Source: **SA1**

Title: **CR to 22.127 on Removal of Visited Network capabilities (Rel-6)**

Document for: **Approval**

Agenda Item: **7.1.3**

Meet	Doc. No.	Spec	CR	Rev	Phase	Cat	Subject	Vers	New Vers	Doc. SA1
SP-22	SP-030702	22.127	068	-	Rel-6	C	Removal of Visited Network capabilities	6.3.0	6.4.0	S1-031150

TSG-SA WG1 #22 **S1-031150**
Bangkok, Thailand, 27 - 31 October 2003 **Agenda Item: 8**

CR-Form-v7

CHANGE REQUEST

⌘ **22.127** **CR** **068** ⌘ **rev** **-** ⌘ Current version: **6.3.0** ⌘

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

Proposed change affects: UICC apps⌘ ☐ ME ☐ Radio Access Network ☐ Core Network ☒

Title:	⌘	Removal of Visited Network capabilities
Source:	⌘	SA1 (Lucent Technologies)
Work item code:	⌘	OSA3
Date:	⌘	25/10/2003
Category:	⌘	**C**
Release:	⌘	Rel-6

Use one of the following categories:
F *(correction)*
A *(corresponds to a correction in an earlier release)*
B *(addition of feature),*
C *(functional modification of feature)*
D *(editorial modification)*
Detailed explanations of the above categories can be found in 3GPP TR 21.900.

Use one 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)
Rel-6 (Release 6)

Reason for change:	⌘	CN Plenary deleted functions for retrieval of visited network capabilities from their workplan. This topic will not complete in Rel-6. The requirements will not accurately reflect OSA in the completed Rel-6.
Summary of change:	⌘	Removal of the requirement for the application to obtain information about the network capabilities of the visited network serving a subscriber.
Consequences if not approved:	⌘	OSA Rel-6 will not be seen to be complete and avoidable confusion could arise.

Clauses affected:	⌘	13.3.6

Other specs affected:	⌘	Y	N		⌘
			X	Other core specifications	
			X	Test specifications	
			X	O&M Specifications	

Other comments:	⌘	This is a tidy-up CR post-CN Plenary #21

13 Functions offered by OSA

Functions that are offered by OSA shall be applicable for a number of different business and applications domains (including besides the telecommunication network operators also service provider, third party service providers acting as HE-VASPs, etc.).

All of these businesses have different requirements, ranging from simple telephony and call routing, virtual private networks to fully interactive multimedia to using MS based applications.

Service Capability Features:

Application/Clients access OSA functions in terms of service capability features via the standardised application interface. This means that service capability features are accessible and visible to application/clients via the method/operation invocations in the interface.

Service capability features shall be defined as much as possible in a generic way to hide the network specific implementation. To achieve this, it is necessary to identify the functionalities that can be provided in different ways by the use of different service capabilities. For example, User Location can be produced in several underlying ways.Each of these functionalities can then be defined as a single generic function and the different underlying service capabilities are not visible to the application. It is important that the generic part becomes as large as possible to enable applications to use functions without the need for knowledge of the underlying network capabilities

When applications use the generic service capability features, these applications become independent of the network domain ie network agnostic. Applications shall however still be able to request service capability features specific to a service capability (e.g. Call Setup from CAMEL) or specific to a particular network domain. This will increase dependency of the application on the used service capability while providing improved optimisation.

Note: The grouping of OSA functions into Service Capability features is out of scope of this document.

Three different types of OSA functions can be distinguished:

- **Framework functions**: provide commonly used utilities, necessary for access control, security, resilience and management of OSA functions;
- **Network functions**: these shall enable the applications to make use of the functionality of the underlying network capabilities.
- **User Data** related functions: these enable applications to access data of a particular user. Such data are e.g. the status of the user, her location or data in the user's User Profile.

13.1 The Framework functions

The framework provides the essential capabilities that allow OSA applications to make use of the service capabilities in the network. The framework shall support the ability for applications to access SCFs in another network.

There are three distinct features that comprise the framework: *Trust and Security Management, Service Registration and Discovery functions* and *Integrity Management*.

