3GPP TSG CN Plenary Meeting #26 **NP-040488**
08-10 December 2004, Athens, GREECE

Source: CN5 (OSA)

Title: Rel-6 Draft TS 23.198 v100 (OSA Stage 2) – for Information

Agenda item: 9.7 (OSA Enhancements [OSA3])

Document for: APPROVAL

Presentation of Technical Specification to TSG CN

Abstract of document:

- This TS covers the Open Service Access (OSA) Stage 2 description, formerly covered by the SA2's

TS 23.127	Virtual Home Environment (VHE) / Open Service Access (OSA)

- Work done against the CN#25 approved WID in NP-040351 (Work Item ID: OSA3).

Purpose of This Specification:

- Since CN#25 (09/2004), CN5 took the responsibility for OSA Stage 2 in addition to OSA Stage 3 specifications.
- The ***OSA APIs*** are designed to enable creation of telephony applications as well as to "telecom-enable" IT applications. IT developers, who develop and deploy applications outside the traditional telecommunications network space and business model, are viewed as crucial for creating a dramatic whole-market growth in next generation applications, services and networks.

Changes since last presentation to TSG-CN

- New.
- Draft TS 23.198 v1.0.0 is an extraction from the latest version of 3GPP SA2's TS 23.127 on VHE/OSA Stage 2 and several CN5 contributions had been added.

Outstanding Issues:

- None. However, a comprehensive review of the draft TS is needed before placing it under change control (CR regime).

Contentious Issues:

- SA2 TS 23.127 covers both OSA&VHE and VHE is not anymore part of Rel6.
- The SA1 VHE TS was downgraded to a TR in pre-Rel-6 releases.

TR 22.121	Service aspects; The Virtual Home Environment; Stage 1

- 23.127 plus CN5 made 23.127 CRs were used as basis for the new OSA-only Stage 2 description in TS 23.198, which due to SA2 handover delays is submitted to CN#26 for Information only.
- SA2 needs to withdraw Rel-6 23.127. Reason: there are currently in Rel-6 two (2) TSs on OSA Stage 2 !
- TS 23.198 final Rel6 OSA Stage 2 for Approval at CN#27 (Mar 2005)

3GPP TS 23.198 V1.0.0 (2004-12)

Technical Specification

3rd Generation Partnership Project; Technical Specification Group Core Network; Open Service Access (OSA); Stage 2 (Release 6)

Keywords
UMTS, OSA,

3GPP

Postal address

3GPP support office address
650 Route des Lucioles - Sophia Antipolis
Valbonne - FRANCE
Tel.: +33 4 92 94 42 00 Fax: +33 4 93 65 47 16

Internet
http://www.3gpp.org

Contents

Foreword

This Technical Specification (TS) 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 Open Service Access (OSA) defines 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 the present document. The requirements for OSA are contained in 3GPP TS 22.127 [2].

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 Application Programming Interface (API) itself the SCFs as identified in the 3GPP requirements and architecture are reflected as ‘service’, in terms like service instance lifecycle manager, service Discovery.

The present document is part of a TS-family as identified below:

22.127: "Service Requirement for the Open Services Access (OSA); Stage 1".
23.198: "Open Service Access (OSA); Stage 2".

Stage 3 Technical Specifications (TSs):

29.198-01:	"OSA API; Part 1:	Overview".	
29.198-02:	"OSA API; Part 2:	Common data".	
29.198-03:	"OSA API; Part 3:	Framework".	
29.198-04:	"OSA API; Part 4:	Call control".	
29.198-04-1:	"OSA API; Part 4:	Call control; Subpart 1:	Common call control data definitions".
29.198-04-2:	"OSA API; Part 4:	Call control; Subpart 2:	Generic call control data SCF".
29.198-04-3:	"OSA API; Part 4:	Call control; Subpart 3:	Multi-party call control data SCF".
29.198-04-4:	"OSA API; Part 4:	Call control; Subpart 4:	Multimedia call control SCF".
29.198-05:	"OSA API; Part 5:	Generic user interaction".	
29.198-06:	"OSA API; Part 6:	Mobility".	
29.198-07:	"OSA API; Part 7:	Terminal capabilities".	
29.198-08:	"OSA API; Part 8:	Data session control".	
29.198-09:	"OSA API; Part 9:	Generic messaging SCF".	
29.198-10:	"OSA API; Part 10:	Connectivity manager SCF".	
29.198-11:	"OSA API; Part 11:	Account management".	

29.198-12: "OSA API; Part 12: Charging".
29.198-13: "OSA API; Part 13: Policy management SCF".
29.198-14: "OSA API; Part 14: Presence and Availability Management (PAM)".
29.198-15: "OSA API; Part 15: Multi-media Messaging (MM) SCF".

29.199-01: "OSA; Parlay X web services; Part 1: Common".
29.199-02: "OSA; Parlay X web services; Part 2: Third party call".
29.199-03: "OSA; Parlay X web services; Part 3: Call notification".
29.199-04: "OSA; Parlay X web services; Part 4: Short messaging".
29.199-05: "OSA; Parlay X web services; Part 5: Multimedia messaging".
29.199-06: "OSA; Parlay X web services; Part 6: Payment".
29.199-07: "OSA; Parlay X web services; Part 7: Account management".
29.199-08: "OSA; Parlay X web services; Part 8: User status".
29.199-09: "OSA; Parlay X web services; Part 9: Terminal location".
29.199-10: "OSA; Parlay X web services; Part 10: Call handling".
29.199-11: "OSA; Parlay X web services; Part 11: Audio call".
29.199-12: "OSA; Parlay X web services; Part 12: Multimedia conference".
29.199-13: "OSA; Parlay X web services; Part 13: Address list management".
29.199-14: "OSA; Parlay X web services; Part 14: Presence".

Technical Reports TRs):

29.998-01: "OSA API Mapping for OSA; Part 1: General issues on API mapping".
29.998-04-1: "OSA API Mapping for OSA; Part 4: Call Control Service Mapping; Subpart 1: API to CAP Mapping".
29.998-04-2: "OSA API Mapping for OSA; Part 4: Call Control Service Mapping; Subpart 2: INAP".
29.998-04-3: "OSA API Mapping for OSA; Part 4: Call Control Service Mapping; Subpart 3: MEGACO mapping".
29.998-04-4: "OSA API Mapping for OSA; Part 4: Call Control Service Mapping; Subpart 4: Multiparty Call Control ISC".
29.998-05-1: "OSA API Mapping for OSA; Part 5: User Interaction Service Mapping; Subpart 1: API to CAP Mapping".
29.998-05-2: "OSA API Mapping for OSA; Part 5: User Interaction Service Mapping; Subpart 2: INAP mapping".
29.998-05-3: "OSA API Mapping for OSA; Part 5: User Interaction Service Mapping; Subpart 3: MEGACO mapping".
29.998-05-4: "OSA API Mapping for OSA; Part 5: User Interaction Service Mapping; Subpart 4: API to SMS Mapping".
29.998-06: "OSA API Mapping for OSA; Part 6: User Location and User Status Service Mapping to MAP".
29.998-08: "OSA API Mapping for OSA; Part 8: Data Session Control Service Mapping to CAP".

1 Scope

The present document specifies the stage 2 of the Open Service Access (OSA).

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 22.101: "Service Aspects; Service Principles".

[2] 3GPP TR 21.905: "Vocabulary for 3GPP Specifications".

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

[4] 3GPP TR 22.121: "Service aspects; The Virtual Home Environment; Stage 1".

[5] 3GPP TS 22.141: "Presence Service; Stage 1".

[6] 3GPP TS 23.228: "IP Multimedia Subsystem (IMS) Stage 2".

[7] 3GPP TS 23.141: "Presence service; Architecture and functional description; Stage 2".

[8] 3GPP TS 23.140: "Multimedia Messaging Service (MMS); Functional description; Stage 2".

3 Definitions and abbreviations

3.1 Definitions

For the purposes of the present document, the terms and given in 3G TS 22.101 [1] and 3G TR 22.905 [2] and the following definitions apply:

Applications: software components providing services to end-users by utilising service capability features.

Home Environment: responsible for overall provision of services to users.

Home Environment Value Added Service Provider: see [4].

Interface: listing and semantics of the methods and attributes provided by an object that belongs to a Service Capability Feature.

OSA API: standardised API used by applications to access service capability features.

