3GPP TR 23.874 V1.3.0 (2000-11)

Technical Report

3rd Generation Partnership Project; Technical Specification Group Services and System Aspects; Feasibility study of architecture for push service (Release 4)

Keywords
<keyword[, keyword]>

3GPP

Postal address

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

Internet
http://www.3gpp.org

Contents

Foreword 5
1 Scope 6
2 References 6
3 Definitions, symbols and abbreviations 7
3.1 Definitions 7
3.2 Symbols 7
3.3 Abbreviations 7
4 Introduction 7
5 Requirements 8
6 General Description 8
6.1 Service Environment and Scenario 8
6.1.1 Dedicated Connection Approach 8
6.1.2 Connectionless Approach 10
6.2 Addressing 10
6.3 Dedicated Connection Establishment 11
6.4 Push Content Delivery 11
6.4.1 Reliable Delivery 12
6.4.1.1 Store and Forward 12
6.4.1.2 Presence Service 12
6.5 Multiple Services 14
6.6 Security and Charging 15
6.7 User Terminal 15
6.8 Roaming Support 15
7 Architecture for GPRS 16
7.1 Introduction 16
7.2 Network requested PDP Context activation with User-ID 17
7.2.1 Functional Architecture 17
7.2.1.1 Application Server (AS) 17
7.2.1.2 Notification Agent (NA) 17
7.2.1.3 MS Address Resolver (AR) 17
7.2.1.4 Mobile Station （MS） 17
7.2.1.5 GPRS Network 18
7.2.3 Protocol Architecture 18
7.2.4 Message Flow 19
7.2.5 Impacts on 3G specifications 20
7.3 PDP context activation triggered by DNS query 20
7.3.1 Definitions 20
7.3.2 Assumptions 21
7.3.3 Requirements 21
7.3.4 General Description 21
7.3.5 Proposed behaviours for DNS queries 22
7.3.5.1 Lifetime of the PDP context 23
7.3.5.2 Choice of T_{ctxt} 23
7.3.6 Proposed behaviours for IP data delivery 23
7.3.7 Example Scenario 23
7.3.8 Alternative PDNS Implementation 24
7.3.9 Alternative GGSN Implementation 25
7.3.10 GGSN with embedded PDNS 27
7.3.11 Avoiding an Application Server Timeout 27
7.3.12 Protocol Architecture 28
7.3.13 Security 28
7.3.14 Roaming Support 28
7.3.15 Error Responses 28

7.4 SMS Push Service 29
7.4.1 Assumptions 29
7.4.2 Basic Service Scenarios 29
7.4.2.1 Short Message Push 29
7.4.2.2 Push Notification with User Connect Scenario 29
7.4.2.3 Push Broadcast Scenario 30
7.4.3 Addressing 31
7.4.4 Subscription, Security, and Charging 31
7.4.5 Roaming 31
7.4.6 Delivery Reliability 31
7.4.7 Protocol Architecture 32
7.5 Push "The internet way" 32
7.6 SIP Push Service 34
7.6.1 IM Subsystem Scenario 34
7.6.2 No IM Subsystem Scenario 35
7.6.3 Roaming 36
7.6.3.1 IM Roaming 36
7.6.3.2 Roaming with SIP Proxy in Home Network 36
7.6.3.3 Roaming with SIP Proxy in Visited Network 36
7.6.4 Protocol Architecture 37
7.6.5 Addressing 38
7.6.5.1 SIP Identity 38
7.6.5.2 IP Address 38
7.6.6 Subscription, Security, and Charging 38
7.6.7 Delivery Reliability 38
7.6.8 Connectionless Push 39
7.6.9 Quality of Service 39
8 Conclusion and Recommendations 39
Annex A (Informative): Comparison of the Push Techniques comparison 40
Annex <X>: Change history 41

Foreword

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

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

Version x.y.z

where:

- x the first digit:
  - 1 presented to TSG for information;
  - 2 presented to TSG for approval;
  - 3 or greater indicates TSG approved document under change control.
- y the second digit is incremented for all changes of substance, i.e. technical enhancements, corrections, updates, etc.
- z the third digit is incremented when editorial only changes have been incorporated in the document.

1 Scope

The purpose of this technical report is to study the feasibility of architecture for push services over Packet Switched Networks.

In the present document, the architecture for the delivery network is examined and the architectures for the user terminal and the application server are out of scope.

2 References

[Editor's note: Chapter to be completed]

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.

[0] 3GPP TR 21.905: " Vocabulary for 3GPP Specifications ".

[1] 3GPP TS 22.060: " General Packet Radio Service (GPRS); Service description; Stage 1 (Release 2000) ".

[2] 3GPP TS 23.039: " Interface protocols for the connection of Short Message Service Centres (SMSCs) to Short Message Entities (SMEs) ".

[3] 3GPP TS 23.040: " Technical realization of the Short Message Service (SMS) ".

[4] 3GPP TS 23.060: " General Packet Radio Service (GPRS); Service description; Stage 2 ".

[5] 3GPP TS 23.228: " IP Multimedia (IM) Subsystem - Stage 2 ".

[6] ITU-T Recommendation E.164: "Numbering plan for ISDN era".

[7] IETF RFC 791: " Internet Protocol "(STD 5).

[8] IETF RFC 1035: "Domain names - implementation and specification "(STD 13).

[9] IETF RFC 2136: "Dynamic Updates in the Domain Name System (DNS UPDATE) ".

[9] IETF RFC 2460: "Internet Protocol, Version 6 (IPv6) Specification".

[10] IETF RFC 2543: "SIP: Session Initiation Protocol".

[11] IETF draft: "Interaction between DHCP and DNS" (draft-ietf-dhc-dhcp-dns-12.txt).

[12] IETF draft: "A Lightweight Presence Information Format (LPIDF)" (draft-rosenberg-impp-lpidf-00.txt)

[13] IETF draft: "SIP Extensions for Presence" (draft-rosenberg-impp-presence-00.txt)

[14] IETF draft: "SIP Extensions for Instant Messaging" (draft-rosenberg-impp-im-00.txt)

[15] WAP Forum: "Wireless Application Protocol Architecture Specification"(1998) URL: http://www.wapforum.org/WAP Specification

[16] SMPP Developers Forum: "Short Message Peer to Peer Protocol Specification v3.4".

3 Definitions, symbols and abbreviations

[Editor's note: Chapter to be completed]

3.1 Definitions

For the purposes of the present document, the following terms and definitions apply.

push service: is the delivery of information (data/multimedia) from a network node to a user equipment for the purpose of activating the UE, providing information from the network and activate e.g. PDP context if needed.

Editor Note: This definition should align with the definition in TS 22.060. An example of push services is stock quote notification.

delivery network: a network that provides connectionless or connection oriented push services. A delivery network may simply be a GPRS network, or it can include additional proxies or equipment (e.g. SIP Proxy, Push Proxy, SMS Service Centre).

application server: a server that provides push services through a delivery network, e.g. via an IP connection

user IP address: an IP address provided by the delivery network that can be used by an application server to access to a push services user. The address can be temporarily assigned to the user so that the network shares the address among multiple users.

user-ID: an identity or name that can be used to deliver push content to a user in a delivery network The format of user-ID is dependent on the protocol for the push services. A telephony number presented in character format an example of a possible user-ID.

user availability: the ability of an delivery network to provide push service to a subscribed user.

user terminal: the end user equipment that receives push content. For a GPRS PLMN, the user terminal is the MS or UE.

3.2 Symbols

For the purposes of the present document, the following symbols apply:

G_{dns}	A new interface defined to allow a PDNS to request a GGSN to activate a PDP context for a specified IMSI. A GGSN will use this interface to provide IP address updates to a PDNS.

3.3 Abbreviations

For the purposes of the present document, the following abbreviations apply:

SC	SMS Service Center

4 Introduction

A number of current and future services require the capability for an external IP network to "Push" data to 3G terminals in PS Domain. R99 specifications allow operators to provide push services by using static IP address (and only when GGSN stores static PDP information for the IP address) or by having long-lasting PDP contexts. However, as mobile application services in the PS Domain are emerging in the future, the following additional service requirements should be considered.

