ETSI TS 129 198-2 V4.1.0 (2001-06)

Technical Specification

Universal Mobile Telecommunications System (UMTS); Open Service Access (OSA); Application Programming Interface (API); Part 2: Common data (3GPP TS 29.198-2 version 4.1.0 Release 4)

Reference

RTS/TSGN-0529198-2Uv4R2

Keywords

UMTS

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://www.etsi.org/tb/status/

If you find errors in the present document, send your comment to:
editor@etsi.fr

Copyright Notification

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://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 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 Technical Specification (TS) has been produced by the ETSI 3rd Generation Partnership Project (3GPP).

The present document may refer to technical specifications or reports using their 3GPP identities, UMTS identities or GSM identities. These should be interpreted as being references to the corresponding ETSI deliverables.

The cross reference between GSM, UMTS, 3GPP and ETSI identities can be found under www.etsi.org/key .

Contents

Foreword

This Technical Specification has been produced by the 3rd Generation Partnership Project (3GPP).

The contents of the present document are subject to continuing work within the TSG and may change following formal TSG approval. Should the TSG modify the contents of the present document, it will be re-released by the TSG with an identifying change of release date and an increase in version number as follows:

Version x.y.z

where:

x the first digit:

1 presented to TSG for information;

2 presented to TSG for approval;

3 or greater indicates TSG approved document under change control.

y the second digit is incremented for all changes of substance, i.e. technical enhancements, corrections, updates, etc.

z the third digit is incremented when editorial only changes have been incorporated in the document.

Introduction

The present document is part 2 of a multi-part TS covering the 3rd Generation Partnership Project: Technical Specification Group Core Network; Open Service Access (OSA); Application Programming Interface (API), as identified below. The **API specification** (3GPP TS 29.198) 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	(not part of 3GPP Release 4)
Part 10:	Connectivity Manager SCF	(not part of 3GPP Release 4)
Part 11:	Account Management SCF	
Part 12:	Charging SCF	

The **Mapping specification of the OSA APIs and network protocols** (3GPP TR 29.998) is also structured as above. 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.

OSA API specifications 29.198-family		OSA API Mapping - 29.998-family	
29.198-1	**Part 1: Overview**	**29.998-1**	**Part 1: Overview**
29.198-2	**Part 2: Common Data Definitions**	29.998-2	Not Applicable
29.198-3	**Part 3: Framework**	29.998-3	Not Applicable
29.198-4	**Part 4: Call Control SCF**	**29.998-4-1**	**Subpart 1: Generic Call Control – CAP mapping**
		29.998-4-2	
29.198-5	**Part 5: User Interaction SCF**	**29.998-5-1**	**Subpart 1: User Interaction – CAP mapping**
		29.998-5-2	
		29.998-5-3	
		29.998-5-4	**Subpart 4: User Interaction – SMS mapping**
29.198-6	**Part 6: Mobility SCF**	**29.998-6**	**User Status and User Location – MAP mapping**
29.198-7	**Part 7: Terminal Capabilities SCF**	29.998-7	Not Applicable
29.198-8	**Part 8: Data Session Control SCF**	**29.998-8**	**Data Session Control – CAP mapping**
29.198-9	Part 9: Generic Messaging SCF	29.998-9	Not Applicable
29.198-10	Part 10: Connectivity Manager SCF	29.998-10	Not Applicable
29.198-11	**Part 11: Account Management SCF**	29.998-11	Not Applicable
29.198-12	**Part 12: Charging SCF**	29.998-12	Not Applicable

1 Scope

The present document is Part 2 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 concepts and the functional architecture for the OSA are contained in 3GPP TS 23.127 [3]. The requirements for OSA are contained in 3GPP TS 22.127 [2].