13.1.1 Trust and Security Mangement

The trust and security management feature provides the necessary mechanisms which define the security parameters in which client applications may access the network. This includes the availability of a framework initial access point through which all client applications are authenticated and authorised and the ability to allow the signing of on-line service level agreements between the client applications and the framework.

13.1.1.1 Authentication

Authentication is used to verify the identity of an entity (user, network, and application).

Three types of authentication are distinguished:

- User-Network Authentication:

Before a user can access her subscribed applications, the user has to be authenticated by the network that provides access to the application. This allows the network to check to what applications the user has subscribed to. User-network authentication *is handled within the network and therefore outside the scope of the present document*.

- Application-Network Authentication:

  Before an application can use the capabilities from the network, a service agreement has to be established between the application and the network. Establishment of such a service agreement starts with the mutual authentication between application and network. If a service agreement already exists, modification might be needed or a new agreement might supersede the existing.

- User-Application Authentication:

  Before a user can use an application or perform other activities (e.g. modifying profile data) the application must authenticate the user. When the network already authenticates the user, authentication is not needed anymore. When the network is transparent and the user accesses an application directly, authentication is needed between user and application.

13.1.1.2 Authorisation

Authorisation is the activity of determining what an authenticated entity (user, network, and application) is allowed to do.

NOTE: Authentication must therefore precede authorisation.

Two types of authorisation are distinguished:

- Application-Network Authorisation:

  Verifies what non-framework functions the application is allowed to use. Once an application has been authorised to use one, more or all non-framework functions no further authorisation is required as long as the "allowed" non-framework functions are used.

- User-Application Authorisation:

  The application verifies what actions the user is allowed to perform (e.g. deactivation of functionality, modification of application data). This is transparent to the network and therefore outside the scope of the present document.

13.1.2 Service Registration feature

The Registration function enables the non-framework service capability features (i.e. service capability features that contain non-Framework functions) to register with the Framework. Registration must take place before authorised applications can find out from the Framework which non-framework service capability features are available. This means that the non-framework service capability features must be registered before they can be discovered and used by authorised applications. The service capability feature is finally registered if the registration process is successfully completed.

Note that only the non-framework service capability features have to be registered. The Framework service capability features (containing only Framework functions) are available by default since they provide basic mechanisms.

13.1.3 Service De-Registration function

The De-Registration function enables the non-framework service capability features (i.e. service capability features that contain non-Framework functions) to de-register with the Framework. When a service capability feature de-registers the service capability feature shall discontinue providing service to all applications. Further, the service capability feature can no longer be discovered.

13.1.4 Service Discovery feature

The Discovery function enables the application to identify the total collection of service capability features that it can use. Upon request of the application, the Discovery function indicates the non-framework service capability features that are available for use by the application. The list of available service capability features is created through the Registration process described in subclause 12.1.2. This means that a service capability feature must be registered at the Framework before it can be discovered by the application.

13.1.5 Integrity Management function

Integrity management provides the means to allow the framework to query and report conditions that relate to the integrity of the framework, the network service capability features and the client applications. Furthermore an application may query conditions that relate to the integrity of the framework ad the network service capability features and report on it's own conditions. As part of the integrity management functions, the framework may provide:

- supervision of the status of client applications to ensure continued operation, a process known as heartbeating.
- fault management reporting and control. Specific examples include the ability for the framework to notify applications of internal fault conditions as well as faults in the network service capability features and the ability for applications to request specific test executions in the framework.
- operations and maintenance access, more specifically, the ability for the application to synchronise with the system date and time.

13.2 Network functions

The Network functions represent the total collection of network resources.
The following subclauses define generic network functions e.g. for Message Transfer.

13.2.1 Call and Session Control functions

This subclause details with Call and Session Control functions. The purpose of this function is to allow applications to control and monitor calls, packet switched sessions and IM Sessions.

The application may request the quality of service when first negotiated at the start of the call and may also request the network to notify the application of any changes in QoS (conversational, background, interactive and streaming class - see [4]) which take place during the call.

For QoS information, the Call and Session Control Functions allow applications to monitor the amount of used traffic channels and bandwidth (e.g. for HSCSD) and used timeslots (e.g. for GPRS).

13.2.1.1 CS Call Control functions

This subclause details with circuit switched call control functions. The purpose of this function is to allow applications to control and monitor calls.