- Push services should be provided whenever networks can reach mobile users. In other words, even though a bearer connection between network and MS is not established, users should be able to enjoy push services.

- When IPv4 connectivity is used, IP address should be assigned not only statically but also dynamically. Also, in order to use dynamic IP address, other identities than IP address are necessary.

The present document examines the feasibility of architecture for a delivery network that provides push services with the requirements stated in this TR. In addition to the push services principles above, the architecture shall consider the following aspects:

- How common push services can be offered both through an UMTS IP access and through other IP access networks (the work being performed by IETF should be considered to this respect).
- How the service works in a roaming case

5 Requirements

The delivery network architecture that can provide push services on top of its IP connectivity service shall support following requirements:

- Push services should be provided whenever networks can reach mobile users. In other words, even though the bearer connection between network and MS is not established, users should be able to enjoy push services.
- It shall be possible to provide push services to a mobile user with a dynamically assigned IP address.
- A protocol for push services shall be independent of the type of delivery network. The initiation procedure for the push services, except the user-ID, shall be the same regardless of delivery network.
- A delivery network supporting push services shall provide restriction and security mechanism to protect user from unwilling access.

A delivery network may be able to provide user availability status to an application server if requested by the application server. This information may also include UE capabilities and QoS support in this delivery network.

A network may specify a required type of IP connectivity path for a push service at the initiation of the push service. E.g. QoS.

[Editor Note: the push service may provide multicasting to multi-users.]

6 General Description

This section defines the general push architecture concepts and environment. In this reference architecture there are three entities that should be considered: a user (includes the user terminal), an application server, and a delivery network. To clarify the functionality of the delivery network, the relationships among these entities are specified.

6.1 Service Environment and Scenario

To offer a push service to a user through a delivery network, there are two approaches depending on type of contents to be delivered. One content type can be delivered directly to the user with single message. Another content type requires a sequence of messages, e.g. a movie clip that streams for some while.

The single message content type does not require a dedicated IP connection. However, the other content type requires a dedicated IP connection for communication between the application server and the user.

6.1.1 Dedicated Connection Approach

In this approach, an application server offers a push service to a user through a dedicated IP connection. The dedicated IP connection provides an application server with the service necessary for high bandwidth push content.

Figure 6.1 shows the dedication connection push service environment.

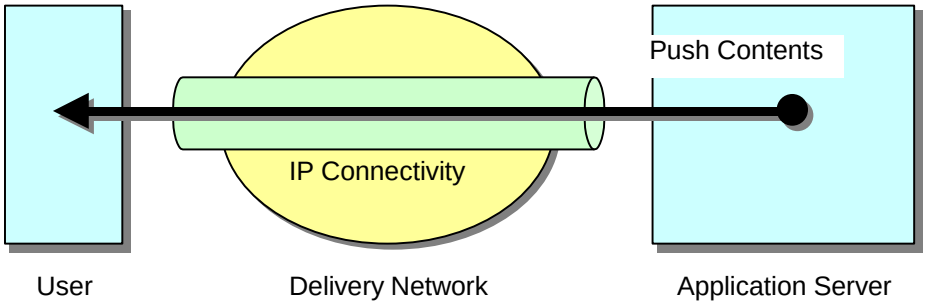

Figure 6.1: Dedicated Connection Reference Architecture Entities

Some networks may have limited resources for services (i.e. a limited number of IP addresses). In such a case, the network may share resources by allocating a dedicated connection at service initiation and releasing the connection when the service completes. Figure 6.2 shows the general service scenario.

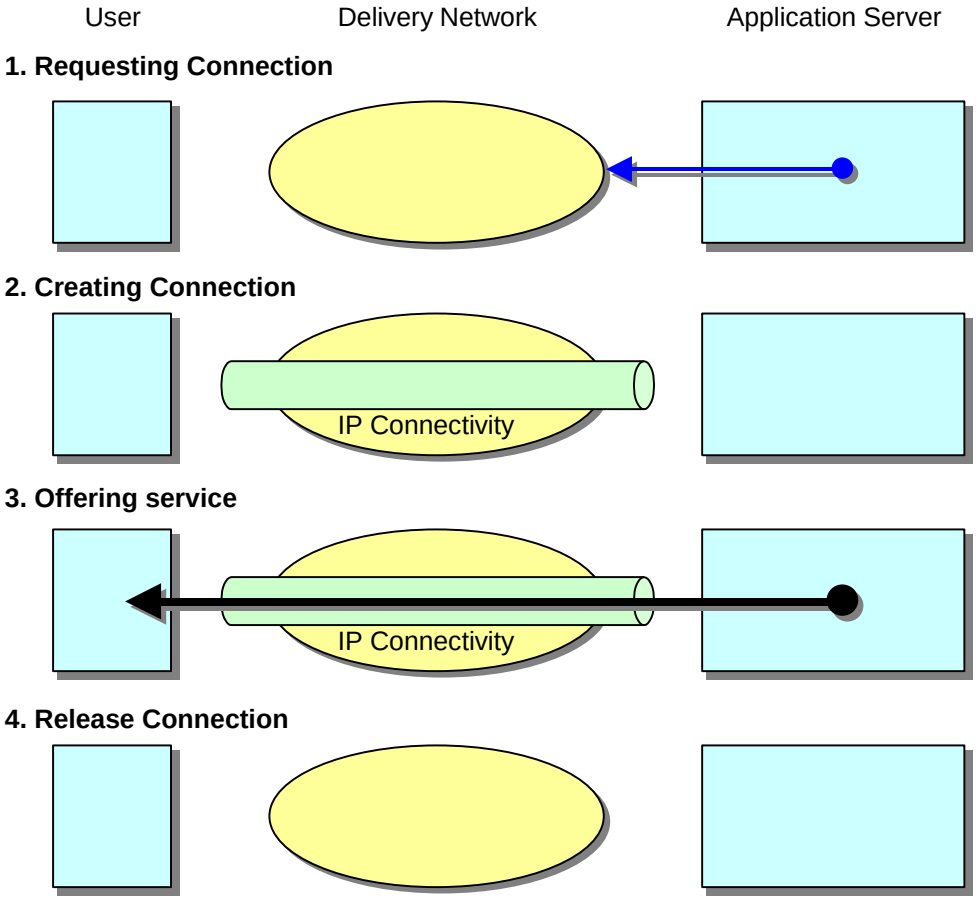

Figure 6.2: General Service Scenario in Dedicated Connection Approach

1) The application server requests a connection to the designated user.

2) The delivery network or the user establishes the IP connection, and returns the IP address for the connection to the application server.

3) The application server delivers the contents using the returned IP address.

4) The application server or the user releases the connection either after completing the delivery of the contents or some time thereafter (application dependent).

6.1.2 Connectionless Approach

An application server offers a push service to a user through a delivery network.

Figure 6.3 shows the connectionless push service environment.

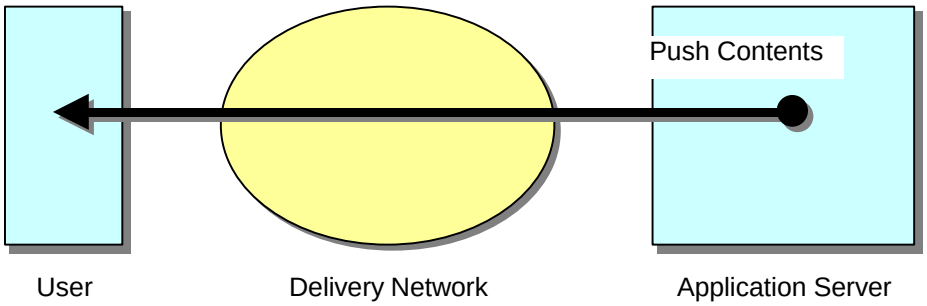

Figure 6.3: Connectionless Reference Architecture Entities

Figure 6.4 shows the general service scenario.