The present document specifies the Common Data definitions of the OSA. The Common Data definitions contain data-types that are common across the rest of the OSA API. All aspects of the Common Data are defined here, these being:

- Data definitions
- IDL Description of the interfaces

This specification has been defined jointly between 3GPP TSG CN WG5, ETSI SPAN 12 and the Parlay Consortium, in co-operation with the JAIN consortium.

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. In the case of a reference to a 3GPP document (including a GSM document), a non-specific reference implicitly refers to the latest version of that document *in the same Release as the present document.*

[1] 3GPP TS 29.198-1: "Open Service Access; Application Programming Interface; Part 1: Overview".

[2] 3GPP TS 22.127: "Stage 1 Service Requirement for the Open Service Access (OSA) (Release 4)".

[3] 3GPP TS 23.127: "Virtual Home Environment (Release 4)".

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

[5] 3GPP TS 22.024: "Description of Charge Advice Information (CAI)".

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

[7] IETF RFC 0822: "Standard for the format of ARPA Internet text messages".

[8] ISO 4217 (1995): "Codes for the representation of currencies and funds ".

[9] ITU-T Recommendation E.164: "The international public telecommunication numbering plan".

[10] ITU-T Recommendation X.400: "Message handling system and service overview".

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

3 Definitions and abbreviations

3.1 Definitions

For the purposes of the present document, the terms and definitions given in TS 29.198-1 [1] apply.

3.2 Abbreviations

For the purposes of the present document, the abbreviations given in TS 29.198-1 [1] apply.

4 Common Data definitions

The following clauses describe each aspect of the Common Data definitions.

The order is as follows:

- The Data definitions clause shows a detailed expansion of each of the data types associated with the methods within the classes.

NOTE: Some data types are used in other methods and classes and are therefore defined within the Common Data types part of the present document.

5 Common System Data definitions

These data definitions are assumed to be provided by the client operating system.

5.1 Standard Data types

The APIs assume that the following Data types can be supported.

5.1.1 TpBoolean

Defines a Boolean data type.

5.1.2 TpInt32

Defines a signed 32-bit integer.

5.1.3 TpInt32Ref

Defines a 5.2.4 Reference to a TpInt32.

5.1.4 TpFloat

Defines a single precision real number

5.1.5 TpFloatRef

Defines a Reference to a TpFloat

5.1.6 TpLongstring

Defines a Byte string, comprising length and data. The length shall be at least a 32-bit integer.

5.1.7 TpLongstringRef

Defines a 5.2.4 Reference to a TpLongstring.

5.1.8 TpOctet

Defines an 8 bit quantity that is not translated during transmission.

5.1.9 TpOctetRef

Defines a Reference to a TpOctet.

5.1.10 TpOctetSet

Defines a Numbered Set of Data elements of TpOctet.

5.1.11 TpString

Defines a Byte string, comprising length and data. The length shall be at least a 16-bit integer.

5.1.12 TpStringRef

Defines a 5.2.4 Reference to a TpString.

5.1.13 TpAssignmentID

This data type is identical to a TpInt32. It specifies a number which identifies an individual event notification enabled by the application or service.

5.1.14 TpAssignmentIDRef

Defines a Reference to type TpAssignmentID.

5.1.15 TpSessionID

Defines a network unique session ID. The API uses this ID to identify sessions, e.g. call or call leg sessions, within an object implementing an interface capable of handling multiple sessions. For the different services, the sessionIDs are unique only in the context of a service manager instantiation (e.g., within the context of one Generic Call Control manager). As such if an application creates two instances of the same service manager it shall use different instantiations of the callback objects which implement the callback interfaces.

The session ID is identical to a TpInt32 type.

5.1.16 TpSessionIDRef

Defines a Reference to a TpSessionID.

5.1.17 TpSessionIDSet

Defines a Numbered_Set_of_Data_Elements of TpSessionID.

5.2 Other Data sorts

The APIs assumes that the following data syntaxes can be supported

5.2.1 Sequence of Data Elements

This describes a sequence of data types. This may be defined as a structure (for example, in C++) or simply a sequence of data elements within a structure.

