3GPP TR 32.808 V0.0.3 (2006-10)

Technical Report

3rd Generation Partnership Project; Technical Specification Group Services and System Aspects; Telecommunication management; Study of Common Profile Storage (CPS) Framework of User Data for network services and management; (Release 7)

The present document has been developed within the 3rd Generation Partnership Project (3GPP TM) and may be further elaborated for the purposes of 3GPP.
The present document has not been subject to any approval process by the 3GPP Organizational Partners and shall not be implemented.
This Specification is provided for future development work within 3GPP only. The Organizational Partners accept no liability for any use of this Specification.
Specifications and reports for implementation of the 3GPP TM system should be obtained via the 3GPP Organizational Partners' Publications Offices.

Keywords

Object Model, Management

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

Copyright Notification

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

© 2006, 3GPP Organizational Partners (ARIB, ATIS, CCSA, ETSI, TTA, TTC).
All rights reserved.

Contents

Foreword....5
Introduction....5
1 Scope....6
2 References....6
3 Definitions and abbreviations....9
3.1 Definitions....9
3.2 Abbreviations....9
4 Rationale for the Analysis of a Common User Model and of the Basic Structure of a Common Profile Storage Framework....11
4.1 Business Case Considerations....11
4.2 Use Cases....11
4.3 Resulting Requirements....11
5 Considerations on a Common User Model....12
5.1 Network Functions and Management Applications Using Subscriber/User Data....12
5.1.1 Network Supporting Services....12
5.1.1.1 Location Register:....12
5.1.1.2 The Equipment Identity Register (EIR)....16
5.1.1.3 The Mobile-services Switching Centre (MSC) Server (Media Control)....17
5.1.2 Enabling Services....17
5.1.2.1 Presence Service....17
5.1.2.1.1 IETF....17
5.1.2.1.2 3GPP....20
5.1.2.1.3 3GPP2....22
5.1.2.1.4 OMA....22
5.1.2.2 Location Service....27
5.1.2.2.1 3GPP....27
5.1.2.2.2 OMA....28
5.1.2.3 XDM....29
5.1.2.3.1 OMA....29
5.1.2.4 Device Management....30
5.1.2.4.1 OMA....30
5.1.2.5 Authorization and Authentication....34
5.1.2.5.1 ITU-T....34
5.1.2.5.2 IETF....38
5.1.2.5.3 3GPP....41
5.1.2.5.4 OMA....44
5.1.2.6 Accounting....44
5.1.2.6.1 ITU-T....44
5.1.2.6.2 IETF....46
5.1.2.6.3 3GPP....50
5.1.2.6.4 OMA....50
5.1.3 Network Hosted Business Services and Network functions....51
5.1.3.1 MM Messaging Services (MMS)....51
5.1.3.2 Content Services....51
5.1.3.3 IMS Application Servers....51
5.1.3.4 Store and Forward Messaging....51
5.1.3.5 P2P and Group Communication....51
5.1.3.6 IN Services....51
5.1.3.7 Online Charging, PrePaid Services....51
5.1.4 Management Applications....51
5.1.4.1 Subscriber Management....51
5.1.4.2 Statistics....51
5.1.4.3 Billing....51

5.1.4.4 System Management ... 51
5.2 Standardization documents containing subscriber/user information ... 52
5.3 Problems in the handling of the current Subscriber/User Data Model ... 53
5.3.1 Inconsistencies in the data between different network functions ... 53
5.3.2 Redundancy problems ... 53
5.3.3 Services requiring different handling of User Data ... 53
5.3.4 Requirements for the Common User Data Model ... 53
5.4 Basics of a Common User Data Model ... 53
5.4.1 Characteristics of an End-User ... 53
5.4.1.1 Types of Data Assigned to an End-User ... 53
5.4.1.2 The Identity of an End-User ... 54
5.4.1.2.1 Representation of the Identity of the End-User through his Keys ... 54
5.4.1.2.2 Identity Management of the End-User ... 54
5.4.1.3 The Relation between an End-User and a Subscriber ... 54
5.4.2 Different Levels of Data Consolidation ... 56
5.4.2.1 Adaptation Layer for partial Data Consolidation ... 56
5.4.2.2 Adaptation Layer for full data consolidation ... 57
5.4.2.3 Mixed Scenarios ... 57
5.4.3 Semantic Identity of Data Entities ... 58
5.4.4 Content of Post Update Triggers ... 58

6 Basic Structure of the Common Profile Storage Framework (CPSF) ... 59
6.1 Logical View ... 59
6.2 Analysis of alternative solutions ... 59
6.3 Tooling ... 59

7 Gap-Analysis ... 60

8 Conclusions ... 61

Annex A: Example for the realization of an end-user database according to the Common Profile Storage Framework ... 62

A.1 The Model Structure ... 62

A.2 The Network Architecture ... 62

Annex B: Change history ... 63

Foreword

This Technical Report 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

This Technical Report studies the problems and potential improvements in the way user data is introduced and handled in the 3GPP based networks. Related to these studies this Technical Report investigates the need to introduce a common end-user profile storage framework into 3GPP conformant communication networks.

The introduction of a Common Profile Storage Framework of User Data ("Common" in the sense of common to all applications) for network services and management applications could significantly enhance the ability of 3GPP based networks to offer complex and combined services in the areas of:

- Multimedia;
- Data services;
- Value Added Services;
- End-to-end applications.

In light of developments both within 3GPP (e.g. IMS, MBMS, OCS, PCC) and outside 3GPP (e.g. NGN, OMA, etc.) with a growing number of physically disjoint but logically correlated user data stored in several data bases a consolidation and co-ordination of these is needed to prevent further redundancy and possible contradiction and to enable operators to administer and provision complex and combined services.

Initial study of the user related data within and outside 3GPP is necessary to assess the properties of such a Common User Profile Storage Framework based on

- the needs of Network Elements like MMS-RS, HLR, HSS, BM-SC and
- features like Service Management, Subscription Management and Charging Management.

On-going efforts within the mobile and NGN community to define advanced services within the network (i.e. 3GPP, TISPAN) should provide valuable drive for developing the Common Profile Storage Framework functionality for 3GPP.

1 Scope

This Technical Report investigates possibilities to introduce an end-user profile storage framework into 3GPP conformant communication networks. To this end the following topics are covered:

- Analysis of the consequences of creating one common data model structure for an end user
  - Existing data specifications concerning an end-user or a subscriber in 3GPP, TISPAN, OMA and other relevant standardization groups are listed and used as a basis for the proposal of a structure for a common end-user data model
  - Topics connected to the fact that one common data model exists, like minimizing data redundancy, access control, providing views for applications are highlighted.
  - The relationship between an end user and a subscriber as defined by 3GPP is revisited in light of the fact that the relations between subscribers (contract holders) and end-users can be n:m.
- Current initiatives connected to subscription information carried out inside and outside 3GPP are covered.
- Analysis of differences and deficiencies concerning 3GPP's current status of specifications (e.g. GUP, SuM, etc.) in comparison to existing/emerging solutions with centralised or distributed storage models.
- Identification of items for standardization as a result of the analysis described in the bullet above.

2 References

The following documents contain provisions which, through reference in this text, constitute provisions of the present document.

- References are either specific (identified by date of publication, edition number, version number, etc.) or non-specific.
- For a specific reference, subsequent revisions do not apply.
- For a non-specific reference, the latest version applies. In the case of a reference to a 3GPP document (including a GSM document), a non-specific reference implicitly refers to the latest version of that document *in the same Release as the present document*.

[1] 3GPP TR 21.905: "Vocabulary for 3GPP Specifications".

[2] 3GPP TS 23.008 V7.2.0 (2006-06): "Organization of Subscriber Data".

[3] 3GPP TS 23.003 V7.0.0 (2006-06): "Numbering, addressing and identification".

[4] 3GPP TS 23.097 V6.0.0 (2004-12): " Multiple Subscriber Profile (MSP) (Phase 2) - Stage 2".

[5] 3GPP TS 23.016 V6.1.0 (2004-03): "Subscriber data management; Stage 2".

[6] 3GPP TS 32.140 V6.3.0 (2004-12): "Subscription Management (SuM) requirements".

[7] 3GPP TS 32.141 V6.1.0 (2004-03): "Subscription Management (SuM) architecture".

[8] 3GPP TS 32.171 V6.1.0 (2004-12): " Subscription Management (SuM) Network Resource Model (NRM) Integration Reference Point (IRP): Requirements".

[9] 3GPP TS 32.172 V6.3.0 (2006-03): " Subscription Management (SuM) Network Resource Model (NRM) Integration Reference Point (IRP): Information Service (IS)".

[10] 3GPP TS 32.175 V6.2.0 (2005-06): " Subscription Management (SuM) Network Resource Model (NRM) Integration Reference Point (IRP): eXtensible Markup Language (XML) definition".

[11] 3GPP TS 23.240 V6.7.0 (2005-03): "3GPP Generic User Profile (GUP); Architecture (Stage 2)".

[12] 3GPP TR 23.941 V6.0.0 (2004-12): "3GPP Generic User Profile (GUP); Stage 2; Data Description Method (DDM)".

[13] 3GPP TS 23.002 V7.1.0 (2006-03): "Network architecture".

[14] 3GPP TS 23.101 V6.0.0 (2004-12): "General Universal Mobile Telecommunications System (UMTS) architecture".

[15] 3GPP TS 33.102 V7.0.0 (2005-12): "3G Security; Security architecture".

[16] 3GPP TS 43.020 V6.4.0 (2006-06): "Security related network functions".

[17] 3GPP TS 23.060 V7 .1.0 (2006-06): "General Packet Radio Service (GPRS); Service description; Stage 2".

[18] 3GPP TS 23.271 V6.13.0 (2005-09): "Functional stage 2 description of Location Services (LCS)".

[19] 3GPP TS 25.305: "Stage 2 functional specification of UE positioning in UTRAN".

[20] 3GPP TS 43.059: "Functional Stage 2 description of Location Services in GERAN".

[21] 3GPP TS 23.141 V6.9.0 (2005-12): "Presence Service; Architecture and functional description".

[22] 3GPP TS 23.228 V7.4.0 (2006-06): "Technical Specification Group Services and System Aspects; IP Multimedia Subsystem (IMS); Stage 2".

[23] 3GPP TS 33.222 V7.1.0 (2006-03):" Generic Authentication Architecture (GAA); Access to network application functions using Hypertext Transfer Protocol over Transport Layer Security (HTTPS)".

[24] 3GPP TS 33.102 V7.0.0 (2005-12): "Technical Specification Group Services and System Aspects; 3G Security; Security Architecture".

[25] Draft ETSI DTS 188 002 V0.0.1 (2006-01):" Telecommunications and Internet Converged Services and Protocols for Advanced Networking (TISPAN); Subscription Management Requirements;"

[26] Draft ETSI TS 188 XXX V0.0.1 (2006-02):"Telecommunications and Internet Converged Services and Protocols for Advanced Networking (TISPAN); NGN Management; Management Information Model Requirements".

[27] TISPAN WG8 interim meeting WG8TD08 Sophia Antipolis, 3 - 5 May 2006: "Remaining Comments".

[28] 3GPP2 S.R0037-0 v3.0 Version Date: August 21, 2003 Version 3.0: "IP Network Architecture Model for cdma2000 Spread Spectrum Systems".

[29] 3GPP2 X.S0027-001-0 Version 1.0 Date: September, 2004: "Presence Service: Architecture and Functional Description".

[30] OMA-TS-Presence_SIMPLE-V1_0-20060418-C Candidate Version 1.0 – 18 Apr 2006: "Presence SIMPLE Specification".

[31] OMA-AD-Presence_SIMPLE-V1_0-20060110-C Candidate Version 1.0 – 10 Jan 2006: "Presence SIMPLE Architecture Document".

[32] OMA-AD_IMS-V1_0-20040420-D Draft Version 1.0 – 20 Apr 2004: "Utilization of IMS capabilities Architecture".

[33] OMA-TS-MLP-V3_2-20051124-C Candidate Version 3.2 – 24 Nov 200:5 "Mobile Location Protocol 3.2".

[34] OMA-AD-MLS-V1_0-20050607-C Candidate Version 1.0 – 07 June 2005: "OMA Mobile Location Service Architecture".

[35] OMA-TS-XDM_Core-V1_0-20060612-A Approved Version 1.0 – 12 Jun 2006: "XML Document Management (XDM) Specification".

[36] OMA-TS-DM_StdObj-V1_2-20060602-C Candidate Version 1.2 – 02 Jun 2006: "OMA Device Management Standardized Objects".

[37] OMA-SyncML-DMStdObj-V1_1_2-20031203-A Approved version 03-December-2003: "SyncML Device Management Standardized Objects, Version 1.1.2".

[38] OMA-Security-CertProf-V1_1-20040615-C Candidate Version 1.1 – 15 Jun 2004: "Certificate and CRL Profiles".

[39] Day et al: RFC 2778:"A Model for Presence and Instant Messaging"; February 2000.

[40] Brownlee & Blount: RFC 2924: "Accounting Attributes and Record Formats"; September 2000.

[41] Calhoun et al: RFC 3588: "RFC 3588 Diameter Base Protocol"; September 2003.

[42] Housley et al: RFC 3280: "Internet X.509 Public Key Infrastructure"; April 2002.

[43] Myers et al: RFC 2560: "PKIX OCSP"; June 1999.

[44] ITU-T X.509 "Information technology – Open systems interconnection – The directory: public-key and attribute certificate frameworks" 08/2005.

[45] ITU-T Q.822 "SPECIFICATIONS OF SIGNALLING SYSTEM No.7 – STAGE 1, STAGE 2 AND STAGE 3 DESCRIPTION FOR THE Q3 INTERFACE – PERFORMANCE MANAGEMENT" 04/1994.

3 Definitions and abbreviations

3.1 Definitions

For the purposes of the present document, the terms and definitions given in TR 21.905 [1] and the following apply.
A term defined in the present document takes precedence over the definition of the same term, if any, in TR 21.905 [1].

Adaptation Layer Function: Function provided by the Common Profile Storage Framework to decouple any Network Function or Management Application from the actual data store.

Application Front End: Data access part of a Network Function or Management Application to the Common Profile Storage (CPS) Framework.

Data entities: a logical data model is an abstract DBMS-independent representation of a set of data entities and their relationships within the scope of a system. A logical data model typically includes all the entities and their attributes that correspond to a set of specified information requirements, which includes the definition of logical constraints on these attributes

End User: Person actually consuming a service, in 3GPP defined as user in opposition to a subscriber

Network Function: Function within the 3GPP or OMA, Service Architecture or Management Network making use of the Common Profile Storage (CPS) Framework.

NOTE: **Application:** This notion is only used in the context of other respective standard documents. This means that the explanation can be found in the cited reference documents.

3.2 Abbreviations

For the purposes of the present document, the abbreviations given in TR 21.905 [1] and the following apply.
An abbreviation defined in the present document takes precedence over the definition of the same abbreviation, if any, in TR 21.905 [1].

AAA	Authentication, Authorization and Accounting
ACID	Atomicity – Consistency – Isolation – Durability
ALF	Adaptation Layer Function
AS	Application Server
AuC	Authentication Center
AVP	Attribute Value Pair
CA	Certification Authority
CAMEL	Customized Application for Mobile Networks Enhanced Logic
CDMA	Code Division Multiple Access
CN	Core Network
CPS	Common Profile Storage
DBMS	Data Base Management System
EIR	Equipment Identity Register
FE	Front End
GGSN	Gateway GPRS Support Node
GPRS	General Packet Radio System
GPS	Global Positioning System
HLR	Home Location Register
HSS	Home Subscriber Server
HTTP	Hypertext Transfer Protocol
IETF	Internet Engineering Taskforce
IMEI	International Mobile Equipment Identity
IMS	IP Multimedia Subsystem
IMSI	International Mobile Subscriber Identity
IN	Intelligent Network
IP	Internet Protocol
LAI	Local Area Identification

LDR	Location Deferred Request.
LIR	Location Immediate Request
LMSI	Local Mobile Station Identity
MBMS	Multimedia Broadcast and Multicast Service
MIME	Multipurpose Internet Mail Extensions
MMD	Multimedia Domain
MMS	Multimedia Messaging Service
MS	Mobile Station
MSC	Mobile Switching Center
MSISDN	Mobile Station International ISDN Number
NGN	Next Generation Networks
OCS	Online Charging System
OMA	Open Mobile Alliance
PCC	Personal Computing and Communications
PDA	Personal Digital Assistant
PDP	Packet Data Protocol
PLMN	Public Land Mobile Network
PNA	Presence Network Architecture
PoC	Push to talk over Cellular
RAI	Routing Area Identification
RLS	Resource List Server
SDO	Standards Development Organisation
SGSN	Serving GPRS Support Node
SMS	Short Message Service
SIP	Session Initiation Protocol
TMSI	Temporary Mobile Station Identity
UE	User Equipment
UMTS	Universal Mobile Telecommunication System
URI	Uniform Resource Identifier
URL	Uniform Resource Locator
VLR	Visitor Location Register
XCAP	XML Configuration Access Protocol
XDM	XML Document Management

4 Rationale for the Analysis of a Common User Model and of the Basic Structure of a Common Profile Storage Framework

4.1 Business Case Considerations

4.2 Use Cases

4.3 Resulting Requirements

5 Considerations on a Common User Model

Clause 5 has multiple intentions concerning considerations on a common user model.

At the beginning, in clause 5.1, a classification of subscriber/end-user data is given. This clause mainly serves the purpose of showing, what roles the services related to subscriber/end-user data might play and for which kind of services to expect subscriber/end-user data. In addition the realization of these services within a network and the location of the respective end user data within the network are shown. The specific structure of this clause is only of limited importance for the subsequent analysis of the properties of a common user model.

Clause 5.2 lists all relevant standards pertaining to subscriber/end-user data for the network functions treated in 5.2 within and outside of 3GPP.

Clause 5.3 highlights some of the current difficulties when handling subscriber/end-user data in more detail.

Finally clause 5.4 deals with the important points to consider, when putting together a common end-user data model.

5.1 Network Functions and Management Applications Using Subscriber/User Data

In this clause the network functions/services as well as enterprise applications, which make up a 3GPP conformant network, or which could be put onto a 3GPP conformant network and which use or house end-user/subscriber data are listed and classified.

Two types of functions/services using end-user/subscriber data can be distinguished:

- Network functions: These network functions can be found within the 3GPP's, OMA's, IETF's, … network architectures.
- Enterprise applications: These can be found within the OSS/BSS environment of a network/service provider.

Clause 5.1 groups the network functions according to the type of service, they provide or at least support.

The rest of this clause is devoted to an overview of the standardized network scenarios including those network elements, which use or store subscriber data.

5.1.1 Network Supporting Services

5.1.1.1 Location Register:

[13] describes how communication to a mobile station is supported by the network. The information where this mobile station is located is stored in a function named "location register".

According to [13] the location register is handled by four different entities.

- The Home Location Register (HLR): register to which a mobile subscriber is assigned for record purposes such as subscriber information.
- The Visitor Location Register (VLR): register for Circuit Switched (CS) services, other than the HLR, used by an MSC to retrieve information for, e.g. handling of calls to or from a roaming mobile station currently located in its area.
- The Serving GPRS Support Node (SGSN): register function in the SGSN storing subscription information and location information for Packet Switched (PS) services for each subscriber registered in the SGSN (needed only in a PLMN which supports GPRS).
- The Gateway GPRS Support Node (GGSN): register function in the GGSN storing subscription information and routeing information (needed to tunnel packet data traffic destined for a GPRS MS to the SGSN where the MS is registered) for each subscriber for which the GGSN has at least one PDP context active (needed only in a PLMN which supports GPRS)

A basic configuration of a PLMN for UMTS is given in [13]. The following figures show the configurations of the mobile network in the GSM, GPRS and UMTS case.

