider network infrastructure to connect two geographically separated sites of an enterprise

Virtual Private Network (VPN): network that uses a provider network infrastructure to connect geographically separated sites of an enterprise

NOTE: Such a network looks like a private network to the enterprise as the sites are connected using tunnelling and security technologies. With no QoS measures, VPN passes all packets among the sites with a best effort approach.

Virtual Provisioned Network (VPrN): collection of VPrP delivered as a service to a single enterprise network

Virtual Provisioned Pipe (VPrP): service provided by the provider network to the enterprise network, which is a type of virtual leased line (VLL) provisioned with QoS levels

NOTE: VPrP carries enterprise network traffic whose packets are marked with the specific DS Codepoint that is associated with this VPrP. The enterprise operator using APIs can create on-line a VPrP.

3.2 Abbreviations

For the purposes of the present document, the abbreviations given in ES 204 915-1 and the following apply:

CIM	Common Information Model
CM	Connectivity Manager
DiffServ	Differentiated Services
DMTF	Distributed Management Task Force
DS	Differentiated Services
QoS	Quality of Service
SAP	Service Access Point
SLA	Service Level Agreement terms
TOS	Type Of Service
VLL	Virtual Leased Line
VPN	Virtual Private Network
VPrN	Virtual Provisioned Network
VPrP	Virtual Provisioned Pipe

4 Connectivity Manager SCF

Connectivity Manager includes the APIs between the enterprise operator and the provider network for the two parties to establish QoS parameters for enterprise network packets travelling through the provider network.

The Connectivity Manager service provides tools for the enterprise operator to set up a Provisioned QoS service in the provider network. The QoS measures used in the enterprise network are outside the scope of the service. The API does not require any specific QoS method to be used in the enterprise network, nor in the provider network. However, in order for Provisioned QoS service to be applied to packets arriving from the enterprise network into the provider network, the packets have to be marked using DS Codepoint marking. Once the packets are so marked, they can enjoy the QoS service provisioned in the provider network.

APIs provide the enterprise network operator on-line access to provision quality of service measures that control the enterprise's own traffic passing through the provider network. Using APIs the operator can create Virtual Provisioned Pipes (VPrPs) in the provider network to carry the enterprise traffic and support it with pre-specified quality of service attributes. A VPrP can be thought of as a Virtual Leased Line (VLL) provisioned to deliver pre-specified QoS. The provider may offer to the enterprise operator a set of templates that are used by the operator to specify a VPrP. For instance, the provider may offer templates for video conferencing, audio conferencing, Gold Service, Silver Service, etc. Using these templates the operator can select and provision a VPrP that specifies the quality of service attributes for this VPrP.

Elements that can be specified for a VPrP include attributes such as packet delay and packet loss. Characteristics of traffic that enters the VPrP at its access point to the provider network can be also specified with attributes such as maximum rate and burst rate.

The following is an example of a possible scenario:

- The provider prepares a template with *operator-specified* attributes, *provider-specified* attributes, and *unspecified* attributes, one for each QoS level.
- The provider generates for the enterprise network a list of all the current sites and their access points to the provider network.
- Enterprise operator logs into connectivity manager after being authenticated and authorised by the Framework service.
- Operator gets the list of the sites and service access points of the enterprise Virtual Private Network (VPN) already provided to the enterprise by the provider.
- Enterprise operator retrieves the set of templates available to the enterprise (as supported by the SLA), selects one, and requests a template for constructing a new VPrP based upon the selected QoS.
- Enterprise operator completes the VPrP template: i.e. selects a value for delay, loss, jitter and excess traffic treatment action, enters the SLA ID against which the template could be validated, selects endpoints, load parameters and traffic flow direction, and selects the time requirements desired. The enterprise operator can choose or modify those attributes that are *operator-specified* attributes in the template. *Provider-specified* attributes cannot be modified and are inherently part of the service.
- Enterprise operator submits the completed VPrP template for validation by the CM service. Operator creates a new VPrP with *pending-status* that holds these selections.
- The provider responds after validating the requests, which may be an approval or a denial (e.g. the requested service is not available at this access point, or at the specified time).
- If the provider approves service, the operator may send packets marked with the templates DiffServ Codepoint, which identifies together with the endpoints the VPrP that carries these packets.

Some additional clarification points:

- A VPrN is associated with a single network provider.
- A VPrP defines QoS parameters for traffic flowing through this provider network, between two specified enterprise endpoints, optionally during specified date/time period(s).
- The enterprise operator may be (provider's choice) constrained to selecting QoS parameter values from a pre-defined set of values, and selecting endpoints from a predefined set of enterprise sites; where these sets were negotiated off-line between the enterprise and the network provider and possibly documented in a Service Level Agreement (SLA).
- The CM service validates each VPrP request submitted by the enterprise operator. The validation process is not specified here. Validation against the SLA is an example of such possible validation.
- If the CM service accepts the VPrP request, it adds it to the VPrN. The DiffServ Codepoint provided by the enterprise operator is then used by the enterprise for marking all packets belonging to any traffic flow associated with the VPrP.