OSA Internal API: standardised API between framework and service capability servers.

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

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

Service Capabilities: See [3].

Service Capability Feature: See [3].

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

Services: See [4].

User Profile: See [4].

User Services Profile: See [4].

Value Added Service Provider: See [4].

Virtual Home Environment: See [4].

3.2 Abbreviations

For the purposes of the present document, the abbreviations defined in GSM 01.04 and in 3G TR 21.905 and the following apply:

API	Application Programming Interface
HE	Home Environment
HE-VASP	Home Environment Value Added Service Provider
HSS	Home Subscriber Server
IMS	IP Multimedia Core Network Subsystem
ISC	IMS Service Control
MRF	Media Resource Function
MRFC	Media Resource Function Controller
MRFP	Media Resource Function Protocol
OSA	Open Service Access
SCF	Service Capability Feature
SCS	Service Capability Server
S-CSCF	Serving Call Session Control Function
SMS-C	Short Message Service - Center
SOAP	Simple Object Access Protocol
VASP	Value Added Service Provider
VHE	Virtual Home Environment

4 OSA support of VASP

The OSA toolkit may be used by the Home Environment, by Value Added Service Providers (VASPs) and Home Environment Value Added Service Providers (HE-VASPs).

OSA is optimized to support HE-VASPs as the user subscription information is owned and managed by the home environment, i.e. the Home Environment knows which users are subscribed to the service implemented by the OSA application, and if the service is activated or not.

Specific methods are specified in OSA Network Service Capability Features, permitting:

- An OSA application to request user related event notifications pertaining to any subscribed user for which the service implemented by the application is activated.
- The OSA SCS to report user related event notifications in which it explicitly identifies the user to which the event applies.
- An OSA application to request a function to be applied to all current subscribed users for which the service implemented by the application is activated.

The OSA SCS can report user related events to the OSA application, without the application having explicitly subscribed to the event (events to be reported have been agreed between the Home Environment and the HE-VASP by other means, e.g. in their service level agreement).

These VHE-specific extensions applies to all relevant Network Service Capability Features, like call and session control SCFs, user status, and user location.

The OSA toolkit may be used by the Home Environment, by Value Added Service Providers (VASPs) and Home Environment Value Added Service Providers (HE-VASPs).

Extensions shall be made to OSA in order to optimize the support of HE-VASPs. These extensions shall use the fact that user subscription information is owned and managed by the home environment, i.e. the Home Environment knows which users are subscribed to the service implemented by the OSA application, and if the service is activated or not.

Specific methods shall be specified in OSA Service Capability Features, permitting:

- An OSA application to request user related event notifications pertaining to any subscribed user for which the service implemented by the application is activated.
- The OSA SCS to report user related event notifications in which it explicitly identifies the user to which the event applies.
- An OSA application to request a function to be applied to all current subscribed users for which the service implemented by the application is activated.

It shall also be possible for the OSA SCS to report user related events to the OSA application, without the application having explicitly subscribed to the event (events to be reported have been agreed between the Home Environment and the HE-VASP by other means, e.g. in their service level agreement).

These VHE-specific extensions shall apply to all relevant Service Capability Features, like call and session control SCFs, user status, and user location.

5 Open Service Access

In order to be able to implement future applications/end user services that are not yet known today, a highly flexible Framework for Services is required. The Open Service Access (OSA) enables applications implementing the services to make use of network functionality. Network functionality offered to applications is defined in terms of a set of Service Capability Features (SCFs). These SCFs provide functionality of network capabilities which is accessible to applications through the standardised OSA interface upon which service developers can rely when designing new services (or enhancements/variants of already existing ones).

The aim of OSA is to provide a standardised, extensible and scalable interface that allows for inclusion of new functionality in the network in future releases with a minimum impact on the applications using the OSA interface.

Network functionality offered to applications is defined as a set of Service Capability Features (SCFs) in the OSA API, which are supported by different Service Capability Servers (SCS). These SCFs provide access to the network capabilities on which the application developers can rely when designing new applications (or enhancements/variants of already existing ones). The different features of the different SCSs can be combined as appropriate. The exact addressing (parameters, type and error values) of these features is described in stage 3 descriptions. These descriptions (defined using UML, and in three realizations - OMG Interface Description Language™, Java and WSDL) are open and accessible to application developers, who can design services in any programming language, while the underlying core network functions use their specific protocols.

The standardised OSA APIs are secure, independent of vendor specific solutions and independent of programming languages, operating systems etc used in the service capabilities. The OSA APIs are independent of the location within the home environment where service capabilities are implemented, and independent of supported service capabilities in the network. Furthermore, an architecture with open interfaces allows for application developers to rapidly design new and innovative applications.

5.1 Overview of the Open Service Access

The Open Service Access consists of three parts:

- **Applications:** e.g. VPN, conferencing, location based applications. These applications are implemented in one or more Application Servers;

- **Framework:** providing applications with basic mechanisms that enable them to make use of the service capabilities in the network. Examples of framework functions are Authentication, and Registration and Discovery. Service Capability Features made available to applications are Registered in the Framework. Before an application can use the network functionality made available through Service Capability Features, authentication between the application and framework is needed. After authentication, the discovery function enables the application to find out which network service capability features are provided by the Service Capability Servers. The network service capability features are accessed by the methods defined in the OSA interfaces;

- **Service Capability Servers:** providing the applications with service capability features, which are abstractions from underlying network functionality. Examples of service capability features offered by the Service Capability Servers are Call Control and User Location. Similar service capability features may possibly be provided by more than one Service Capability Server. For example, Call Control functionality might be provided by SCSs on top of CAMEL and MExE.

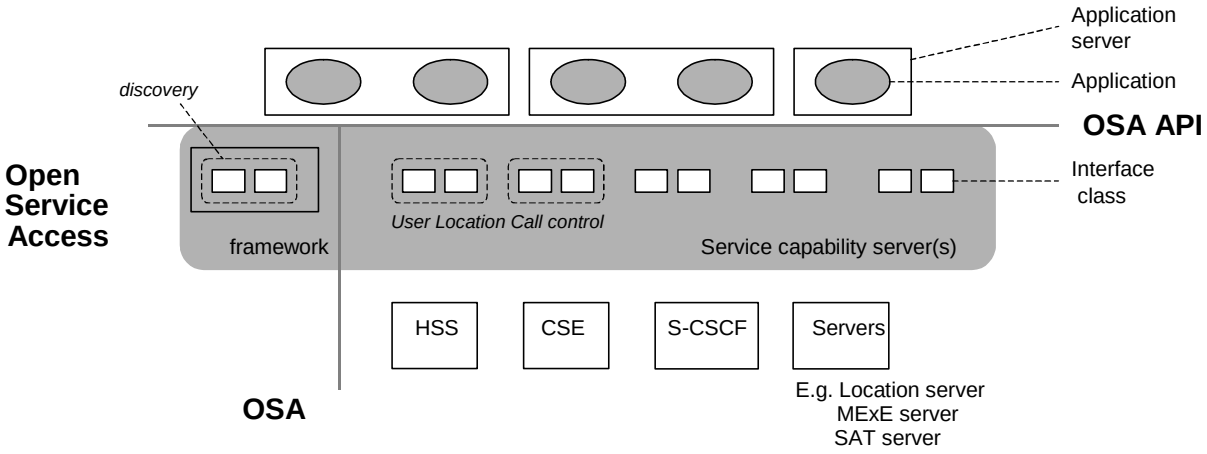

Figure 1: Overview of Open Service Access

The present document, together with the associated stage 3 specification, defines the OSA APIs. OSA does not mandate any specific platform or programming language.

The Service Capability Servers that provide the OSA interfaces are functional entities that can be distributed across one or more physical nodes. For example, the User Location interfaces and Call Control interfaces might be implemented on a single physical entity or distributed across different physical entities. Furthermore, a service capability server can be implemented on the same physical node as a network functional entity or in a separate physical node. For example, Call Control interfaces might be implemented on the same physical entity as the CAMEL protocol stack (i.e. in the CSE) or on a different physical entity.

Several options exist:

Option 1

The OSA interfaces are implemented in one or more physical entity, but separate from the physical network entities. Figure 2 shows the case where the OSA interfaces are implemented in one physical entity, called "gateway" in the figure. Figure 3 shows the case where the SCSs are distributed across several "gateways".