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

```
typedef struct {
  TpAddressPlan       Plan;
  TpString         AddrString;
  TpString         Name;
  TpAddressPresentation.....Presentation;
....TpAddressScreening........Screening;
....TpString..................SubAddressString;
} TpAddress;
```

5.2.2 Tagged Choice of Data Elements

This describes a data type which actually evaluates to one of a choice of a number of data elements. This data element contains two parts: a tag data type (the *tag* part) which is used to identify the chosen data type, and the chosen data type itself (the *union* part). This form of data type is also referred to as a tagged union.

This data type can be implemented (for example, in C++) as a structure containing an integer for the *tag* part, and a union for the *union* part.

This data type is implementation specific. Please refer to the appropriate IDL documents (and the resulting language mappings) to see how this data type is implemented.

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

```
typedef struct {
  TpCallErrorType Tag;
  union {
    TpCallErrorInfoUndefined     Undefined;
    TpCallErrorInfoRoutingAborted  RoutingAborted;
    TpCallErrorInfoCallAbandoned   CallAbandoned;
    TpCallErrorInfoInvalidAddress  InvalidAddress;
    TpCallErrorInfoInvalidState   InvalidState;
    TpCallErrorInfoInvalidCriteria  InvalidCriteria;
  } callErrorInfo;
} TpCallError;
```

5.2.3 Numbered Set of Data Elements

This describes a data type which comprises an integer which indicates the total number of data elements in the set (the *number* part), and an **unordered** set of data elements (the *data* part). *Set* data types do not contain duplicate data elements.

EXAMPLE: The TpAddressSet data type may be defined in MIDL as:

```
typedef struct TpAddressSet
{
TpInt32 Number; [size_is(Number)] TpAddress Set[];
}
TpAddressSet;
```

5.2.4 Reference

This describes a reference (or pointer) to a data type. This is primarily used to describe 'out' method parameters.

This data type may be implemented (for example, in C++) as a pointer. However, in some languages it may not be necessary for 'out' parameters to be implemented as pointers.

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

```
typedef TpAddress *   TpAddressRef
```

5.3 Interface Related Data definitions

5.3.1 IpInterface

Defines the address of a generic interface instance.

5.3.2 IpInterfaceRef

Defines a 5.2.4 Reference to type IpInterface.

5.3.4 IpInterfaceRefRef

Defines a 5.2.4 Reference to type IpInterfaceRef.

5.4 Method Result Data definitions

5.4.1 TpResult

Defines the 5.2.1 Sequence of Data Elements that specify the result of a method call. All methods in the APIs return a result of type TpResult.

Sequence Element Name	Sequence Element Type
ResultType	TpResultType

ResultFacility	TpResultFacility
ResultInfo	TpResultInfo

5.4.2 TpResultType

Defines whether the method was successful or not.

Name	Value	Description
P_RESULT_FAILURE	0	Method failed
P_RESULT_SUCCESS	1	Method was successful

5.4.3 TpResultFacility

Defines the facility code of a result. In phase 2 of the APIs, only P_RESULT_FACILITY_UNDEFINED shall be used.

Name	Value	Description
P_RESULT_FACILITY_UNDEFINED	0	Undefined

5.4.4 TpResultInfo

Defines further information relating to the result of the method, such as error codes.

