

Source: CN5 (OSA)  
Title: Rel-5 CRs 29.198-xy (OSA API) Addition of support for Java API technology realisation  
Agenda item: 8.2  
Document for: APPROVAL

---

Doc-1 <sup>st</sup> -Level	Spec	CR	R v	Pha	Subject	Cat	Ver Curr	Ver New	Doc-2 <sup>nd</sup> -Level	Work item
NP-020181	29.198-01	005	-	Rel-5	Addition of support for Java API technology realisation	B	4.3.1	5.0.0	N5-020366	OSA2
NP-020181	29.198-02	014	-	Rel-5	Addition of support for Java API technology realisation	B	4.4.0	5.0.0	N5-020367	OSA2
NP-020181	29.198-03	033	-	Rel-5	Addition of support for Java API technology realisation	B	4.5.0	5.0.0	N5-020368	OSA2
NP-020181	29.198-04	037	-	Rel-5	Addition of support for Java API technology realisation	B	4.4.0	5.0.0	N5-020369	OSA2
NP-020181	29.198-05	009	-	Rel-5	Addition of support for Java API technology realisation	B	4.4.0	5.0.0	N5-020370	OSA2
NP-020181	29.198-06	009	-	Rel-5	Addition of support for Java API technology realisation	B	4.4.0	5.0.0	N5-020371	OSA2
NP-020181	29.198-12	016	-	Rel-5	Addition of support for Java API technology realisation	B	4.3.0	5.0.0	N5-020372	OSA2

## CHANGE REQUEST

⌘ **29.198-01 CR 005** ⌘ rev **-** ⌘ Current version: **4.3.1** ⌘

For **HELP** on using this form, see bottom of this page or look at the pop-up text over the ⌘ symbols.

**Proposed change affects:** ⌘ (U)SIM  ME/UE  Radio Access Network  Core Network

<b>Title:</b>	⌘ Addition of support for Java API technology realisation		
<b>Source:</b>	⌘ CN5		
<b>Work item code:</b>	⌘ OSA2	<b>Date:</b>	⌘ 17/05/2002
<b>Category:</b>	⌘ <b>B</b>	<b>Release:</b>	⌘ REL-5
	Use <u>one</u> of the following categories: <b>F</b> (correction) <b>A</b> (corresponds to a correction in an earlier release) <b>B</b> (addition of feature), <b>C</b> (functional modification of feature) <b>D</b> (editorial modification) Detailed explanations of the above categories can be found in 3GPP TR 21.900.		Use <u>one</u> of the following releases: 2 (GSM Phase 2) R96 (Release 1996) R97 (Release 1997) R98 (Release 1998) R99 (Release 1999) REL-4 (Release 4) REL-5 (Release 5)

<b>Reason for change:</b>	⌘ It has been acknowledged that OSA can be realised using different technologies. One of these proposed technologies is Java. This change is necessary in order to acknowledge Java as another technology realisation of OSA.
<b>Summary of change:</b>	⌘ Indicate that OSA is realised though both IDL and Java. Provide an Informative Annex that specifies how the Java is created from UML.
<b>Consequences if not approved:</b>	⌘ Lack of application portability between OSA gateways that include Java as a realisation of OSA.

<b>Clauses affected:</b>	⌘ 2, 3.2, 5, B, C		
<b>Other specs affected:</b>	⌘ <input type="checkbox"/> Other core specifications <input type="checkbox"/> Test specifications <input type="checkbox"/> O&M Specifications	⌘	
<b>Other comments:</b>	⌘		

### How to create CRs using this form:

Comprehensive information and tips about how to create CRs can be found at: [http://www.3gpp.org/3G\\_Specs/CRs.htm](http://www.3gpp.org/3G_Specs/CRs.htm). Below is a brief summary:

- 1) Fill out the above form. The symbols above marked ⌘ contain pop-up help information about the field that they are closest to.
- 2) Obtain the latest version for the release of the specification to which the change is proposed. Use the MS Word "revision marks" feature (also known as "track changes") when making the changes. All 3GPP specifications can be downloaded from the 3GPP server under <ftp://ftp.3gpp.org/specs/>. For the latest version, look for the directory name with the latest date e.g. 2001-03 contains the specifications resulting from the March 2001 TSG meetings.
- 3) With "track changes" disabled, paste the entire CR form (use CTRL-A to select it) into the specification just in front of the clause containing the first piece of changed text. Delete those parts of the specification which are not relevant to the change request.