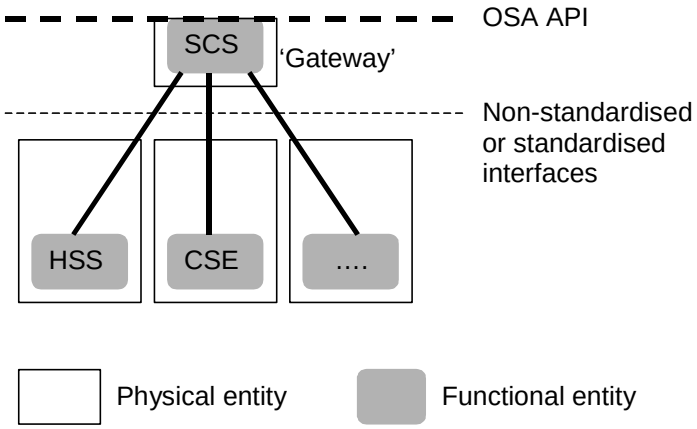

Figure 2: SCSs and network functional entities implemented in separate physical entities

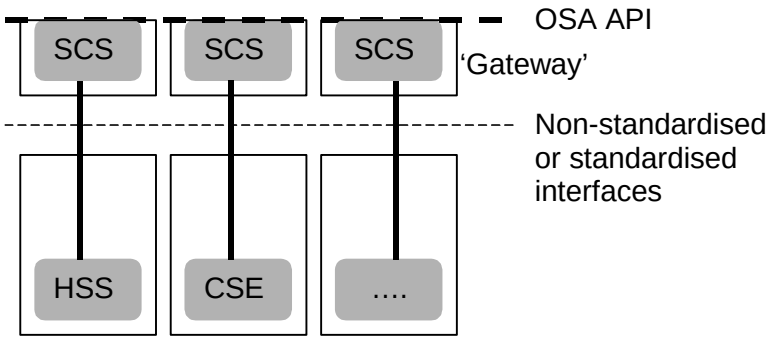

Figure 3: SCSs and network functional entities implemented in separate physical entities, SCSs distributed across several 'gateways'

Option 2

The OSA interfaces are implemented in the same physical entities as the traditional network entities (e.g. HSS, CSE), see figure 4.

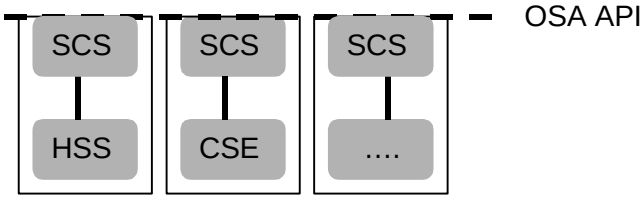

Figure 4: SCSs and network functional entities implemented in same physical entities

Option 3

Option 3 is the combination of option 1 and option 2, i.e. a hybrid solution.

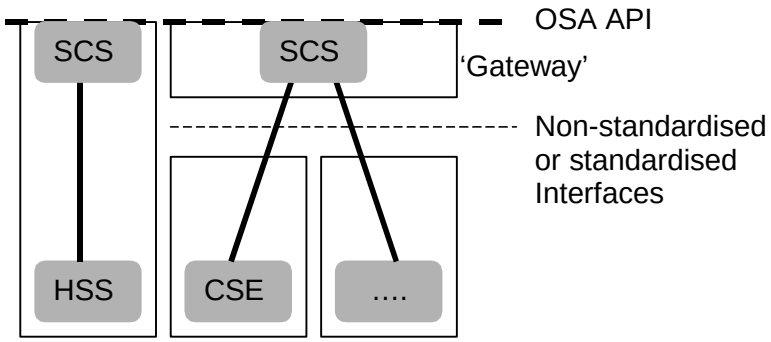

Figure 5: Hybrid implementation (combination of option 1 and 2)

It shall be noted that in all cases there is only one Framework. This framework may reside within one of the physical entities containing an SCS or in a separate physical entity.

From the application point of view, it shall make no difference which implementation option is chosen, i.e. in all cases the same network functionality is perceived by the application. The applications shall always be provided with the same set of interfaces and a common access to framework and service capability feature interfaces. It is the framework that will provide the applications with an overview of available service capability features and how to make use of them.

The implementation of applications that make use of the OSA interfaces is not constrained or limited to a particular programming language; middleware choice or physical architecture by the OSA interfaces themselves. Applications may be realised as functional entities that can be distributed across one or more physical entities, e.g. platforms, processes etc. For example, a logical application providing call routing capability may be realised as a number of discrete physical applications in order to support differentiated application behaviour; application scalability or application resilience. However, the logical relationship established between applications, framework and SCFs, must always be maintained such that the integrity, security and use of the OSA interfaces remains consistent.

A range of options exist that allows applications to be realised as a number of physical entities in a manner that maintains the integrity of the OSA architecture:

Option 1

The application is produced as a single physical entity.

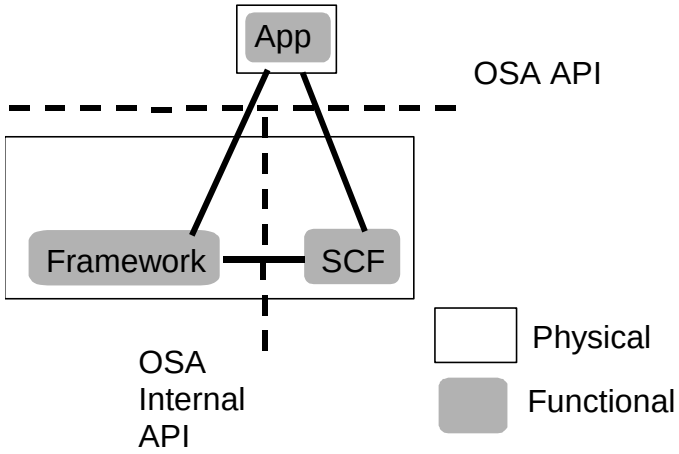

Figure 6: Application function implemented as a single physical entity

Option 2

The functionality of the application is realised using multiple physical entities. Figure 7 and Figure 8 represent alternative solutions where the same application obtains a reference to the same SCF interface. In Figure 7, each physical application uses a unique set of interfaces supplied by the Framework for each application access session, whereas in Figure 8 the Framework shares the same interfaces. In either case the Framework resolves the application sessions to a single FW-Svc session, and a single service manager is provided to the functional application.

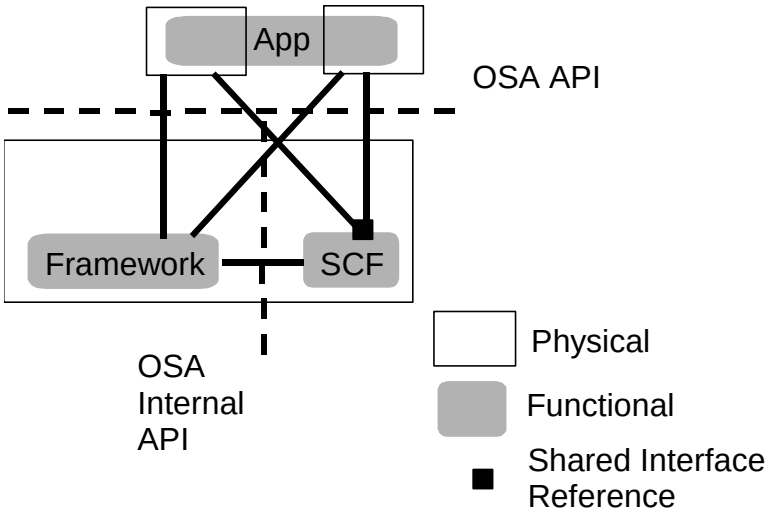

Figure 7: Application function implemented in several physical entities (unique Framework Interface References for each application entity)

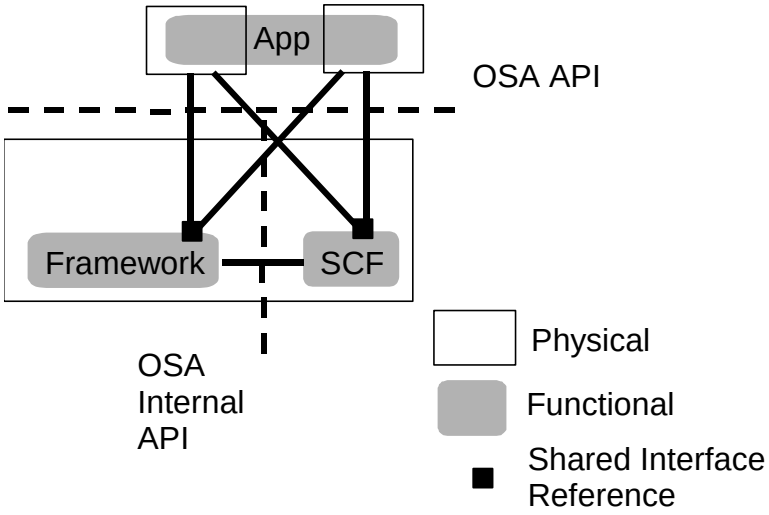

Figure 8: Application function implemented in several physical entities (sharing common Framework Interface References)

The application examples above depict an OSA Gateway with both Framework and SCF functions supported on a single physical entity. Further Gateway architectures in which the Framework and SCFs are supported in multiple discrete physical entities are also possible.