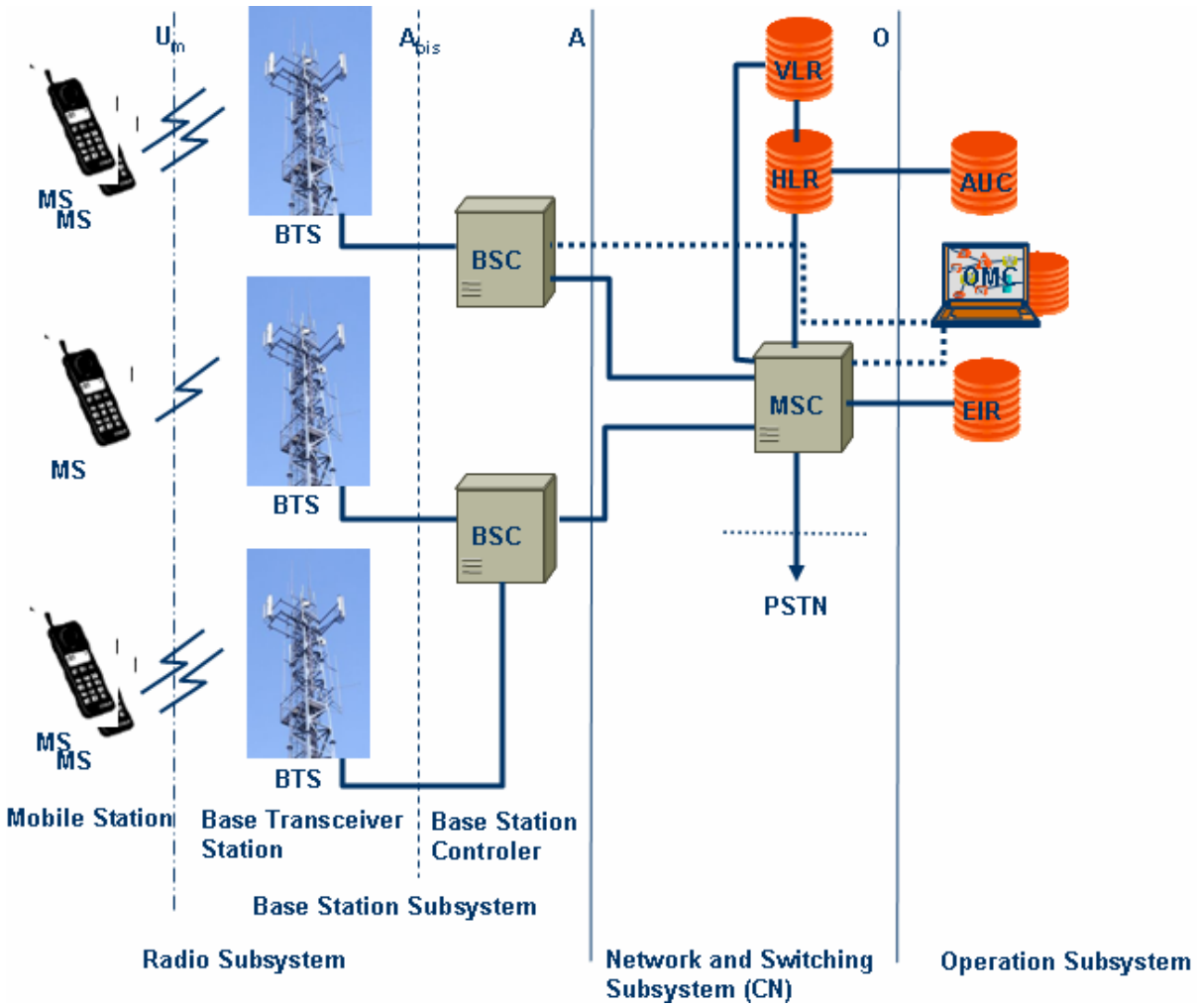


Figure 5.1.1.1.1: Basic Configuration of a GSM PLMN supporting CS services and interfaces

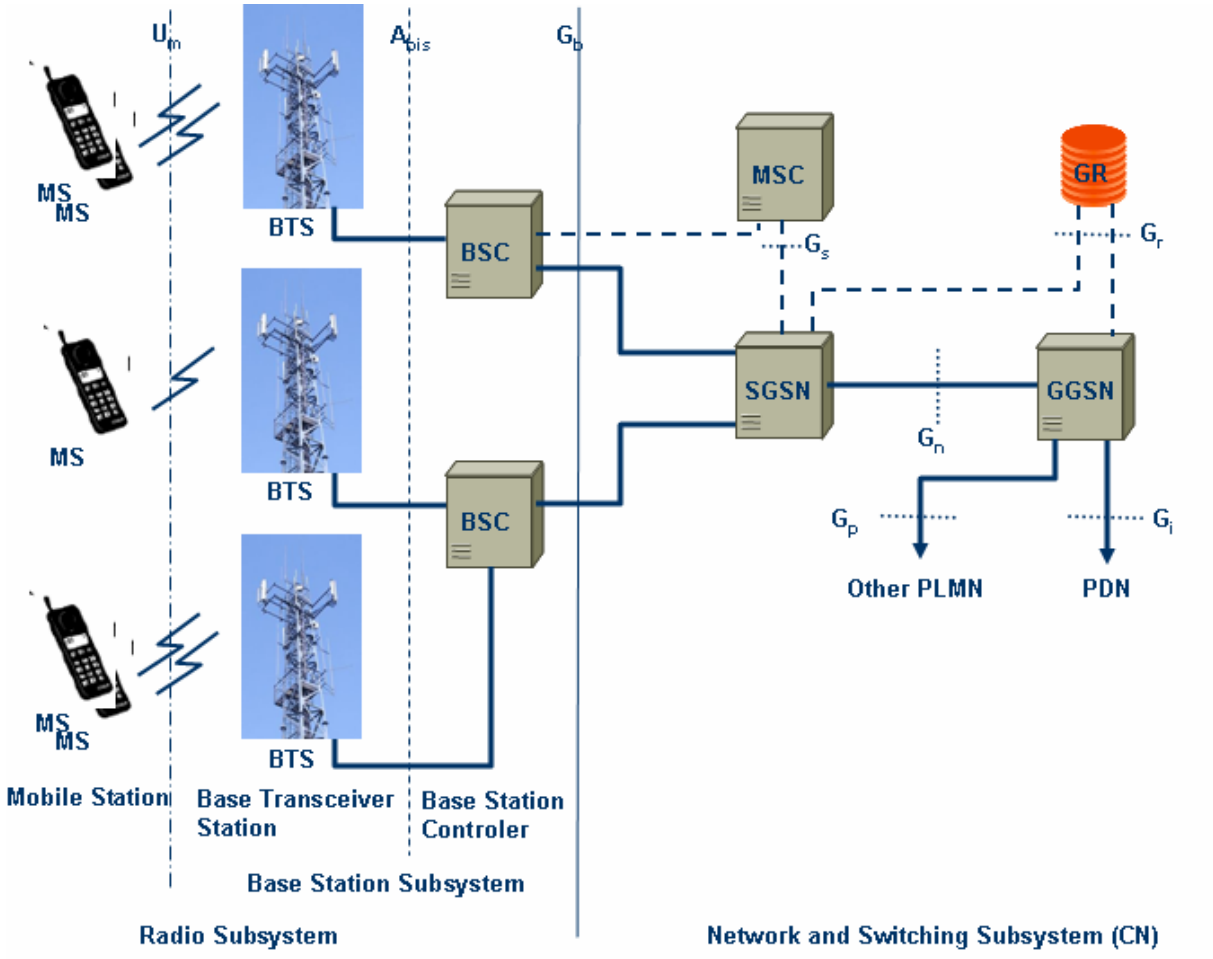


Figure 5.1.1.1.2: Basic Configuration of a GPRS PLMN supporting PS services and interfaces

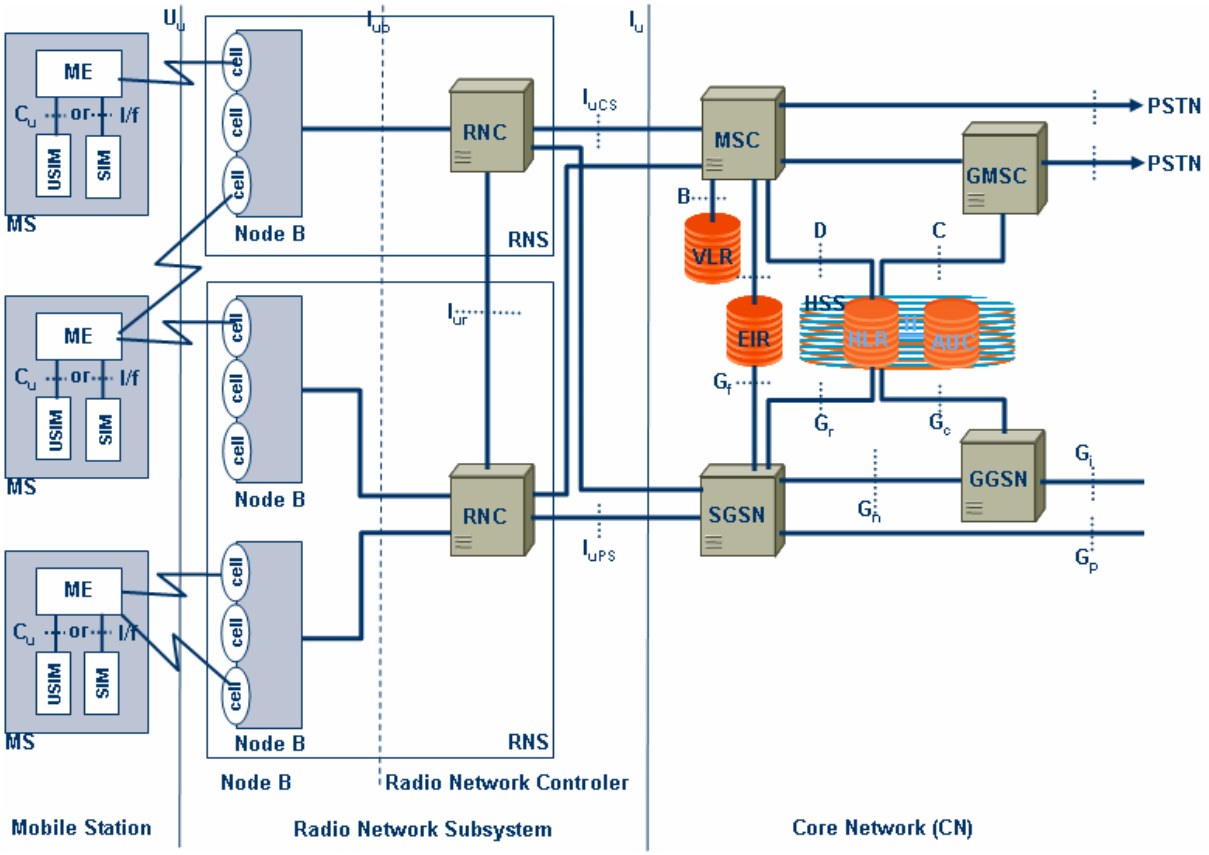


Figure 5.1.1.1.3: Basic Configuration of a UMTS PLMN supporting CS and PS services and interfaces [13]

The HSS (see figure 5.1.1.1.3) is the master database for a given user. It is the entity containing the subscription-related information to support the network entities actually handling calls/sessions.

The HSS houses the following user related information:

- User Identification, Numbering and addressing information.
- User Security information: Network access control information for authentication and authorization
- User Location information at inter-system level: the HSS supports the user registration, and stores inter-system location information, etc.
- User profile information

According to [13] the HSS has the following functions:

- IP multimedia functionality to provide support to control functions of the IM subsystem. It enables subscriber usage of the IM CN subsystem services.
- The subset of the HLR/AUC functionality required by the PS Domain.
- The subset of the HLR/AUC functionality required by the CS Domain, if it is desired to enable subscriber access to the CS Domain or to support roaming to legacy GSM/UMTS CS Domain networks.

Thus the HSS is a realization of parts of the location register functionality mentioned above.

The VLR contains the information needed to handle the calls set-up or received by the MSs registered in its data base including the following elements:

- the International Mobile Subscriber Identity (IMSI);
- the Mobile Station International ISDN number (MSISDN);
- the Mobile Station Roaming Number (MSRN),
- the Temporary Mobile Station Identity (TMSI), if applicable;
- the Local Mobile Station Identity (LMSI), if used;
- the location area where the mobile station has been registered;
- the identity of the SGSN where the MS has been registered. (applicable to PLMNs supporting GPRS and a Gs interface between MSC/VLR and SGSN),
- the last known location and the initial location of the MS.

The VLR also contains supplementary service parameters attached to the mobile subscriber and received from the HLR. The organisation of the subscriber data is outlined in TS 23.008 [2].

The location register function in the SGSN contains:

- subscription information:
    - the IMSI;
    - one or more temporary identities;
    - zero or more PDP addresses.
- location information:
    - depending on the operating mode of the MS, the cell or the routeing area where the MS is registered;
    - the VLR number of the associated VLR (if the Gs interface is implemented);
    - the GGSN address of each GGSN for which an active PDP context exists.

The organisation of the subscriber data in the SGSN is defined in [2] and [17].

The location register function in the GGS stores subscriber data received from the HLR and the SGSN in order to be able to handle originating and terminating packet data transfer:

- subscription information:
    - the IMSI;
    - zero or more PDP addresses.
- location information:
    - the SGSN address for the SGSN where the MS is registered.

The organisation of the subscriber data in the GGSN is defined in [2] and [17].

5.1.1.2 The Equipment Identity Register (EIR)

The EIR is part of the GSM system, responsible for storing in the network the International Mobile Equipment Identities (IMEIs), which may be classified as "white listed", "grey listed" and "black listed" and therefore may be stored in three separate lists.

5.1.1.3 The Mobile-services Switching Centre (MSC) Server (Media Control)

The MSC is the interface between the radio system and the fixed networks handling the circuit switched services to and from the mobile stations. The MSC Server mainly comprises the call control (CC) and mobility control parts of a MSC. To this end it also contains a VLR to hold the mobile subscriber's service data and CAMEL related data.

The MSC Server controls the parts of the call state that pertain to connection control for media channels in a CS-MGW (thus the name Media Control).

5.1.2 Enabling Services

5.1.2.1 Presence Service

5.1.2.1.1 IETF

In [39] IETF describes the Presence Service facilities as accepting information, storing it, and distributing it. Architecturally it has two distinct sets of "clients":

- Presentities: provide presence information to be stored and distributed.
- Watchers: receive presence information from the service.

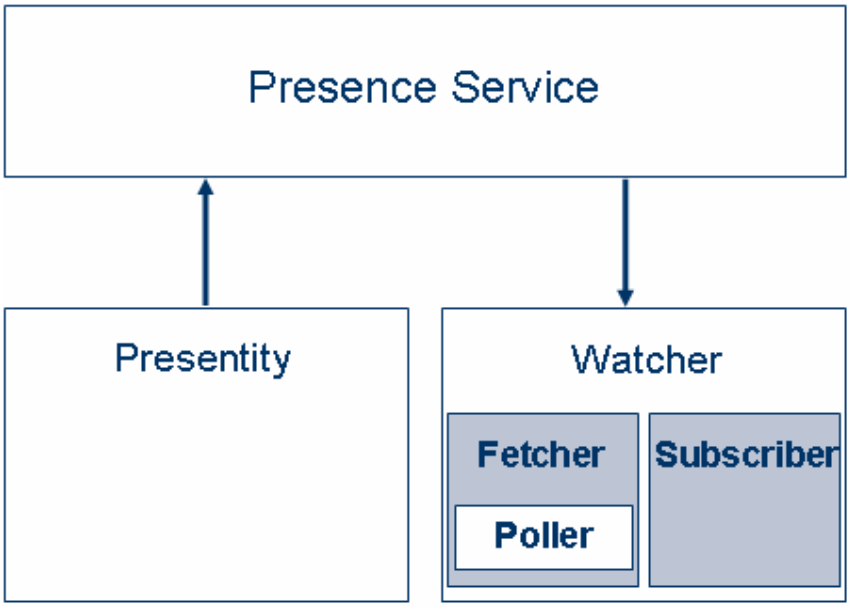


Figure 5.1.2.1.1.1: Overview over presence service support defined by IETF [39]

There are two kinds of Watchers:

- Fetchers: requests the current value of some Presentity's presence information from the Presence Service. A special kind of Fetcher (the Poller) is one that fetches information on a regular basis.
- Subscribers: requests notifications from the Presence Service of changes in some Presentity's presence information.

IETF's model for Presence Information consists of an arbitrary number of elements, called "presence tuples". Each such element consists of:

- a status marker: (e.g. online/offline/busy/away/do not disturb);
- a communication address: optional (containing communication means and communication address);

- other presence markup: optional.

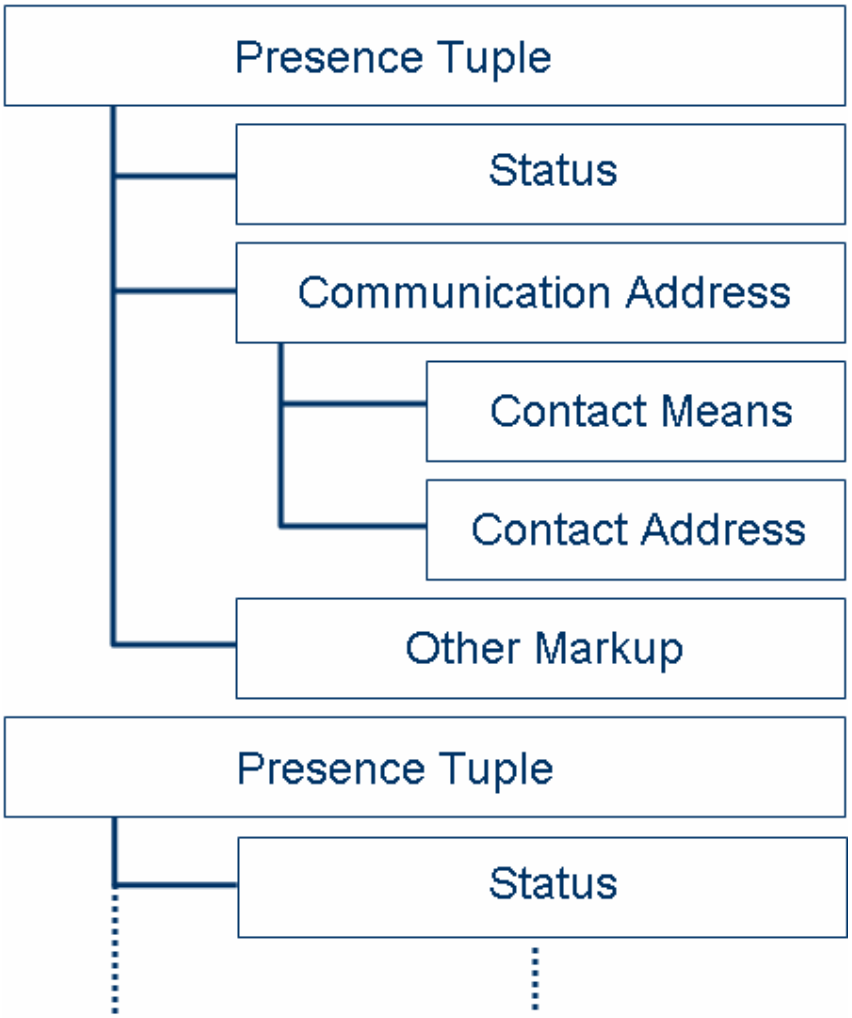


Figure 5.1.2.1.1.2: Overview over presence information defined by IETF [39]

The IETF model includes additional elements that are useful in characterizing how the protocol and markup work:

- Principals: people, groups, and/or software in the "real world" outside the system using the system as a means of coordination and communication.

A Principal interacts with the system via one of several user agents:

- INBOX USER AGENT,
- SENDER USER AGENT,
- PRESENCE USER AGENT,
- WATCHER USER AGENT.