## 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 TR 21.905: "3G Vocabulary".
- [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] 3GPP TS 23.078: "CAMEL Phase 3, stage 2".
- [5] 3GPP TS 22.101: "Universal Mobile Telecommunications System (UMTS): Service Aspects; Service Principles".
- [6] World Wide Web Consortium Composite Capability/Preference Profiles (CC/PP): A user side framework for content negotiation ([www.w3.org](http://www.w3.org)).
- [7] 3GPP TS 29.002: "Mobile Application Part (MAP)".
- [8] 3GPP TS 29.078: "CAMEL Phase 3, , CAMEL Application Part (CAP) Specification".
- [9] Wireless Application Protocol (WAP), Version 1.2, UAProf Specification ([www.wapforum.org](http://www.wapforum.org)).
- [10] Wireless Application Protocol (WAP), version 1.2, WAP Service Indication specification, ([www.wapforum.org](http://www.wapforum.org)).
- [11] Wireless Application Protocol (WAP), version 1.2, WAP Push Architecture Overview ([www.wapforum.org](http://www.wapforum.org)).
- [12] Wireless Application Protocol (WAP), version 1.2, WAP Architecture ([www.wapforum.org](http://www.wapforum.org)).
- [13] ~~SUN~~ Java IDL Compiler (~~[www.javasoft.com/products/jdk/idl/index.html](http://www.javasoft.com/products/jdk/idl/index.html)~~<http://java.sun.com/products/jdk/idl/index.html>).
- [14] UML Unified Modelling Language ([www.rational.com/uml](http://www.rational.com/uml)).
- [15] Object Management Group ([www.omg.org](http://www.omg.org)).
- [16] 3GPP TS 22.002: "Circuit Bearer Services supported by a PLMN".
- [17] 3GPP TS 22.003: "Circuit Teleservices supported by a PLMN".
- [18] 3GPP TS 24.002: "Public Land Mobile Network (PLMN) Access Reference Configuration".
- [19] ITU-T Q.763: "Signalling System No. 7 – ISDN user part formats and codes".
- [20] ITU-T 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] 3GPP TS 22.121: "Service aspects; The Virtual Home Environment (Release 4)".

- [24] ["The Parlay Group homepage" \(http://www.parlay.org\)](http://www.parlay.org)
- [25] ["JAIN Community homepage" \(http://java.sun.com/products/jain\)](http://java.sun.com/products/jain)
- [26] 3GPP TS 23.057: "Mobile Station Application Execution Environment (MExE)".
- [27] ["JSR Overview" \(http://jcp.org/jsr/overview/index.en.jsp\)](http://jcp.org/jsr/overview/index.en.jsp)
- [28] ["Java 2 SDK, Standard Edition" \(http://java.sun.com/j2se/1.4/docs/relnotes/features.html\)](http://java.sun.com/j2se/1.4/docs/relnotes/features.html)
- [29] ["Java Community Process" \(http://jcp.org/\)](http://jcp.org/)

---

## 3 Definitions and abbreviations

### 3.1 Definitions

For the purposes of the present document, the terms and definitions given in 3GPP TS 22.101 [5] and the following apply.

**Applications:** Services, which are designed using Service Capability Features (SCFs).

**Gateway:** Synonym for Service Capability Server (SCS). From the viewpoint of applications, an SCS can be seen as a gateway to the core network.

**HE-VASP:** Home Environment Value Added Service Provider. 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:** A 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 (PSE):** contains personalised information defining how subscribed services are provided and presented towards the user. 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. These are within networks and under network control.

**Service Capability Feature (SCF):** Functionality offered by service capabilities that are accessible via the standardised OSA interface.

**Service Capability Server (SCS):** Functional Entity providing OSA interfaces towards an application.

**Service:** term used as an 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:** This is a 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:** A 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 3GPP TR 21.905 [1] and the following apply.

API	Application Programming Interface
CAMEL	Customised Application for Mobile network Enhanced Logic
CAP	CAMEL Application Part
CSE	CAMEL Service Environment
FW	Framework
HE	Home Environment
HE-VASP	Home Environment - Value Added Service Provider
HLR	Home Location Register
INAP	Intelligent Networks Application Part
IDL	Interface Description Language
<u>JSR</u>	<u>Java Specification Request</u>
MAP	Mobile Application Part
ME	Mobile Equipment
MEExE	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
SCS	Service Capability Server
SIM	Subscriber Identity Module
SMS	Short Message Service
SMTP	Simple Mail Transfer Protocol
<u>SPA</u>	<u>Service Provider API</u>
UE	User Equipment
USIM	Universal Subscriber Identity Module
VLR	Visited Location Register
VASP	Value Added Service Provider
VHE	Virtual Home Environment
WAP	Wireless Application Protocol
WGP	Wireless Gateway Proxy
WPP	Wireless Push Proxy