The following is a summary of interfaces and methods supported by connectivity manager. The syntax method (interface) is used for this description.

There are Passive CM interface functions (CM1, CM2, CM4) that are used to retrieve information (read) relative to VPN, VPrN, and QoS templates provided by the service provider, and active (CM3) functions (read/write) used to provision new services.

CM1: Retrieve information on a Virtual Private Network, its sites and their service access points:

```
getEnterpriseNetwork(IpConnectivityManager)
getSiteList(IpEnterpriseNetwork)
getSite(IpEnterpriseNetwork)
getSAPList(IpEnterpriseNetworkSite)
getSiteID(IpEnterpriseNetworkSite)
getSiteLocation(IpEnterpriseNetworkSite)
getSiteDescription(IpEnterpriseNetworkSite)
getSAPIPSubnet(IpEnterpriseNetworkSite)
getSiteIPSubnet(IpEnterpriseNetworkSite)
```

CM2: Retrieve QoS services offered by provider, stored in QoS templates:

```
getQoSMenu(IpConnectivityManager)
getTemplateList(IpQoSMenu)
getTemplate(IpQoSMenu)
getTemplateType(IpQoSTemplate)
getDescription(IpQoSTemplate)
getPipeQoSInfo(IpQoSTemplate)
getValidityInfo(IpQoSTemplate)
getProvisionedQoSInfo(IpQoSTemplate)
getDsCodepoint(IpQoSTemplate)
```

CM3: Set up a new Virtual Provisioned Pipe:

```
createVPrP(IpVPrN)
deleteVPrP(IpVPrN)
setSlaID(IpQoSTemplate)
setPipeQoSInfo(IpQoSTemplate)
setValidityInfo(IpQoSTemplate)
setProvisionedQoSInfo(IpQoSTemplate)
```

CM4: Retrieve information on a Virtual Provisioned Network and its Virtual Provisioned Pipes:

```

getVPrN(IpEterpriseNetwork)
getVPrPList(IpVPrN)
getVPrP(IpVPrN)
getVPrPID(IpVPrP)
getSlaID(IpVPrP)
getStatus(IpVPrP)
getProvisionedQoSInfo(IpVPrP)
getPipeQoSInfo(IpVPrP)
getDsCodepoint(IpVPrP)

```

Two typical scenarios:

- 1) To set up a new VPrP:
 - The enterprise operator retrieves information regarding enterprise's existing VPN (sites and SAPs), using as needed the methods listed in CM1 above.
 - Enterprise operator retrieves information on provider's offered QoS services, using methods listed in CM2 above as needed.
 - Enterprise operator submits a request to set up a new VPrP, using methods listed in CM3 above as needed.
 - Enterprise operator checks the status of the request using the methods listed in CM4 above as needed.
 - If the request was approved by the network provider, the VPrP is put in an active mode, and packets that are marked in the enterprise network with appropriate marking in their packet header will travel the provider's network through the new Virtual Provisioned Pipe that supports the requested QoS levels.
- 2) Retrieve information on current enterprise network services delivered to the enterprise by the provider network:
 - Enterprise operator retrieves information on current QoS services delivered to the enterprise network using CM4 methods listed above as needed.
 - Enterprise operator checks if the provider offers new QoS services using CM2 listed above as needed.

The following clauses describe each aspect of the Connectivity Manager Service Capability Feature (SCF).

The order is as follows:

- The Sequence diagrams give the reader a practical idea of how each of the SCF is implemented.
- The Class relationships clause shows 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 transition between states in the SCF. The states and transitions are well-defined; either methods specified in the Interface specification or events occurring in the underlying networks cause state transitions.
- The Data Definitions clause shows 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 ES 204 915-2.

An implementation of this API which supports or implements a method described in the present document, shall support or implement the functionality described for that method, for at least one valid set of values for the parameters of that method. Where a method is not supported by an implementation of a Service interface, the exception P_METHOD_NOT_SUPPORTED shall be returned to any call of that method.

5 Sequence Diagrams

5.1 Operator Selects Service Components and creates a new VPrP

The following sequence diagram shows how an enterprise operator client collects the information required to select a service, and then selects the service parameters, and finally submits it to the connectivity manager.



5.2 Operator Browses Virtual Provisioned Pipe

The following shows an enterprise operator client browses and collect information pertinent to an existing virtual provisioned pipe, including all the QoS parameters that have been set for this pipe.