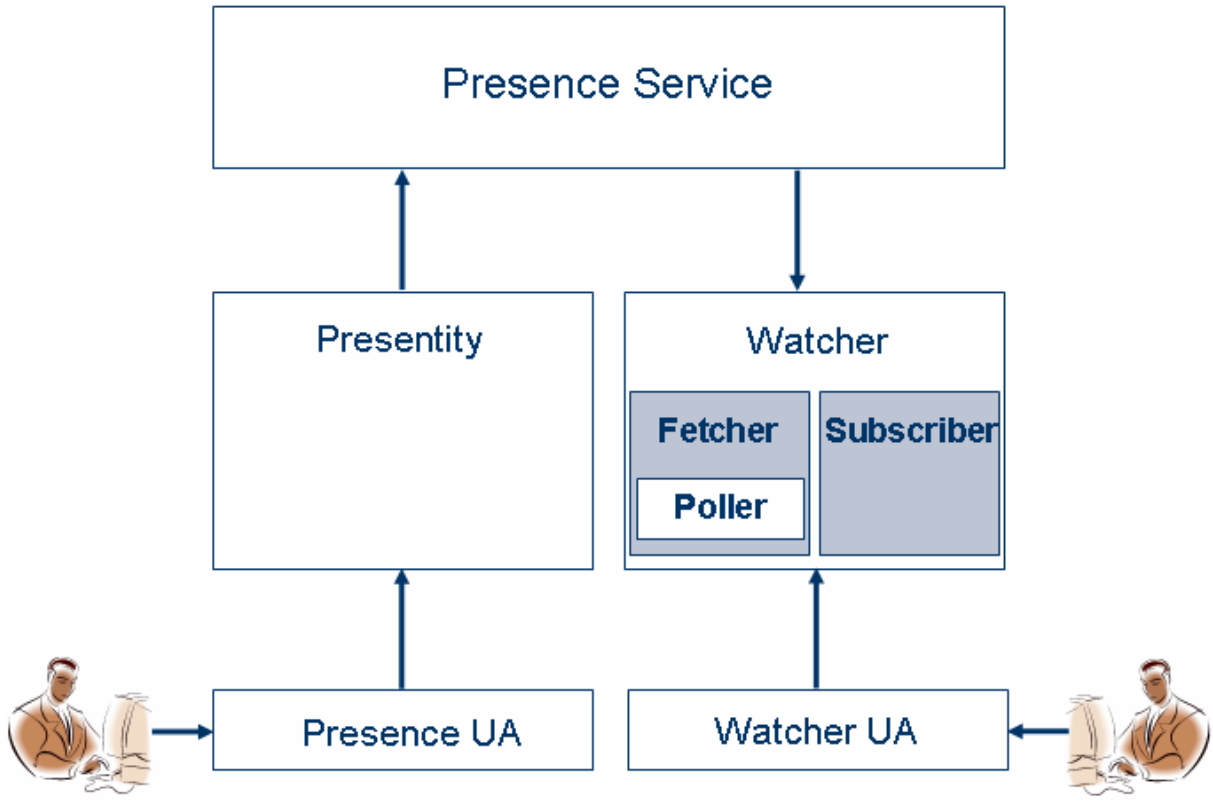


Figure 5.1.2.1.1.3: Overview over presence system defined by IETF [39]

Information Model of [39]:

- Access Rules: constraints on how a presence service makes presence information available to Watchers. For each presentity's presence information, the applicable Access Rules are manipulated by the Presence User Agent of a Principal that controls the Presentity.
- Communication Address: consists of
  - Communication Means: indicates a method whereby communication can take place (e.g. instant message service).
  - Contact Address: a specific point of contact via some Communication Means (e.g. for an instant message service it is the instant inbox address)
- Fetcher: a form of Watcher that has asked the Presence Service to for the Presence Information of one or more Presentities, but no Subscription to be created.
- Notification: a message sent from the Presence Service to a Subscriber when there is a change in the Presence Information of some Presentity of interest, as recorded in one or more Subscriptions.
- Other Presence Markup: any additional information included in the Presence Information of a Presentity. [39] does not define this further.
- Poller: a Fetcher that requests Presence Information on a regular basis.
- Presence Information: consists of one or more Presence Tuples.

- Presence Service: accepts, stores, and distributes Presence Information.
    - May require authentication of Presentities, and/or Watchers.
    - May have different authentication requirements for different Presentities.
    - May have different authentication requirements for different Watchers, and may also have different authentication requirements for different Presentities being watched by a single Watcher.
    - May have an internal structure involving multiple servers and/or proxies. There may be complex patterns of redirection and/or proxying while retaining logical connectivity to a single Presence Service.
    - the service may be implemented as direct communication among Presentity and Watchers.
    - May have an internal structure involving other Presence Services, which may be independently accessible in their own right as well as being reachable through the initial Presence Service.

- Presence Tuple: consists of a Status, an optional Communication Address, and optional Other Presence Markup.

- Presence User Agent: means for a Principal to manipulate zero or more Presentities.

- Presentity (presence entity): provides Presence Information to a Presence Service.

  Note that the presentity is not (usually) located in the presence service: the presence service only has a recent version of the presentity's presence information. The presentity initiates changes in the presence information to be distributed by the presence service.

- Principal: human, program, or collection of humans and/or programs that chooses to appear to the Presence Service as a single actor, distinct from all other Principals.

- Proxy: a server that communicates Presence Information, subscriptions and/or notifications to another server. Sometimes a Proxy acts on behalf of a Presentity or a Watcher.

- Server: an indivisible unit of a Presence Service.

- Status: a distinguished part of the Presence Information of a Presentity. Status has at least the mutually-exclusive values OPEN and CLOSED. There may be other values of Status that may be combined with OPEN and CLOSED or they may be mutually-exclusive with those values.

- Subscriber: a form of Watcher that has asked the Presence Service to notify it immediately of changes in the Presence Information of one or more Presentities.

- Subscription: the information kept by the Presence Service about a Subscriber 's request to be notified of changes in the Presence Information of one or more Presentities.

- Visibility Rules: constraints on how a Presence Service makes Watcher Information available to Watchers. For each Watcher 's Watcher Information, the applicable Visibility Rules are manipulated by the Watcher User Agent of a Principal that controls the Watcher.

- Watcher: requests Presence Information about a Presentity, or Watcher Information about a Watcher, from the Presence Service. Special types of Watcher are Fetcher, Poller, and Subscriber.

- Watcher Information: information about Watchers that have received Presence Information about a particular Presentity within a particular recent span of time. Watcher Information is maintained by the Presence Service, which may choose to present it in the same form as Presence Information.

- Watcher User Agent: means for a Principal to manipulate zero or more Watchers controlled by that Principal.

5.1.2.1.2 3GPP

The Presence Service for GSM and GPRS [21] provides the ability for the home network to manage presence information of a user's device, service or service media even whilst roaming. A user's presence information may be obtained through input from the user, information supplied by network entities or information supplied by elements external to the home network. Consumers of presence information, watchers, may be internal or external to the home network.

The principal architecture of the 3GPP presence service is summarized in the following figure.

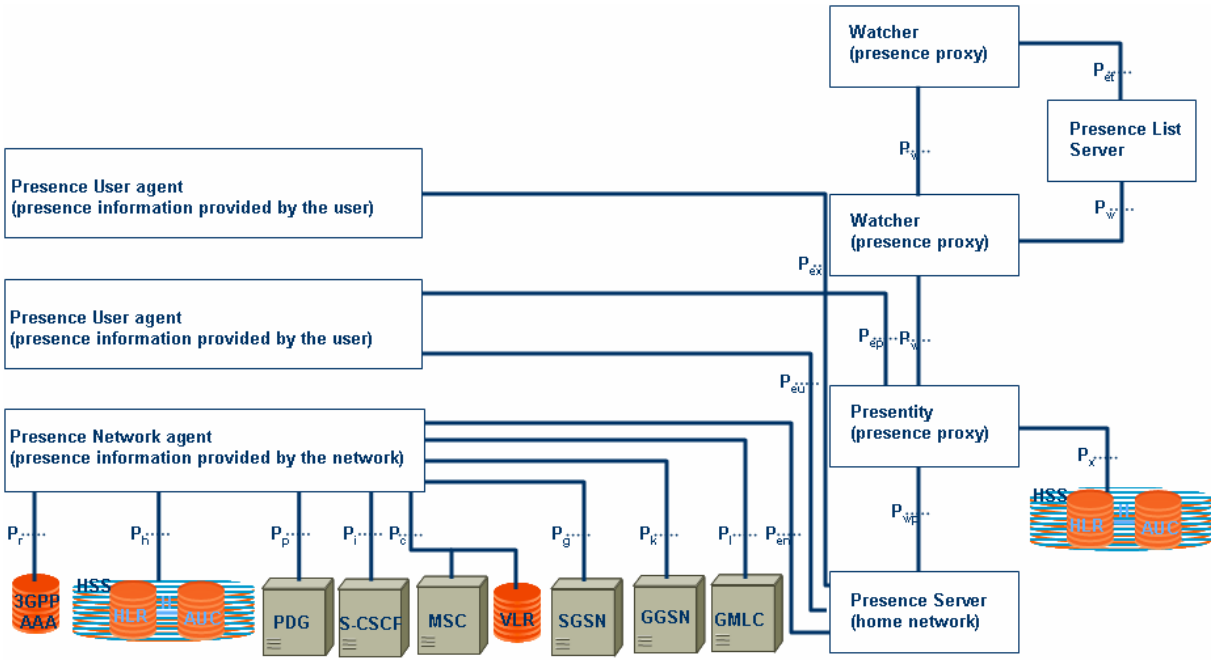


Figure 5.1.2.1.2.1: Reference Architecture for Presence service support defined by 3GPP [21]

[21] describes a 3GPP subscriber by the following attributes: *subscriber's status, communication means status, one or more communication address(es)* (containing *communication means* and *contact address*), *location (subscriber provided location and/or network provided location), priority, text.*

The attributes can be categorised as

- communication means and contact address specific information: *communication means* status, communication means, contact address, priority and text or
- generic information: subscriber's status, location and text

In [21] these attributes are assigned the following values:

- Generic information attributes:
  - Subscriber's status (willing, willing with limitations, not willing, not disclosed),
    The subscriber's status attribute is not intended to be used when interworking with IM clients. Subscribers are able to provide more detailed willingness information as well as other information through the generic Text attribute, and the communication means and contact address specific Text attribute.
  - Location (Last known CGI/SAI and/or geographic co-ordinates and/or free format text and timestamp),
  - Text (free format text).
- Communication means and contact address specific information attributes, if these attributes are used as part of any tuple they shall use following values (values in parenthesis) to enable interoperability:
  - communication means status (online, offline),
  - communication means (Service type (e.g. telephony, SMS, email, multimedia messaging service, instant messaging service)),
  - contact address (E.164 (e.g. MSISDN), SIP URL, Email, Instant message address e.g. IM:name@domain name),
  - Priority (Priority order for each of the defined communication means and contact address),
  - Text (free format text).

5.1.2.1.3 3GPP2

The principal architecture of the 3GPP2 presence service is summarized in the following figure.

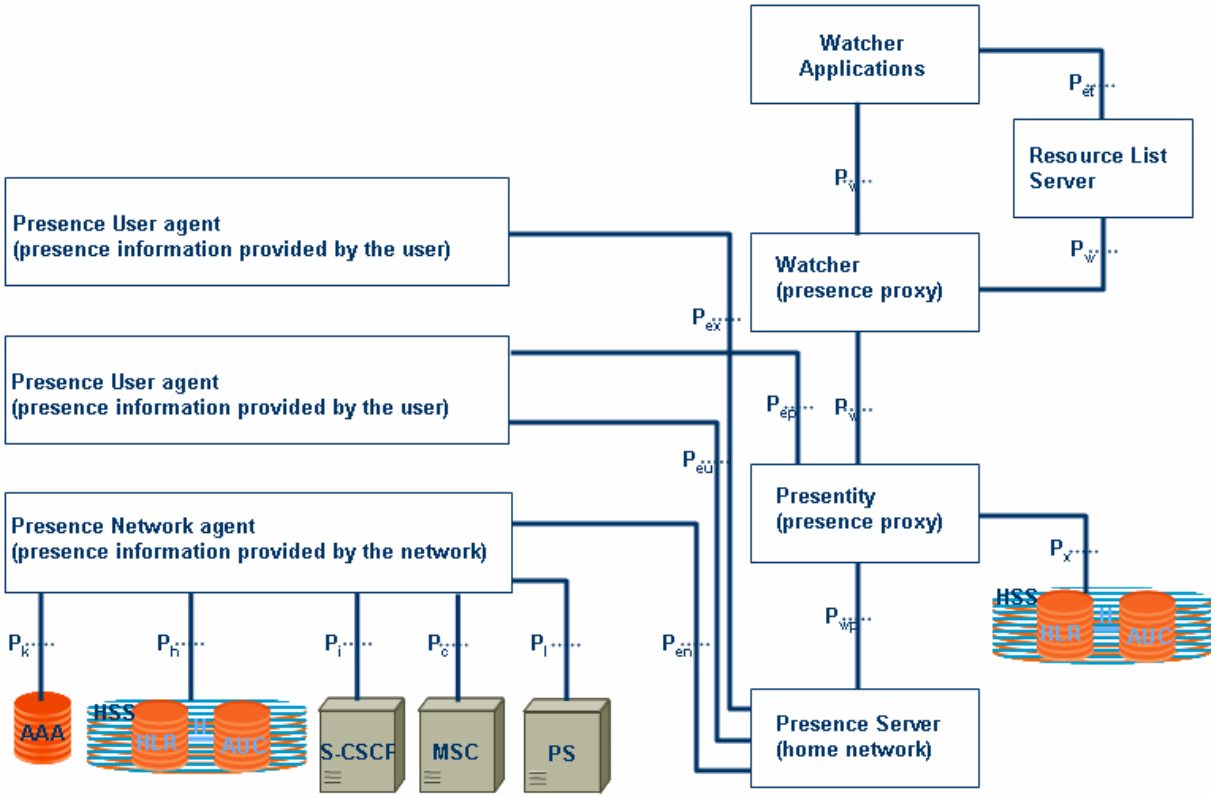


Figure 5.1.2.1.3.1: Reference Architecture for Presence service support defined by 3GPP2 [29]

[29] describes a 3GPP2 subscriber by the following attributes:

- subscriber's status,
- communication means status,
- one or more communication address(es) (containing communication means and contact address),
- location (subscriber provided location and/or network provided location),
- priority,
- text.

The only information element, which differs from 3GPP in the name is the element "contact information", from "communication means" , which replaces "contact address".

5.1.2.1.4 OMA

OMA's Presence Framework leverages IETF's protocols and formats for presence in its SIMPLE (SIP Instant Messaging and Presence Leveraging Extensions) activity (see [RFC3856]).

3GPP [21] and 3GPP2 [29] have defined an aligned Presence Service framework both in the

- "network layer": communication that is required between the Presence Service functional elements (e.g. Presence Server) and various network elements as they are defined in the network architectures of 3GPP and 3GPP2 (e.g. MSC, HLR) and

- "application layer": communication that is required between the various Presence Service elements (e.g. Presence Server and Presence Source), which includes the "application layer" functional entities

defining end-to-end presence information flows.

In addition OMA's Presence Architecture supports presence services that do not leverage core network infrastructure as defined by 3GPP and 3GPP2, but which are still relevant to the mobile domain. The following picture shows the SIMPLE reference architecture on which it is based.

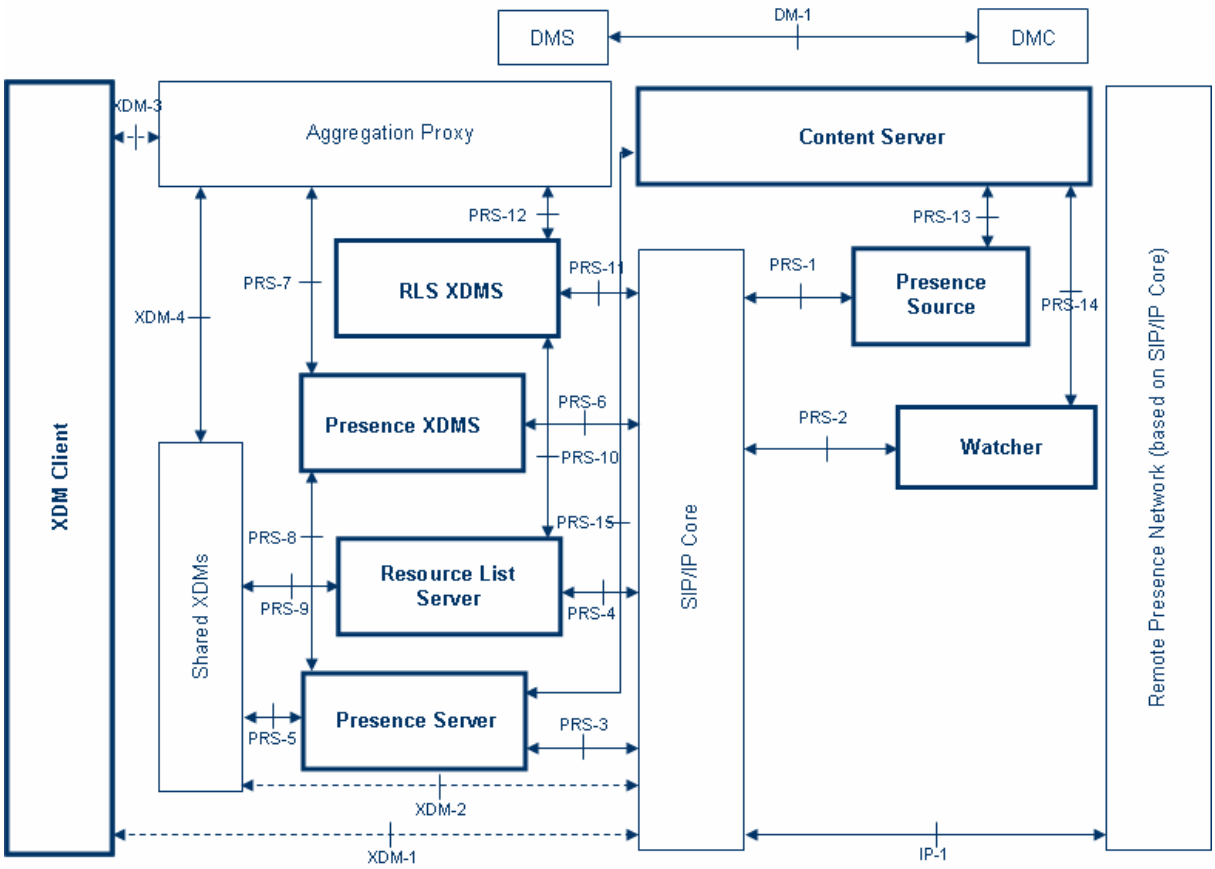


Figure 5.1.2.1.4.1: SIMPLE Reference Architecture for Presence service support based on IETF [31]

Presence Source

The Presence Source is an entity that provides presence information to a Presence Service [31]. It can be located in the user's terminal or within a network entity.

When the SIP/IP Core network corresponds with 3GPP IMS or 3GPP2 MMD networks, the Presence Source can be implemented in a UE or an AS as defined in 3GPP and 3GPP2 respectively.

The following figure shows an architecture for the PNA in the case that no 3GPP/3GPP2 network represents the core.