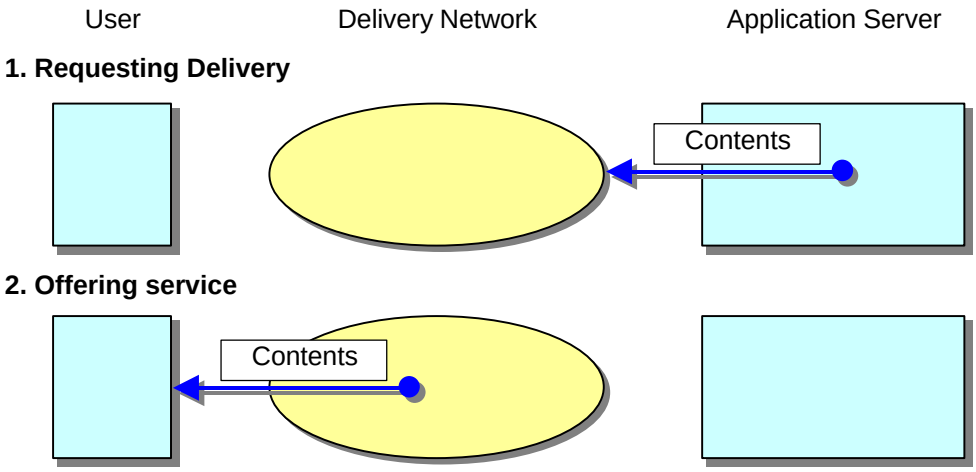

Figure 6.4: General Service Scenario in Connectionless Approach

1) The application server provides the user address and push contents in the same request.

2) The delivery network delivers the content to the designated user.

6.2 Addressing

An application server identifies the user by a user ID or address. The user ID is either a globally unique ID or it may be locally unique within the delivery network when the application server has the ability to uniquely identify the delivery network as well. For example, an Internet E-mail address is an example of a user ID. The user ID may be used to request a connection (step 1, figure 6.2) or to request delivery in a connectionless push (step 1, figure 6.4).

There are multiple methods for addressing push services users. Each addressing method is associated with a specific architecture alternative. The methods identified for addressing the push user are:

- Send push content as an IP packet addressed directly to user's IP address (requires a static IP address).
- Send SIP Invite to end user with user's SIP identity to establish a session, then use the returned IP address to send push content over the SIP bearer connection.

- For connectionless delivery, a SIP Notify may be sent to the SIP identity with the push content embedded in the Notify message body.
- Send a DNS query with the user's Domain Name. Use the returned IP address to deliver push content to the user.
- Send SIP Invite to new PLMN server's SIP identity with the user's push address (e.g. MSISDN) embedded in the Invite message body. Use the returned IP address to send push content after a bearer connection is established.
- Send a request to a new PLMN server with a unique user ID using a push protocol. The PLMN server will return an IP address to the originating application server to allow use of a dedicated connection for push content delivery by the application server.
- For connectionless delivery, the push request to the PLMN server includes the entire push message contents as well as the user ID in the push protocol. The PLMN delivers the push message directly using the user ID (i.e. without returning the IP address to the application server).
- Send push content to the SMS SC (IP address) with the user's SMS address (e.g. MSISDN) embedded in the message delivered to the SMS SC. This is a connectionless push only.

Each addressing method is discussed in detail later in this document.

In dedicated connection case, an IP address for the user is required so that the server can transfer push contents over IP. The architecture shall allow the delivery network to share resource, e.g. IP address. . The application server requests a connection to the user at service initiation and the server or the user may release the connection (and the address) when the service completes. This IP address is used to route push contents in the third phase of figure 6.2. Thus the push services network is responsible for translating the User-ID and supporting allocation of an IP address for the dedicated connection.

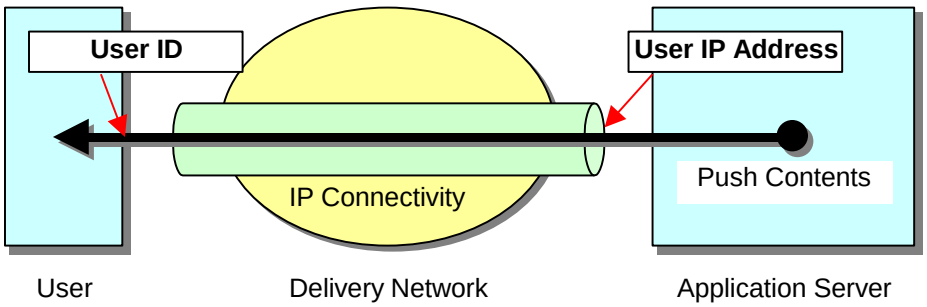

Figure 6.5: User-ID and User IP Address

6.3 Dedicated Connection Establishment

At times, the designated user may not have an active IP connection when the application server initiates a push request. In this case, the network and/or the user will establish a new IP connection.

The application server may provide QoS parameters with the initial push request.

6.4 Push Content Delivery

In the dedicated connection approach, push content is delivered over the established IP connection.

In the connectionless approach, the content is delivered over an existing delivery path. An existing delivery path may be SMS, or it may be an IP connection using a static IP address, or it could be an established SIP message signalling path.

6.4.1 Reliable Delivery

If a user is not available (e.g. not attached to the network) when the application server attempts a push delivery, the delivery would fail. One option for the application server is to simply to retransmit until the user becomes available. Another option is a store and forward mechanism in the delivery network. The third option is presence notification from the user terminal or delivery network.

6.4.1.1 Store and Forward

If the user terminal is not available when the application server wants to push the contents, the delivery network may store the contents and try to send them later Figure 6.6 shows a service scenario with store and forward.

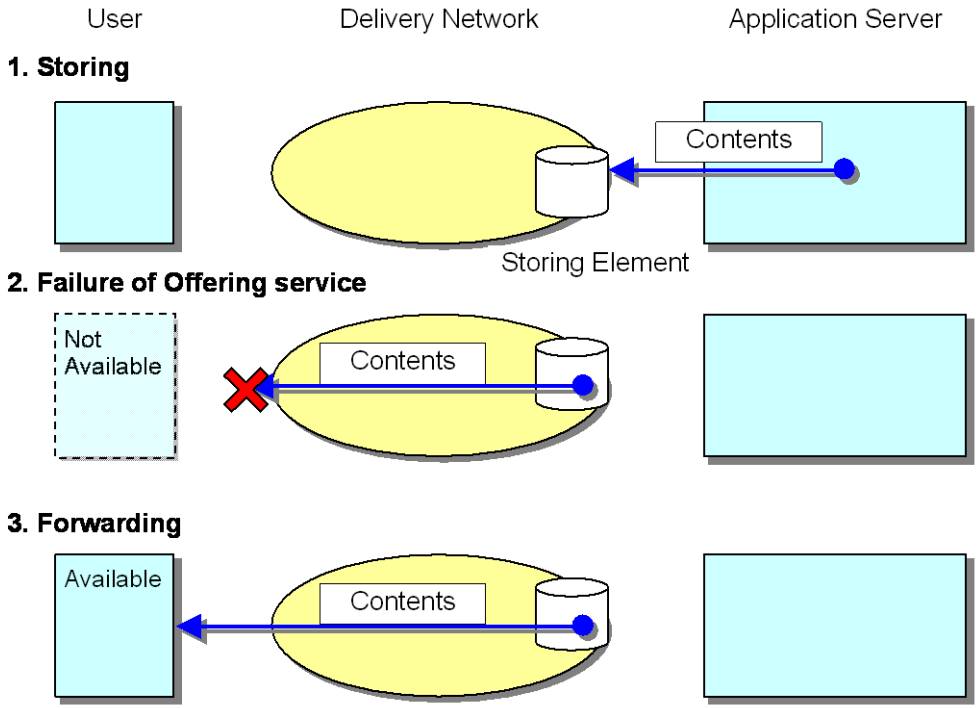

Figure 6.6: Service Scenario with Store and Forward

6.4.1.2 Presence Service

Presence service allows the application server to receive notification when the user becomes available. This notification could come directly from the user terminal in the form of a direct application level registration, or it could come from the delivery network using some form of presence service indication.