Applications should have the ability to :

- Release Calls:

  This provides the ability for the application to force the release of a call. The application may provide an indication of the reason for release of the call.

- Control Calls:

  This provides the ability for an application to modify the information pertaining to the call at the time of establishment of the call. The application may also allow the call to continue with or without the modified information pertaining to the call. The application shall have the ability to request call events of the call under control to be observed by the network and reported back to the application.

- Request call information:

This provides the ability for an application to request information relating to the call of interest specified in advance. Requested information includes at least call duration, call end time.

- Monitor calls:

  This provides the ability for an application to monitor for call duration and tariff switching moments. An application may specify a threshold for the duration of a call or a part thereof. The application shall have the ability to grant new thresholds when the expiry of a previously set threshold has been reported to the application. When an event is subject to be monitored and the event is met, the application shall get informed accompanied with sufficient information.

- Presentation of, or restriction of, information associated with a party involved in a call (e.g. calling line ID, calling name);

- Relinquish control over a call

  This allows an application to relinquish control over a call but still allowing the established call to continue. Once the control of the call has been relinquished, the application may not be able to regain control over that call.

- Interact with a user

  This provides the ability for an application to interact with a user. An application may be able to send specific information to the user and may request the collection of data from the user. Sending information to the user or collecting information from the user shall be supported when the user is engaged in a call (e.g. USSD, DTMF) or call-unrelated (e.g. USSD, SMS). The information sent to the user may include an indication of an announcement, text or user specific data.

Note 1: Call related user interaction may e.g. be used to play/record/customise user specific announcements while through call-unrelated user interaction e.g. service preferences may be managed.

For each call it shall be possible to specify:

- the events on which monitoring is required ([10]).

Note 2: The mapping to service capabilities is for further study (it shall be investigated to which extend the requirements above fit to CAMEL, MEXE and other service capabilities).

13.2.1.2 PS Session Control functions

This subclause details with PS session control functions. The purpose of this function is to allow applications to control and monitor PS sessions. A PS Session may consists of one or more GPRS PDP context.

Applications should have the ability to :

- Release a PDP context:

  This provides the ability for the application to force a PDP context to be released. The application may provide an indication of the reason for release of the PDP context.

- Control a PDP context:

  This provides the ability for an application to modify the information pertaining to the PDP context at the time of establishment. The application may also allow the PDP context to continue with or without the modified information pertaining to the PDP context. The application shall have the ability to request events to be observed by the network and reported back to the application.

- Monitor a PDP context:

  This provides the ability for an application to monitor for PDP context duration and tariff switching moments.. An application may specify a threshold for the duration of a PDP context or a part thereof. The application shall have the ability to grant new thresholds when the expiry of a previously set threshold has been reported to the application.

- Monitor a PS session:

This provides the ability for an application to monitor for PS session data volume. An application may specify a threshold for the amount of data allowed to be transferred within a PS session. The application shall have the ability to grant new thresholds when the expiry of a previously set threshold has been reported to the application.

13.2.1.3 IM Session Control functions

IM Session Control

Create IM Sessions
The application shall be able to establish IM sessions between two or more parties with certain media capabilities. The application may add or remove parties at any time for any session. An application may add additional sessions with certain media capabilities between any parties already involved in an session. Sessions with multiple parties may lead to the creation of a multi-media Conference Call. This can either be an ad-hoc conference creation or it can refer to resources that were reserved in advance.

Release IM Sessions
This provides the ability for an application to force the release of an IM session. This may be limited to the release of certain parties from the session or may be the release of all the parties.

Relinquish control over an IM session
This allows an application to relinquish control over IM sessions.

Party join/leave control
The application shall be informed when a new call party wants to join/leave the conference. It shall be possible for the application to allow or reject the inclusion of the new party to a conference.

Presentation of, or restriction of, information associated with a party involved in a session (e.g. calling line ID, calling name);

Media Control

Control media channels
The application shall have the ability to control media channels originated by (or on behalf of) a user or media channels terminated to a user. This control includes, but is not limited to the barring of a media channel request, allowing the media channel establishment to continue with or without modified information, addition or removal of additional media channels, temporarily suspend a media channel (place on hold), open, close or modify the parameters of the media channels.