5.2 Basic mechanisms in the Open Service Access

This subclause explains which basic mechanisms are executed in OSA prior to offering and activating applications.

Some of the mechanisms are applied only once (e.g. establishment of service agreement), others are applied each time a user subscription is made to an application (e.g. enabling the call attempt event for a new user).

Basic mechanisms between Application and Framework:

- **Authentication:** Once an off-line service agreement exists, the application can access the authentication function. The authentication model of OSA is a peer-to-peer model. The application must authenticate the framework and vice versa. The application must be authenticated before it is allowed to use any other OSA function.

- **Authorisation:** Authorisation is distinguished from authentication in that authorisation is the action of determining what a previously authenticated application is allowed to do. Authentication must precede authorisation. Once authenticated, an application is authorised to access certain service capability features.

- **Discovery of framework functions and network service capability features:** After successful authentication, applications can obtain available framework functions and use the discovery function to obtain information on authorised network service capability features. The Discovery function can be used at any time after successful authentication.

- **Establishment of service agreement:** Before any application can interact with a network service capability feature, a service agreement must be established. A service agreement may consist of an off-line (e.g. by physically exchanging documents) and an on-line part. The application has to sign the on-line part of the service agreement before it is allowed to access any network service capability feature.

- **Access to network service capability features:** The framework must provide access control functions to authorise the access to service capability features or service data for any API method from an application, with the specified security level, context, domain, etc.

Basic mechanism between Framework and Service Capability Server:

- **Registering of network service capability features**. SCFs offered by a Service Capability Server can be registered at the Framework. In this way the Framework can inform the Applications upon request about available service capability features (Discovery). For example, this mechanism is applied when installing or upgrading a Service Capability Server.

Basic mechanisms between Application Server and Service Capability Server:

- **Request of event notifications**. This mechanism is applied when a user has subscribed to an application and that application needs to be invoked upon receipt of events from the network related to the user. For example, when a user subscribes to an incoming call screening application, the application needs to be invoked when the user receives a call. It will therefore request to be notified when a call setup is performed, with the user number as Called Party Number.

5.3 Handling of end-user related security

Once OSA basic mechanisms have ensured that an application has been authenticated and authorised to use network service capability features, it is important to also handle end-user related security aspects. These aspects consist of the following.

- End-user authorisation to applications, limiting the access of end-users to the applications they are subscribed to.

- Application authorisation to end-users, limiting the usage by applications of network capabilities to authorised (i.e. subscribed) end-users.

- End-user's privacy, allowing the user to set privacy options.

These aspects are addressed in the following subclauses.

5.3.1 End-user authorisation to applications

An end-user is authorised to use an application only when he or she is subscribed to it.

In the case where the end-user has subscribed to the application before the application accesses the network SCFs, then the subscription is part of the Service Level Agreement signed between the HE and the HE-VASP.

After the application has been granted access to network SCFs, subscriptions are controlled by the Home Environment. Depending on the identity of an authenticated and authorised end-user, the Home Environment may use any relevant policy to define and possibly restrict the list of services to which a particular end-user can subscribe. At any time, the Home Environment may decide, unilaterally or after agreement with the HE-VASP, to cancel a particular subscription.

Service subscription and activation information need to be shared between the Home Environment and the HE-VASP, so that the HE-VASP knows which end-users are entitled to use its services. Appropriate online and/or offline synchronisation mechanisms (e.g. SLA re-negotiation) can be used between the HE and the HE-VASP, which are not specified in OSA release 5.

End-to-end interaction between a subscribed end-user and an application may require the usage of appropriate authentication and authorisation mechanisms between the two, which are independent from the OSA API, and therefore not in the scope of OSA standardisation.

5.3.2 Application authorisation to end-users

The Home Environment is entitled to provide service capabilities to an application with regard to a specific end-user if the following conditions are met:

1) the end-user is subscribed to the application;

2) the end-user has activated the application;

3) the usage of this network service capability does not violate the end-users privacy settings (see next subclause).

The service capability server ensures that the above conditions are met whenever an application attempts to use a service capability feature for a given end-user, and to respond to the application accordingly, possibly using relevant error parameters). The mechanism used by the SCS to ensure this is internal to the HE (e.g. access to user profile) and is not standardised in OSA release 5.

5.3.3 End-user's privacy

The Home Environment may permit an end-user to set privacy options. For instance, it may permit the end-user to decide whether his or her location may be provided to 3rd parties, or whether he or she accepts information to be pushed to his or her terminal. Such privacy settings may have an impact on the ability of the network to provide service capability features to applications (e.g. user location, user interaction). Thus, even if an application is authorised to use an SCF and the end-user is subscribed to this application and this application is activated, privacy settings may still prevent the HE from fulfilling an application request.

The service capability server ensures that a given application request does not violate an end-users privacy settings or that the application has relevant privileges to override them (e.g. for emergency reasons). The mechanism used by the SCS to ensure this is internal to the HE and is not standardised in OSA release 5.

6 Framework service capability features

6 Framework

6.1 Trust and Security Management Functions

The Trust and Security Management functions provide:

- the first point of contact for an application to access a network via the OSA APIs;

- the authentication methods for the application and network to perform a mutual authentication;
- the application with the ability to select a service capability feature to make use of;
- the application with a portal to access other framework functions.

The process by which the application accesses the network via the OSA APIs has been separated into 3 stages, each supported by a different framework function:

1) Initial Contact with the framework;
2) Authentication to the framework;
3) Access to framework functions and service capability features.

6.1.1 Initial Contact

The application gains a reference to the OSA Initial Contact function for the network that they wish to access. This may be gained through a URL, a Naming or Trading Service or an equivalent service, a *stringified* object reference, etc. At this stage, the application has no guarantee that this is a reference to the network, so it this reference to initiate an authentication process.

Initial Contact supports a particular method to allow the authentication process to take place (using the Authentication SCF defined in subclause 6.1.2). This method must be the first invoked by the application. Invocations of other methods will fail until authentication has been successfully completed.

Once the application has authenticated with the network, it can gain access to other framework functions and service capability features. This is done by invoking a method, by which the application requests a certain type of access service capability feature. The OSA Access function is defined in subclause 6.1.3.

6.1.2 Authentication

Once the application has made initial contact with the network, and any time during their interactions, authentication of the application and network may be required.

The OSA APIs supports multiple authentication techniques. The procedure used to select an appropriate technique for a given situation is described below. The authentication mechanisms may be supported by cryptographic processes to provide confidentiality, and by digital signatures to ensure integrity. The inclusion of cryptographic processes and digital signatures in the authentication procedure depends on the type of authentication technique selected. In some cases strong authentication may need to be enforced by the network to prevent misuse of resources. In addition it may be necessary to define the minimum encryption key length that can be used to ensure a high degree of confidentiality.

Editor's note: Chelo: this needs to be double checked to align with last security updates

The application must authenticate with the framework before it is able to use any of the other interfaces supported by the framework. Invocations on other interfaces will fail until authentication has been successfully completed.

6.1.3 OSA Access

This function supports stage 1 requirements related to authorization and service registration.

During an authenticated session accessing the Framework, the application will be able to select and access an instance of a framework function or network service capability feature.

In order to use network SCFs, the application must first be authorised to do so by establishing a service agreement with the network. The application uses the discovery SCF to retrieve the ID of the SCF they wish to use. They may then check that they are authorised to use the SCF. The network is informed that the application wishes to use the SCF. Finally, a service agreement is signed digitally between the two parties.