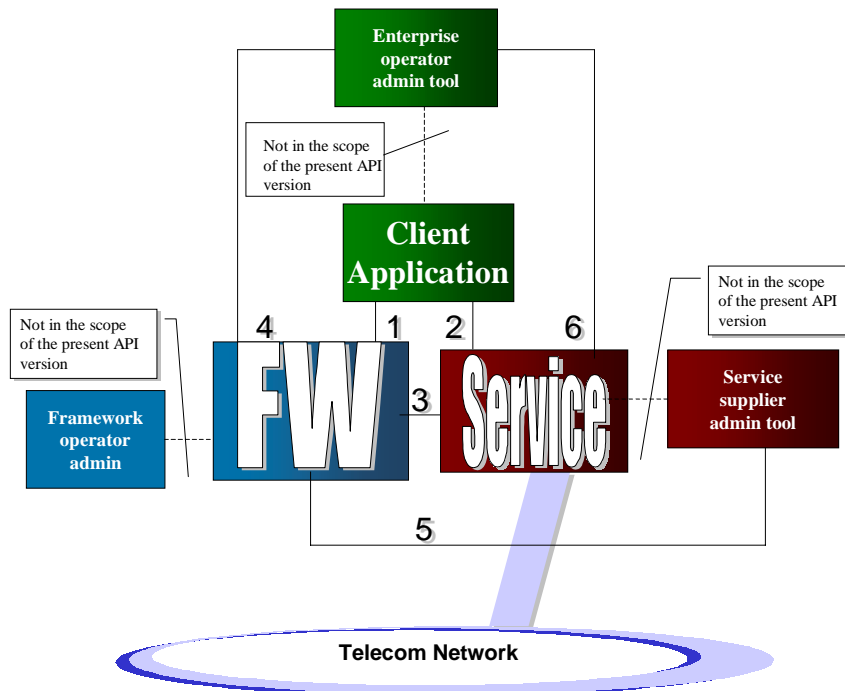
# 4 Open Service Access APIs

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 APIs. The network functionality is describes as Service Capability Features (SCFs) or Services. The OSA Framework is a general component in support of Services (Service Capabilities) and Applications. 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 OSA API is split into three types of interface classes, Service and Framework (FW).

- Interface classes between the Applications and the Framework (FW), 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 SCFs, 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 (FW) and the SCFs, that provide the mechanisms necessary for a multi-vendor environment.

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



**Figure 1:**

Within the OSA concept a set of Service Capability Features (SCFs) 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 and the following Parts contain the description of the SCFs.

**NOTE:** The terms ‘Service’ and ‘Service Capability Feature’ are used as alternatives for the same concept in the present document. In the OSA API itself the SCFs as identified in the 3GPP requirements and architecture are reflected as ‘service’, in terms like serviceFactory, serviceDiscovery.

## 5 Structure of the OSA API (29.198) and Mapping (29.998) documents

The Open Service Access (OSA) Application Programming Interface (API) specifications consist of two sets of documents:

### **API specification** (3GPP TS 29.198)

The Parts of 29.198 - apart from Part 1 (the present document) and Part 2 - define the interfaces, parameters and state models that belong to the API specification. UML (Unified Modelling Language) 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 (Interface Description Language). Reference is made to the Java API specification of the interfaces.

### **Mapping specification of the OSA APIs and network protocols** (3GPP TR 29.998)

The Parts of 29.998 contain a possible mapping from the APIs defined in 29.198 to various network protocols (i.e. MAP [7], CAP [8], etc.). 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 SCFs to the application. The specific underlying network and its protocols are transparent to the application.

The **API specification** (3GPP TS 29.198) is structured in the following Parts:

29.198-1	Part 1:	Overview
29.198-2	Part 2:	Common Data Definitions
29.198-3	Part 3:	Framework
29.198-4	Part 4:	Call Control SCF
29.198-5	Part 5:	User Interaction SCF
29.198-6	Part 6:	Mobility SCF
29.198-7	Part 7:	Terminal Capabilities SCF
29.198-8	Part 8:	Data Session Control SCF
29.198-9	Part 9:	Generic Messaging SCF
29.198-10	Part 10:	Connectivity Manager SCF
29.198-11	Part 11:	Account Management SCF
29.198-12	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.

## Structure of the Parts of 29.198

The Parts with API specification themselves are structured 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 progression of internal processes either in the application, or Gateway.
- The Data definitions clauses show a detailed expansion of each of the data types associated with the methods within the classes. It is to be noted 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).
- Reference to the Java API description of the interface (informative 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 follow 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*.