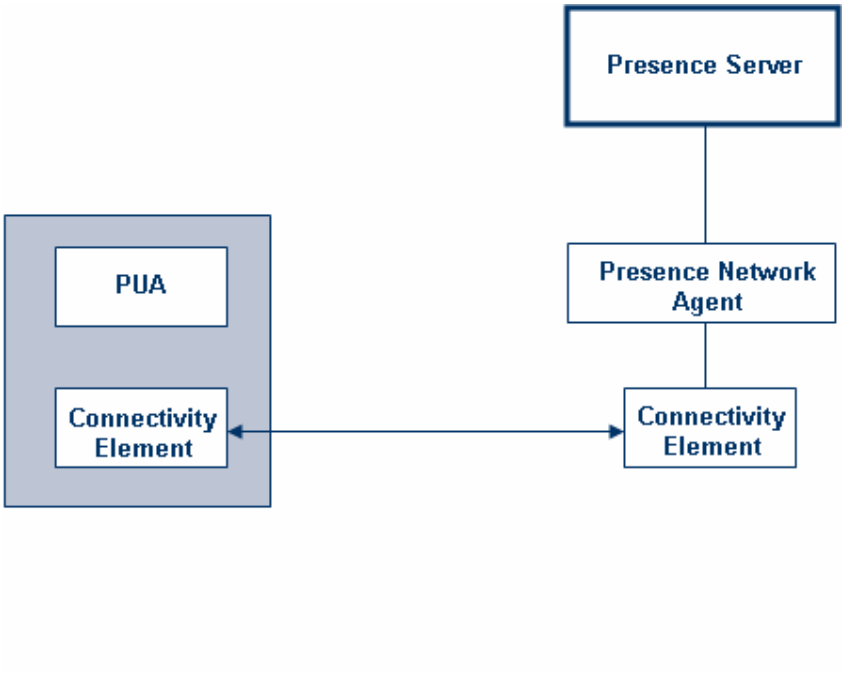


Figure 5.1.2.1.4.2: OMA Reference Architecture for Presence service support in a non-3GPP/3GPP2 case [30]

Watcher

The watcher is an entity that subscribes to presence information about a presentity or list of presentities (i.e. presence list) [30].

When the SIP/IP Core network corresponds with 3GPP IMS or 3GPP2 MMD networks, the watcher can be implemented in a UE or an AS as defined in 3GPP and 3GPP2.

Presence Server

The Presence Server (PS) is an entity that accepts, stores and distributes presence information. The PS performs the following functions:

- Handles publications from one or multiple Presence Source(s) of a certain presentity. This includes
  - refreshing presence information,
  - replacing existing presence information with newly published information, or
  - removing presence information, for a given Presence Source
- Composes the presence information received from one or multiple Presence Source(s) into a single presence document.
- Handles subscriptions from watchers to presence information and generates notifications about the presence information state changes.
- Handles subscriptions from watcher information subscribers to watcher information and generates notifications about the watcher information state changes.
- Authorizes the watcher's subscription to the presentity's presence information and applies policies.
- Applies the watcher's event notification filtering preferences, as appropriate.
- Applies rate control mechanisms to the notifications, as appropriate.

When the SIP/IP Core network corresponds with 3GPP IMS or 3GPP2 MMD networks, the PS is implemented in an AS as defined in 3GPP and 3GPP2 respectively.

Resource List Server

The Resource List Server (RLS) performs the following functions:

- Accepts subscriptions to presence lists.
- Authorizes the watcher's usage of the presence list.
- Creates and manages back-end subscriptions to all presentities in the presence list, on behalf of the watcher.
- Sends notifications to the watcher, based on information received from the back-end subscriptions.
- Applies aggregation and rate control mechanisms to the notifications, as appropriate.

When the SIP/IP Core network corresponds with 3GPP IMS or 3GPP2 MMD networks, the RLS has to be implemented in an AS as defined in 3GPP and 3GPP2 respectively.

XDM Client

The XDM Client has to support the XDM Client procedures and the XCAP application usages.

Presence XDMS

The Presence XDMS has to support the XDM Server procedures and the XCAP application usages.

RLS XDMS

The RLS XDMS has to support the XDM Server procedures and the XCAP application usages.

Content Server

The Content Server has to support IETF's HTTP GET and PUT methods [30], and the procedures defined in [30].

The Content Server has to store a MIME object when receiving it in an HTTP PUT request behind the HTTP URI therein. .

The Content Server has to return a MIME object in a 200 OK response to an HTTP GET request. The Content Server has to fetch the MIME object from the Request URI of the HTTP GET request.

The Content Server can be used by Presence Sources, Watchers and the Presence Server as described in figure xx.

The Presence data model

The Presence Data Model defined by IETF (see above) is categorized in four key components: the Presentity URI, the Person, the Service and the Device:

- The Presentity's URI component indicating the Presentity's identifier (e.g. SIP URI, tel. URI)
- The Person components model the information about the Presentity: The "person" component models information about the Presentity whom the presence data is trying to describe. It has the following components:
    - Overriding Willingness
    - Activity
    - Location
    - Time Zone
    - Mood
    - Icon
    - Class

- Geographical Location (location info, usage rules)
- Note
- Timestamp

The model supports only one "person" component per presentity, or a group, which appears to the watcher as a single Presentity.

Note: according to [30] more than one "person" component instance may exist in the Presence document in cases where composition policy in the PS cannot clearly semantically differentiate between the multiple instances of the same component.

- The Service components model the forms of communication used by the Presentity and has potentially access to:
  - Application-specific Availability (registration state, barring state)
  - Application-specific willingness
  - Icon
  - Session Participation
  - Service Description
  - Class
  - Per service device identifier (device-id)
  - Communication address (contact)
  - Timestamp

One important characteristic of each "service" might be the devices on which that service executes. Each device is uniquely identified by the device identifier <deviceID>. A service may contain zero or more <deviceID> elements to indicate which devices that service is available on. The Presence document may contain information on each device, but this is a separate part of the document modelled by the "device" component.

The "service" component has to be mapped to the <tuple> element specified in [31].

- The Device components model the physical pieces of equipment used by the Presentity:
  - Network Availability
  - Geographical Location (location-info, location-rules)
  - Device identifier (device-id)
  - Timestamp

Examples of Presence information that can be represented by "device" elements include mobile phones, PCs and PDAs.

The mapping of services to devices is many to many. Devices are uniquely identified with a device identifier. The model supports only one "device" component per device identifier, however the Presence Sources publish their own "device" component instances. The PS composes the multiple instances into one component and resolves conflicts among the Presence Sources according to [30].

The "device" component has to be mapped to the <device> element, which is specified in [31].

For a given presentity, the value of the <deviceID> element of the <device> element have to be unique for each device used by the presentity.

A version 4 UUID as defined in [30] has to be used for <deviceID> to uniquely identify the device. This is a purely random identifier, providing uniqueness. It is not allowed to change over the lifetime of the device and

has to be stored in a non-volatile memory. It has to be used in all the Presence publications requiring the use of <deviceID>.

The relationship between the data elements is shown in the following figure xx.

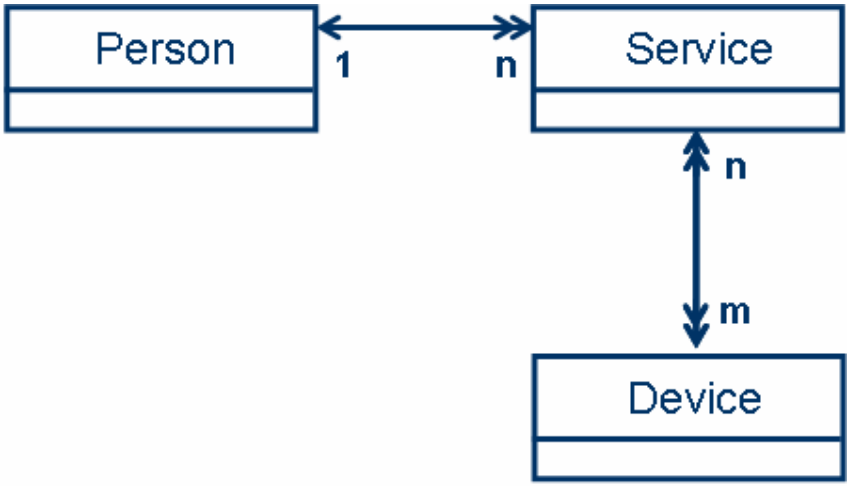


Figure 5.1.2.1.4.3: Presence Object Model defined by OMA [31]

5.1.2.2 Location Service

5.1.2.2.1 3GPP

The functional model defined in [18] and presented in the following figure includes functional entities for both CS and PS related LCS. In addition, it consists of all the entities needed for different positioning methods:

- i.e. network based,
- mobile based,
- mobile assisted, and
- network assisted positioning, exploiting
    - either uplink or
    - downlink measurements.

NOTE: The UE may use e.g. the GPS positioning mechanism, but still demand e.g. auxiliary measurements from the serving network. RAN specific functional entities are specified in TS 25.305 [19] for UTRAN and in TS 43.059 [20] for GERAN.

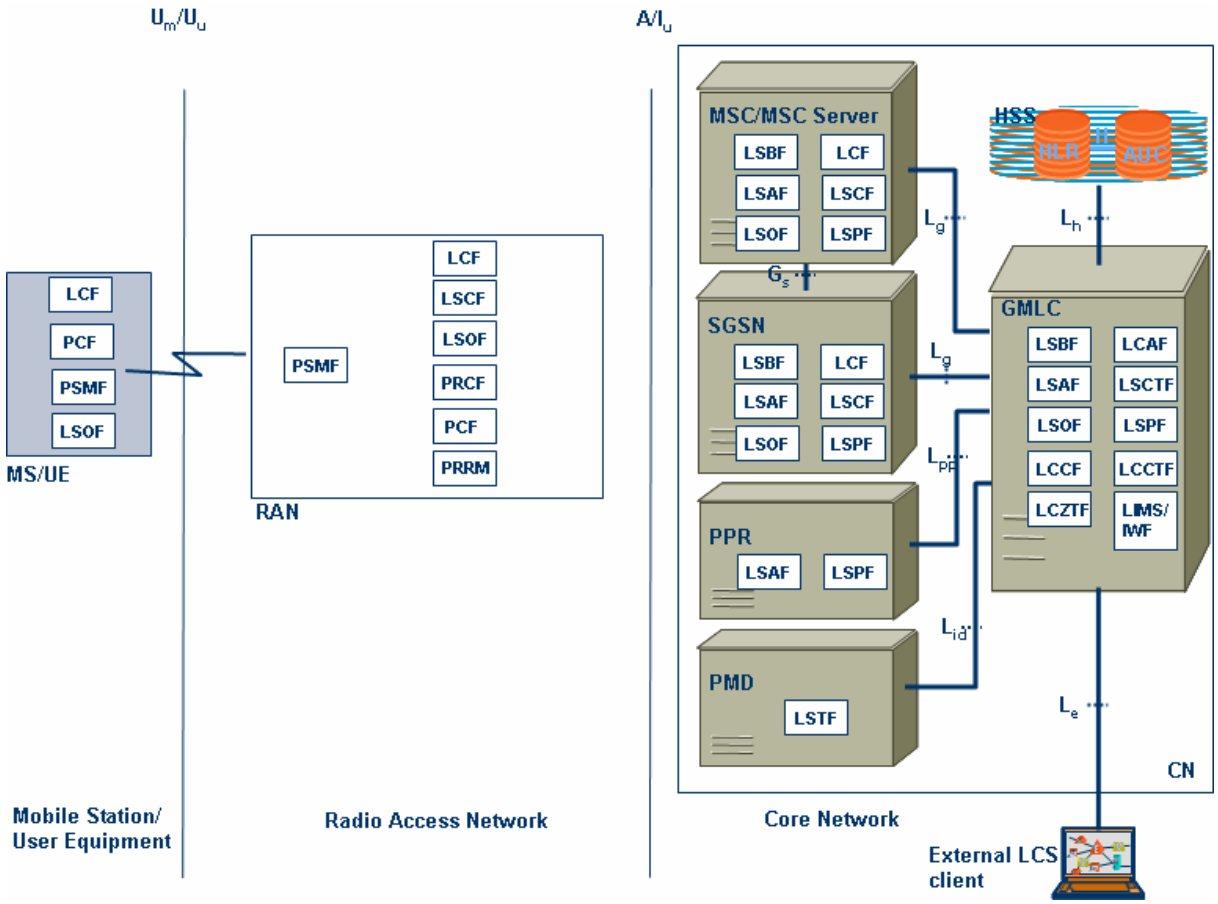


Figure 5.1.2.2.1.1: Generic Logical LCS Architecture defined by 3GPP [18]

The LCS client Subscription profile shall contain a minimum set of parameters assigned on per LCS client basis for an agreed contractual period. The LCS client profile shall contain the following set of access parameters:

- LCS client identity;
- allowed LCS request types (i.e. LIR, LDR or both) (see note);
- maximum number of subscribers allowed in a single LCS request;
- priority;
- position override indicator;
- state(s);
- event(s) (applicable to LDR requests only);
- local coordinate system;
- LCS client access barring list (optional);
- PLMN access barring list applicability.

5.1.2.2.2 OMA

For certain authorized LCS clients internal to the PLMN, a subscription profile is unnecessary. These clients are empowered to access any defined service that is not barred for an UE subscriber. This permits positioning of emergency calls without the need for pre-subscription.

The OMA Mobile Location Service V1.0 (MLS V1.0 [34]) consists of a set of location specifications complying with 3GPP Release 6 LCS Specification [18].

The Architecture of MLS V1.0 as described in [34] defines four reference points Le, Lr, Lpp and Lid. It also describes the components in the architecture as shown in Figure xx:

- Location Privacy Checking Entity that is described in [18] clause 6.3.11 and 6.3.12.
- MLS Client that is described in [18] clause 6.3.2.
- Requesting Location Server, Home Location Server and Visited Location Server that are described in [18] clause 6.3.3.

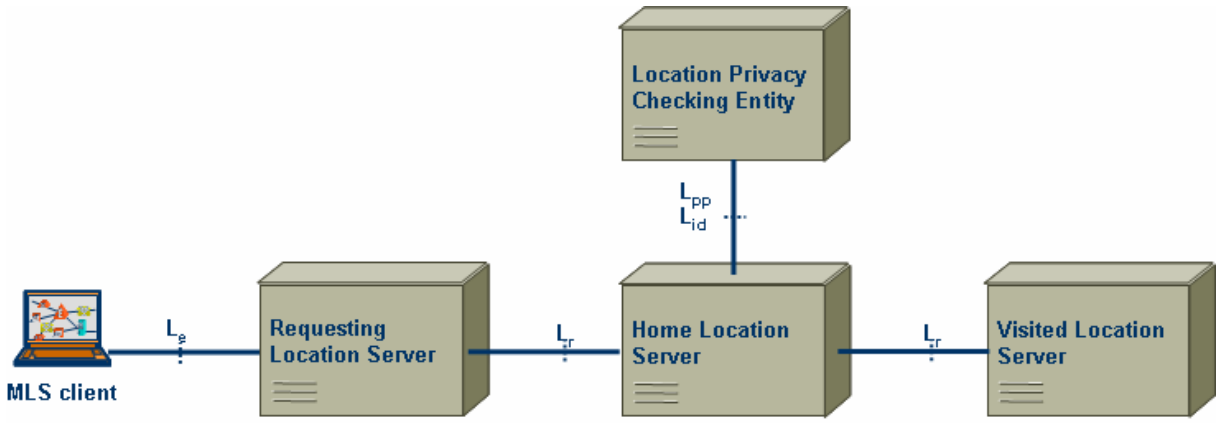


Figure 5.1.2.2.2.1: Logical LCS Architecture defined by OMA [34]

5.1.2.3 XDM

5.1.2.3.1 OMA

Various network functions such as, Presence, Push to Talk Over Cellular (PoC), Instant Messaging (IM), etc. need to access and manipulate certain information (e.g. the list of PoC participants who can take part in a PoC session as well as additional PoC-specific properties, the lists of PoC callers who are allowed/not allowed to call a given user, a list of users who are potential presentities, so that this list can be used to collectively subscribe to the presence status of each member in that list, an access control policy for Presence, which specifies whether a particular watcher is authorized to subscribe to a certain set of events).

Note: such information is not always composed of pure lists (of principals), but can be a combination of lists together with other properties that define an end-user's personalization of the service behaviour.

The XDM enabler [35] specifies documents that can be shared by multiple enablers. One such case is a particular type of list, the URI List, which is a convenient way for a principal to group together a number of end users (e.g., "Friends" or "Family) or other resources, where such a list is expected to be reused for a number of different enablers. Such a list can be re-used wherever a principal has a need to collectively refer to a group of other end users or resources.

URI lists: The end-user can store URI information about other end-users to later initiate URI with them or to subscribe for their presence. A URI list is an essential basis for other OMA enabler (such as, PoC or messaging) as the addresses on the URI list are used to set up a session.

Group lists: Groups described in [35] are used for communication sessions (e.g. PoC sessions) or immediate messaging sessions for chat rooms. The group has properties (e.g., "open" or "restricted") that sets the rules for the communication.

Access Control list: According to [35] it shall be possible for Access Control lists to be created, modified and deleted by the subscriber or another authorised end-user.

XDM Client:

- Has to support the XDM Client procedures (document management, subscription to changes in XML documents), and
- the XCAP application usages (XCAP-Server Capabilities, XML Documents Directory).

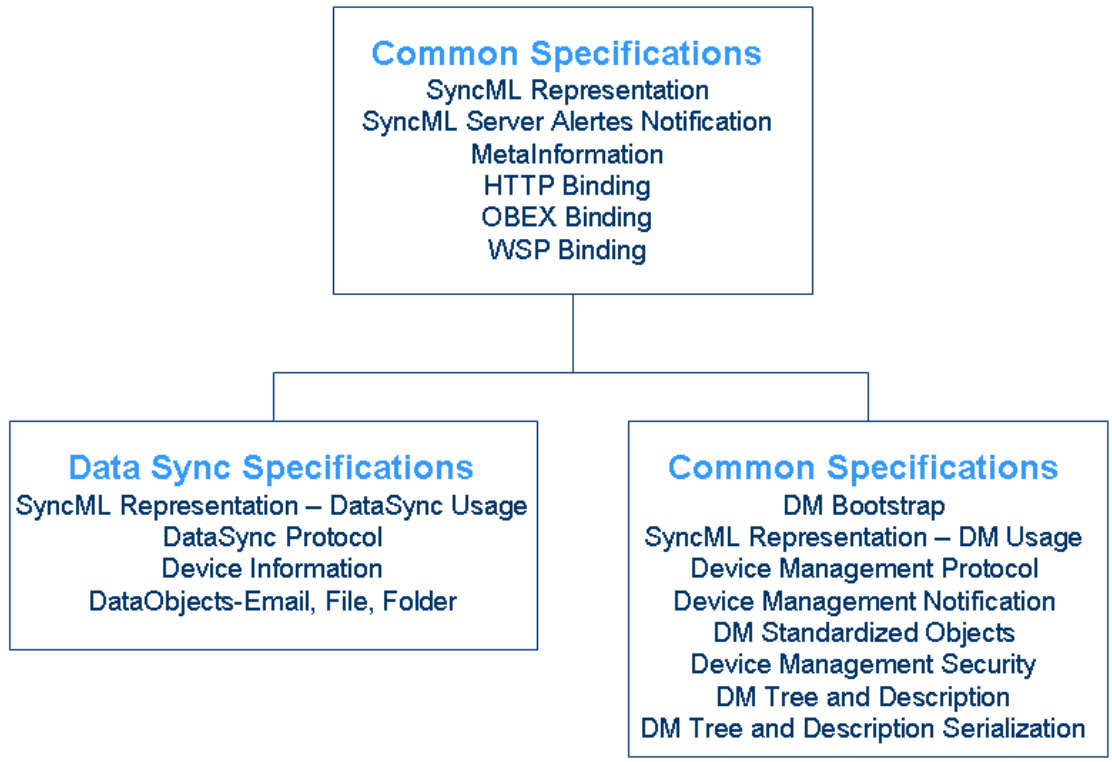
When the SIP/IP Core network corresponds with 3GPP IMS or 3GPP2 MMD networks, the XDM Client can be implemented in a UE or an AS as defined in [22].