Establishing a service agreement is a business level transaction, which requires the HE-VASP that owns the application to agree terms for the use of an SCF with the Home Environment. Service agreements can be reached using either off-line or on-line mechanisms. Off-line agreements will be reached outside of the scope of OSA interactions, and so are

not described here. However, applications can make use of service agreements that are made off-line. Some Home Environments may only offer off-line mechanisms to reach service agreements.

After a service agreement has been established between the application and the Home Environment domains, the application will be able to make use of this agreement to access the SCF.

6.2 Discovery

Before a network SCF can be discovered, the application must know what "types" of SCFs are supported by the Framework and what "properties" are applicable to each SCF type. Once the HE-VASP finds out the desired set of SCFs supported by the network, it subscribes (a sub-set of) these SCFs using the Subscription framework function. The HE-VASP (or the applications in its domain) can find out the set of SCFs available to it (i.e., the SCFs that it can use).

6.3 Integrity Management functions

6.3.1 Load Manager

The Load Manager function permits to manage the load on both the application and network sides.

The framework should allow the load to be distributed across multiple machines and across multiple component processes, according to a load balancing policy. The separation of the load balancing mechanism and load balancing policy ensures the flexibility of the load balancing functionality. The load balancing policy identifies what load balancing rules the framework should follow for the specific application. It might specify what action the framework should take as the congestion level changes. For example, some real-time critical applications will want to make sure continuous service is maintained, below a given congestion level, at all costs, whereas other applications will be satisfied with disconnecting and trying again later if the congestion level rises. Clearly, the load balancing policy is related to the QoS level to which the application is subscribed.

6.3.2 Fault Manager

The Fault Manager function is used by the application to inform the framework of events which affect the integrity of the framework and SCFs, and to request information about the integrity of the system.

6.3.3 Heartbeat Management

The Heartbeat Management function allows the initialisation of a heartbeat supervision of the client application. In case of SCF supervision, it is the framework's responsibility to check the health status of the respective SCF.

Since the OSA APIs are inherently synchronous, the heartbeats themselves are synchronous for efficiency reasons.

6.3.4 OAM

The OAM function is used to query the system date and time. The application and the framework can synchronise the date and time to a certain extent. Accurate time synchronisation is outside the scope of the OSA APIs.

6.5

Editor’s note: Chelo: this section is deleted because PM is not part of the Framework. But we still need a section on PM somewhere in the stage 2 document.

7 Network service capability features

Network service capability features are provided to the applications by service capability servers to enable access to network resources.

7.1 Call Control

The Call Control SCF supports stage 1 requirements related to CS call control, IMS session control and call/session charging.

The Call Control network service capability feature supports the following functionality:

1) management function for call/session-related issues, e.g. enable or disable call/session-related event notifications.

2) call/session control, e.g. route, disconnect.

7.1.1 Mapping of OSA APIs in CS domain

In the CS domain the OSA Call Control SCF may be mapped to CAP and MAP protocols.

7.1.2 Mapping of OSA APIs in IMS

OSA SCS is one of the three types of "application servers" communicating with S-CSCF in the IMS [8]. OSA Application Server is connected by OSA API to OSA Service Capability Server (SCS) that is connected through ISC interface to S-CSCF and through Sh interface to HSS. ISC interface is based on the use of SIP protocol, see TS 23.218 [10]. The details and functionality of the Sh interface are for further study in TS 23.228 [8].

OSA functions for IMS session control are supported by the following entities:

- The Serving-CSCF (S-CSCF), which performs session control services for an originating or terminating party.
- The Media Resource Function (MRF), which performs conference control and media control functions for multiparty multimedia sessions.

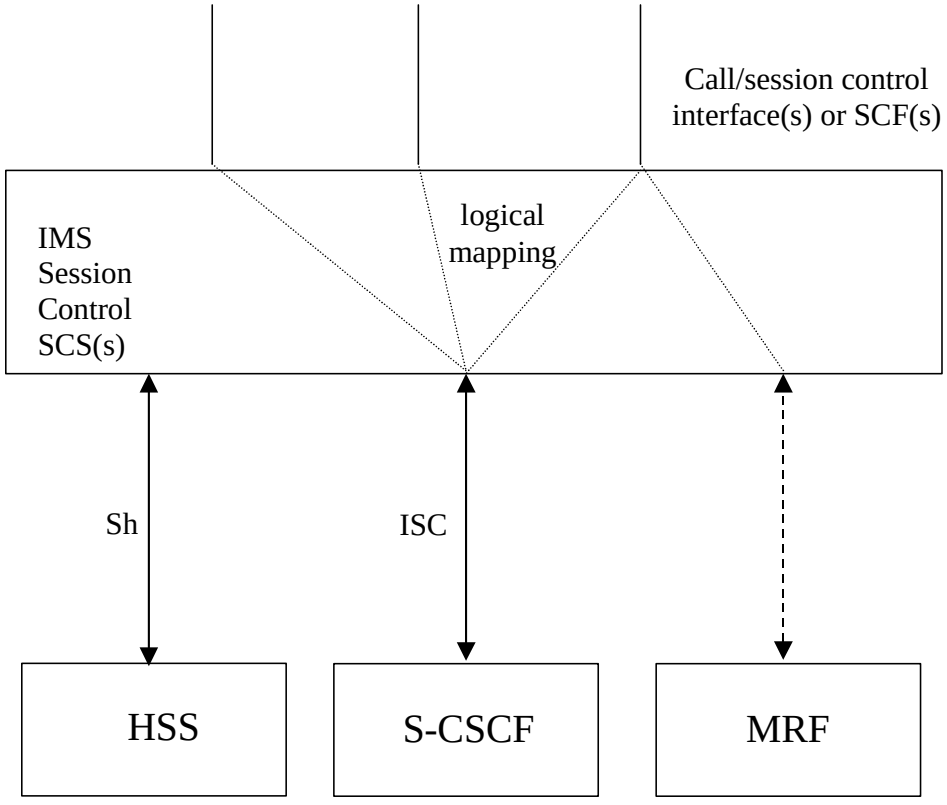

Figure 6: Mapping of OSA IMS session control on the IMS

The stage 3 specification of OSA for IMS session control shall take into account this distribution of responsibilities between the S-CSCF and the MRF, by specifying specific OSA SCF(s) or interface(s) for the S-CSCF, and specific OSA SCF(s) or interface(s) for the MRF. This is to permit clear mapping of OSA on the corresponding entities' functionality, as well as allowing multivendorship.

IMS session control SCF(s) or interface(s) applicable to the S-CSCF shall be mapped onto the IMS Service Control (ISC).

The MRF is either controlled by the OSA SCS by (1) using SIP 3rd party call control via the S-CSCF or (2) using a direct interface to the MRF. These two options are still under investigation.

TS 22.127 [3] classifies IMS session control functions as follows:

- session control requirements;
- media control requirements;
- information requirements.

IMS session control SCF(s) or interface(s) applicable to the S-CSCF shall support session control and information requirements applicable to the originating or terminating party of 2-party session.

These OSA SCF(s) or interface(s) and their implementation shall take into account that the S-CSCF:

- Is an entity that is dynamically associated to the user when she registers to the IMS;
- May behave as a SIP registrar, proxy server, and user agent;
- May receive the request from a session party to initiate an ad-hoc conference (to be associated to an MRF);
- May generate CDRs.

IMS session control SCF(s) or interface(s) applicable to the MRF shall support all session control, media control, and information requirements.

These OSA SCF(s) or interface(s) and their implementations shall take into account that the MRF:

- May support both ad-hoc and pre-arranged conferences;
- Controls media stream resources associated to the conference;
- Behaves as a SIP user agent with regard to each party of the conference;
- Supports conference booking and floor control;
- Is divided into Media Resource Function Controller (MRFC) and Media Resource Function Protocol (MRFP), which interface via an H.248 fully compliant interface.

7.2 Data Session Control

The Data Session Control SCF supports stage 1 requirements related to PS call control.

The Data Session Control network service capability feature supports the following functionality:

1) management functions for data session related issues, e.g. enable or disable data session-related event notifications
2) session control, e.g. route, disconnect.

7.3 Mobility

The Mobility SCF addresses stage 1 requirements for user location and user statusbased on network-related information.