Name	Value	Description
P_RESULT_INFO_UNDEFINED	0000h	No further information present
P_INVALID_DOMAIN_ID	0001h	Invalid client ID
P_INVALID_AUTH_CAPABILITY	0002h	Invalid authentication capability
P_INVALID_AGREEMENT_TEXT	0003h	Invalid agreement text
P_INVALID_SIGNING_ALGORITHM	0004h	Invalid signing algorithm
P_INVALID_INTERFACE_NAME	0005h	Invalid interface name
P_INVALID_SERVICE_ID	0006h	Invalid service ID
P_INVALID_EVENT_TYPE	0007h	Invalid event type
P_SERVICE_NOT_ENABLED	0008h	The service ID does not correspond to a service that has been enabled
P_INVALID_ASSIGNMENT_ID	0009h	The assignment ID is invalid
P_INVALID_PARAMETER	000Ah	The method has been called with an invalid parameter
P_INVALID_PARAMETER_VALUE	000Bh	A method parameter has an invalid value
P_PARAMETER_MISSING	000Ch	A mandatory parameter has not been specified in the method call
P_RESOURCES_UNAVAILABLE	000Dh	The required resources in the network are not available
P_TASK_REFUSED	000Eh	The requested method has been refused
P_TASK_CANCELLED	000Fh	The requested method has been cancelled
P_INVALID_DATE_TIME_FORMAT	0010h	Invalid date and time format provided
P_NO_CALLBACK_ADDRESS_SET	0011h	The requested method is refused because no callback address is set
P_INVALID_SIGNATURE	0012h	Invalid digital signature
P_INVALID_SERVICE_TOKEN	0013h	The service token has not been issued, or it has expired.
P_ACCESS_DENIED	0014h	The client is not currently authenticated with the framework
P_INVALID_PROPERTY	0015h	The framework does not recognise the property supplied by the client
P_METHOD_NOT_SUPPORTED	0016h	The method is not allowed or supported within the context of the current service agreement.
P_NO_ACCEPTABLE_AUTH_CAPABILITY	0017h	An authentication mechanism, which is acceptable to the framework, is not supported by the client
P_INVALID_INTERFACE_TYPE	0018h	The interface reference supplied by the client is the wrong type.

Name	Value	Description
P_INVALID_ACCESS_TYPE	0019h	The framework does not support the type of access interface requested by the client.
P_SERVICE_ACCESS_DENIED	001Ah	The client application is not allowed to access this service.
P_USER_NOT_SUBSCRIBED	0030h	An application is unauthorised to access information and request services with regards to users that are not subscribed to the application.
P_APPLICATION_NOT_ACTIVATED	0031h	An application is unauthorised to access information and request services with regards to users that have deactivated that particular application.
P_USER_PRIVACY	0032h	An application is unauthorised to access information and request services with regards to users that have set their privacy flag regarding that particular service.

Name	Value	Description
P_GCCS_SERVICE_INFORMATION_MISSING	0100h	Information relating to the Call Control service could not be found
P_GCCS_SERVICE_FAULT_ENCOUNTERED	0101h	Fault detected in the Call Control service
P_GCCS_UNEXPECTED_SEQUENCE	0102h	Unexpected sequence of methods, i.e., the sequence does not match the specified state diagrams for the call or the call leg.
P_GCCS_INVALID_ADDDRESS	0103h	Invalid address specified
P_GCCS_INVALID_CRITERIA	0104h	Invalid criteria specified
P_GCCS_INVALID_NETWORK_STATE	0105h	Although the sequence of method calls is allowed by the gateway, the underlying protocol can not support it. E.g., in some protocols some methods are only allowed by the protocol, when the call processing is suspended, e.g., after reporting an event that was monitored in interrupt mode.

5.5 Date- and Time-related Data definitions

5.5.1 TpDate

This data type is identical to a TpString. It specifies the data in accordance with International Standard ISO 8601 [4]. This is defined as the string of characters in the following format:

YYYY-MM-DD

where the date is specified as:

YYYY	four digits year
MM	two digits month
DD	two digits day

The date elements are separated by a hyphen character (-).

EXAMPLE: The 4 December 1998, is encoded as the string:

1998-12-04

5.5.2 TpTime

This data type is identical to a TpString. It specifies the time in accordance with International Standard ISO 8601 [4]. This is defined as the string of characters in the following format:

HH:MM:SS.mmm

or

HH:MM:SS.mmmZ

where the time is specified as:

HH	two digits hours (24h notation)
MM	two digits minutes
SS	two digits seconds
mmm	three digits fractions of a second (i.e. milliseconds)

The time elements are separated by a colon character (:). The date and time are separated by a space. Optionally, a capital letter Z may be appended to the time field to indicate Universal Time Co-ordinated (UTC). Otherwise, local time is assumed.