Aggregation Proxy:

The Aggregation Proxy is the contact point for the XDM Client implemented in an UE to access XML documents stored in any XDMS. When realized with 3GPP IMS or 3GPP2 MMD networks, the Aggregation Proxy shall act as an Authentication Proxy defined in [23].

5.1.2.4 Device Management

5.1.2.4.1 OMA



Note: The OMA DM v1.2 specifications are based on the OMA Device Management (DM) v1.1.2 specifications and make use of the OMA SyncML Common v1.2 specifications as specified in the OMA SyncML common specification.

Figure 5.1.2.4.1.1: Document Structure for Device Management defined by OMA [35]

The SyncML Initiative, Ltd. was a not-for-profit corporation formed by a group of companies who co-operated to produce an open specification for data synchronization and device management.

The SyncML Initiative merged with the Open Mobile Alliance in November 2002. The SyncML legacy specifications were converted to the OMA format with the 1.1.2 versions of OMA SyncML Common, OMA Data Synchronization and OMA Device Management in May 2002.

The SyncML specification contains the following main components:

- An XML-based representation protocol
- A synchronization protocol and a device management protocol
- Transport bindings for the protocol
- A device description framework for device management

OMA DM management objects are defined using the OMA DM Device Description Framework [DMTND], or DDF.

The following figure gives an example of OMA's definition of device information.

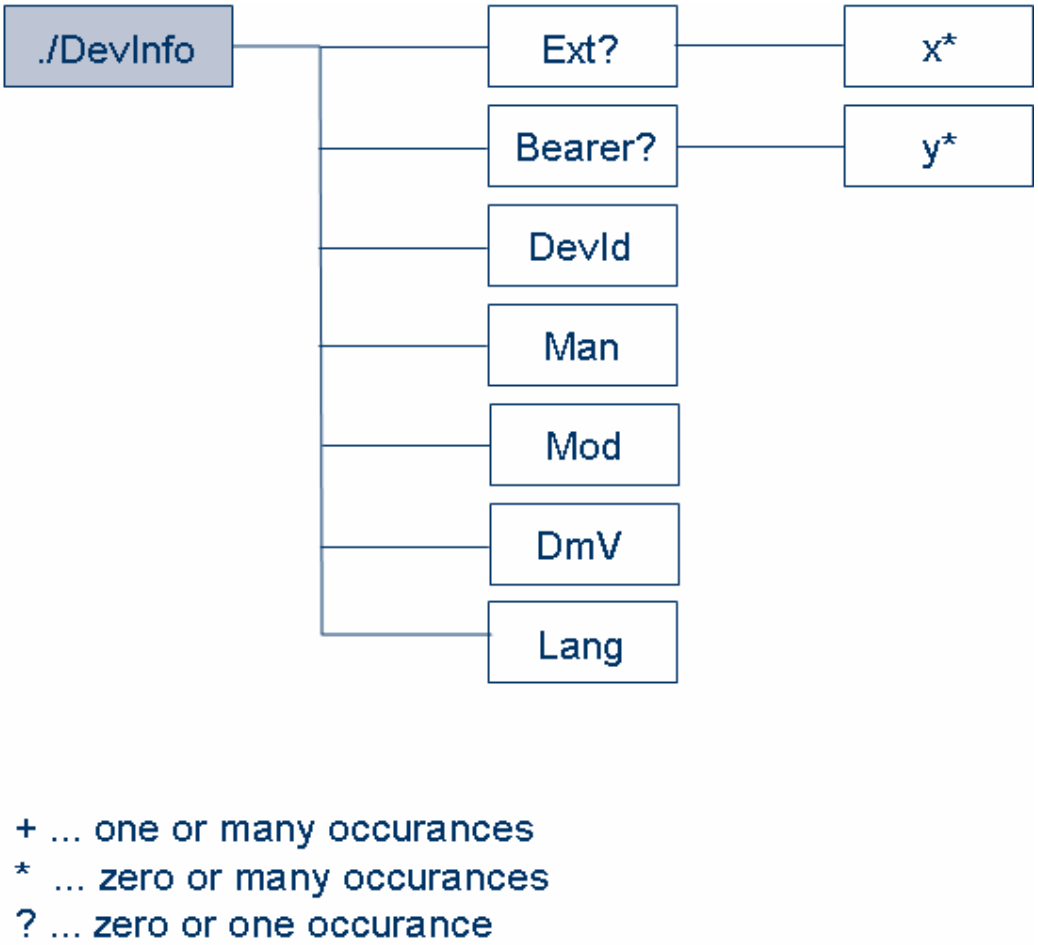


Figure 5.1.2.4.1.2: Example of a management object for Device Management defined by OMA [36]

The above figure provides some points to note:

- All the blocks with names in place occur exactly once, except Ext and Bearer that are optional and may not be present at all.
- The named node DevInfo has child nodes and is an interior node. With the exception of Ext and Bearer, which can have children, the other named nodes are leaf nodes.
- The un-named leaf nodes marked with * can be represented by any number of instantiated nodes at run-time, including none. The only limit is that the node names must be unique and memory must be available to store the nodes.

As part of the common definitions OMA [36] provides the following management objects:

DMAcc: The management object is used to manage settings for the OMA DM protocol.

DevInfo:

- Ext: An optional, interior node, designating the only branch of the DevInfo sub tree into which extensions can be added, permanently or dynamically.
- Bearer: An optional, interior node of the DevInfo sub tree in which items related to the bearer (CDMA, etc.) are stored. Use of this sub tree can be mandated by other standards.

- DevId: A unique identifier for the device, which should be globally unique and must be formatted as a URN as defined in RFC2141.
- Man: The manufacturer identifier.
- Mod: A model identifier (manufacturer specified string).
- DmV: OMA device management client version identifier (manufacturer specified string).
- Lang: The current language setting of the device. The syntax of the language tags and their use are defined in RFC1766]. Language codes are defined by ISO in the standard ISO639.

DevDetail:

- Ext: An optional, interior node, designating the only branch of the DevDetail sub tree into which extensions can be added, permanently or dynamically.
- Bearer: An optional, interior node, designating a branch of the DevDetail sub tree into which items related to the bearer (CDMA, etc.) are stored. Use of this sub tree can be mandated by other standards.
- URI/MaxDepth: Specifies the maximum depth of the management tree supported by the device. The maximum depth of the tree is defined as the maximum number of URI segments that the device supports. The value is a 16 bit, unsigned integer encoded as a numerical string. The value ‘0' means that the device supports a tree of ‘unlimited' depth.
- URI/MaxTotLen: Specifies the maximum total length of any URI used to address a node or node property. The maximum total length of a URI is defined as the largest total number of characters making up the URI which the device can support. Note that depending on the character set this might not be the same as the number of bytes. The value is a 16 bit, unsigned integer encoded as a numerical string. The value ‘0' means that the device supports URI of ‘unlimited' length.
- URI/MaxSegLen: Specifies the maximum total length of any URI segment in a URI used to address a node or node property. The maximum total length of a URI segment is defined as the largest number of characters which the device can support in a single URI segment. Note that depending on the used character set this might not be the same as the number of bytes. The value is a 16 bit, unsigned integer encoded as a numerical string. The value ‘0' means that the device supports URI segments of ‘unlimited' length.
- DevTyp: Device type, e.g. PDA, pager, or phone.
- OEM: Original Equipment Manufacturer of the device.
- FwV: Firmware version of the device.
- SwV: Software version of the device.
- HwV: Hardware version of the device.
- LrgObj: Indicates whether the device supports the OMA DM Large Object Handling specification, as defined in [DMPRO].

Inbox:

In some circumstances a Management Object's URI is not the preferred addressing method and the management object identifier is enough information for the device to resolve a suitable location for that Management Object. In that case the URI: "./Inbox" is a reserved location for this purpose.

As part of the SyncML definitions OMA [37] provides the following management objects:

SyncML DM:

The SyncML DM management object consists of two parts.

- The first part is the DMAcc node which is where the SyncML DM specific settings are stored. These settings are collectively referred to as a SyncML DM account.

- The second part is Con, which is used for connectivity settings needed to communicate with a SyncML DM server. The sub tree of the Con node is similar to what a generic connectivity management object might look like and there is also substantial overlap with WAP Provisioning parameters here.

DevInfo: see above

DevDetail: see above

5.1.2.5 Authorization and Authentication

5.1.2.5.1 ITU-T

ITU-T Rec. X.509 [44] defines a framework for public-key certificates, which includes the specification of

- data objects used to represent the certificates themselves as well as
- revocation notices for issued certificates that should no longer be trusted.

In addition ITU-T Rec. X.509 [44] defines a framework for attribute certificates, which includes specification of

- data objects used to represent the certificates themselves as well as
- revocation notices for issued certificates that should no longer be trusted.

In order for a user to be able to trust a public-key for another user, for instance to authenticate the identity of that user, ITU-T Rec. X.509 [44] defines how the public-key is to be obtained from a trusted source. Such a source, called a Certification Authority (CA), certifies a public key by issuing a public-key certificate which binds the public-key to the entity which holds the corresponding private-key.

The following ASN.1 data type has been defined by ITU-T Rec. X.509 [44] to represent certificates:

```
Certificate                 ::=     SIGNED { SEQUENCE {
    version             [0]  Version DEFAULT v1,
    serialNumber             CertificateSerialNumber,
    signature                AlgorithmIdentifier,
    issuer                   Name,
    validity                 Validity,
    subject                  Name,
    subjectPublicKeyInfo     SubjectPublicKeyInfo,
    issuerUniqueIdentifier  [1]  IMPLICIT UniqueIdentifier OPTIONAL,
                                 -- if present, version shall be v2 or v3
    subjectUniqueIdentifier [2]  IMPLICIT UniqueIdentifier OPTIONAL,
                                 -- if present, version shall be v2 or v3
    extensions              [3]  Extensions OPTIONAL
                                 -- If present, version shall be v3 -- } }

    Version                 ::=     INTEGER { v1(0), v2(1), v3(2) }

    CertificateSerialNumber ::=     INTEGER

AlgorithmIdentifier         ::=     SEQUENCE {
algorithm           ALGORITHM.&id ({SupportedAlgorithms}),
parameters          ALGORITHM.&Type ({SupportedAlgorithms}{ @algorithm}) OPTIONAL }
-- Definition of the following information object set is deferred, perhaps to standardized
-- profiles or to protocol implementation conformance statements. The set is required to
-- specify a table constraint on the parameters component of AlgorithmIdentifier.
-- SupportedAlgorithms      ALGORITHM           ::=  { ... }

Validity                    ::=     SEQUENCE {
    notBefore       Time,
    notAfter        Time }

SubjectPublicKeyInfo                ::=     SEQUENCE {
    algorithm               AlgorithmIdentifier,
    subjectPublicKey        BIT STRING }
```

```
Time ::= CHOICE {
    utcTime            UTCTime,
    generalizedTime    GeneralizedTime }

Extensions ::= SEQUENCE OF Extension

Extension ::=  SEQUENCE {
    extnId      EXTENSION.&id ({ExtensionSet}),
    critical    BOOLEAN DEFAULT FALSE,
    extnValue   OCTET STRING
                -- contains a DER encoding of a value of type &ExtnType
                -- for the extension object identified by extnId -- }

ExtensionSet  EXTENSION      ::=      { ... }
```

- version: the version of the encoded certificate. If the extensions component is present in the certificate, version shall be v3. If the issuerUniqueIdentifier or subjectUniqueIdentifier component is present version shall be v2 or v3.
- serialNumber: is an integer assigned by the CA to each certificate. The value of serialNumber shall be unique for each certificate issued by a given CA (i.e., the issuer name and serial number identify a unique certificate).
- signature: contains the algorithm identifier for the algorithm and hash function used by the CA in signing the certificate
- issuer: identifies the entity that has signed and issued the certificate.
- validity: time interval during which the CA warrants that it will maintain information about the status of the certificate.
- Subject: identifies the entity associated with the public-key found in the subject public key field.
- subjectPublicKeyInfo: carries the public key being certified and identifies the algorithm which this public key is an instance of .
- issuerUniqueIdentifier: uniquely identifies an issuer in case of name re-use.
- subjectUniqueIdentifier: uniquely identifies a subject in case of name re-use.
- Extensions: field, which allows addition of new fields to the structure without modification to the ASN.1 definition. An extension field consists of an extension identifier, a criticality flag, and an encoding of a data value of an ASN.1 type associated with the identified extension.

If revocation lists are published, they the following information:

- Certificate revocation list;
- Authority revocation list;
- Delta revocation list;
- Attribute certificate revocation list;
- Attribute authority revocation list.

The following ASN.1 data type has been defined by ITU-T Rec. X.509 [44] to represent revocation lists:

```
CertificateList          ::=     SIGNED { SEQUENCE {
    version                      Version OPTIONAL,
                                 -- if present, version shall be v2
    signature                    AlgorithmIdentifier,
    issuer                       Name,
    thisUpdate                   Time,
    nextUpdate                   Time OPTIONAL,
    revokedCertificates          SEQUENCE OF SEQUENCE {
        serialNumber                     CertificateSerialNumber,
```

```
        revocationDate                  Time,
        crlEntryExtensions          Extensions OPTIONAL } OPTIONAL,
    crlExtensions     [0]           Extensions OPTIONAL }}
```

- version: version of the encoded revocation list. If the extensions component flagged as critical is present in the revocation list, version shall be v2. If no extensions component flagged as critical is present in the revocation list, version may either be absent or present as v2.
- Signature: contains the algorithm identifier for the algorithm used by the authority to sign the revocation list.
- Issuer: identifies the entity that has signed and issued the revocation list.
- thisUpdate: date/time at which this revocation list was issued.
- nextUpdate: if present, indicates the date/time by which the next revocation list in this series will be issued.
- revokedCertificates: identifies certificates that have been revoked. The revoked certificates are identified by their serial numbers.
- crlExtensions: if present, contains one or more CRL extensions.

The attribute certificate framework defined in ITU-T Rec. X.509 [44] provides a foundation upon which Privilege Management Infrastructures (PMI) can be built. These infrastructures can support applications such as access control.

The binding of a privilege to an entity is provided by an authority through a digitally signed data structure called an attribute certificate or through a public-key certificate containing an extension defined explicitly for this purpose.

An attribute certificate using system needs to validate a certificate prior to using that certificate for an application. Procedures for performing that validation are also defined here, including verifying the integrity of the certificate itself, its revocation status, and its validity with respect to the intended use.

An attribute certificate is a separate structure from a subject's public key certificate. A subject may have multiple attribute certificates associated with each of its public key certificates. There is no requirement that the same authority create both the public key certificate and attribute certificate(s) for a user; in fact separation of duties will frequently dictate otherwise.

The attribute certificate is defined as follows ITU-T Rec. X.509 [44].

```
AttributeCertificate ::= SIGNED {AttributeCertificateInfo}

AttributeCertificateInfo ::= SEQUENCE
    {
    version                     AttCertVersion, --version is v2
    holder                      Holder,
    issuer                      AttCertIssuer,
    signature                   AlgorithmIdentifier,
    serialNumber                CertificateSerialNumber,
    attrCertValidityPeriod      AttCertValidityPeriod,
    attributes                  SEQUENCE OF Attribute,
    issuerUniqueID              UniqueIdentifier OPTIONAL,
    extensions                  Extensions    OPTIONAL
    }

AttCertVersion ::= INTEGER { v2(1) }

Holder    ::=   SEQUENCE
            {
            baseCertificateID       [0] IssuerSerial        OPTIONAL,
                -- the issuer and serial number of the holder's Public Key Certificate
            entityName              [1] GeneralNames        OPTIONAL,
                -- the name of the entity or role
            objectDigestInfo        [2] ObjectDigestInfo    OPTIONAL
                -- used to directly authenticate the holder, e.g. an executable
-- at least one of baseCertificateID, entityName or objectDigestInfo shall be present --}
```

```
ObjectDigestInfo    ::= SEQUENCE {
    digestedObjectType  ENUMERATED {
        publicKey               (0),
        publicKeyCert           (1),
        otherObjectTypes        (2) },
    otherObjectTypeID           OBJECT IDENTIFIER  OPTIONAL,
    digestAlgorithm             AlgorithmIdentifier,
    objectDigest                BIT STRING }

AttCertIssuer ::=     [0]   SEQUENCE {
issuerName                      GeneralNames OPTIONAL,
baseCertificateID           [0] IssuerSerial OPTIONAL,
objectDigestInfo            [1] ObjectDigestInfo OPTIONAL }
-- At least one component shall be present
    ( WITH COMPONENTS { ..., issuerName PRESENT } |
    WITH COMPONENTS { ..., baseCertificateID PRESENT } |
    WITH COMPONENTS { ..., objectDigestInfo PRESENT } )

IssuerSerial  ::=  SEQUENCE {
    issuer        GeneralNames,
    serial        CertificateSerialNumber,
    issuerUID     UniqueIdentifier OPTIONAL }

AttCertValidityPeriod  ::= SEQUENCE {
    notBeforeTime         GeneralizedTime,
    notAfterTime          GeneralizedTime }
```

- version: differentiates between different versions of the attribute certificate. For attribute certificates issued in accordance with the syntax in this Specification, version shall be v2.
- Holder: field which conveys the identity of the attribute certificate's holder.
    - baseCertificateID: if present, identifies a particular public-key certificate that is to be used to authenticate the identity of this holder when asserting privileges with this attribute certificate.
    - entityName: if present, identifies one or more names for the holder.
    - objectDigestInfo: if present, is used directly to authenticate the identity of a holder, including an executable holder (e.g. an applet)..
- issuer field: conveys the identity of the AA that issued the certificate.
    - issuerName: if present, identifies one or more names for the issuer.
    - baseCertificateID: if present, identifies the issuer by reference to a specific public-key certificate for which this issuer is the subject.
    - objectDigestInfo: if present, identifies the issuer by providing a hash of identifying information for the issuer.
- Signature: identifies the cryptographic algorithm used to digitally sign the attribute certificate.
- serialNumber: uniquely identifies the attribute certificate within the scope of its issuer.
- attrCertValidityPeriod: conveys the time period during which the attribute certificate is considered valid, expressed in GeneralizedTime format.
- attributes: contains the attributes associated with the holder that are being certified (e.g. the privileges).
- issuerUniqueID: may be used to identify the issuer of the attribute certificate in instances where the issuer component is not sufficient.
- extensions: allows addition of new fields to the attribute certificate.

5.1.2.5.2 IETF

The Diameter base protocol [41] provides an Authentication, Authorization and Accounting (AAA) framework for network functions such as network access or IP mobility. Diameter is applicable in both local Authentication, Authorization and Accounting and roaming situations.

Diameter can provide two different types of services to network functions:

- Authentication and authorization, optionally making use of accounting.
- Accounting only

[41] defines the following AVPs for Authentication and Authorization.