The Mobility SCF provides terminal location information and general terminal status monitoring. The following information is reported when requested provided that the network is able to support the corresponding capability:

- user whom the report concerns;
- VLR number;
- Cell Global Identification or Location Area Identification;
- location number (network specific, refer to ITU-T Q.763);
- geographical location (e.g. in terms of universal latitude and longitude co-ordinates);
- accuracy (value depending on local regulatory requirements and level of support in serving/home networks; note that the accuracy of the serving network might differ from that in the home environment);
- age of location information (last known date/time made available in GMT);
- status of the user's terminal.

Connection of an external LCS client by means of OSA API to GMLC is shown in TS 23.271 [13], figure 6.1: General arrangement of LCS.

An application uses the Mobility SCF to perform the following:

- user location requests;
- requests for starting (or stopping) the generation by the network of periodic user location reports;
- requests for starting (or stopping) the generation by the network of user location reports based on location changes;
- report of location information;
- notification of location update.

The application can also for each user start/stop receipt of notifications and modify the required accuracy by selecting another option from the network provided options.

7.4 Terminal Capabilities

The Terminal Capabilities SCF provides applications information about the terminal capabilities of the user. It shall be possible for an application to request Terminal Capabilities as defined by MExE (MExE User Profile) [1]. The terminal capabilities are provided by a MExE compliant terminal to the MExE Service Environment either on request or by the terminal itself.

Terminal Capabilities are available only after a capability negotiation has previously taken place between the user´s MExE terminal and the MExE Service environment as specified in [1].

NOTE: For Release 5 only WAP and MExE devices can supply terminal capabilities.

7.5 User Interaction

The User Interaction SCFs support stage 1 requirements for information transfer.

There are two user interaction SCFs:

- Generic User Interaction: used by applications to interact with end users;
- Call User Interaction: used by applications to interact with end users participating to a call.

7.6 Charging

The Charging SCF addresses stage 1 requirements for charging related to service usage (and not call/session control).

This SCF permits an application to access subscriber accounts maintained by the network and charge subscribers for service usage.

Provided, that these functions are supported by the underlying network an application providing a service to the subscriber can use the Charging SCF to:

- Check, if – for the service to be provided by the application – the charge is covered by the subscribers account or credit limit.
- Reserve – for the service to be provided by the application – a charge in the subscribers account, that can be deduced from the account after service delivery.
- Deduct an amount from the subscriber's account.
- Release a reservation acquired earlier.
- Add non-monetary units to a subscriber's account.
- Deduct non-monetary units from a subscriber's account.

Reverse a completed charge transaction, e.g. after repudiation.

7.7 Account Management

The Account Management SCF addresses stage 1 requirements related to the features to monitor subscriber's account:

- retrieval of transaction history for a certain subscriber's account;
- query of the balance of the account of one or several subscriber's;
- request of notifications on certain criteria for one or several subscribers.

7.8 Presence

The Presence SCF addresses stage 1 requirements on presence related capability functions.

OSA shall allow an application access to presence capabilities within the network. Presence related information may be requested or supplied by an OSA application and may include, but not be limited to presence information pertaining to the presence service or user availability. Presence information, i.e. a set of attributes characterising current properties of a presentity, is described in TS 22.141 [11].

An OSA application shall be able:

- to register as a watcher, to request a presentity's presence information and to be notified of changes in the presence information.
- to register as a presentity, to publish presence information, to retrieve watcher information and to manage related parameters (e.g. access rules). Presence management may include the setting of user preferences, the update of access rules…etc.

7.8.1 Mapping of OSA APIs

The Presence OSA APIs can be mapped to reference points Peu and Pw of the Presence Server.

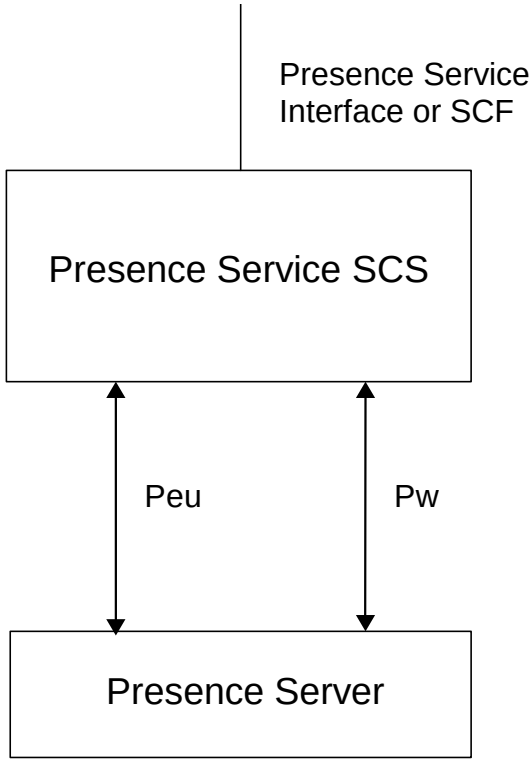

Figure 7: Mapping of OSA Presence APIs

Reference points Peu (i.e. between a Presence User Agent and the Presence Server) and Pw (i.e. between Watcher Applications and the Presence Server) are described in TS 23.141 [14].

7.9 Multi Media Messaging (MMM)

The Multi Media Messaging SCF addresses the stage 1 requirement for multimedia messaging.

The Multi Media Messaging SCF allows applications to:

- send and receive messages both within and outside the context of a session (for session-based and single-shot messaging respectively)
- put messages in the mailbox for storage or for sending by the messaging system (with a copy in the mailbox)
- cancel a message previously sent or query the status of a message previously sent
- manipulate folders and messages in the mailbox (e.g. copy, move, delete)
- list messages in the mailbox and retrieve complete messages, message headers, message body or parts of the message body

7.9.1 Mapping of OSA APIs in MMM

The Messaging SCF can interface to various messaging network elements or contain those network elements. Examples of network elements are SMS-C, MMS-C, WAP Push Proxy or an e-mail server. OSA Multi Media Messaging SCF does not mandate what network protocols to use to interface to those network elements. However, a typical example of the interface used to interface to MMS-C is MM7 [15].

8 OSA Internal API

The OSA internal API between framework and service capability servers (SCSs) supports registering of network service capability features (SCFs), permits the framework to retrieve a network SCF manager interface when an application is granted access to a network SCF, and enables integrity management by means of load management, heartbeat management and fault management.

8.1 OSA Access and Discovery

To support registration, the OSA Access and Discovery interfaces shall be supported at the OSA internal API.

8.2 Registration of network service capability features at the framework

The Framework needs to know the Service Capability Features provided by the SCSs, in order to make them available to applications. For this purpose network service capability features have to be registered with the Framework, and they need to be registered in such a way that applications can discover them.

NOTE: Framework and Service Capability Servers are located within the same trusted domain. Therefore no authentication mechanisms are required between them.

8.2.1 Service Registration

The Service Registration interface provides the methods used for the registration of network SCFs at the framework.

8.2.2 Service Factory

The Service Factory interface allows the framework to get access to a manager interface of a network SCF. It is used, in order to return an SCF manager interface reference to the application. Each SCF has a manager interface that is the initial point of contact for the network SCF.

8.3 Integrity Management

Integrity Management interfaces allow the framework to perform load management, heartbeat management and fault management.

8.3.1 Load Management

Load management enables the framework to manage the load allowing it to be distributed across multiple SCSs by means of load balancing.

8.3.2 Heartbeat Management

Heartbeat management allows the initialisation of a heartbeat supervision of a network SCF.

8.3.3 Fault Management

Fault management allows to inform the framework of events which affect the integrity of the framework and network SCFs, and to request information about the integrity of the system.

9 Parlay X Web Services: OSA at a higher level of abstraction

9.1 General

The general architecture of a solution including Web Services and/or OSA links in deployment allows a number of deployment configurations. These configurations are derivatives of a basic architecture model, enabling a variety of deployment options.

A typical Parlay X Web Services deployment model is shown in the Figure 9.1. This model shows the publication of Parlay X Web Services through a registry, making those Web Services available for discovery, and for applications to use Web Services access methods to interact with the Gateway, where the Web Service interfaces are implemented.