---

## 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 this specification.

```
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= P_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): Java API

### B.1 Tools and Languages

The Java language is used as a means to programmatically define the interfaces. Java files are either generated manually from class diagrams or by using a UML tool and editing scripts. Either way, the Java files are generated by the JAIN Community [25] in accordance with the Parlay UML to Java API Rulebook [24], which define a set of rules that are used to rapidly generate the Java APIs from the Parlay UML.

The generated Java files are verified using Java compilers such as javac [28]. The Java API specifications are designed to be compatible with the Java 2 SDK, Standard Edition, version 1.4.0 [28] or later. The Java API Realizations of the Parlay/OSA APIs are known as the JAIN Service Provider APIs (JAIN SPA).

---

### B.2 JAIN SPA Overview

JAIN SPA is a local Java API realization of the Parlay UML specifications. The benefits of providing a local API (in addition to a distribution or remote API, such as the Parlay OMG-IDL or the Parlay W3C WSDL) is that the API is tailored to a particular programming language (in this case it's Java), which is distribution mechanism independent, meaning that, providing the necessary adapters are put in place, Java applications can be written to this local API that use any form of technology (e.g. CORBA, SOAP, RMI) for the purpose of distributing this API. With remote APIs, although the programmer may be free to write in multiple programming languages, he needs knowledge of, and is committed to, the particular distribution mechanism (e.g. CORBA, SOAP, RMI).

As the Parlay UML assumes a remote API, many optimizations have been made to the specifications, which, although acceptable to a "specialist" programmer taking distribution into account, would appear alien to the large community of "regular" Java programmers. As such, the JAIN SPA specifications are tailored to the Java language by following Java language naming conventions, design patterns and object oriented practices for a local Java API, while reusing as much Java codebase as possible. JAIN Service Provider APIs are developed by the JAIN Community [25] under the Java Community Process (JCP) [29]. Within the JCP, each JAIN Service Provider API is developed by submitting a Java Specification Request (JSR) [27]. Each JAIN Service Provider API is assigned a JSR number, and an associated webpage, that can be used to identify it.

Each JSR webpage contains a table identifying the relationships between the different versions of the Parlay, ETSI/OSA, 3GPP/OSA and JAIN SPA specifications. In addition, each JAIN SPA specification version indicates to which Parlay, ETSI/OSA and 3GPP/OSA specification versions it corresponds to.

## CHANGE REQUEST

⌘ **29.198-02 CR 014** ⌘ rev **-** ⌘ Current version: **4.4.0** ⌘

For **HELP** on using this form, see bottom of this page or look at the pop-up text over the ⌘ symbols.

**Proposed change affects:** ⌘ (U)SIM  ME/UE  Radio Access Network  Core Network

<b>Title:</b>	⌘ Addition of support for Java API technology realisation		
<b>Source:</b>	⌘ CN5		
<b>Work item code:</b>	⌘ OSA2	<b>Date:</b>	⌘ 17/05/2002
<b>Category:</b>	⌘ <b>B</b>	<b>Release:</b>	⌘ REL-5
	Use <u>one</u> of the following categories: <b>F</b> (correction) <b>A</b> (corresponds to a correction in an earlier release) <b>B</b> (addition of feature), <b>C</b> (functional modification of feature) <b>D</b> (editorial modification) Detailed explanations of the above categories can be found in 3GPP TR 21.900.		Use <u>one</u> of the following releases: 2 (GSM Phase 2) R96 (Release 1996) R97 (Release 1997) R98 (Release 1998) R99 (Release 1999) REL-4 (Release 4) REL-5 (Release 5)

<b>Reason for change:</b>	⌘ It has been acknowledged that OSA can be realised using different technologies. One of these proposed technologies is Java. This change is necessary in order to acknowledge Java as another technology realisation of OSA
<b>Summary of change:</b>	⌘ Indicate that OSA is realised though both IDL and Java. Provide an Informative Annex that references the corresponding Java specification.
<b>Consequences if not approved:</b>	⌘ Lack of application portability between OSA gateways that include Java as a realisation of OSA.

<b>Clauses affected:</b>	⌘ B (new)		
<b>Other specs affected:</b>	⌘ <input type="checkbox"/> Other core specifications <input type="checkbox"/> Test specifications <input type="checkbox"/> O&M Specifications	⌘	
<b>Other comments:</b>	⌘ <b>Another CR will be submitted to nullify this CR if the referenced Java specification does not reach Public Review status before 1 September 2002</b>		