Figure 6.7 shows a delivery network based presence service scenario. Presence can be delivered, for example, by SMS or SIP.

Note: For certain presence services the scenario may be optimized by inclusion of a request for notification at the time of the connection request.

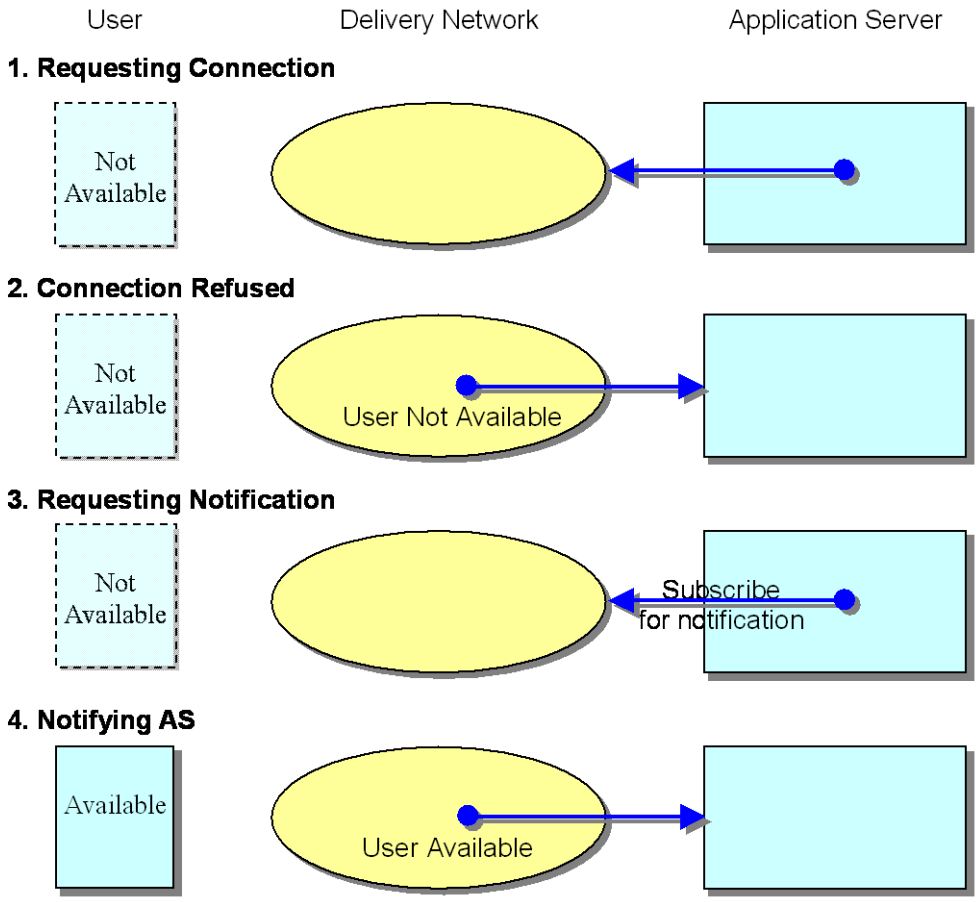

Figure 6.7: Delivery Network Based Presence Service Scenario

Figure 6.8 (below) shows a user terminal based presence service scenario. In this scenario, the user terminal provides a notice to the application server when it becomes available. Since the user terminal is not available when the application server attempts delivery, there is no opportunity for the application server to subscribe with the user terminal at that point for subsequent notification.

User terminal presence is managed end-to-end at the application level. Details of such application level negotiation are outside of the scope of this specification. However, as an example, user terminal presence may be provided in one of the following ways:

- Application protocol requires the user terminal to always "register" with the application server when the User becomes available. In this case, step 0 shown in figure 6.8 is not included.
- When a user requests a specific application/service that requires reliable delivery, the application server negotiates presence notifications to be provided by the user terminal when the User becomes available (optional step 0 below). These would continue to be required until the application server re-negotiates to turn this option off.

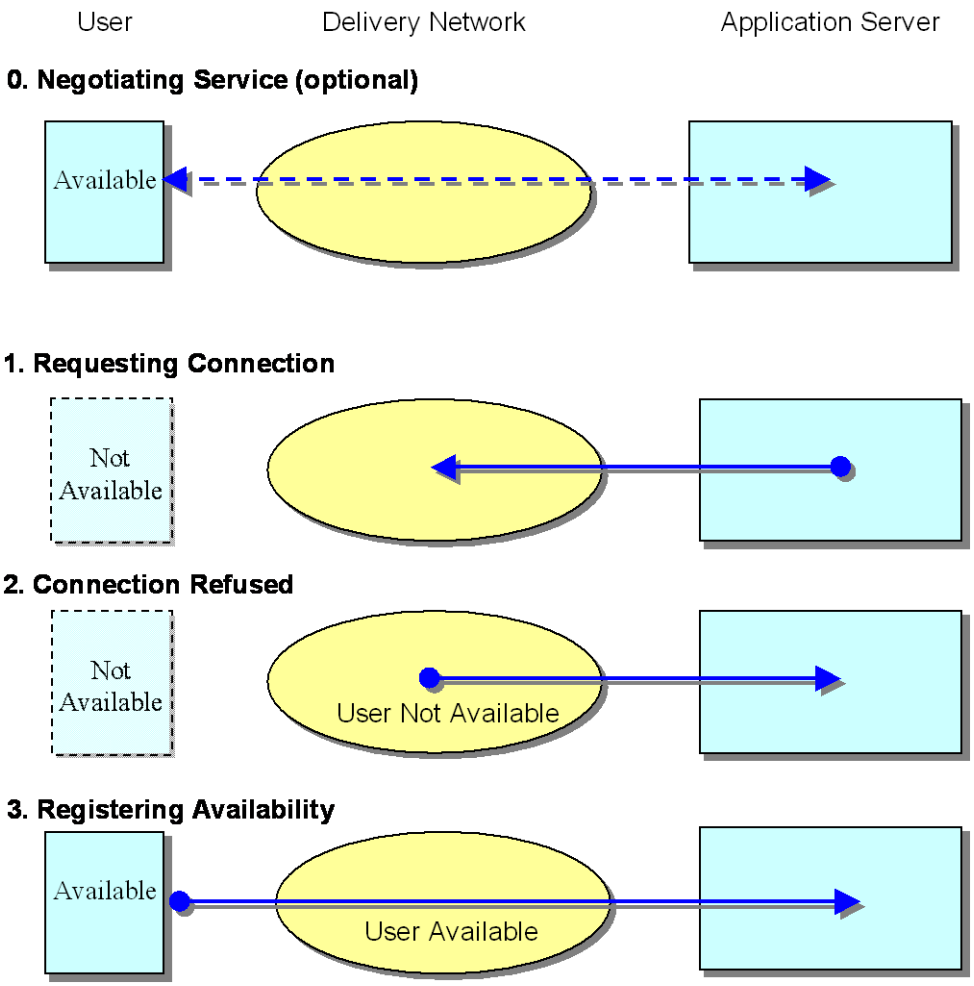

Figure 6.8: User Based Presence Service Scenario

When user based presence is provided, the user terminal is responsible for delivery of each end-to-end application level registration/notification. The user terminal must know which applications require registration, and it must store information for each application server that has negotiated presence notification.

6.5 Multiple Services

A user may subscribe push services provided by multiple application servers. The delivery network shall support delivery of push content from multiple sources simultaneously. This includes support for multiple push service connections.