- Origin-State-Id AVP: Origin-State-Id is used to allow rapid detection of terminated sessions.
- Auth-Request-Type AVP: The Auth-Request-Type AVP is included in network function-specific auth requests to inform the peers whether a user is to be authenticated only, authorized only or both.
- Session-Id AVP: The Session-Id AVP is used to identify a specific session.
- Authorization-Lifetime AVP: The Authorization-Lifetime AVP contains the maximum number of seconds of service to be provided to the user before the user is to be re-authenticated and/or re-authorized
- Auth-Grace-Period AVP: The Auth-Grace-Period AVP contains the number of seconds the Diameter server will wait following the expiration of the Authorization-Lifetime AVP before cleaning up resources for the session.
- Auth-Session-State AVP: The Auth-Session-State AVP specifies whether state is maintained for a particular session.
- Re-Auth-Request-Type AVP: The Re-Auth-Request-Type AVP is included in network function-specific auth answers to inform the client of the action expected upon expiration of the Authorization-Lifetime.
- Session-Timeout AVP: The Session-Timeout AVP contains the maximum number of seconds of service to be provided to the user before termination of the session.
- User-Name AVP: The User-Name AVP (AVP Code 1) [RADIUS] is of type UTF8String, which contains the User-Name, in a format consistent with the NAI specification.
- Termination-Cause AVP: The Termination-Cause AVP is used to indicate the reason why a session was terminated on the access device.
- Origin-State-Id AVP: The Origin-State-Id AVP is a monotonically increasing value that is advanced whenever a Diameter entity restarts with loss of previous state, for example upon reboot.
- Session-Binding AVP: The Session-Binding AVP may be present in network function-specific authorization answer messages.
- Session-Server-Failover AVP: The Session-Server-Failover AVP may be present in network function-specific authorization answer messages that either do not include the Session-Binding AVP or include the Session-Binding AVP with any of the bits set to a zero value.
- Multi-Round-Time-Out AVP: The Multi-Round-Time-Out AVP should be present in network function-specific authorization answer messages whose Result-Code AVP is set to DIAMETER_MULTI_ROUND_AUTH. This AVP contains the maximum number of seconds that the access device must provide the user in responding to an authentication request.
- Class AVP: The Class AVP is used to by Diameter servers to return state information to the access device.
- Event-Timestamp AVP: The Event-Timestamp may be included in an Accounting-Request and Accounting-Answer messages to record the time that the reported event occurred, in seconds since January 1, 1900 00:00 UTC.

RFC 3280 [42] profiles the format and semantics of certificates and Certificate Revocation Lists (CRLs) for the Internet PKI.
The users of certificates will operate in a wide range of environments with respect to their communication topology, especially users of secure electronic mail. This profile supports users without high bandwidth, real-time IP connectivity, or high connection availability.

The following figure gives an overview over the basic building blocks of IETF's PKI infrastructure:

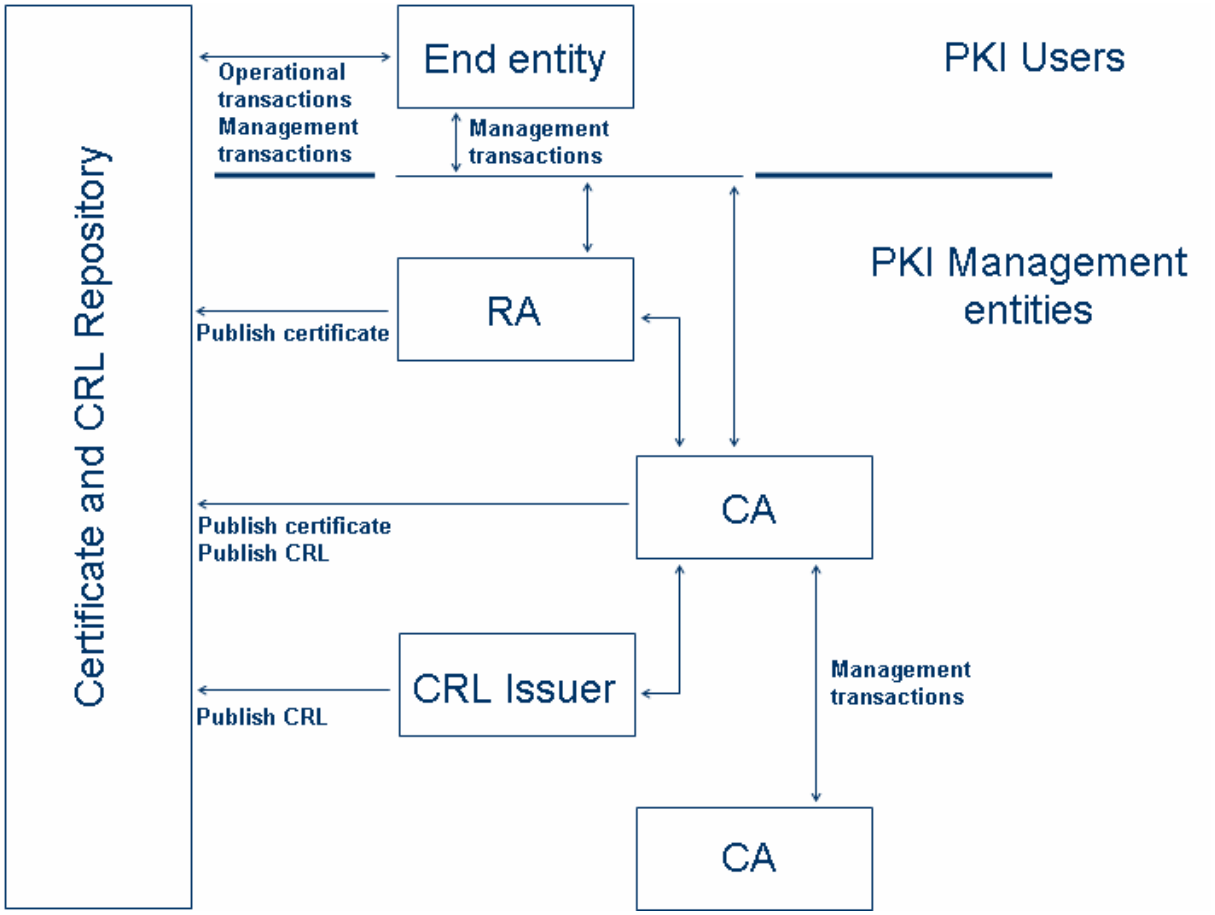


Figure 5.1.2.5.2.1: PKI Entities defined by IETF [42]

The following paragraphs present the profile for public key certificates according to RFC 3280 [42] that will foster interoperability and a reusable PKI. This clause is based upon the X.509 v3 certificate format and the standard certificate extensions defined in ITU-T Rec. X.509 [44].

- Certificate: The Certificate is a SEQUENCE of three required fields.
    - tbsCertificate: The field contains the names of the subject and issuer, a public key associated with the subject, a validity period, and other associated information.
    - SignatureAlgorithm: The signatureAlgorithm field contains the identifier for the cryptographic algorithm used by the CA to sign this certificate.
    - signatureValue: The signatureValue field contains a digital signature computed upon the ASN.1 DER encoded tbsCertificate. The ASN.1 DER encoded tbsCertificate is used as the input to the signature function.
- TBSCertificate: The sequence TBSCertificate contains information associated with the subject of the certificate and the CA who issued it. Every TBSCertificate contains the names of the subject and issuer, a

public key associated with the subject, a validity period, a version number, and a serial number; some MAY contain optional unique identifier fields.

- Version: This field describes the version of the encoded certificate.
- Serial number: The serial number must be a positive integer assigned by the CA to each certificate. It must be unique for each certificate issued by a given CA (i.e., the issuer name and serial number identify a unique certificate).
- Signature: This field contains the algorithm identifier for the algorithm used by the CA to sign the certificate.
- Issuer: The issuer field identifies the entity that has signed and issued the certificate. The issuer field MUST contain a non-empty distinguished name (DN). The issuer field is defined as the X.501 type Name.
- Validity: The certificate validity period is the time interval during which the CA warrants that it will maintain information about the status of the certificate. The field is represented as a SEQUENCE of two dates: the date on which the certificate validity period begins (notBefore) and the date on which the certificate validity period ends (notAfter).
- Subject: The subject field identifies the entity associated with the public key stored in the subject public key field.
- Subject Public Key Info: This field is used to carry the public key and identify the algorithm with which the key is used (e.g., RSA, DSA, or Diffie-Hellman).
- Unique Identifiers: These fields MUST only appear if the version is 2 or 3
- Extensions: This field must only appear if the version is 3.

In addition IETF uses the extensions defined for X.509 v3 certificates, which provide methods for associating additional attributes with users or public keys and for managing a certification hierarchy.

As discussed in [42], one goal of this X.509 v2 CRL profile is to provide the basis for the creation of an interoperable and reusable Internet PKI. The profile defines

- a set of information that can be expected in every CRL and
- common locations within the CRL for frequently used attributes as well as common representations for these attributes.

The following items describe the use of the X.509 v2 CRL in the Internet PKI.

- CertificateList: The CertificateList is a sequence of three required fields.
    - tbsCertList: This field is itself a sequence containing the name of the issuer, issue date, issue date of the next list, the optional list of revoked certificates, and optional CRL extensions.
    - signatureAlgorithm: The signatureAlgorithm field contains the algorithm identifier for the algorithm used by the CRL issuer to sign the CertificateList.
    - signatureValue: The signatureValue field contains a digital signature computed upon the ASN.1 DER encoded tbsCertList.
- Certificate List "To Be Signed": The or TBSCertList is a sequence of required and optional fields. The required fields identify the CRL issuer, the algorithm used to sign the CRL, the date and time the CRL was issued, and the date and time by which the CRL issuer will issue the next CRL.
    - Version: This optional field describes the version of the encoded CRL.
    - Signature: This field contains the algorithm identifier for the algorithm used to sign the CRL.
    - Issuer Name: The issuer name identifies the entity that has signed and issued the CRL.

- This Update: This field indicates the issue date of this CRL.
- Next Update: This field indicates the date by which the next CRL will be issued.
- Revoked Certificates: When there are revoked certificates, they are listed by their serial numbers.
- Extensions: This field may only appear if the version is 2. If present, this field is a sequence of one or more CRL extensions.

5.1.2.5.3 3GPP

For 3G networks the following security features related to entity authentication are provided [24]:

- user authentication: the property that the serving network corroborates the user identity of the user;
- network authentication: the property that the user corroborates that he is connected to a serving network that is authorised by the user's HE to provide him services; this includes the guarantee that this authorisation is recent.

To achieve these objectives, 3GPP assumes that entity authentication occurs at each connection set-up between the user and the network. To this end two mechanisms have been defined:

- an authentication mechanism using an authentication vector delivered by the user's HE to the serving network, and
- a local authentication mechanism using the integrity key established between the user and serving network during the previous execution of the authentication and key establishment procedure.

The SN may request the MS to send it the IMEI or IMEISV of the terminal. The IMEI should be securely stored in the terminal.

User to USIM authentication restricts access to the USIM until the USIM has authenticated the user. Thereby, it is ensured that access to the USIM can be restricted to an authorised user or to a number of authorised users.

To accomplish this feature, user and USIM must share a secret (e.g. a PIN) that is stored securely in the USIM. The user gets access to the USIM only if he/she proves knowledge of the secret.

USIM to terminal authentication ensures that access to a terminal or other user equipment can be restricted to an authorised USIM.

To this end, the USIM and the terminal must share a secret that is stored securely in the USIM and the terminal.

Assignment of temporary identities allows the identification of a user on the radio access link.

These are

- a temporary mobile subscriber identity (TMSI/P-TMSI), which has local significance only in the location area or routing area in which the user is registered and which
- outside that area should be a accompanied by an appropriate
  - Location Area Identification (LAI) or
  - Routing Area Identification (RAI) in order to avoid ambiguities.

The association between the permanent and temporary user identities is kept by the Visited Location Register (VLR/SGSN) in which the user is registered.

Authentication by key agreement achieves mutual authentication by the user and the network.

Both parties show knowledge of

- a secret key K which is shared between and available only to the USIM and the AuC in the user's HE.
- In addition the USIM and the HE keep track of counters
  - SQN_{MS} and

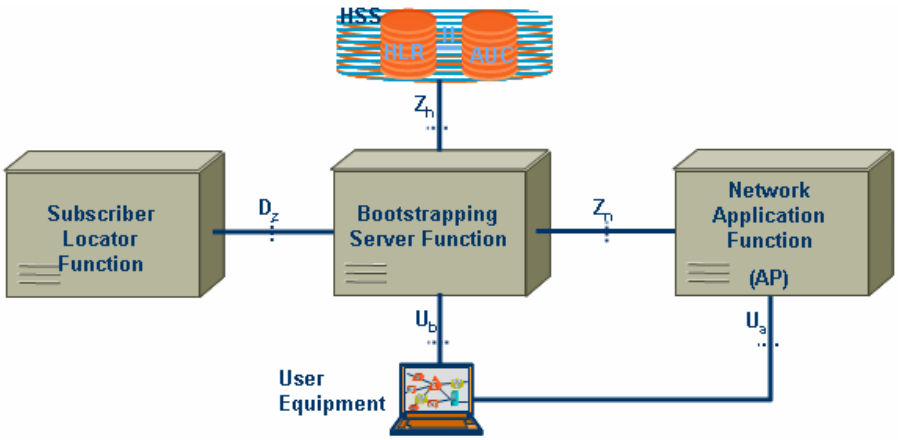
- SQN_{HE} respectively to support network authentication.

The sequence number SQN_{HE} is an individual counter for each user and the sequence number SQN_{MS} denotes the highest sequence number the USIM has accepted.

Upon receipt of a request from the VLR/SGSN, the HE/AuC sends an ordered array of *n* authentication vectors (the equivalent of a GSM "triplet") to the VLR/SGSN. The authentication vectors are ordered based on sequence number. Each authentication vector consists of the following components:

- a random number RAND,
- an expected response XRES,
- a cipher key CK,
- an integrity key IK and an authentication token AUTN.

Authorization of IMS-SIP based services has the following basic architecture if the NAF is based in the home network:



BSF Bootstrapping Server Functionality
HSS Home Subscriber System
NAF Operator-controlled network application function functionality
UE User Equipment

Figure 5.1.2.5.2.2: Simple network architecture for bootstrap in the home network defined by 3GPP [23]

Authorization of IMS-SIP based services has the following basic architecture if the NAF is based in the visited network:

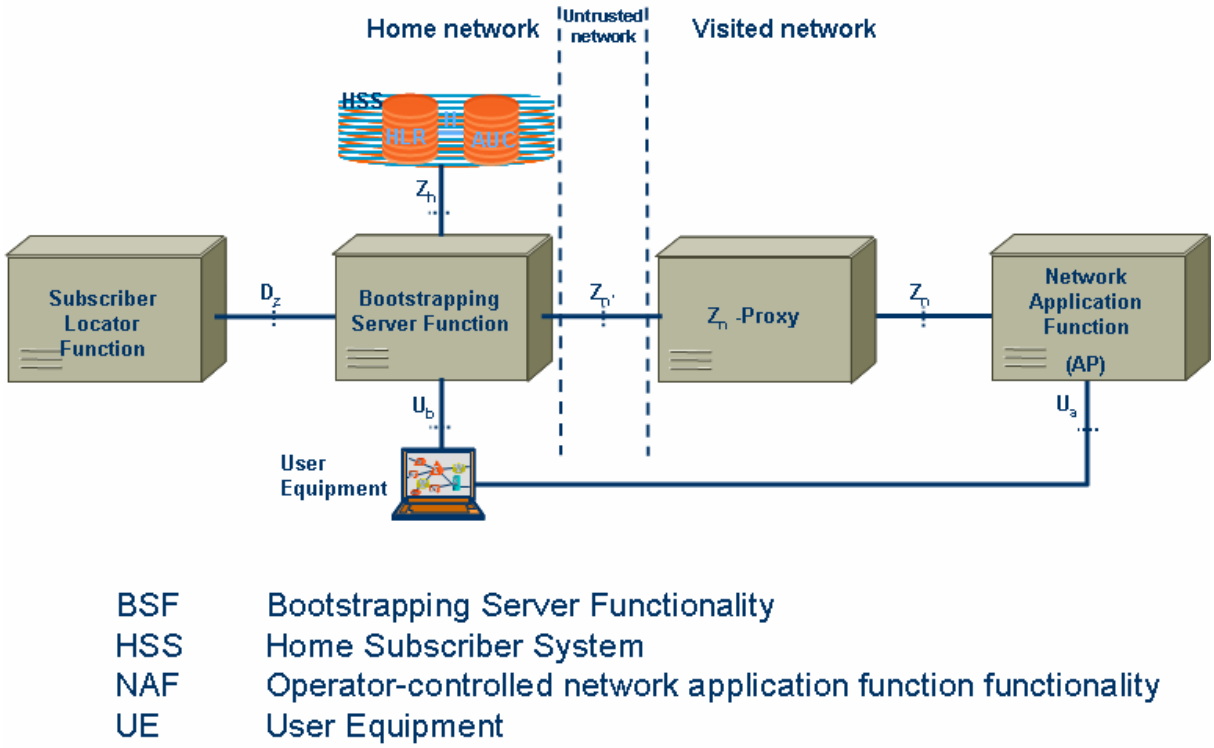


Figure 5.1.2.5.2.3: Simple network architecture for bootstrap in the visited network defined by 3GPP [23]

As described in [23], the generic Bootstrapping Server Function (BSF) and the UE shall mutually authenticate using the AKA protocol, and agree on session keys that are afterwards applied between UE and a Network Application Function (NAF).

After the bootstrapping has been completed, the UE and a NAF can run some application specific protocol where the authentication of messages will be based on those session keys generated during the mutual authentication between UE and BSF.

In the case where UE has contacted a NAF that is operated in another network than home network, this visited NAF shall use a Zn-Proxy of the NAFs network to communicate with subscriber's BSF (i.e. home BSF).

The set of all user security settings (USSs), i.e. GUSS, is stored in the HSS. In the case where the subscriber has multiple subscriptions, i.e. multiple ISIM or USIM applications on the UICC, the HSS shall contain one or more GUSSs that can be mapped to one or more private identities, i.e. IMPIs and IMSIs.

The required functionalities from the UE are:

- the support of HTTP Digest AKA protocol;
- the capability to use both a USIM and an ISIM in bootstrapping;
- the capability to select either a USIM or an ISIM to be used in bootstrapping, when both of them are present;
- the capability for a Ua application on the ME to indicate to the GBA Function on the ME the type or the name of UICC application to use in bootstrapping;
- the capability to derive new key material to be used with the protocol over Ua interface from CK and IK;
- support of NAF-specific application protocol.

The SLF:

- is queried by the BSF in conjunction with the Zh interface operation to get the name of the HSS containing the required subscriber specific data.
- is accessed via the Dz interface by the BSF.

5.1.2.5.4 OMA

In [38] OMA defines WAP certificate profiles. The profiles are mainly based on the Internet Certificate Profile defined in RFC 3280 [42], which in turn is based on the format defined in ITU-T Rec. X.509 [44]. Specification [38] provides, for each certificate type discussed, additional details regarding the contents of some individual fields in the certificate.

Network functions must recognize all the following required distinguished name attributes:

- countryName,
- organizationName,
- organizationalUnitName,
- stateOrProvinceName,
- commonName,
- domainComponent ,
- serialNumber.

CAs are not required to issue certificates with the

- serialNumber attribute,

but they should be able to do so.

All network functions must recognize all required distinguished name attributes listed above. Further, they must recognize the serialNumber attribute.

According to [38] certificate-processing network functions must recognize the following standard extensions:

- keyUsage,
- extKeyUsage,
- certificatePolicies,
- subjectAltName, and
- basicConstraints.

If the keyUsage

extension is included, it shall have the

- digitalSignature bit

set if the public key is an RSA key.

5.1.2.6 Accounting

5.1.2.6.1 ITU-T

ITU-T Recommendation Q.825 [45] specifies how CDRs (Call Detail Records) are produced and managed in Network Elements for POTS, ISDN and IN (Intelligent Networks).

The information model for Call Detail Recording is formally described in terms of an Entity-Relationship model, and an object model specified in terms of GDMO templates (Guidelines for the Definition of Managed Objects).

[40] defines the following attributes with very brief only, see [45] for the complete text.

1 accessDelivery: Indicates that the call was delivered to the called subscriber

2 accountCodeInput: Account code (for billing), supplied by subscriber.

78 additionalParticipantInfo: (No details given)

5 b-PartyCategory: Subscriber category for called subscriber.

4 bearerService: Bearer capability information (only for ISDN calls).

13 cDRPurpose: Reason for triggering this Call Data Record.

70 callDetailDataId: Unique identifier for the CallDetailData object.

79 callDuration: Duration of call

6 callIdentificationNumber: Identification number for call; all records produced for this call have the same callIdenfificationNumber.

73 callStatus: Identifies whether the call was answered or not.

9 calledPartyNumber: Telephone number of the called subscriber (may be a "diverted-to" or "translated" number).