EXAMPLE: 10:30 and 15 seconds is encoded as the string:

`10:30:15.000`

for local time, or in UTC it would be: `10:30:15.000Z`

5.5.3 TpDateAndTime

This data type is identical to a TpString. It specifies the data and time in accordance with International Standard ISO 8601 [4]. This is defined as the string of characters in the following format:

`YYYY-MM-DD HH:MM:SS.mmm`

or

`YYYY-MM-DD HH:MM:SS.mmmZ`

where the date is specified as:

YYYY	four digits year
MM	two digits month
DD	two digits day

The date elements are separated by a hyphen character (-).

The time is specified as:

HH	two digits hours (24h notation)
MM	two digits minutes
SS	two digits seconds
mmm	three digits fractions of a second (i.e. milliseconds)

The time elements are separated by a colon character (:).The date and time are separated by a space. Optionally, a capital letter Z may be appended to the time field to indicate Universal Time Co-ordinated (UTC). Otherwise, local time is assumed.

EXAMPLE: The 4 December 1998, at 10:30 and 15 seconds is encoded as the string:

`1998-12-04 10:30:15.000`

for local time, or in UTC it would be:

`1998-12-04 10:30:15.000Z`

5.5.4 TpDateAndTimeRef

Defines a 5.2.4 Reference to type TpDateAndTime.

5.5.5 TpDuration

This data type is a TpInt32 representing a time interval in milliseconds. A value of "-1" defines infinite duration and a value of "-2" represents a default duration.

5.5.6 TpTimeInterval

Defines the Sequence of Data Elements that specify a time interval.

Sequence Element Name	Sequence Element Type
StartTime	TpDateAndTime
StopTime	TpDateAndTime

5.6 Address-related Data definitions

5.6.1 TpAddress

Defines the 5.2.1 Sequence of Data Elements that specify an address.

Sequence Element Name	Sequence Element Type
Plan	TpAddressPlan
AddrString	TpString
Name	TpString
Presentation	TpAddressPresentation
Screening	TpAddressScreening
SubAddressString	TpString

The AddrString defines the actual address information and the structure of the string depends on the Plan.
The following table gives an overview of the format of the AddrString for the different address plans.

Address Plan	AddrString Format Description	Example
P_ADDRESS_PLAN_NOT_PRESENT	Not applicable	
P_ADDRESS_PLAN_UNDEFINED	Not applicable	
P_ADDRESS_PLAN_IP	For Ipv4 the dotted quad notation is used. Also for IPv6 the dotted notation is used. The address can optionally be followed by a port number separated by a colon.	"127.0.0.1:42"
P_ADDRESS_PLAN_MULTICAST	An Ipv4 class D address or Ipv6 equivalent in dotted notation.	"224.0.0.0"
P_ADDRESS_PLAN_UNICAST	A non multicast or broadcast IP address in dotted notation.	"127.0.0.1"
P_ADDRESS_PLAN_E164	An international number without the international access code, including the country code and excluding the leading zero of the area code.	"31161249111"
P_ADDRESS_PLAN_AESA	The ATM End System Address in binary format (40 bytes)	01234567890ABCDEF01234567890ABCDEF01234567
P_ADDRESS_PLAN_URL	A uniform resource locator as defined in IETF RFC 1738 [6]	"http://www.parlay.org"
P_ADDRESS_PLAN_NSAP	The binary representation of the Network Service Access Point	490001AA000400010420
P_ADDRESS_PLAN_SMTP	An e-mail address as specified in IETF RFC822 [7]	"webmaster@parlay.org"
P_ADDRESS_PLAN_MSMAIL	Identical to P_ADDRESS_PLAN_SMTP	"john.doe@hitech.com"
P_ADDRESS_PLAN_X400	The X400 address structured as a set of attribute value pairs separated by semicolons.	"C=nl;ADMD=;PRMD=uninet;O=parlay;S=Doe;I=S;G=John'

5.6.2 TpAddressRef

Defines a 5.2.4 Reference to type TpAddress.