Relinquish control over specific media channels
This allows an application to relinquish control over the media stream. When it relinquishes control over certain media channels it does not lose control over the entire session.

Reserve/Free conference resources
The application shall be able to reserve resources in the network or free earlier reserved resources for a conference in advance.

Information

Request Notification of Media channel events
The application shall be able to request notification of certain events associated with a type of media channel. Events include, but not limited to: a user initiating or closing a session, an incoming IM session request to user or a terminating user unable to accept an incoming IM session request.

Monitoring of Media channels
The application shall be able to request all the media channels currently available on a IM session In addition the application must be able to monitor the opening and closing of channels for media for a specified session.

13.2.2 Void

13.2.3 Void

13.2.4 Charging functions

Call and Event Charging

Call and Event Charging functions enable the application to instruct the network and inform the user with charging information and to add some additional charging information to the network generated Call Detail Records. Some of the following charging facilities are available only if the network either controls the account or has access to it.

The OSA Call and Event Charging function shall enable an application to:

- define and manage thresholds (e.g. session duration, data volume) for a call;
- provide additional charging information to be included in the Call Detail Record. This may contain information transparent to the network;
- transfer Advice of Charge data (as defined in [5]) to the terminal.

Service Usage

These charging functions shall enable applications to augment subscriber accounts maintained by the network and to charge subscribers for using services. These applications are not necessarily telecommunication related. In addition to charging subscribers for service usage, these functions could also be used for payments in an online purchase scenario.

Provided, that these functions are supported by the underlying network an application shall be able 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. If a reservation has been made earlier, this amount should be taken from the reserved amount.
- Request the network to split the deduction of an amount among several subscribers accounts or other chargeable entities according to a specified partitioning. It shall be possible to notify an individual subscriber's account or other chargeable entity about the percentage of the total amount, to which the deduction has been performed Note: this requirement also covers the case when the total amount to be deduced is calculated by the network.
- Release a reservation acquired earlier. If part of a reservation has been deducted already, just release the remaining reservation.
- 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.

The functions for charging application usage shall meet the following general requirements:

- Hide payment policy (e.g. prepaid/postpaid) from applications
- Hide payment type (credit card, cash, bank withdrawal) from applications
- Hide subscriber's identity towards the application. This would provide anonymity (like for prepaid customers).
- Support prepaid subscribers. This requires that the application immediately gets informed if the subscriber's account covers the service usage costs. If not, the application may reject serving the subscriber.
- Allow for Multi-currency support. This shall allow service providers to request charging in their preferred currency

General Account functions

These functions provide access to sensitive data. They shall be restricted to client applications that had been granted additional privileges via suitable means, i.e. as authorised by the framework function.

The OSA general Account function shall enable an application to:

- retrieve a transaction history of a subscriber's account, this may include
  - the retrieval of a list of monetary or non-monetary amounts that have been debited from or credited to a subscribers online account,
  - the request of additional information on the specific transaction (e.g. the application or service description provided with the actual transaction).
- check a subscriber's current account balance.
- monitor the subscribers account and may request to get informed of any change.
- ask the charged user for an explicit, interactive confirmation before any charging operation is performed. The General Account function will support a procedure to obtain confirmation by the user. Such a procedure shall be under the control of the network.

Note: There is no requirement to standardise this procedure.

In case an application retrieves a list for a subscribers' transaction history, it shall specify the time interval for which the transaction history shall be retrieved.

13.2.5 User-Application Authentication functions

The User-Application Authentication functions provide to applications support for authentication of their users. It also provides an "application-specific user identifier" to be used as a parameter in invocation of other OSA Network functions, when requested by the application.

The User-Application Authentication functions shall authenticate an user upon requests of an application; this requires the application to provide as an input the subscriber's credentials, which enable secure method of authentication (e.g. subscriber's certificates).

The User-Application Authentication functions shall return to the invoking application an "application-specific user identifier" (a true identity or alias) that identifies the authenticated user, when requested by the application. The identifier may be used by the application to recognize a user through several accesses to the application; it may also be used by the application as a parameter in invocation of other OSA network functions (e.g., for User Location function).