### How to create CRs using this form:

Comprehensive information and tips about how to create CRs can be found at: [http://www.3gpp.org/3G\\_Specs/CRs.htm](http://www.3gpp.org/3G_Specs/CRs.htm). Below is a brief summary:

- 1) Fill out the above form. The symbols above marked ⌘ contain pop-up help information about the field that they are closest to.
- 2) Obtain the latest version for the release of the specification to which the change is proposed. Use the MS Word "revision marks" feature (also known as "track changes") when making the changes. All 3GPP specifications can be downloaded from the 3GPP server under <ftp://ftp.3gpp.org/specs/> For the latest version, look for the directory name with the latest date e.g. 2001-03 contains the specifications resulting from the March 2001 TSG meetings.
- 3) With "track changes" disabled, paste the entire CR form (use CTRL-A to select it) into the specification just in front of the clause containing the first piece of changed text. Delete those parts of the specification which are not relevant to the change request.

---

## Annex B (informative): Java API Description of the Common Data definitions

The Java API representation of this specification can be obtained from the following URL:

- JAIN Common (<http://jcp.org/jsr/detail/145.jsp>)

Each JSR webpage contains a table identifying the relationships between the different versions of the Parlay, ETSI/OSA, 3GPP/OSA and JAIN SPA specifications. In addition, each JAIN SPA specification version indicates to which Parlay, ETSI/OSA and 3GPP/OSA specification versions it corresponds to.

## CHANGE REQUEST

⌘ **29.198-03 CR 033** ⌘ rev **-** ⌘ Current version: **4.5.0** ⌘

For **HELP** on using this form, see bottom of this page or look at the pop-up text over the ⌘ symbols.

Proposed change affects: ⌘ (U)SIM  ME/UE  Radio Access Network  Core Network

<b>Title:</b>	⌘ Addition of support for Java API technology realisation		
<b>Source:</b>	⌘ CN5		
<b>Work item code:</b>	⌘ OSA2	<b>Date:</b>	⌘ 17/05/2002
<b>Category:</b>	⌘ <b>B</b> Use <u>one</u> of the following categories: <b>F</b> (correction) <b>A</b> (corresponds to a correction in an earlier release) <b>B</b> (addition of feature), <b>C</b> (functional modification of feature) <b>D</b> (editorial modification) Detailed explanations of the above categories can be found in 3GPP TR 21.900.	<b>Release:</b>	⌘ <b>REL-5</b> Use <u>one</u> of the following releases: <b>2</b> (GSM Phase 2) <b>R96</b> (Release 1996) <b>R97</b> (Release 1997) <b>R98</b> (Release 1998) <b>R99</b> (Release 1999) <b>REL-4</b> (Release 4) <b>REL-5</b> (Release 5)

<b>Reason for change:</b>	⌘ It has been acknowledged that OSA can be realised using different technologies. One of these proposed technologies is Java. This change is necessary in order to acknowledge Java as another technology realisation of OSA
<b>Summary of change:</b>	⌘ Indicate that OSA is realised though both IDL and Java. Provide an Informative Annex that references the corresponding Java specification.
<b>Consequences if not approved:</b>	⌘ Lack of application portability between OSA gateways that include Java as a realisation of OSA.

<b>Clauses affected:</b>	⌘ B (new)
<b>Other specs affected:</b>	⌘ <input type="checkbox"/> Other core specifications ⌘ <input type="checkbox"/> Test specifications <input type="checkbox"/> O&M Specifications
<b>Other comments:</b>	⌘ <b>Another CR will be submitted to nullify this CR if the referenced Java specifications do not reach Public Review status before 1 September 2002</b>

### How to create CRs using this form:

Comprehensive information and tips about how to create CRs can be found at: [http://www.3gpp.org/3G\\_Specs/CRs.htm](http://www.3gpp.org/3G_Specs/CRs.htm). Below is a brief summary:

- 1) Fill out the above form. The symbols above marked ⌘ contain pop-up help information about the field that they are closest to.
- 2) Obtain the latest version for the release of the specification to which the change is proposed. Use the MS Word "revision marks" feature (also known as "track changes") when making the changes. All 3GPP specifications can be downloaded from the 3GPP server under <ftp://ftp.3gpp.org/specs/> For the latest version, look for the directory name with the latest date e.g. 2001-03 contains the specifications resulting from the March 2001 TSG meetings.
- 3) With "track changes" disabled, paste the entire CR form (use CTRL-A to select it) into the specification just in front of the clause containing the first piece of changed text. Delete those parts of the specification which are not relevant to the change request.

---

## Annex B (informative): Java API Description of the Framework

The Java API representation of this specification can be obtained from the following URLs:

- JAIN SPA Framework Access Session (<http://jcp.org/jsr/detail/24.jsp>)
- JAIN SPA Framework to Application (<http://jcp.org/jsr/detail/119.jsp>)

Each JSR webpage contains a table identifying the relationships between the different versions of the Parlay, ETSI/OSA, 3GPP/OSA and JAIN SPA specifications. In addition, each JAIN SPA specification version indicates to which Parlay, ETSI/OSA and 3GPP/OSA specification versions it corresponds to.

## CHANGE REQUEST

⌘ **29.198-04 CR 037** ⌘ rev **-** ⌘ Current version: **4.4.0** ⌘

For **HELP** on using this form, see bottom of this page or look at the pop-up text over the ⌘ symbols.

**Proposed change affects:** ⌘ (U)SIM  ME/UE  Radio Access Network  Core Network

<b>Title:</b>	⌘ Addition of support for Java API technology realisation		
<b>Source:</b>	⌘ CN5		
<b>Work item code:</b>	⌘ OSA2	<b>Date:</b>	⌘ 17/05/2002
<b>Category:</b>	⌘ <b>B</b>	<b>Release:</b>	⌘ REL-5
	Use <u>one</u> of the following categories: <b>F</b> (correction) <b>A</b> (corresponds to a correction in an earlier release) <b>B</b> (addition of feature), <b>C</b> (functional modification of feature) <b>D</b> (editorial modification) Detailed explanations of the above categories can be found in 3GPP TR 21.900.		Use <u>one</u> of the following releases: 2 (GSM Phase 2) R96 (Release 1996) R97 (Release 1997) R98 (Release 1998) R99 (Release 1999) REL-4 (Release 4) REL-5 (Release 5)

<b>Reason for change:</b>	⌘ It has been acknowledged that OSA can be realised using different technologies. One of these proposed technologies is Java. This change is necessary in order to acknowledge Java as another technology realisation of OSA
<b>Summary of change:</b>	⌘ Indicate that OSA is realised though both IDL and Java. Provide an Informative Annex that references the corresponding Java specification.
<b>Consequences if not approved:</b>	⌘ Lack of application portability between OSA gateways that include Java as a realisation of OSA.

<b>Clauses affected:</b>	⌘ B (new)		
<b>Other specs affected:</b>	⌘ <input type="checkbox"/> Other core specifications <input type="checkbox"/> Test specifications <input type="checkbox"/> O&M Specifications	⌘	
<b>Other comments:</b>	⌘ <b>Another CR will be submitted to nullify this CR if the referenced Java specification does not reach Public Review status before 1 September 2002</b>		

### How to create CRs using this form:

Comprehensive information and tips about how to create CRs can be found at: [http://www.3gpp.org/3G\\_Specs/CRs.htm](http://www.3gpp.org/3G_Specs/CRs.htm). Below is a brief summary:

- 1) Fill out the above form. The symbols above marked ⌘ contain pop-up help information about the field that they are closest to.
- 2) Obtain the latest version for the release of the specification to which the change is proposed. Use the MS Word "revision marks" feature (also known as "track changes") when making the changes. All 3GPP specifications can be downloaded from the 3GPP server under <ftp://ftp.3gpp.org/specs/> For the latest version, look for the directory name with the latest date e.g. 2001-03 contains the specifications resulting from the March 2001 TSG meetings.
- 3) With "track changes" disabled, paste the entire CR form (use CTRL-A to select it) into the specification just in front of the clause containing the first piece of changed text. Delete those parts of the specification which are not relevant to the change request.



---

## Annex B (informative): Java API Description of the Call Control SCFs

The Java API representation of this specification can be obtained from the following URL:

- Java Call Control (<http://jcp.org/jsr/detail/21.jsp>)

Each JSR webpage contains a table identifying the relationships between the different versions of the Parlay, ETSI/OSA, 3GPP/OSA and JAIN SPA specifications. In addition, each JAIN SPA specification version indicates to which Parlay, ETSI/OSA and 3GPP/OSA specification versions it corresponds to.

## CHANGE REQUEST

⌘ **29.198-05 CR 009** ⌘ rev **-** ⌘ Current version: **4.4.0** ⌘

For **HELP** on using this form, see bottom of this page or look at the pop-up text over the ⌘ symbols.

**Proposed change affects:** ⌘ (U)SIM  ME/UE  Radio Access Network  Core Network

<b>Title:</b>	⌘ Addition of support for Java API technology realisation		
<b>Source:</b>	⌘ CN5		
<b>Work item code:</b>	⌘ OSA2	<b>Date:</b>	⌘ 17/05/2002
<b>Category:</b>	⌘ <b>B</b>	<b>Release:</b>	⌘ REL-5
	Use <u>one</u> of the following categories: <b>F</b> (correction) <b>A</b> (corresponds to a correction in an earlier release) <b>B</b> (addition of feature), <b>C</b> (functional modification of feature) <b>D</b> (editorial modification) Detailed explanations of the above categories can be found in 3GPP TR 21.900.		Use <u>one</u> of the following releases: 2 (GSM Phase 2) R96 (Release 1996) R97 (Release 1997) R98 (Release 1998) R99 (Release 1999) REL-4 (Release 4) REL-5 (Release 5)