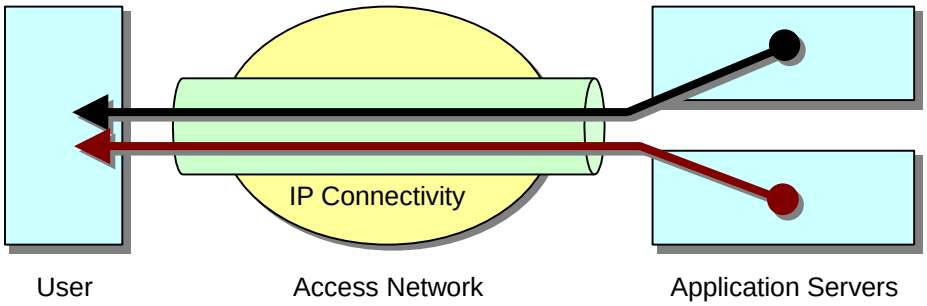

Figure 6.8: multiple push services over single IP connectivity path

6.6 Security and Charging

A delivery network shall protect a user from unwanted attack by application servers. The most basic level of security will be refusal of connection or push content. This may be accomplished via a firewall at the boundary of the delivery network. In addition, push architecture alternatives may include additional subscription control on a per user basis. The delivery network may deny access from application servers that this user has not subscribed to or does not desire content from, based on the registration. The network operator may also charge based on user subscription to specific services.

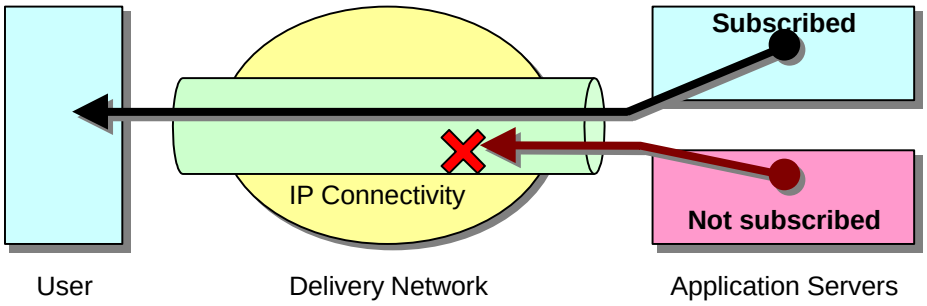

Figure 6.9: Denial of Service Based on Subscription

6.7 User Terminal

A user terminal capable of push services must support the application protocols used for push content. Additional user terminal requirements vary depending on the push architecture.

The push application in the user terminal may be activated by the reception of an initial message from an application server or during an initialization/provisioning procedure initiated by the delivery network.

6.8 Roaming Support

PLMNs support roaming service. Push service shall be available to subscribed users when they roam. The method used to deliver or follow a user when he roams is dependent on the push architecture. However, each alternative architecture uses either a redirection method or a forwarding method.

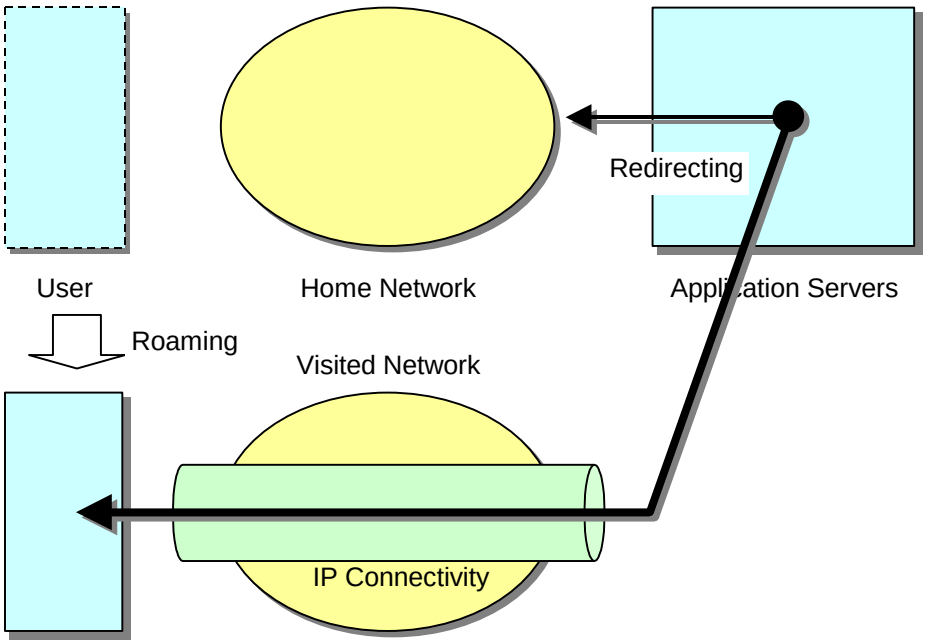

Figure 6.10: Roaming Support by Redirecting

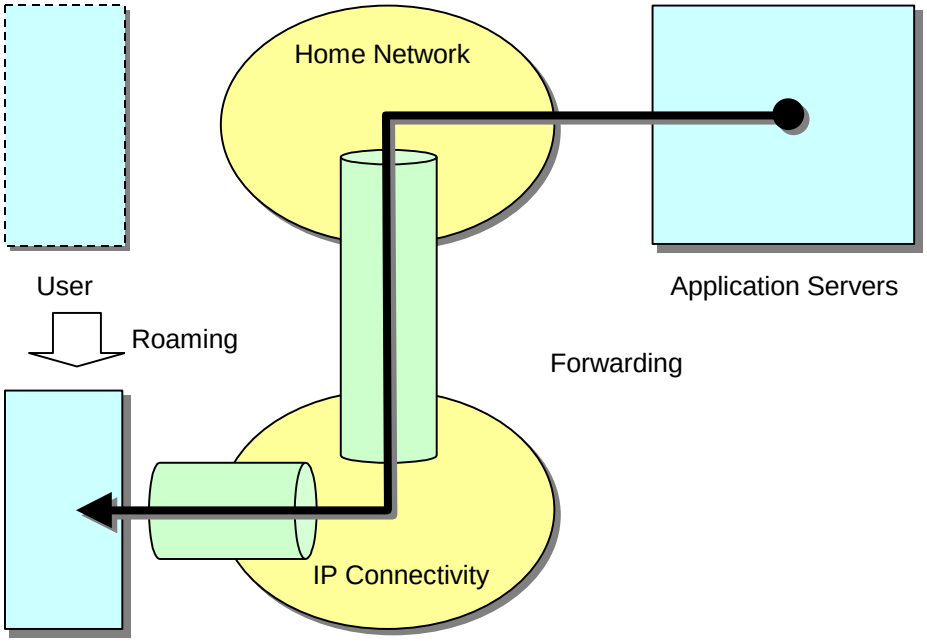

Figure 6.11: Roaming Support by Forwarding

7 Architecture for GPRS

7.1 Introduction

This section describes various solutions to be applicable to the GPRS PLMN. The principles in section 6 shall be applied

[Editor's note: Checking whether all principles have been considered is needed.]

7.2 Network requested PDP Context activation with User-ID

7.2.1 Functional Architecture

The architecture includes the following entities: Application server (AS) in the external PDN that wants to communicate with GRPS MS, a GRPS Mobile Station (MS) that waits requests from ASs, Notification Agent (NA) in GGSN that processes the requests from the ASs, Address Resolver (AR) that keeps relations between user-ids and their correspond IMSI, and other GPRS network entities (see Figure 7.2.1).