The User-Application Authentication functions shall support privacy settings defined by the user.

If the subscriber's privacy settings so require, the "application-specific user identifier", returned by User-Application Authentication function to the invoking application, shall be an alias. Otherwise, the "application-specific user identifier" shall be the true identity of the subscriber (e.g. MSISDN).

When the application invokes OSA Network functions related to subscriber (e.g. Location, Presence), the subscriber's identifier shall be included in the request. An application may request it from the User-Application Authentication function.

When an OSA Network function receives the request from the application and the subscriber's identifier is an alias, the OSA Network Function shall invoke the User-Application Authentication function to translate the alias to the subscriber's true identity (e.g. MSISDN).

13.3 User data related functions

13.3.1 User Status functions

The User Status functions enable an application to retrieve the user's status, i.e. to find out on which terminals the user is available.

The following functions shall be provided:

- **retrieval of User Status:**
  - the application shall be able to retrieve the status of the user (e.g. the user is busy, her terminal is attached, or detached).
- **notification of User Status Change:**
  - the application shall receive notifications when the user's terminal attaches or detaches:
    - detach: the user's terminal is switched off or the network initiates detach upon location update failure;
    - attach: the user's terminal is switched on or there has been a successful location update after network initiated detach.
  - the application shall receive notifications when the user's status moves from idle to busy, or from busy to idle.

Attach and detach applies for CS and PS.

The application shall be able for each terminal to start/stop receipt of notifications.

13.3.2 User Location functions

The User Location functions provide an application with information concerning the user's location.

The user location information contains the following attributes:

- **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).

The following functions shall be provided:

- **report of location information:**
  - the application shall be able to request user location information;
  - by default the location information is provided once; the application may also request periodic location reporting (i.e. multiple reports spread over a period of time).
- **notification of location update:**
  - the application shall be able to request to be notified when the user's location changes, i.e. when:
    - the user enters or leaves a specified geographic area;
    - the user's location changes more than a specified lower boundary. The lower boundary can be selected from the options provided by the network.

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

- **Access control to location information:**
  - the user shall be able to restrict/allow access to the location information. The restriction can be overridden by the network operator when appropriate (e.g. emergency calls).

When an application requests report of location information or notification of location update, it shall be possible for the application to provide an optional requestor identiy, an optional service identity and an optional codeword (as defined in [9]). If an application provides one or more of the above optinal privacy information, the information shall be brought to the location service capabilities attention and used to comply with privacy policies of the subscriber the request relates to.

13.3.3 User Profile Management functions

The User Profile Management functions enables the (authorised) applications to access the User Profile data, checking before the application's rights related to each separate part of the User Profile. The User Profile data accessed by the application could be independent of specific application but necessary to personalise the application according to the user preferences (an example could be the preferred language of end-user).

Depending on the authorisation, the User Profile Management functions may permit the VAS to read from and/or to add to and/or to modify the User Profile or parts of it. This decision is based on:

- Subscriber identity
- Access information on specific part of the User Profile of the subscriber
- Application identity
- Access type (read, add or modify)

Access information shall contain the user specific access rights per application. These may be given either for individual parts of the User Profile or for a group of data or even all data in the User Profile.

13.3.4 Void

13.3.5 Terminal Capabilities functions

The Terminal Capabilities functions enable the application to determine the capabilities of the user's terminal .

Note 1: The ability to support this function is dependent on the ability of a terminal (through e.g. MExE or WAP) to notify its terminal capabilities. Therefore this function will *not* be able to supply terminal capabilities for terminals not supporting notification of their terminal capabilities.

Note 2: "Terminal" covers both (mobile) equipment and USIM.

The following functions shall be provided:

- **retrieval of Terminal Capabilities:**
  - the application shall be able to retrieve the capabilities of the terminal. This includes:
    - the media that the terminal is capable to deal with (e.g. audio, video, ; this information is needed by the application e.g. when the user wants to download messages from the mailbox);
    - the number of calls/sessions that the terminal can deal with simultaneously.