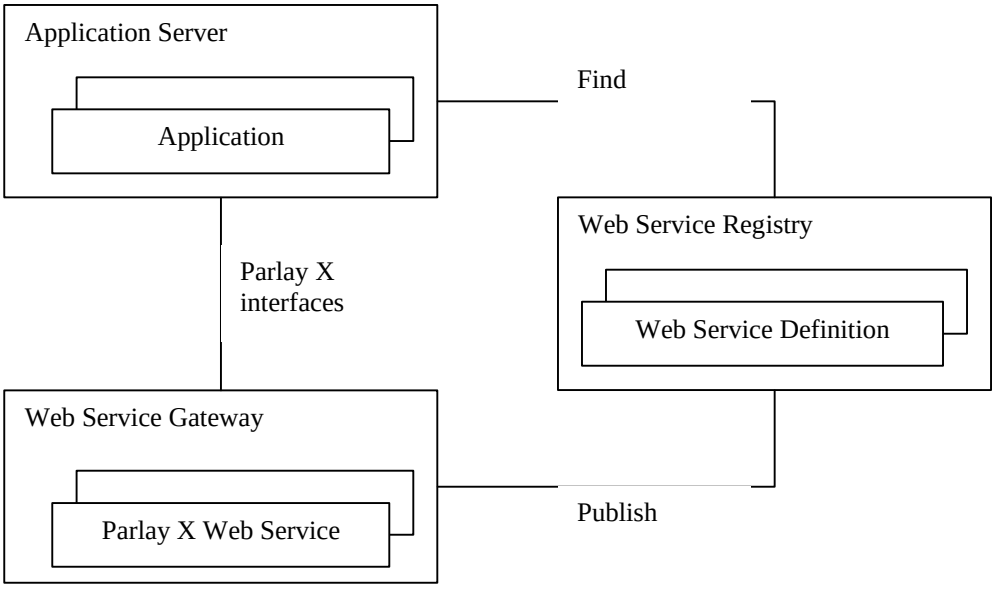

Figure 9.1 : Parlay X Web Services deployment model

Interfaces to the Web Services Registry are drawn in Figure 9.1 for consistency with Web Service architectures, but they are not in the scope of the Parlay X Web Services.

This architecture may be combined with existing OSA deployment configurations, providing the overall architecture as illustrated in Figure 9.2.

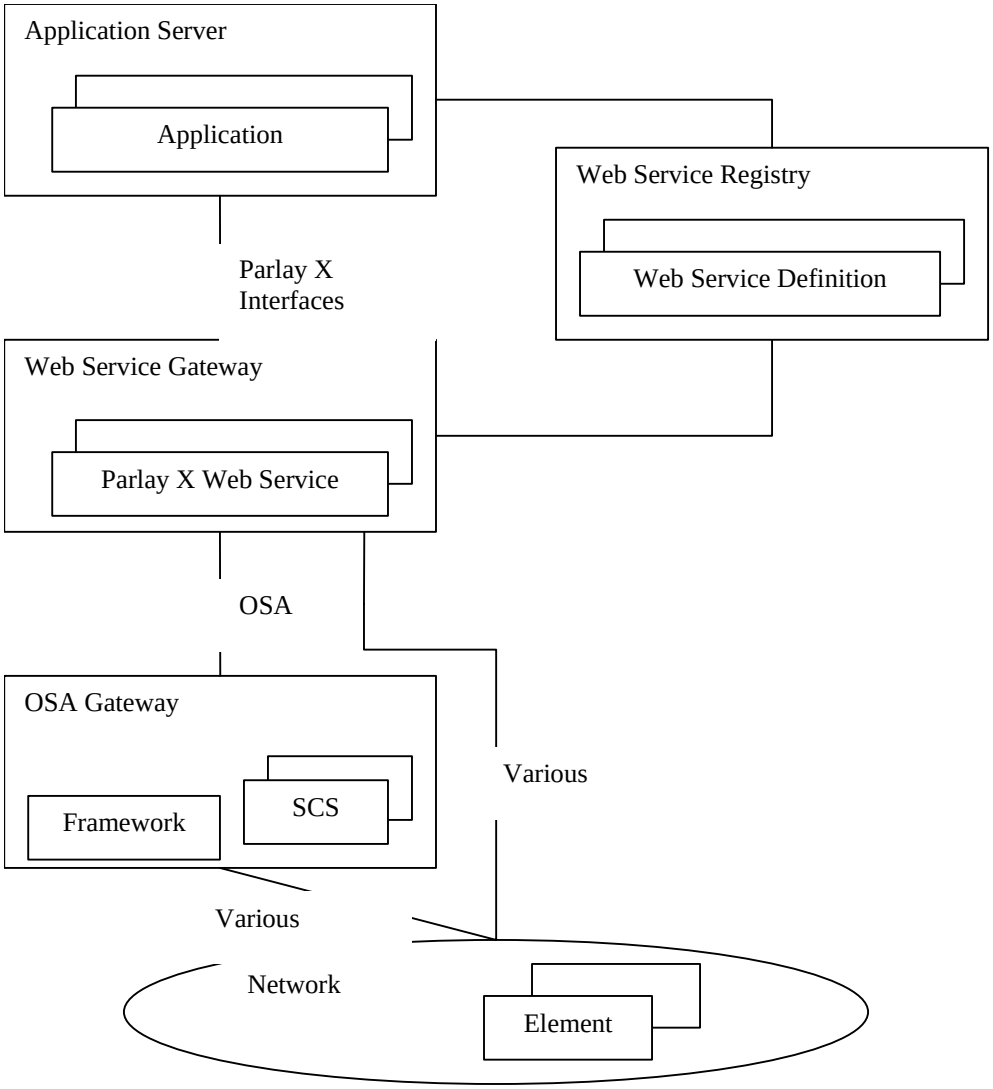

Figure 9.2 : The overall Parlay X Web Services architecture

9.1.1 Deployment Scenario A: Web Services to OSA

This scenario addresses solutions that combine Web Services interfaces facing the exterior of the network with OSA interfaces facing the interior of the network.

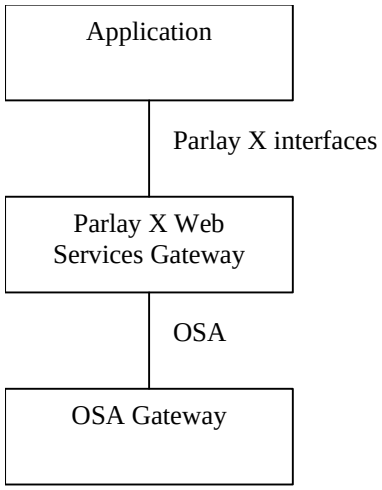

Figure 9.3 Deployment Scenario A: Web Services to OSA

Environment Description

The Application will utilize Web Services to discover and interact with the network, and will not have visibility to the OSA implementation behind the Parlay X Web Services Gateway. The Parlay X Web Services Gateway attaches to the OSA Gateway through an OSA interface. The information published to the Web Services Registry provides the Application with the connection information required to connect with the Parlay X Web Services Gateway.

9.1.2 Deployment Scenario B: Web Services to Network Element

This scenario addresses solutions that combine Web Services interfaces facing the exterior of the network with network element specific interfaces facing the interior of the network.

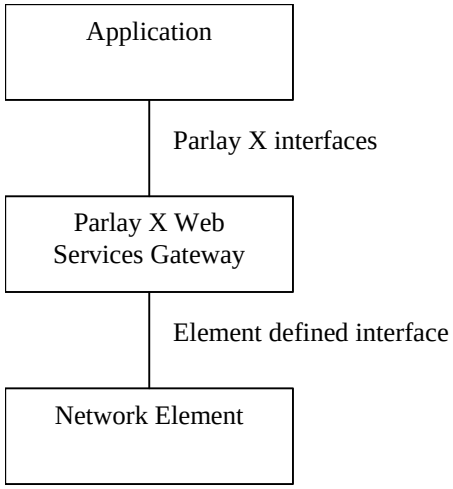

Figure 9.4 : Deployment Scenario B: Web Services to Network Element

Environment Description

The Application will utilize Web Services to discover and interact with the network, and will not have visibility to the implementation behind the Parlay X Web Services Gateway. The Parlay X Web Services Gateway attaches to the Network Element through an interface defined by the Network Element. These interfaces (i.e. Element defined interface) are not in the scope of this specification. The information published to the Web Services Registry provides the Application with the connection information required to connect with the Parlay X Web Services Gateway.

The next clauses describe the Web Services supported by OSA Release 6.