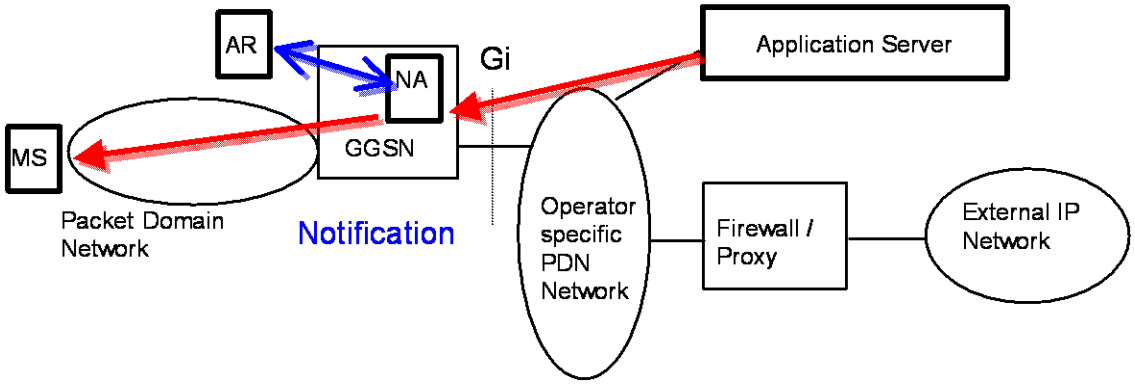

Figure 7.2.1: architecture for PDP context activation with User-ID through GPRS

7.2.1.1 Application Server (AS)

AS serves application that requests MS to communicate with the server over GPRS like VoIP or push application. AS may or may not be able to know in advance that there is no PDP context for the MS. If AS wants to be aware of the status of the user's PDP context, it is a necessary procedure for GPRS network to inform AS, but the procedure is FFS. As one possibility, there is a method that AS decides the status of users PDP context by means of the status of other session to the same user. AS sends application's PDUs to the user's address (it is PDP address for the user) that NA assigns while PDP context activation procedure and is sent to AS by NA.

7.2.1.2 Notification Agent (NA)

NA in GGSN controls the users PDP context activation with dynamic PDP address requested by AS. The GGSN receiving the request may be chosen statically or may change dynamically on session basis depending on the load of PDN or GGSN etc. To achieve dynamic GGSN selection, there may be DNS in the external PDN and AS inquires the IP address for GGSN to the DNS.

NA identifies the requested MS by means of AR that resolves its IMSI from user-ID and activates network requested PDP context activation for MS to invite PDP context activation with dynamic IP address. This delays the PDP address allocation as far as possible and it enables the efficient use of GGSN PDP address or other Gn I/F resources. After assigning the address, NA sends it to the AS.

7.2.1.3 MS Address Resolver (AR)

AR keeps the relations between external User-IDs and IMSIs and provides the information for NA to identify the requested MS. AR may be integrated with GGSN. In case of the type of user-ID is MSISDN, it is realistic for HLR to integrate AR. By this integration, a visited network or a GGSN in the visited network via which AS wants to connect to a MS doesn't have to equip AR for the visited MS.

7.2.1.4 Mobile Station（MS）

MS may deactivate a PDP context but still keep the application active when the application enters the state waiting requests from the server. This helps the GPRS network to save the resources. When some applications run at the same time in the MS, the coordination function in the MS may be required.

7.2.1.5 GPRS Network

GRPS network may release a PDP context of the MS for which the radio connection becomes broken, then NA in the GGSN notifies AS that the PDP address for the MS shall be released and AS enters the state for the MS that there is no valid PDP context.

7.2.3 Protocol Architecture

According to the service scenario in figure 6.2, it seems that two protocol stacks shall be identified.

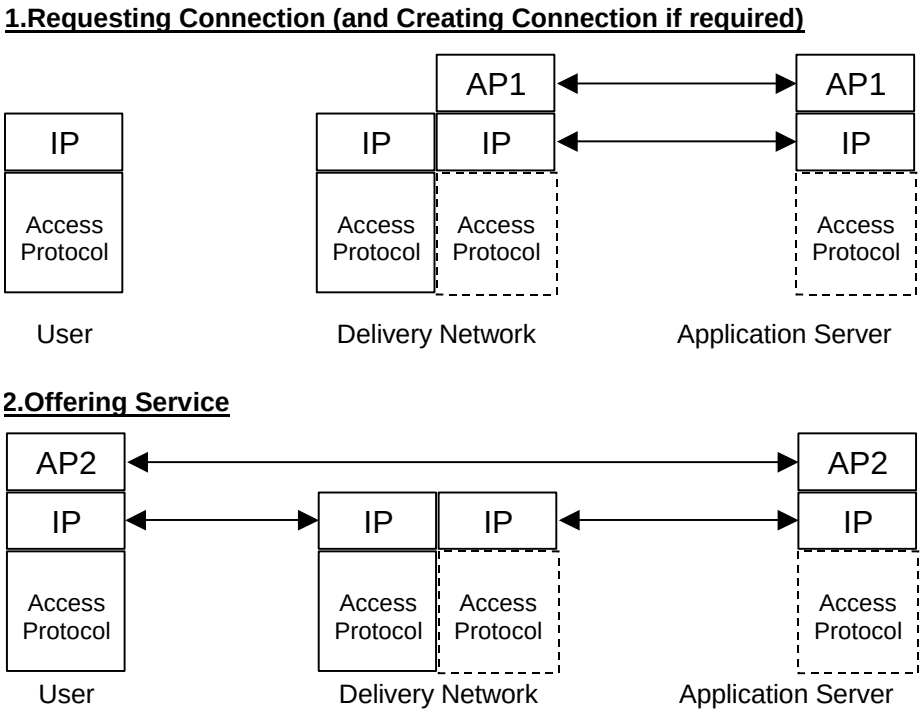

Figure 7.2.2:Protocol Stacks for general push service.

In figure 7.2.2, AP1 is a protocol for requesting connection and AP2 is one for offering push service. AP1 may be capable of requesting connection and of specifying the transport type and the protocol to offer a push service to a user.

Regarding push service offering as a session, SIP is a candidate for AP1 protocol. Figure 7.2.3 shows the push service sequence. In the figure, SIP is chosen as AP1.