13.3.6 ~~Functions for retrieval of Visited Network Capabilities~~Void

~~OSA applications make use of network capabilities offered through the abstraction of the service capability features. As a user may be served by network capabilities in a VPLMN, applications may benefit from knowing the differences that exist between the home and visited network capabilities. Such information may provide the ability for an application to tailor its behaviour according to the capabilities of the visited network.~~

~~The functions for retrieval of Visited Network Capabilities shall enable the application to obtain information about the network capabilities of the visited network serving a subscriber.~~

~~The information provided to the application shall contain the following, if available:~~

- ~~Available network toolkits, including level of support (e.g. CAMEL Phase X)~~
- ~~Supported Network access, (e.g. GPRS, CS, IMS), and in case of no support, detailed information (unknown support, roaming not allowed, …).~~

13.4 Void

13.5 Presence related capability functions

13.5.1 Relationship to Release 6 Presence Service

Void.

13.5.2 Functions

The OSA interface 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 limited to presence information pertaining to the presence service as described in [7] or user availability.

An OSA application may act as a requester of presence information (i.e. act as a watcher) and/or act as a supplier of presence information (i.e. act as a presentity). All the capabilities offered to presence service watchers and presentities are described in [7] and may be offered to OSA applications. In addition to the authorisation performed by the OSA Framework, the presence service checks that the application is permitted to access the presence service.

An OSA application may manage or query availability status and/or preferences of a user which may be associated with one or more services (e.g. voice call, IMS sessions, MMS …etc.). Such availability may be determined from a range of existing capabilities.

The following OSA capabilities shall be supported for an application:

- **register as a presentity and/or watcher:**
  - the application shall be able to request the registration as a presentity and/or as a watcher in the presence service. This registration shall include the ability to establish as well as cancel a registration.
- **supply presence related information to the network:**
  - the application shall be able to supply and/or update presence related information (presence information or availability) at any time. An application may modify the availability of a user. - **request the querying and/or modification of presence related data:**
  - the application shall be able to request the querying and/or modification of data other than presence information related to watchers and/or presentities. Such data includes, but is not limited to any access rules pertaining to the presentity to be modified. An application may be able to request the management of availability preferences of a user. Management includes the setting, modification and deletion of availability preferences.
- **request Presence related Information :**
  - the application shall be able to request presence related information. The application shall be able to request presence information about a presentity or may request the availability of a user. Such requests may be for the current information, on a periodic basis or for future changes in the presence related information (e.g. arming of event notifications).
- **retrieve watcher information:**
  - the application shall be able to request watcher information about a presentity.

13.6 IP session function

The IP session function enables applications to access information about IP sessions in progress between a UE and IP networks (i.e., the MSISDN and Session Correlation identifier) using the IP address of the UE. An IP session comprises a flow or a set of flows through a network element during a certain time interval. An IP flow is defined to be a stream of packets that have a set of common properties. The properties include source IP address/port and destination IP address/port, protocol type etc. Flows can be grouped into sessions by specifying wildcards for properties (e.g. the set of flows going to port 80, or the set of flows with target IP address X.X.X.X.)

Applications shall have the ability to :

- Release flows in an IP session:

  This provides the ability for an application to force the termination of an IP session. The application may provide an indication of the reason for release of the IP session.

- Control an IP session:

  This provides the ability for an application to request the modification of the parameters of an IP session both during establishment of the session and while the sessions are in progress. The application may also allow the IP Session to continue with or without the modified information pertaining to the IP Session. This may also include the ability to refuse session establishment, to request modification of Quality of Service parameters, to request modification of the destination IP address (including the IP port) and the modification of volume thresholds (e.g. to allow an application to change the threshold at which a notification is raised).

- Monitor an IP Session:

  This provides the ability for an application to monitor an IP session. The application will specify a particular IP session and event condition. When the condition is met an event is generated and the application shall be informed accompanied with sufficient information. For example, an application could be notified when the data volume threshold of a particular user (defined by source IP address) is exceeded.

- Request flow Information

  This provides the ability for an application to request information about the session of interest. This includes quality of service parameters, target IP address and port, duration of session, and data volume of session

The access to the data, which is typically stored within a network authentication server, is obtained via the OSA gateway (i.e., through this SCF). The IP session information/data shall be released based on specific defined policies between the network operator and the application service provider.

13.7 Multimedia Messaging function

The Multimedia Messaging function enables applications to receive and send multi-media messages.