5.6.3 TpAddressSet

Defines a Numbered_Set_of_Data_Elements of TpAddress.

5.6.4 TpAddressSetRef

Defines a 5.2.4 Reference to type TpAddressSet.

5.6.5 TpAddressPresentation

Defines whether an address can be presented to an end user.

Name	Value	Description
P_ADDRESS_PRESENTATION_UNDEFINED	0	Undefined
P_ADDRESS_PRESENTATION_ALLOWED	1	Presentation Allowed
P_ADDRESS_PRESENTATION_RESTRICTED	2	Presentation Restricted
P_ADDRESS_PRESENTATION_ADDRESS_NOT_AVAILABLE	3	Address not available for presentation

5.6.6 TpAddressScreening

Defines whether an address can be presented to an end user.

Name	Value	Description
P_ADDRESS_SCREENING_UNDEFINED	0	Undefined
P_ADDRESS_SCREENING_USER_VERIFIED_PASSED	1	user provided address verified and passed
P_ADDRESS_SCREENING_USER_NOT_VERIFIED	2	user provided address not verified
P_ADDRESS_SCREENING_USER_VERIFIED_FAILED	3	user provided address verified and failed
P_ADDRESS_SCREENING_NETWORK	4	Network provided address (see Note)
NOTE: Even though the application may provide the address to the gateway, from the end-user point of view it is still regarded as a network provided address.		

5.6.7 TpAddressPlan

Defines the address plan (or numbering plan) used. It is also used to indicate whether an address is actually defined in a TpAddress data element.

Name	Value	Description
P_ADDRESS_PLAN_NOT_PRESENT	-1	No Address Present
P_ADDRESS_PLAN_UNDEFINED	0	Undefined
P_ADDRESS_PLAN_IP	1	IP
P_ADDRESS_PLAN_MULTICAST	2	Multicast
P_ADDRESS_PLAN_UNICAST	3	Unicast
P_ADDRESS_PLAN_E164	4	E.164
P_ADDRESS_PLAN_AESA	5	AESA
P_ADDRESS_PLAN_URL	6	URL
P_ADDRESS_PLAN_NSAP	7	NSAP
P_ADDRESS_PLAN_SMTP	8	SMTP
P_ADDRESS_PLAN_MSMAIL (see Note)	9	Microsoft Mail
P_ADDRESS_PLAN_X400	10	X.400
NOTE: This value is not used in the scope of 3GPP.		

For the case where the P_ADDRESS_PLAN_NOT_PRESENT is indicated, the rest of the information in the TpAddress is not valid.

5.6.8 TpAddressError

Defines the reasons why an address is invalid.

Name	Value	Description
P_ADDRESS_INVALID_UNDEFINED	0	Undefined error
P_ADDRESS_INVALID_MISSING	1	Mandatory address not present
P_ADDRESS_INVALID_MISSING_ELEMENT	2	Mandatory address element not present
P_ADDRESS_INVALID_OUT_OF_RANGE	3	Address is outside of the valid range
P_ADDRESS_INVALID_INCOMPLETE	4	Address is incomplete
P_ADDRESS_INVALID_CANNOT_DECODE	5	Address cannot be decoded

5.6.9 TpAddressRange

This type is identical to TpAddress with the difference that the AddrString can contain wildcards.

Two wildcards are allowed: * which matches zero or more characters and ? which matches exactly one character. The wildcards are only allowed at the end or at the beginning of the AddrString.