7 callingPartyCategory: Calling subscriber category.

8 callingPartyNumber: Telephone number of the calling party.

10 callingPartyNumberNotScreened: An additional, user-provided (not screened) number to the calling party.

11 callingPartyType: Calling subscriber type.

74 carrierId: Carrier ID to which the call is sent.

12 cause: Cause and location value for the termination of the call.

14 chargedDirectoryNumber: Charged directory number (where the charged participant element can't indicate the number).

16 chargedParticipant: Participant to be charged for the usage.

15 chargingInformation: Charging information generated by a Network Element which is capable of charging.

17 configurationMask: Time consumption, e.g. from B-answer to termination time, between partial call records, etc.

18 conversationTime: Time consumption from B-answer to end of call.

19 creationTriggerList: List of trigger values which will create Call Detail data objects.

75 dPC: Destination point code (for analysis purposes).

20 dataValidity: Indicates that the NE is having problems, contents of the generated Call Detail record is not reliable.

23 durationTimeACM: Time consumption from seizure until received ACM.

21 durationTimeB-Answer: Time consumption from seizure until B-answer.

22 durationTimeNoB-Answer: Time from seizure to termination when no B-answer was received.

25 exchangeInfo: Identity of exchange where Call Detail record was generated.

26 fallbackBearerService: Fallback bearer capability information for a call.

27 glare: Indicates if a glare condition was encountered.

31 iNServiceInformationList: Contains information about the use of IN (Intelligent Network) services.

32 iNSpecificInformation: Contains information about the use of one IN service.

33 iSUPPreferred: Indicate whether an ISUP preference was requested.

28 immediateNotificationForUsageMetering: Indicates that the Call Detail records requires immediate data transfer to the Operations System.

34 maxBlockSize: Maximum number of Call Detail records in a block.

35 maxTimeInterval: Maximum latency allowable for near-real-time Call Detail data delivery.

36 networkManagementControls: Indicates which Traffic Management Control has affected the call.

37 networkProviderId: Indicates the Network Provider for whom the CDR is generated.

76 oPC: Originating point code for a failed call (for analysis purposes).

38 operatorSpecific1AdditionalNumber

40 operatorSpecific2AdditionalNumber

42 operatorSpecific3AdditionalNumber: Operator-defined additional participant information.

39 operatorSpecific1Number

41 operatorSpecific2Number

43 operatorSpecific3Number: Operator-defined participant information.

44 originalCalledNumber: Telephone number of the original called party.

45 partialGeneration: Included if the CDR (Call Detail record) output is partial. Such CDRs have a field indicating their partial record number.

77 participantInfo: (No details given).

46 percentageToBeBilled: Percentage to be billed when normal billing rules are not to be followed.

47 periodicTrigger: Defines the intervals at which the CDR file should be created.

48 personalUserId: Internationally unique personal User Identity (for UPT calls).

49 physicalLineCode: Identifies the call subscriber's physical line.

50 progress: Describes an event which occurred during the life of a call.

51 queueInfo: Used to record usage of queuing resources with IN calls.

52 receivedDigits: The digits dialled by the subscriber. (Normally only included for customer care purposes).

53 recordExtensions: Information elements added by network operators and/or manufacturers in addition to the standard ones above.

5.1.2.6.2 IETF

Figure xx shows the main architectural components of IETF's way to collect data [40]. Based on this architecture IETF defines the following accounting attributes:

RADIUS:

Each RADIUS attribute is identified by an 8-bit number, the RADIUS Type field. Up-to-date values of this field can be found in the most recent Assigned Numbers RFC [ASG-NBR], at the time of writing the Technical Report is as follows:

1 User-Name

2 User-Password

3 CHAP-Password

4 NAS-IP-Address

5 NAS-Port

6 Service-Type

7 Framed-Protocol

8 Framed-IP-Address

9 Framed-IP-Netmask

10 Framed-Routing

11 Filter-Id

12 Framed-MTU

13 Framed-Compression

14 Login-IP-Host

15 Login-Service

16 Login-TCP-Port

17 (unassigned)

18 Reply-Message

19 Callback-Number

20 Callback-Id

21 (unassigned)

22 Framed-Route

23 Framed-IPX-Network

24 State

25 Class

26 Vendor-Specific

27 Session-Timeout

28 Idle-Timeout

29 Termination-Action

30 Called-Station-Id

31 Calling-Station-Id

32 NAS-Identifier

33 Proxy-State

34 Login-LAT-Service

35 Login-LAT-Node

36 Login-LAT-Group

37 Framed-AppleTalk-Link

38 Framed-AppleTalk-Network

39 Framed-AppleTalk-Zone

40 Acct-Status-Type

41 Acct-Delay-Time

42 Acct-Input-Octets

43 Acct-Output-Octets

44 Acct-Session-Id

45 Acct-Authentic

46 Acct-Session-Time

47 Acct-Input-Packets

48 Acct-Output-Packets

49 Acct-Terminate-Cause

50 Acct-Multi-Session-Id

51 Acct-Link-Count

52 Acct-Input-Gigawords

53 Acct-Output-Gigawords

54 Unused

55 Event-Timestamp

60 CHAP-Challenge

61 NAS-Port-Type

62 Port-Limit

63 Login-LAT-Port

64 Tunnel-Type

65 Tunnel-Medium-Type

66 Tunnel-Client-Endpoint

67 Tunnel-Server-Endpoint

68 Acct-Tunnel-Connection

69 Tunnel-Password

70 ARAP-Password

71 ARAP-Features

72 ARAP-Zone-Access

73 ARAP-Security

74 ARAP-Security-Data

75 Password-Retry

76 Prompt

77 Connect-Info

78 Configuration-Token

79 EAP-Message

80 Message-Authenticator

81 Tunnel-Private-Group-ID

82 Tunnel-Assignment-ID

83 Tunnel-Preference

84 ARAP-Challenge-Response

85 Acct-Interim-Interval

87 NAS-Port-Id

88 Framed-Pool

90 Tunnel-Client-Auth-ID

91 Tunnel-Server-Auth-ID

DIAMETER [40]:

The DIAMETER framework [41] defines a policy protocol used by clients to perform Policy, AAA and Resource Control allowing a single server to handle policies for many services. DIAMETER defines a base protocol that specifies the header formats, security extensions and requirements as well as a small number of mandatory commands and AVPs.

In the list below attribute numbers which are used for RADIUS attributes but not for DIAMETER are indicated with a star (*). RADIUS attributes used by DIAMETER are not listed again here.

4 (unassigned, *)

17 (unassigned)

21 (unassigned)

24 (unassigned, *)

25 (unassigned, *)

27 (unassigned, *)

32 (unassigned, *)

33 (unassigned, *)

280 Filter-Rule

281 Framed-Password-Policy

480 Accounting-Record-Type

481 ADIF-Record

482 Accounting-Interim-Interval

483 Accounting-Delivery-Max-Batch

484 Accounting-Delivery-Max-Delay

485 Accounting-Record-Number

600 SIP-Sequence

601 SIP-Call-ID

602 SIP-To

603 SIP-From

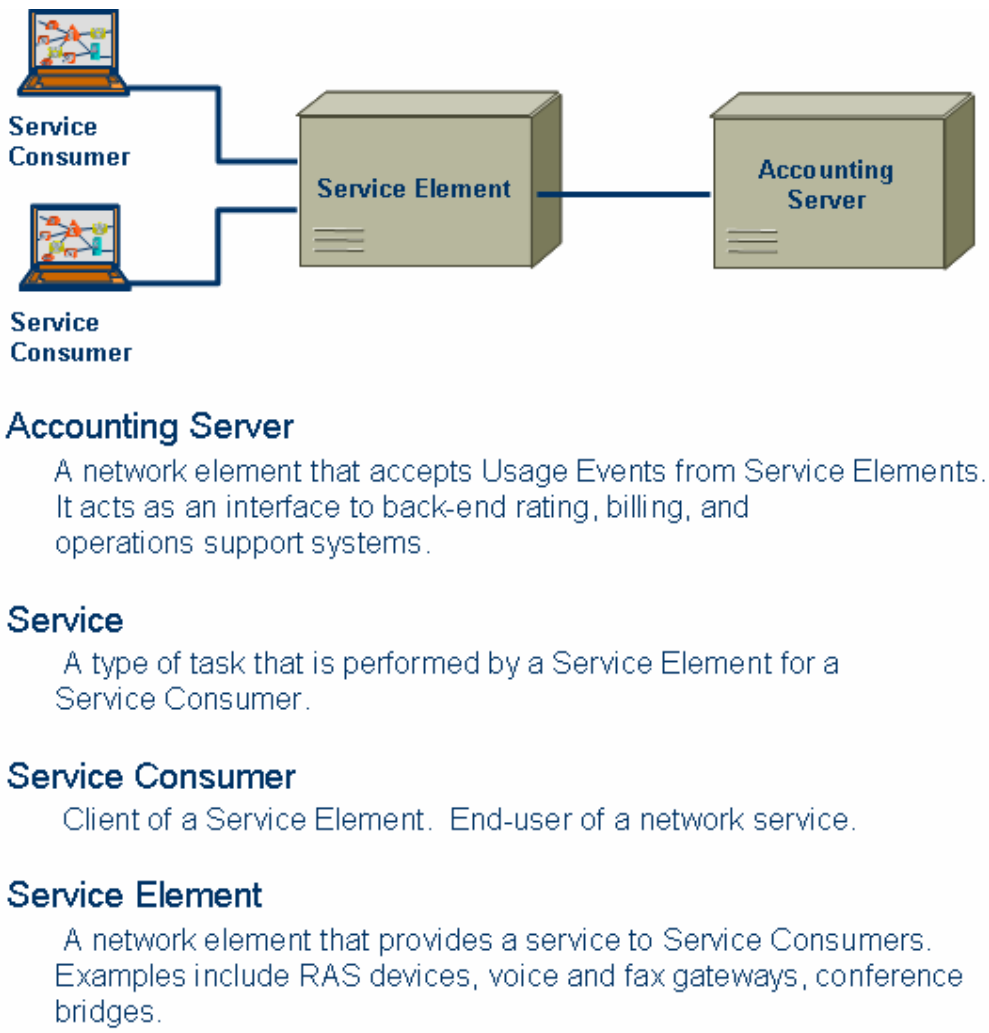


Figure 5.1.2.6.2.1: Accounting architecture defined by IETF [40]

5.1.2.6.3 3GPP

5.1.2.6.4 OMA

5.1.3 Network Hosted Business Services and Network functions

5.1.3.1 MM Messaging Services (MMS)

5.1.3.2 Content Services

5.1.3.3 IMS Application Servers

5.1.3.4 Store and Forward Messaging

5.1.3.5 P2P and Group Communication

5.1.3.6 IN Services

5.1.3.7 Online Charging, PrePaid Services

5.1.4 Management Applications

5.1.4.1 Subscriber Management

5.1.4.2 Statistics

5.1.4.3 Billing

5.1.4.4 System Management

5.2 Standardization documents containing subscriber/user information

This clause lists the standards, which deal with the end-user/subscriber information for the network functions introduced in clause 5.1.

Table 5.2.1: List of standardization documents containing subscriber/user information

SDO	Number	Title
3GPP	TS 23.008 V7.2.0 (2006-06)	ORGANIZATION OF SUBSCRIBER DATA
	TS 23.003 V7.0.0 (2006-06)	NUMBERING, ADDRESSING AND IDENTIFICATION
	TS 23.097 V6.0.0 (2004-12)	MULTIPLE SUBSCRIBER PROFILE (MSP) (PHASE 2) - STAGE 2
	TS 23.016 V6.1.0 (2004-03)	SUBSCRIBER DATA MANAGEMENT; STAGE 2;
	TS 32.140 V6.3.0 (2004-12)	SUBSCRIPTION MANAGEMENT (SUM) REQUIREMENTS
	TS 32.141 V6.1.0 (2004-03)	SUBSCRIPTION MANAGEMENT (SUM) ARCHITECTURE
	TS 32.171 V6.1.0 (2004-12)	SUBSCRIPTION MANAGEMENT (SUM) NETWORK RESOURCE MODEL (NRM) INTEGRATION REFERENCE POINT (IRP): REQUIREMENTS
	TS 32.172 V6.3.0 (2006-03)	SUBSCRIPTION MANAGEMENT (SUM) NETWORK RESOURCE MODEL (NRM) INTEGRATION REFERENCE POINT (IRP): INFORMATION SERVICE (IS)
	TS 32.175 V6.2.0 (2005-06)	SUBSCRIPTION MANAGEMENT (SUM) NETWORK RESOURCE MODEL (NRM) INTEGRATION REFERENCE POINT (IRP): EXTENSIBLE MARKUP LANGUAGE (XML) DEFINITION
	TS 23.240 V6.7.0 (2005-03)	3GPP GENERIC USER PROFILE (GUP); ARCHITECTURE (STAGE 2);
	TR 23.941 V6.0.0 (2004-12)	3GPP GENERIC USER PROFILE (GUP); STAGE 2; DATA DESCRIPTION METHOD (DDM)
	TS 33.102 V7.0.0 (2005-12)	3G SECURITY; SECURITY ARCHITECTURE
	TS 43.020 V6.4.0 (2006-06)	SECURITY RELATED NETWORK FUNCTIONS
	TS 23.060 V7 .1.0 (2006-06)	GENERAL PACKET RADIO SERVICE (GPRS); SERVICE DESCRIPTION; STAGE 2
	TS 23.271 V6.13.0 (2005-09)	FUNCTIONAL STAGE 2 DESCRIPTION OF LOCATION SERVICES (LCS);
	TS 25.305	STAGE 2 FUNCTIONAL SPECIFICATION OF UE POSITIONING IN UTRAN
	TS 43.059	FUNCTIONAL STAGE 2 DESCRIPTION OF LOCATION SERVICES IN GERAN
	TS 23.141 V6.9.0 (2005-12)	PRESENCE SERVICE; ARCHITECTURE AND FUNCTIONAL DESCRIPTION
	TS 23.228 V7.4.0 (2006-06)	IP MULTIMEDIA SUBSYSTEM (IMS); STAGE 2
	TS 33.222 V7.1.0 (2006-03)	GENERIC AUTHENTICATION ARCHITECTURE (GAA); ACCESS TO NETWORK APPLICATION FUNCTIONS USING HYPERTEXT TRANSFER PROTOCOL OVER TRANSPORT LAYER SECURITY (HTTPS)
	TS 33.102 V7.0.0 (2005-12)	3G SECURITY; SECURITY ARCHITECTURE
3GPP2	S.R0037-0 V3.0 VERSION DATE: AUGUST 21, 2003 VERSION 3.0	IP NETWORK ARCHITECTURE MODEL FOR CDMA2000 SPREAD SPECTRUM SYSTEMS
	X.S0027-001-0 VERSION 1.0 DATE: SEPTEMBER, 2004	PRESENCE SERVICE: ARCHITECTURE AND FUNCTIONAL DESCRIPTION
IETF	RFC 2778 FEBRUARY 2000	A MODEL FOR PRESENCE AND INSTANT MESSAGING
	RFC 2924 SEPTEMBER 2000	ACCOUNTING ATTRIBUTES AND RECORD FORMATS
	RFC 3588 SEPTEMBER 2003	DIAMETER BASED PROTOCOL
	RFC 3280 APRIL 2002	INTERNET X.509 PUBLIC KEY INFRASTRUCTURE
	RFC 2560 JUNE 1999	PKIX OCSP
ITU-T	ITU-T X.509 08/2005	INFORMATION TECHNOLOGY – OPEN SYSTEMS INTERCONNECTION – THE DIRECTORY: PUBLIC-KEY AND ATTRIBUTE CERTIFICATE FRAMEWORKS
	ITU-T Q.822 04/1994	SPECIFICATIONS OF SIGNALLING SYSTEM NO.7 – STAGE 1, STAGE 2 AND STAGE 3 DESCRIPTION FOR THE Q3 INTERFACE – PERFORMANCE MANAGEMENT
TISPAN	Draft ETSI DTS 188 002 V0.0.1 (2006-01)	Subscription Management Requirements
	Draft ETSI TS 188 XXX V0.0.1 (2006-02)	NGN Management; Management Information Model Requirements
	WG8 interim meeting WG8TD08 Sophia Antipolis, 3 - 5 May 2006	Remaining Comments

SDO	Number	Title
OMA	OMA-AD_IMS-V1_0-20040420-D DRAFT VERSION 1.0 – 20 APR 2004	UTILIZATION OF IMS CAPABILITIES ARCHITECTURE
	OMA-TS-PRESENCE_SIMPLE-V1_0-20060418-C CANDIDATE VERSION 1.0 – 18 APR 2006	PRESENCE SIMPLE SPECIFICATION
	OMA-AD-PRESENCE_SIMPLE-V1_0-20060110-C CANDIDATE VERSION 1.0 – 10 JAN 2006	PRESENCE SIMPLE ARCHITECTURE DOCUMENT
	OMA-TS-MLP-V3_2-20051124-C CANDIDATE VERSION 3.2 – 24 NOV 2005	MOBILE LOCATION PROTOCOL 3.2
	OMA-AD-MLS-V1_0-20050607-C CANDIDATE VERSION 1.0 – 07 JUNE 2005	OMA MOBILE LOCATION SERVICE ARCHITECTURE
	OMA-TS-XDM_CORE-V1_0-20060612-A APPROVED VERSION 1.0 – 12 JUN 2006	XML DOCUMENT MANAGEMENT (XDM) SPECIFICATION
	OMA-TS-DM_STDOBJ-V1_2-20060602-C CANDIDATE VERSION 1.2 – 02 JUN 2006	OMA DEVICE MANAGEMENT STANDARDIZED OBJECTS
	OMA-SYNCML-DMSTDOBJ-V1_1_2-20031203-A APPROVED VERSION 03-DECEMBER-2003	SYNCML DEVICE MANAGEMENT STANDARDIZED OBJECTS, VERSION 1.1.2
	OMA-SECURITY-CERTPROF-V1_1-20040615-C CANDIDATE VERSION 1.1 – 15 JUN 2004	CERTIFICATE AND CRL PROFILES

5.3 Problems in the handling of the current Subscriber/User Data Model

5.3.1 Inconsistencies in the data between different network functions

5.3.2 Redundancy problems

5.3.3 Services requiring different handling of User Data

5.3.4 Requirements for the Common User Data Model

5.4 Basics of a Common User Data Model

5.4.1 Characteristics of an End-User

This clause gives a characterization of the basic building blocks of a data model describing an end-user.

5.4.1.1 Types of Data Assigned to an End-User

An end-user can have three different types of data:

- **Profile data:** This is data characterizing the services assigned to the end-user.
- **Completely private data:** This is data stored by the end-user and only seen by him.
- **Private data influencing the call processing:** This is data, which is also privately owned by the end-user, but will influence the way services are executed (e.g. private e-mail distribution list).

5.4.1.2 The Identity of an End-User

5.4.1.2.1 Representation of the Identity of the End-User through his Keys

Depending on the service used by the end-user, identification of the same might be achieved by different keys (examples of which can be found in [2], [3]): IMSI, MSISDN, PK, etc. Each of these keys describes part of the identity of the end-user. In order to identify an end-user completely and uniquely some tag, (e.g. UID) has to be defined.