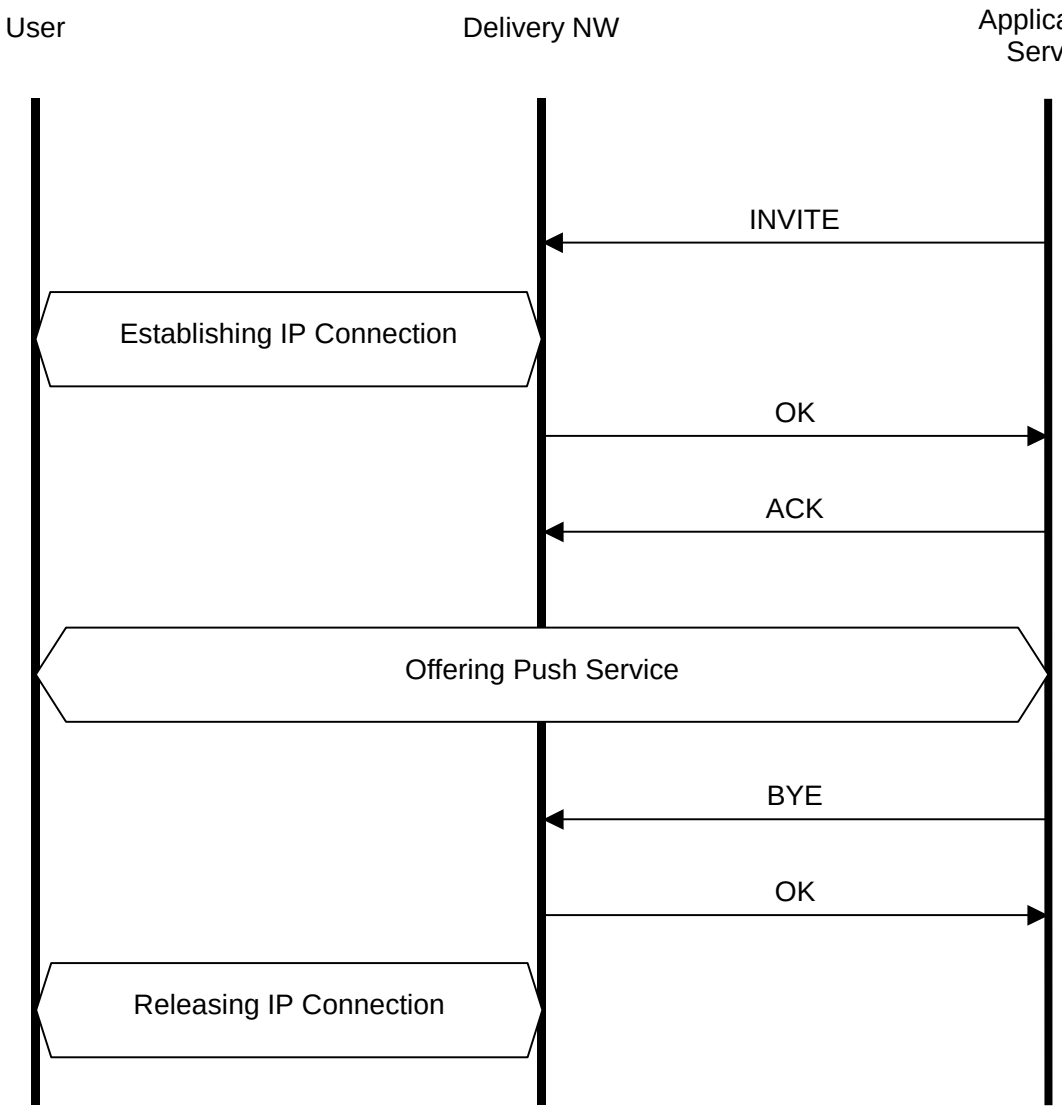

Figure 7.2.3: General Sequence of push service with SIP.

An application server sends an INVITE method to an access network that is derived from the user ID. The server may request the property of required IP connection for the service. The access network receiving the INVITE method establishes the required IP connection for the user and the network return the user IP address by OK response. Then the server can initiate the push service. When the service finishes the server sends BYE method to the network. At the moment the user may release the connection if it is not necessary any longer.

7.2.4 Message Flow

MS to activate PDP context with APN and PDP type and without PDP address. SGSN sends this request to the MS and MS replies it with the same APN and the PDP type and without PDP address. GGSN assigns the PDP address for the MS when it receives the requests and sends it both the MS and the AS. With this PDP address MS and AS are able to communicate with each other via GPRS network.

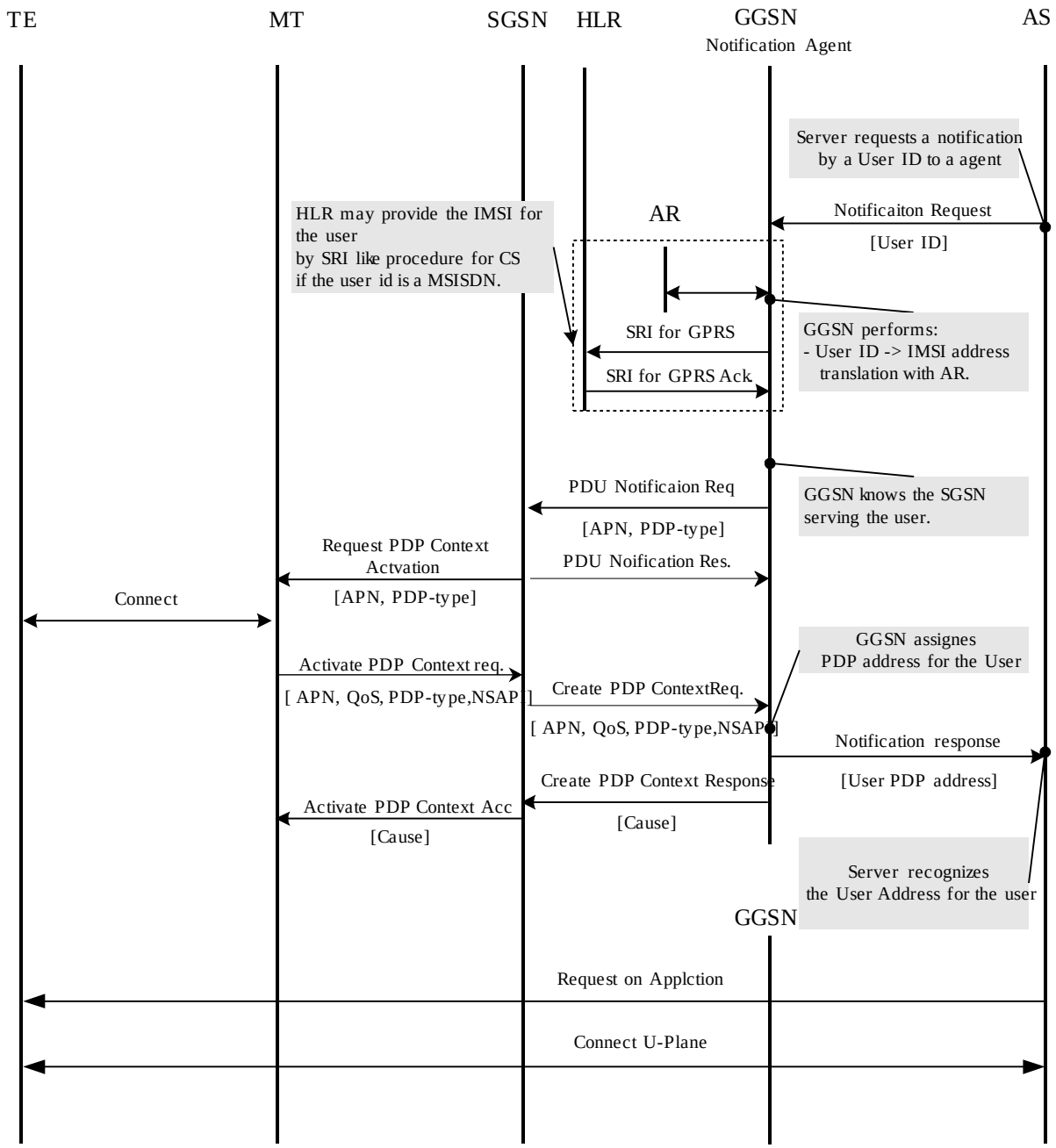

Figure 7:Network requested PDP Context Activation procedure with delayed PDP address allocation

7.2.5 Impacts on 3G specifications

[Editor's note: Chapter to be completed]

7.3 PDP context activation triggered by DNS query

7.3.1 Definitions

New definitions that are introduced by the current reference architecture

Domain Name (DN): a textual string used in Internet to identify a host or a set of hosts. The string shall contain host name followed by the network name (e.g:ggsnanme.gprsnetwork.com, firstname_lastname.gprsnetwork.com or msisdn.gprsnetwork.com).

DNS: an Internet service that translates DNs into IP addresses.

PDNS server: A DNS server that implements the G_{dns} interface. The server database will be modified to also hold the target MS's DN and GGSN.

G_{dns}: a new interface defined to allow a PDNS to request a GGSN to activate a PDP context for a specified IMSI. A GGSN will use this interface to provide IP address updates to a PDNS.

TTL: the duration during which the IP address returned by a DNS or PDNS server is valid. If TTL expires the Application Server must send a new request to the DNS to resolve the target MS's DN to an IP address.

7.3.2 Assumptions

The following assumptions are being made by the reference architecture

- The application server shall not use an MS's IP address if after the addresses TTL expires. If the TTL expires, the application shall perform another DNS lookup to resolve MS's DN to IP address.
- The MS is responsible for requesting appropriate QoS after setting up a PDP context. The procedure for negotiating QoS parameters is outside the scope of this document.
- The QoS of the network initiated PDP context should be interactive or better.
- With IPV6 carriers shall be able to assign static IP address to MS and so might be able to offer push services without needing a PDNS.

7.3.3 Requirements

The following new requirements are satisfied by the reference architecture along with all the requirements discussed in Section "Requirements".

- The access network shall be able to dynamically assign IP addresses to target MSs.
- Push service shall be transparent to the Application Server.
  - The Server shall be able to deliver the service in the same way to a users on wired and wireless networks.
  - Delivery shall be the same for MSs with statically and dynamically assigned IP addresses.
- The Application shall be able to use an identifier for the user that can be resolved to an IP address in a standard way.
- An Application Server must be able to specify a required type of IP connectivity for a push. E.g. QoS (This requirement is satisfied by QoS negotiation at the application level)
- The architecture must be extendable to support peer to peer communication like instant messaging.