<b>Reason for change:</b>	⌘ It has been acknowledged that OSA can be realised using different technologies. One of these proposed technologies is Java. This change is necessary in order to acknowledge Java as another technology realisation of OSA
<b>Summary of change:</b>	⌘ Indicate that OSA is realised though both IDL and Java. Provide an Informative Annex that references the corresponding Java specification.
<b>Consequences if not approved:</b>	⌘ Lack of application portability between OSA gateways that include Java as a realisation of OSA.

<b>Clauses affected:</b>	⌘ B (new)		
<b>Other specs affected:</b>	⌘ <input type="checkbox"/> Other core specifications <input type="checkbox"/> Test specifications <input type="checkbox"/> O&M Specifications	⌘	
<b>Other comments:</b>	⌘ <b>Another CR will be submitted to nullify this CR if the referenced Java specification does not reach Public Review status before 1 September 2002</b>		

### How to create CRs using this form:

Comprehensive information and tips about how to create CRs can be found at: [http://www.3gpp.org/3G\\_Specs/CRs.htm](http://www.3gpp.org/3G_Specs/CRs.htm). Below is a brief summary:

- 1) Fill out the above form. The symbols above marked ⌘ contain pop-up help information about the field that they are closest to.
- 2) Obtain the latest version for the release of the specification to which the change is proposed. Use the MS Word "revision marks" feature (also known as "track changes") when making the changes. All 3GPP specifications can be downloaded from the 3GPP server under <ftp://ftp.3gpp.org/specs/> For the latest version, look for the directory name with the latest date e.g. 2001-03 contains the specifications resulting from the March 2001 TSG meetings.
- 3) With "track changes" disabled, paste the entire CR form (use CTRL-A to select it) into the specification just in front of the clause containing the first piece of changed text. Delete those parts of the specification which are not relevant to the change request.

---

## Annex B (informative): Java API Description of the User Interaction SCF

The Java API representation of this specification can be obtained from the following URL:

- JAIN User Interaction (<http://jcp.org/jsr/detail/103.jsp>)

Each JSR webpage contains a table identifying the relationships between the different versions of the Parlay, ETSI/OSA, 3GPP/OSA and JAIN SPA specifications. In addition, each JAIN SPA specification version indicates to which Parlay, ETSI/OSA and 3GPP/OSA specification versions it corresponds to.

## CHANGE REQUEST

⌘ **29.198-06 CR 009** ⌘ rev **-** ⌘ Current version: **4.4.0** ⌘

For **HELP** on using this form, see bottom of this page or look at the pop-up text over the ⌘ symbols.

**Proposed change affects:** ⌘ (U)SIM  ME/UE  Radio Access Network  Core Network

<b>Title:</b>	⌘ Addition of support for Java API technology realisation		
<b>Source:</b>	⌘ CN5		
<b>Work item code:</b>	⌘ OSA2	<b>Date:</b>	⌘ 17/05/2002
<b>Category:</b>	⌘ <b>B</b>	<b>Release:</b>	⌘ REL-5
	Use <u>one</u> of the following categories: <b>F</b> (correction) <b>A</b> (corresponds to a correction in an earlier release) <b>B</b> (addition of feature), <b>C</b> (functional modification of feature) <b>D</b> (editorial modification) Detailed explanations of the above categories can be found in 3GPP TR 21.900.		Use <u>one</u> of the following releases: 2 (GSM Phase 2) R96 (Release 1996) R97 (Release 1997) R98 (Release 1998) R99 (Release 1999) REL-4 (Release 4) REL-5 (Release 5)

<b>Reason for change:</b>	⌘ It has been acknowledged that OSA can be realised using different technologies. One of these proposed technologies is Java. This change is necessary in order to acknowledge Java as another technology realisation of OSA
<b>Summary of change:</b>	⌘ Indicate that OSA is realised though both IDL and Java. Provide an Informative Annex that references the corresponding Java specification.
<b>Consequences if not approved:</b>	⌘ Lack of application portability between OSA gateways that include Java as a realisation of OSA.

<b>Clauses affected:</b>	⌘ B (new)		
<b>Other specs affected:</b>	⌘ <input type="checkbox"/> Other core specifications <input type="checkbox"/> Test specifications <input type="checkbox"/> O&M Specifications	⌘	
<b>Other comments:</b>	⌘ <b>Another CR will be submitted to nullify this CR if the referenced Java specification does not reach Public Review status before 1 September 2002</b>		

### How to create CRs using this form:

Comprehensive information and tips about how to create CRs can be found at: [http://www.3gpp.org/3G\\_Specs/CRs.htm](http://www.3gpp.org/3G_Specs/CRs.htm). Below is a brief summary:

- 1) Fill out the above form. The symbols above marked ⌘ contain pop-up help information about the field that they are closest to.
- 2) Obtain the latest version for the release of the specification to which the change is proposed. Use the MS Word "revision marks" feature (also known as "track changes") when making the changes. All 3GPP specifications can be downloaded from the 3GPP server under <ftp://ftp.3gpp.org/specs/> For the latest version, look for the directory name with the latest date e.g. 2001-03 contains the specifications resulting from the March 2001 TSG meetings.
- 3) With "track changes" disabled, paste the entire CR form (use CTRL-A to select it) into the specification just in front of the clause containing the first piece of changed text. Delete those parts of the specification which are not relevant to the change request.

---

## Annex B (informative): Java API Description of the Mobility SCFs

The Java API representation of this specification can be obtained from the following URL:

- JAIN User Location and Status (<http://jcp.org/jsr/detail/98.jsp>)

Each JSR webpage contains a table identifying the relationships between the different versions of the Parlay, ETSI/OSA, 3GPP/OSA and JAIN SPA specifications. In addition, each JAIN SPA specification version indicates to which Parlay, ETSI/OSA and 3GPP/OSA specification versions it corresponds to.

## CHANGE REQUEST

⌘ **29.198-12 CR 016** ⌘ rev **-** ⌘ Current version: **4.3.0** ⌘

For **HELP** on using this form, see bottom of this page or look at the pop-up text over the ⌘ symbols.

**Proposed change affects:** ⌘ (U)SIM  ME/UE  Radio Access Network  Core Network

<b>Title:</b>	⌘ Addition of support for Java API technology realisation		
<b>Source:</b>	⌘ CN5		
<b>Work item code:</b>	⌘ OSA2	<b>Date:</b>	⌘ 17/05/2002
<b>Category:</b>	⌘ <b>B</b>	<b>Release:</b>	⌘ REL-5
	Use <u>one</u> of the following categories: <b>F</b> (correction) <b>A</b> (corresponds to a correction in an earlier release) <b>B</b> (addition of feature), <b>C</b> (functional modification of feature) <b>D</b> (editorial modification) Detailed explanations of the above categories can be found in 3GPP TR 21.900.		Use <u>one</u> of the following releases: 2 (GSM Phase 2) R96 (Release 1996) R97 (Release 1997) R98 (Release 1998) R99 (Release 1999) REL-4 (Release 4) REL-5 (Release 5)

<b>Reason for change:</b>	⌘ It has been acknowledged that OSA can be realised using different technologies. One of these proposed technologies is Java. This change is necessary in order to acknowledge Java as another technology realisation of OSA
<b>Summary of change:</b>	⌘ Indicate that OSA is realised though both IDL and Java. Provide an Informative Annex that references the corresponding Java specification.
<b>Consequences if not approved:</b>	⌘ Lack of application portability between OSA gateways that include Java as a realisation of OSA.

<b>Clauses affected:</b>	⌘ B (new)		
<b>Other specs affected:</b>	⌘ <input type="checkbox"/> Other core specifications <input type="checkbox"/> Test specifications <input type="checkbox"/> O&M Specifications	⌘	
<b>Other comments:</b>	⌘ <b>Another CR will be submitted to nullify this CR if the referenced Java specification does not reach Public Review status before 1 September 2002</b>		

### How to create CRs using this form:

Comprehensive information and tips about how to create CRs can be found at: [http://www.3gpp.org/3G\\_Specs/CRs.htm](http://www.3gpp.org/3G_Specs/CRs.htm). Below is a brief summary:

- 1) Fill out the above form. The symbols above marked ⌘ contain pop-up help information about the field that they are closest to.
- 2) Obtain the latest version for the release of the specification to which the change is proposed. Use the MS Word "revision marks" feature (also known as "track changes") when making the changes. All 3GPP specifications can be downloaded from the 3GPP server under <ftp://ftp.3gpp.org/specs/> For the latest version, look for the directory name with the latest date e.g. 2001-03 contains the specifications resulting from the March 2001 TSG meetings.
- 3) With "track changes" disabled, paste the entire CR form (use CTRL-A to select it) into the specification just in front of the clause containing the first piece of changed text. Delete those parts of the specification which are not relevant to the change request.

---

## Annex B (informative): Java API Description of the Charging SCF

The Java API representation of this specification can be obtained from the following URL:

- Java Pay (<http://jcp.org/jsr/detail/182.jsp>)

Each JSR webpage contains a table identifying the relationships between the different versions of the Parlay, ETSI/OSA, 3GPP/OSA and JAIN SPA specifications. In addition, each JAIN SPA specification version indicates to which Parlay, ETSI/OSA and 3GPP/OSA specification versions it corresponds to.