9.2 Third Party Call

This Web Service supports the functionality to create and manage a call initiated by an application (third party call). Using the Third Party Call Web Service, applications can invoke call handling functions without detailed telecommunication knowledge. The functionality provided is:

- Make a call which sets up a call between two addresses
- Get call information gives information about how the call progressed in the network
- End call will cease the call
- Cancel Call Request allows the network to prevent call setup before completion

The third party application can suggest a charge against the receiving subscriber's account.

9.3 Network-Initiated Third Party Call

These functions are for notification or even handling of calls initiated by a subscriber in the network. A (third party) application can determine how the call should be treated. The overall scope of this Web service is to provide simple call

control related functions to application developers. Using the Network-Initiated Third Party Call Web Service, application developers can apply simple logic to network-initiated calls without specific Telco knowledge. The Web Services allow the application to handle the following conditions occurring in the set-up of a call:

- Destination busy
- Address is not Reachable
- Destination is not answering
- A specific number has been called by subscriber

The third party application can suggest a charge against the subscriber's account on which behalf these services were rendered.

9.4 SMS

The overall scope of this Web Service is to provide to applications the means to handle SMS in a simple way. For receiving a message from the network, the application may use either polling or notification mechanisms. The notification mechanism is more common: network-initiated messages are sent to autonomous application-side web services. Both mechanisms are supported, but the provisioning of the notification-related criteria is not specified. Services are specified to enable an application to:

- Send any SMS
- Send a logo embodied in an SMS
- Send a ringtone embodied in an SMS
- Retrieve the delivery status of an SMS
- Request to be notified of received SMSs
- Request to be notified of delivery receipt of a sent SMS
- Retrieve SMS messages sent to an address

For reasons of efficiency, SMSs, whether their payload is text, a logo or ringtone, can be sent to groups of recipients. In order for such groups to be reusable; a group with recipients is uniquely identified and addition/deletion of group members is allowed.

The third party application can suggest a charge against the receiving subscriber's account.

9.5 Multimedia Message

This is additional to the SMS web services and handles more general messaging case. Multimedia Message services provide generic messaging features (including SMS) to send and receive messages. For receiving a message from the network the application may use either polling or notification mechanisms. Network-initiated messages are sent to autonomous application-side web services. The following functions are supported:

- Send a message to an address
- Retrieve the delivery status of a message
- Retrieve by polling for received messages
- Retrieve message parts by URI references
- Retrieve whole messages as SOAP attachments
- Notification to the application that a message has been received for a specific address.

For reasons of efficiency, MMSes can be sent to groups of recipients. In order for such groups to be reusable; a group with recipients is uniquely identified and addition/deletion of group members is allowed.

The third party application can suggest a charge against the receiving subscriber's account.

9.6 Payment

The Payment Web Services support payment reservation, pre-paid payments, and post-paid payments. They support charging of both volume and currency amounts, a conversion function, and a settlement function in case of a financially resolved dispute. The functions supported enable the application to:

- Charge/refund an account by a currency amount
- Charge/refund an account by volume (e.g. minutes)
- Calculate a currency amount from a volume for a specific account
- Reserve a currency amount on an account
- Charge a prior reservation to the account
- Release a reservation by returning to an account the amount remaining in a reservation
- Reserve a volume amount of an account

9.7 Account Management

The Parlay X Account Management supports account querying, direct recharging and recharging through vouchers. The application can manage various aspects of an account using the functionality to:

Return the currency balance on an account

- Request what date the credit on an account is due to expire
- Update the currency/account balance on an account
- Return the transaction history on an account

9.8 Terminal Status

The Parlay X Terminal Status Web Service is used for getting terminal status information. The functionality supported is simple:

- Requests a subscriber’s terminal's status subject to the subscriber's policies.
- Request to be notified of terminal status change.

For reasons of efficiency, Terminal status can be retrieved from groups of subscribers. In order for such groups to be reusable; a group with recipients is uniquely identified and addition/deletion of group members is allowed.

9.9 Terminal Location

The Parlay X Terminal Location Web Service is used for getting location information, it does not require specific telecommunication skills, but some knowledge of location co-ordinates is required. One service is specified to:

- Request the location of one subscriber’s terminal subject to subscriber’s privacy policies.
- Request to be notified of terminal location change.

For reasons of efficiency, Terminal location can be retrieved from groups of subscribers. In order for such groups to be reusable; a group with recipients is uniquely identified and addition/deletion of group members is allowed.

9.10 Audio Call

The Parlay X Audio Call Web Service supports the creation of a call with associated audio content that is used when the call is completed. The third party application can suggest a charge against the receiving subscriber's account. The function supported is:

- Play audio, setting up a call and using provided audio content to communicate with the callee
- Retrieval of status
- Application ability to end the call

9.11 Call Handling

The Parlay X Call Handling Web Service enables call handling rules to be provisioned, allowing third party applications to specify how to handle calls for addresses without requiring the application to handle network interactions, simplifying access to this capability for application developers. The function supported is:

- Provision rules for accepting, blocking, forwarding and answering calls for an address
- Query rules associated with an address
- Remove rule processing for an address or group of addresses

For reasons of efficiency, call handling rules can be provisioned for groups of addresses. In order for such groups to be reusable; a group with recipients is uniquely identified and addition/deletion of group members is allowed.

9.12 Multimedia Conferencing

The Multimedia Conferencing is a simple Web Service that allows the creation of a multimedia conference and the dynamic management of the participants and the media involved. The interface can be used by an application for creating a multimedia conference call and for dynamically managing the participants and the media involved in the call:

- Create a conference without participants
- Query the conference status/ participants
- Add/delete a participant
- Add/delete media per participant
- Query participant status
- End the conference

9.13 Presence

The Parlay X Presence Web Service allows for presence information to be obtained about one or more users and to register presence for the same. The service supports three interfaces: a watcher interface for requesting and subscribing presence data, a watcher notification interface in order to receive presence events, and a presentity interface for supplying presence data and managing subscriptions:

- Requests used by the watcher to obtain presence data. After subscripting to presence data, the watcher can select between a polling mode and a notification mode for receiving the presence data.
- Requests offered by the application to receive presence notifications.
- Requests used to provision presence data and manage access to the data by its watchers.

Support for groups of addresses is essential, e.g. when managing acess to presence data. In order for such groups to be reusable; a group with recipients is uniquely identified and addition/deletion of group members is allowed.

Annex A (informative): Change History

Change history							
Date	TSG#	Vers	CR	Tdoc SA/CN	New Vers	Subject/Comment	
	S-16	5.1.0	043r1	SP-020314	5.2.0	Reduction of scope of OSA Rel5	
	S-18	5.2.0	044	SP-020777	6.0.0	Mapping of OSA APIs to Presence	
	S-24	6.0.0	047r1	SP-040315	6.1.0	Add descriptions of OSA high-level abstraction interfaces	
Sep 2004	N-25	6.1.0	N5-040504	048	D6.2.0	Add descriptions of OSA high-level abstraction interfaces	NP-040353
Sep 2004	N-25	6.1.0	N5-040524	049	D6.2.0	Correct descriptions of OSA high-level abstraction interfaces	NP-040353
Sep 2004	N-25	6.1.0	N5-040586	050	D6.2.0	Add OSA Multi Media Messaging SCF - stage 2 description	NP-040353
Nov 2004	N5-29	0.0.1	N5-040715	N5-040778	0.0.2	Rapporteur: Chelo ABARCA (chelo.abarca@alcatel.fr) Update of N5-040715. This is the new BASELINE Stage 2 specifications	
Nov 2004	--	0.0.2	N5-040778	N5-040877	0.1.0	Zoicas: Added Introduction clause Accepted all rev. marks in line with N5-040708r1 DRAFT A Report_CN5_29_Barcelona. Incorporated changes from CN#29 agreed contributions: N5-040729 SA2 input to New OSA Stage 2 JWG Chair N5-040730 Proposed Modificaitons to Clause 5 of OSA Stage 2 Alcatel N5-040731 Proposed Modifications to Clause 6 of OSA Stage 2 Alcatel N5-040732 Rel 6 CR 23.198 OSA Application HA Support AePONA N5-040800 CR 23.198 Add Request to be notified of delivery receipt of a sent SMS France Telecom Not dealt with.	

Change history							
Date	TSG #	TSG Doc.	CR	Rev	Subject/Comment	Old	New
Dec 2004	CN_26	NP-040488	--	--	Submitted to CN#26 for Information	1.0.0	