Some examples for E164 addresses:

- "123" matches specified number;
- "123*" matches all numbers starting with 123 (including 123 itself);
- "123??*" matches all numbers starting with 123 and at least 5 digits long;
- "123???" matches all numbers starting with 123 and exactly 6 digits long;

For e-mail style addresses, the wildcards are allowed at the beginning of the AddrString:

- "*@parlay.org" matches all email addresses in the parlay.org domain.

The following address ranges are illegal:

- 1?3
- 1*3
- ?123*

Legal occurrences of the '*' and '?' characters in AddrString should be escaped by a '\' character. To specify a '\' character '\\' shall be used.

5.6.10 TpURL

This data type is identical to a TpString and contains a URL address. The usage of this type is distinct from TpAddress, which can also hold a URL. The latter contains a user address which can be specified in many ways: IP, e-mail, URL etc. On the other hand, the TpURL type does not hold the address of a user and always represents a URL. This type is used in user interaction and defines the URL of the test or stream to be sent to an end-user. It is therefore inappropriate to use a general address here.

5.7 Price-related Data definitions

5.7.1 TpPrice

This data type is identical to a TpString. It specifies price information. This is defined as a string of characters (digits) in the following format:

DDDDDD.DD

5.7.2 TpAoCInfo

Defines the Sequence of Data Elements that specify the Advice Of Charge information to be sent to the terminal.

Sequence Element Name	Sequence Element Type	Description
ChargeOrder	TpAoCOrder	Charge order
Currency	TpString	Currency unit according to ISO-4217:1995 [8]

5.7.3 TpAoCOrder

Defines the Tagged Choice of Data Elements that specify the charge plan for the call.

	Tag Element Type	
	TpAoCOrderCategory	

Tag Element Value	Choice Element Type	Choice Element Name
P_CHARGE_ADVICE_INFO	TpChargeAdviceInfo	ChargeAdviceInfo
P_CHARGE_PER_TIME	TpChargePerTime	ChargePerTime
P_CHARGE_NETWORK	TpString	NetworkCharge

5.7.4 TpCallAoCOrderCategory

Name	Value	Description
P_CHARGE_ADVICE_INFO	0	Set of GSM Charge Advice Information elements according to 3GPP TS 22.024 [5]
P_CHARGE_PER_TIME	1	Charge per time
P_CHARGE_NETWORK	2	Operator specific charge plan specification, e.g. charging table name / charging table entry

5.7.5 TpChargeAdviceInfo

Defines the Sequence of Data Elements that specify the two sets of Advice of Charge parameters. The first set defines the current tariff. The second set may be used in case of a tariff switch in the network.

Sequence Element Name	Sequence Element Type	Description
CurrentCAI	TpCAIElements	Current tariff
NextCAI	TpCAIElements	Next tariff after tariff switch

5.7.6 TpCAIElements

Defines the Sequence of Data Elements that specify the Charging Advice Information elements according to 3GPP TS 22.024 [5].

Sequence Element Name	Sequence Element Type	Description
UnitsPerInterval	TpInt32	Units per interval
SecondsPerTimeInterval	TpInt32	Seconds per time interval
ScalingFactor	TpInt32	Scaling factor
UnitIncrement	TpInt32	Unit increment
UnitsPerDataInterval	TpInt32	Units per data interval
SegmentsPerDataInteral	TpInt32	Segments per data interal
InitialSecsPerTimeInterval	TpInt32	Initial secs per time interval

5.7.7 TpChargePerTime

Defines the Sequence of Data Elements that specify the time based charging information.

Sequence Element Name	Sequence Element Type	Description
InitialCharge	TpInt32	Initial charge amount (in currency units * 0.0001)
CurrentChargePerMinute	TpInt32	Current tariff (in currency units * 0.0001)
NextChargePerMinute	TpInt32	Next tariff (in currency units * 0.0001) after tariff switch Only used in setAdviceOfCharge()

5.7.8 TpLanguage

This data type is identical to a TpString, and defines the language. In case an indication for the language is not needed an empty string shall be used. In other cases valid language strings are defined in ISO 639 [11].