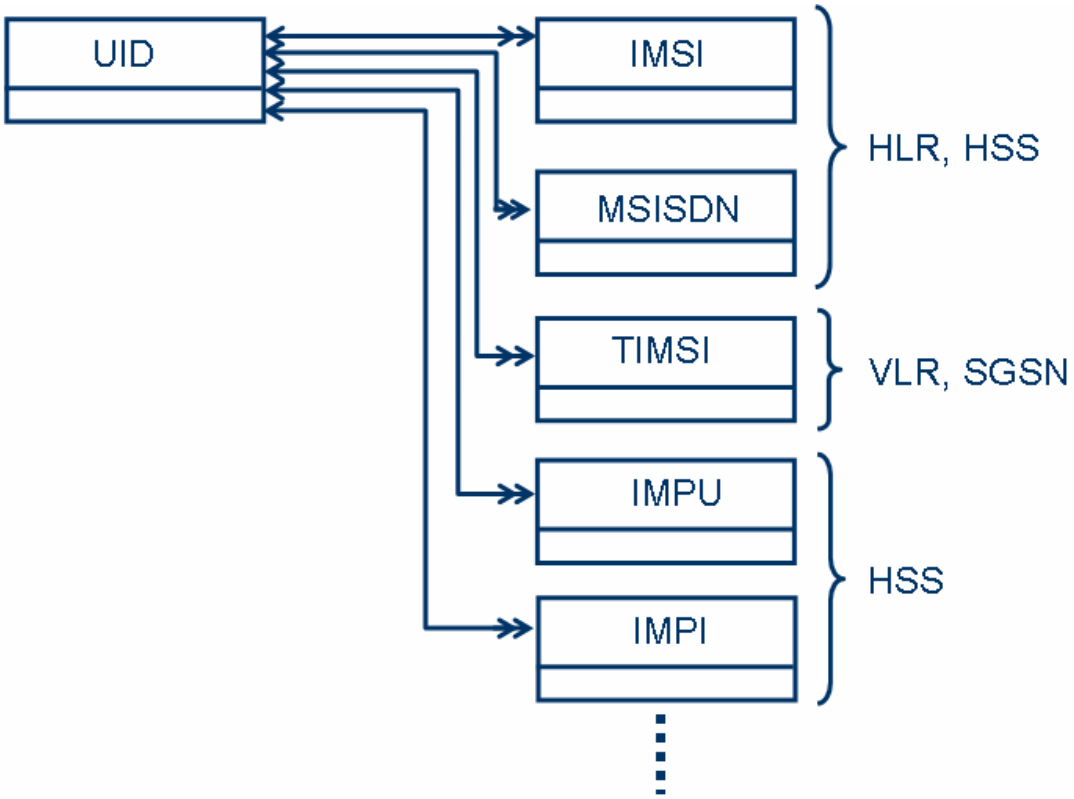


Figure 5.4.1.2.1.1: Characterization of one end-user

5.4.1.2.2 Identity Management of the End-User

5.4.1.3 The Relation between an End-User and a Subscriber

In clause 4.3.1.2 the relation of the end-user to his various key attributes has been discussed. This clause highlights the relationship between a contract holder and an end-user.

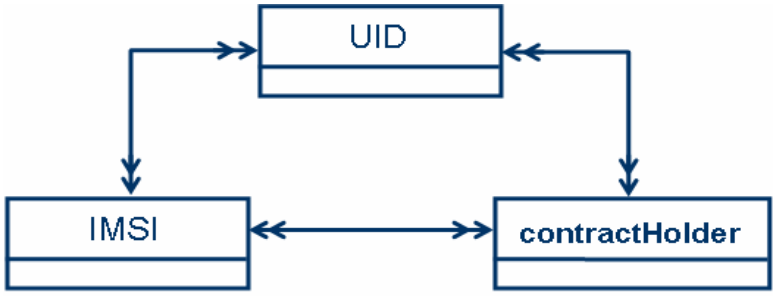


Figure 5.4.1.3.1: General Picture of the relationship between an end-user and a contract holder

The above figure shows the general situation: An end-user can have n cards with the same or different services assigned. The end-user may also be a contract holder (subscriber), but does not need to be so. A contract holder can be assigned to one or more cards (for which he is billed) and to one or more end-users.

The following figure gives a simple example:

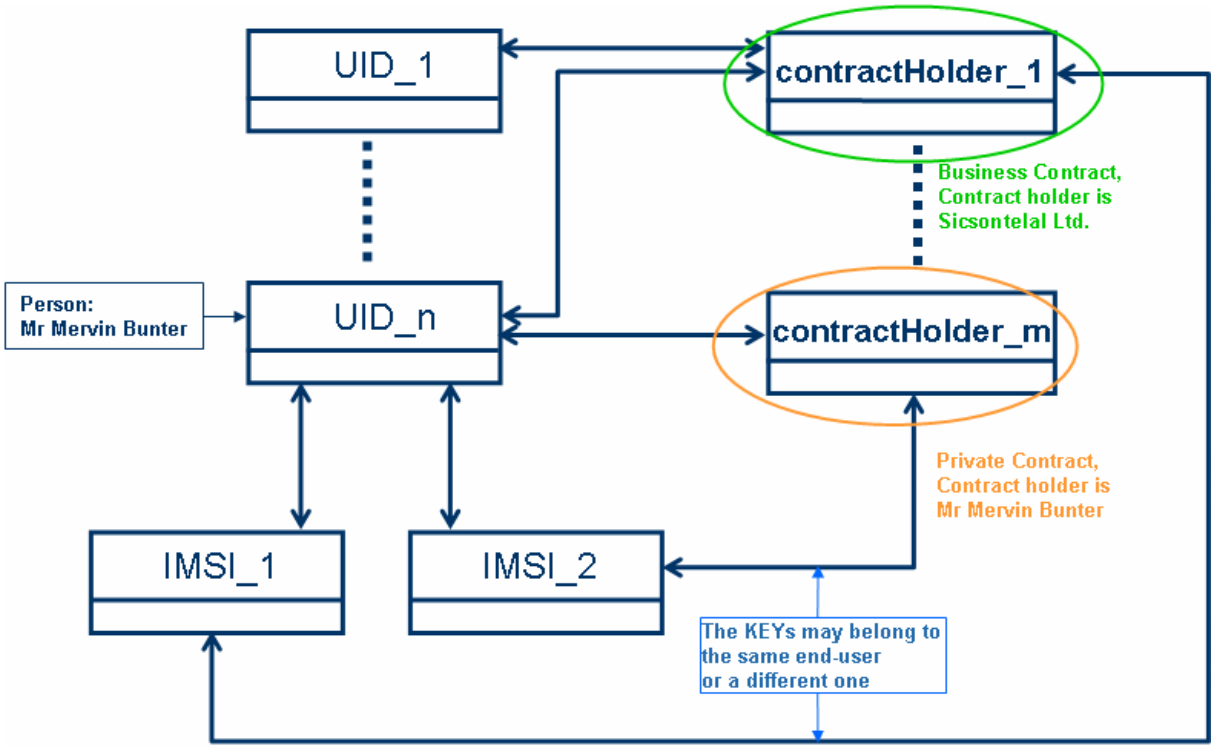


Figure 5.4.1.3.2: Specific example of the relationship between an end-user and a contract holder

Contract holder 1 is a company, which has n employees, with one mobile phone each (paid for by the company). Employee n (UID_n) is also a contract holder (contract holder m), because in addition to the company phone he has a pre-paid mobile for his private use. Thus UID_n is assigned two different keys (IMSI_1, IMSI_2).

5.4.2 Different Levels of Data Consolidation

This clause discusses two completely different approaches to the design of a complete data model of an end-user:

- **Approach 1:** The complete model is exposed to all network functions and only in the cases, where there are semantically identical data model entities handled by different network functions, Adaptation Layer Functionality is used to guarantee the integrity of these entities for all involved network functions.
- **Approach 2:** The complete model is hidden from the network functions (perhaps, because it reflects the business processes of a provider) and Adaptation Layer Functionality has to be provided for a high percentage of the entities of the data model in order to satisfy the network functions.

5.4.2.1 Adaptation Layer for partial Data Consolidation

The complete structure of the end-user data model is, in principle, exposed to any network function. A specific network function is

- either able to ignore data, which are not relevant for its function,
- or the data base requests are tailored so that only data relevant for this network function are manipulated.

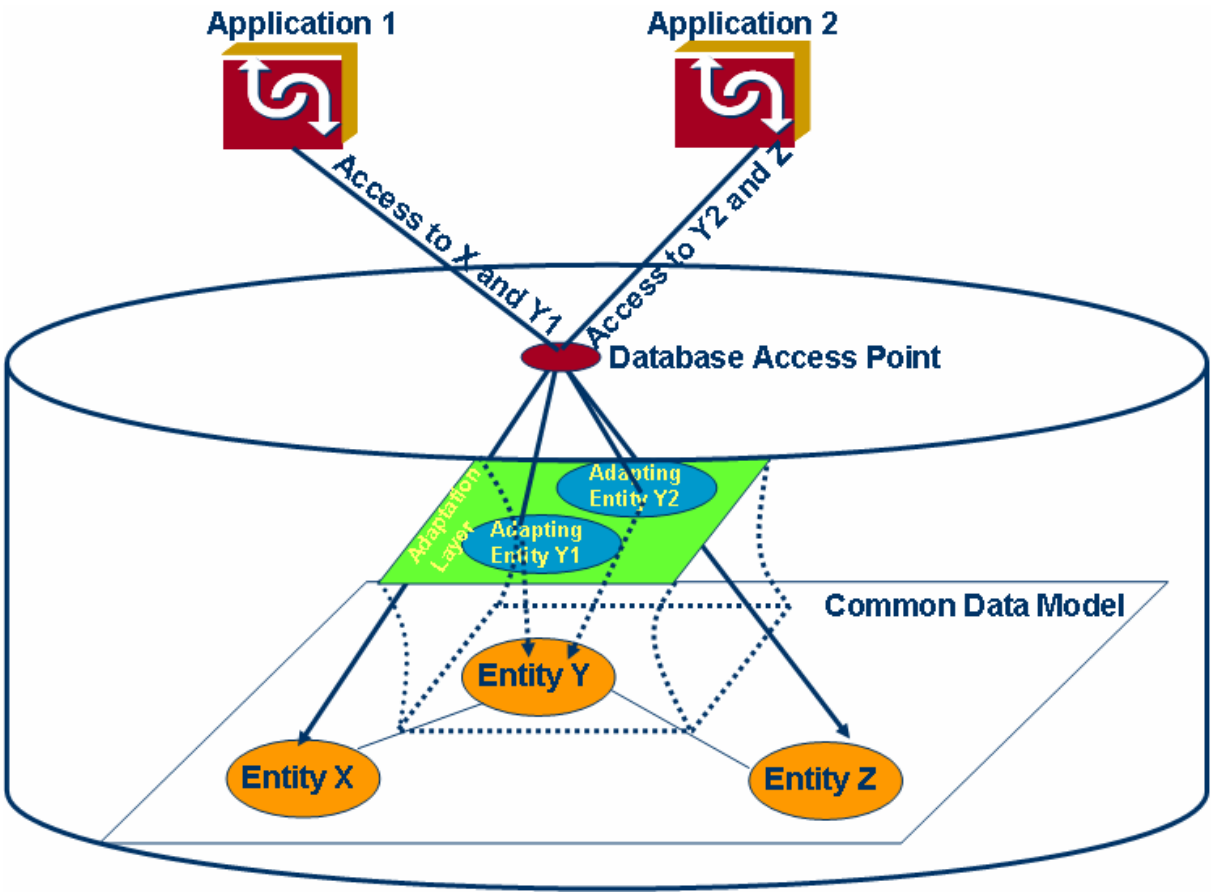


Figure 5.4.2.1.1: Overview over a common object model in case of a partial data consolidation

In case two network functions share a common data entity (Entity Y in Figure x) then it may be necessary to define Adapting Entities Y1 and Y2 under the conditions described in clause 4.3.3.

5.4.2.2 Adaptation Layer for full data consolidation

The structure of the common data model is completely hidden from the network functions for reasons discussed in more detail in clause 5.

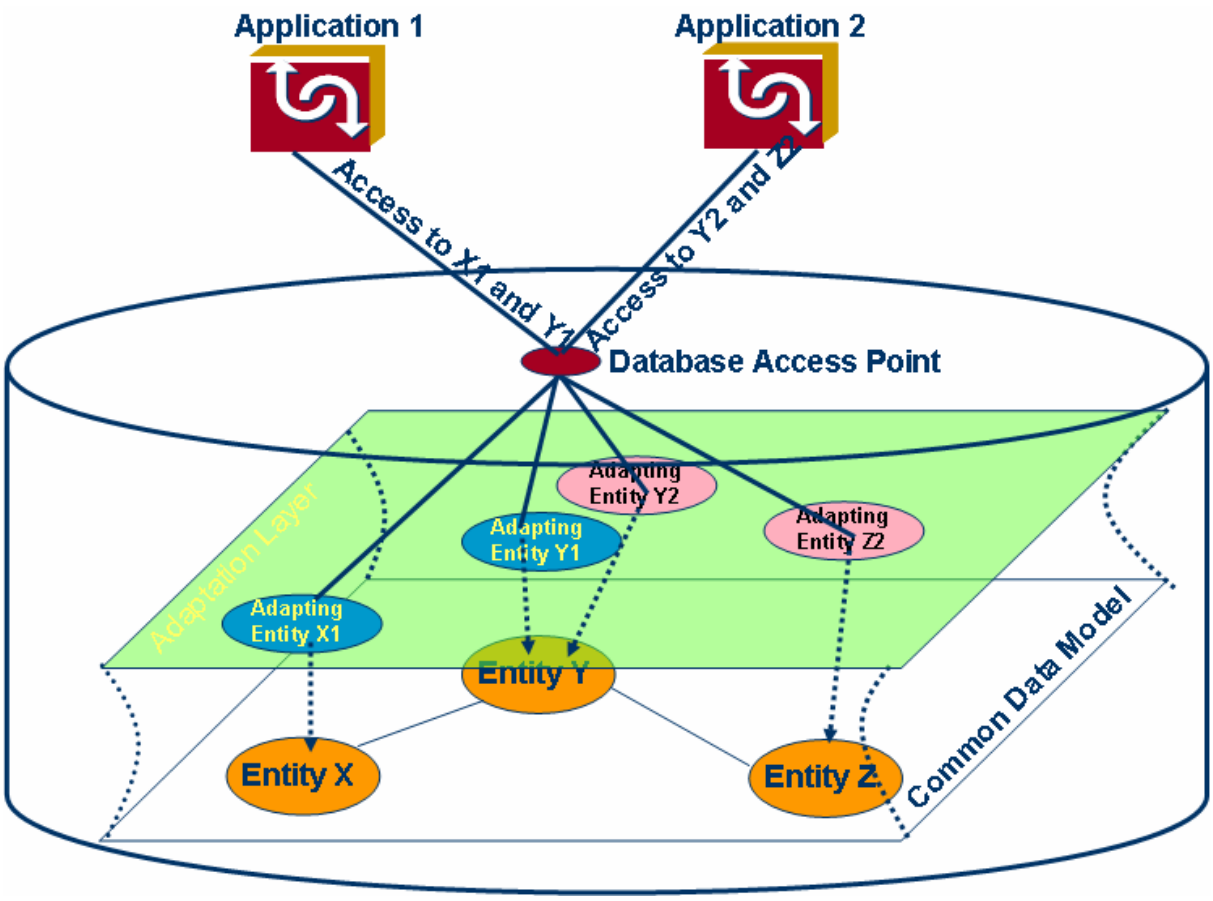


Figure 5.4.2.2.1: Overview over a common object model with a fat Adaptation Layer

5.4.2.3 Mixed Scenarios

Depending on the criteria for the definition of the word "common" in common data model, the common part might, semantically seen, contain only a subset of data needed by one of the network functions.

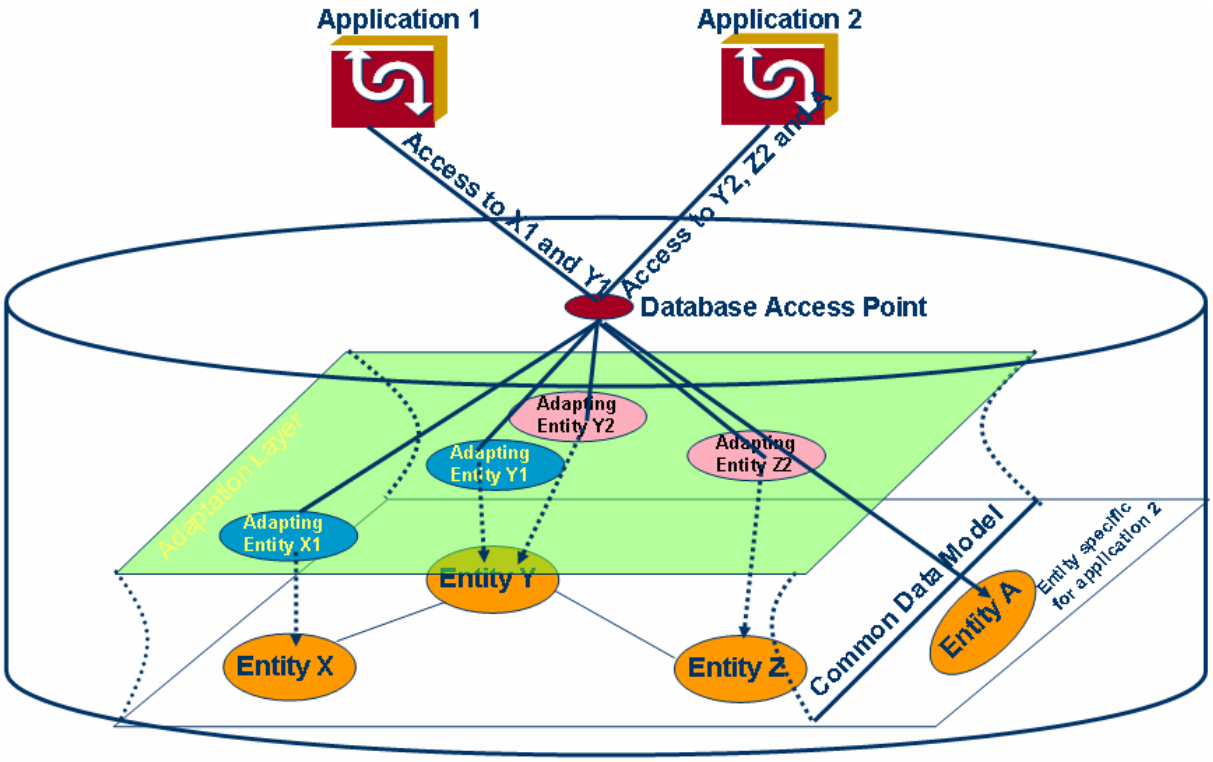


Figure 5.4.2.3.1: Overview over a mixed architecture

5.4.3 Semantic Identity of Data Entities

One idea of a common data model is to avoid redundant definition and storage of data model entities. So all data, which are used by more than one network function in the same way ("semantic identity") shall only be defined once.

5.4.4 Content of Post Update Triggers

In any of the cases described in 4.3.2, in which two or more network functions use the same data entity as described in 4.3.3, it must be possible for all of these network functions to be informed about any change happening to the data entity in question.

As soon as the creation, deletion or modification of a data model item common to two or more network functions has been carried out by the end-user database, a triggering mechanism should be able to inform any interested network function about it.

Content: Name of the data entity, old value, new value

6 Basic Structure of the Common Profile Storage Framework (CPSF)

6.1 Logical View

6.2 Analysis of alternative solutions

6.3 Tooling

7 Gap-Analysis

8 Conclusions

Annex A:
Example for the realization of an end-user database according to the Common Profile Storage Framework

A.1 The Model Structure

A.2 The Network Architecture

Annex B:
Change history

Change history								
Date	**TSG #**	**TSG Doc.**	**CR**	**Rev**	**Subject/Comment**	**Cat**	**Old**	**New**
Jul 2006	SA5_48	S5-060712	--	--	First delivery of draft version	--	0.0.0	0.0.1
Jul 2006	--	--	--	--	MCC clean-up	--	0.0.1	0.0.2
Oct 2006	SA5_50	S5-061435	--	--	MCC clean-up. Input to SA5#50	--	0.0.2	0.0.3
Dec 2005	SA_30	SP-050730	--	--	UID 320006; Submitted to TSG SA#30 for Information		1.0.1	
Jun 2006	SA_32	SP-060262	--	--	Submitted to TSG SA#32 for Approval	--	2.0.0	