5.8 Exception Classes

5.8.1 TpCommonExceptions

Defines the structure of the exception class which is applicable to all methods.

Structure Element Name	Structure Element Type	Structure Element Description
exceptionType	TpInt32	Carries a constant from the list in the table below
extraInformation	TpString	Carries extra information to help identify the source of the exception, e.g. a parameter name

5.8.2 Constants associated with TpCommonExceptions

Name	Value	Description
P_RESOURCES_UNAVAILABLE	000Dh	The required resources in the network are not available
P_TASK_REFUSED	000Eh	The requested method has been refused
P_TASK_CANCELLED	000Fh	The requested method has been cancelled
P_NO_CALLBACK_ADDRESS_SET	0011h	The requested method is refused because no callback address is set
P_METHOD_NOT_SUPPORTED	0016h	The method is not allowed or supported within the context of the current service agreement.
P_INVALID_STATE	0306h	Unexpected sequence of methods, i.e., the sequence does not match the specified state diagrams.

5.8.3 Exceptions available to all methods on all interfaces

The following are the list of exception classes which are available to all interfaces of the API.

Name	Description
P_APPLICATION_NOT_ACTIVATED	An application is unauthorised to access information and request services with regards to users that have deactivated that particular application.
P_INFORMATION_NOT_AVAILABLE	An application is unauthorised to access information and request services with regards to users that have set their privacy flag regarding that particular service.
P_INVALID_ADDDRESS	Invalid address specified
P_INVALID_AMOUNT	Invalid amount specified.
P_INVALID_ASSIGNMENT_ID	The assignment ID is invalid

Name	Description
P_INVALID_CRITERIA	Invalid criteria specified
P_INVALID_CURRENCY	Invalid currency specified.
P_INVALID_EVENT_TYPE	Invalid event type
P_INVALID_INTERFACE_NAME	Invalid interface name
P_INVALID_INTERFACE_TYPE	The interface reference supplied by the client is the wrong type.
P_INVALID_NETWORK_STATE	Although the sequence of method calls is allowed by the gateway, the underlying protocol can not support it. E.g., in some protocols some methods are only allowed by the protocol, when the call processing is suspended, e.g., after reporting an event that was monitored in interrupt mode.
P_INVALID_SESSION_ID	Invalid session ID.
P_INVALID_TIME_AND_DATE_FORMAT	Invalid date and time format provided
P_SET_LENGTH_EXCEEDED	The maximum set size is exceeded in a method parameter value.
P_UNKNOWN_SUBSCRIBER	An application is unauthorised to access information and request services with regards to users that are not subscribed to the application.

Annex A (normative): OMG IDL Description of the Common Data definitions

The OMG IDL representation of this specification is contained in a text file (osa.idl contained in archive 2919802IDL.ZIP) which accompanies the present document.

Annex B (informative): Change history

Change history							
Date	**TSG #**	**TSG Doc.**	**CR**	**Rev**	**Subject/Comment**	**Old**	**New**
Mar 2001	CN_11	NP-010134	047	--	CR 29.198: for moving TS 29.198 from R99 to Rel 4 (N5-010158)	3.2.0	4.0.0
Jun 2001	CN_12	NP-010330	001	--	Corrections to OSA API Rel4 (Exception handling mechanism without ambiguity - Replace TpGeneralException and TpResultInfo with detailed exception classes which can be thrown for each method (N5-010261)	4.0.0	4.1.0
Jun 2001	CN_12	NP-010333	002	--	Introduction of TpOctet (In order to make sure that some data is sent over the “distributed wire” untouched a new data type is needed) (N5-010304)	4.0.0	4.1.0

History

Document history		
V4.0.0	March 2001	Publication
V4.1.0	June 2001	Publication