7.3.4 General Description

This section describes the reference network architecture and behaviours that enable push services (Figure 7.3.1). A carrier can enable network initiated services by adding one or more DNSs (called PDNSs) to its wireless system. The PDNSs offer a standard DNS interface to the Packet data network. The carrier further must assign a DN to each target MS. For each target MS, the carrier must provision a single PDNS with the MS's DN, its IMSI and the GGSN that is to be used for push services to that MS.

NOTE: Provisioning of GGSNs in the PDSN is not needed if there is a one-to-one correspondence between the PDNS and GGSN.

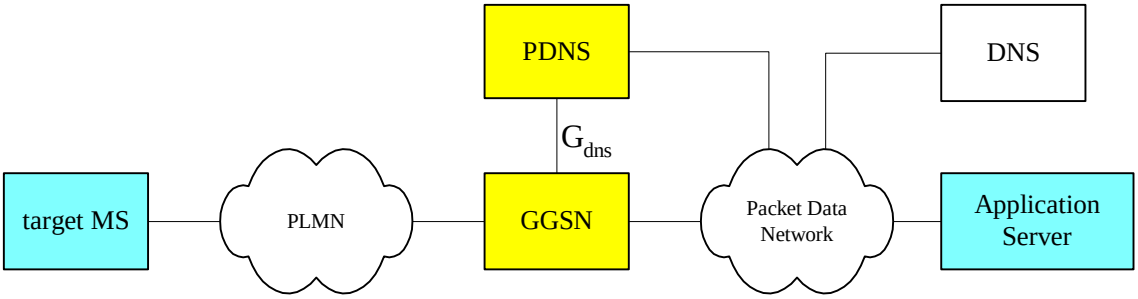

Figure 7.3.1: Reference Push Service Architecture.

A DNS is added to the PLMN (the PDNS). Each target MS is represented in the PDNS by a DN. The PDNS is further provisioned with the MS's IMSI and the GGSN to be used for push services to the MS. The Application Server then can query the DNS to resolve the MS's IP address. If the DNS cannot provide a current IP address, it uses the G_{dns} interface to query the GGSN. If needed, the GGSN will force the activation of a PDP context for the MS, and will report the resulting IP address.

New messages are defined between a PDNS and a GGSN to allow the assignment of an IP address to a target MS for push services.

The proposed method is completely transparent to the Application Server. The Server will use the same address resolution mechanism – standard DNS lookup - for wired users and for wireless users with statically assigned address and for wireless users with a dynamically assigned address. Indeed, an end user can even move from a wireless device to a wired one as long as the user keeps the same DN. For certain types of push service, the Application Server may not need to be aware of the device capabilities.

A new timer - T_{ctxt} - is defined in the GGSN that defines a duration that a dynamically assigned IP address remains reserved for an MS. The timer is started when a PDP context for that IP address becomes inactive. During this period the GGSN must maintain the IP address - IMSI correlation. It cannot assign the IP address to another MS. The value of T_{ctxt} is configurable.

7.3.5 Proposed behaviours for DNS queries

An Application Server that wants to push data to an MS must first query a DNS with the MS's DN.

- When the Application Server queries its local DNS with the MS's DN, the query will be forwarded to the PDNS by established DNS methods.
  If the PDNS contains an IP address for the DN and the TTL indicates that the address has not expired, the PDNS will immediately return the IP address and the remaining TTL. The TTL in the PDNS can be set to infinite for an MS with a statically assigned IP address. For an MS with dynamic addresses it is managed more carefully. This is discussed below.
  The Application Server shall store the value of the remaining TTL and shall use the IP address only while the TTL has not expired.

- When the PDNS does not contain an up-to-date IP address for the DN it retrieves the IMSI and the GGSN for the DN. It then queries the GGNS with the MS's IMSI, using a new message on the new G_{dns} interface. The message is called an "IP Address Query Message".

NOTE: In this implementation a single GGSN is associated with each name. Other contributions to this document show that a GGSN may be chosen from a number of available GGSNs. The choice may for example depend on the GGSN loads. A problem with this approach in the context of this implementation is that one prefers to find the GGSN that already has a PPD context for the MS – if any.

- If the MS has a PDP context that allows for IP push services, the GGSN will return the corresponding IP address over the G_{dns} interface, together with the TTL. If the address is dynamic, the GGSN returns a TTL of T_{ctxt}, for a static address is returns an infinite value. The PDNS then stores this information and responds to the Application Server with the IP address and the TTL.

- If the MS does not have an active PDP context, the GGSN will use the MS's IMSI to force the MS to activate one. The GGSN returns IP address and the TTL to the PDNS after successful creation of the PDP context.
  If the PDP context cannot be activated, the GGSN reports failure to the PDNS. The PDNS then reports failure

to the Application Server.
Alternatively the GGSN can reserve an unused IP address for that IMSI without paging the MS to initiate a PDP context.

NOTE: The TTL value is that of the lease timer T_{ctxt}

Note that under this approach a second application that wants to push data to the same MS will be able to find and use an already reserved IP address, or an already established PDP context - if one exists.

The PDP context and the IP address that are created for the push operation can also be used for other traffic. For example, and Application Server may want to push advertisements to an MS that cause the user to start a browse section on the newly-activated PDP context. Obviously the browser is able to access servers other than the Application Server.

None of the procedures mentioned above allow the Application Server to establish a specific QoS for its push services. QoS must be negotiated explicitly after the IP address of the MS has been resolved.

It may be that the MS already has an active PDP context with another GGSN at the time that the Application Server sends its DNS query. In order for the push to succeed the MS must be able to handle more than one IP address (multi-homing stack).

7.3.5.1 Lifetime of the PDP context

Since IPv4 addresses are valuable resources, a GGSN may want to control the maximum duration of a PDP context that uses a dynamic address. The GGSN may define a timer $T_{ctxtLim}$. For PDP contexts that have been activated as the result of a PDNS query or as result of reception of a packet from the PDN. The GGSN can monitor the traffic associated with the PDP context. It may deactivate the context if there is no traffic for $T_{ctxtLim}$. After an additional duration of T_{ctxt}, the IP address of the deactivated PDP context can then be assigned to another MS. The PDP context may also be inactivated by the MS, for example when all existing sessions on the context have ended. The PDP context may also be inactivated by an external event, such as an MS detach.

7.3.5.2 Choice of T_{ctxt}

The timer T_{ctxt} value is configured by the operator. It impacts system operation in several ways.

A larger timer value will increase the time that an unused reserved dynamic IP address cannot be assigned to another MS. This decreases the efficiency of IP address space management

The TTL returned to the PDNS is equal to the timer value, and impacts the TTL values returned to the Application Server. After TTL expiration at the Application Server, the server will have to make another DNS query. Thus, a larger timer value result in less DNS-related traffic.

7.3.6 Proposed behaviours for IP data delivery

Once the Application Server resolves the MS's IP address, it can push data to the target MS. Data packets will be routed to the associated GGSN.

- If the GGSN receives an IP packet on the IP address of the target MS and the PDP context still exists, the GGSN will forward the packet to the target MS in the legacy way. The MS can send packets to the Application Server.

- If for some reason the PDP context is no longer active and the IP address is a statically assigned one, the GGSN will try to activate a new PDP context in the legacy way.
  If the IP address is dynamic, and the packet is received within T_{ctxt} from PDP context expiration, the GGSN must look up the IMSI for the IP address must try to activate a PDP context. The GGSN must reassign the original IP address. The mechanisms used are very similar to those used for a static address.
  If the packet is received later than T_{ctxt} after PDP context expiration, the packet is dropped. The PDP context is not restored.

7.3.7 Example Scenario

Figure 7.3.2 shows an example of pushing data to a target Ms. In this scenario the MS has neither a statically assigned IP address nor an active PDP context.

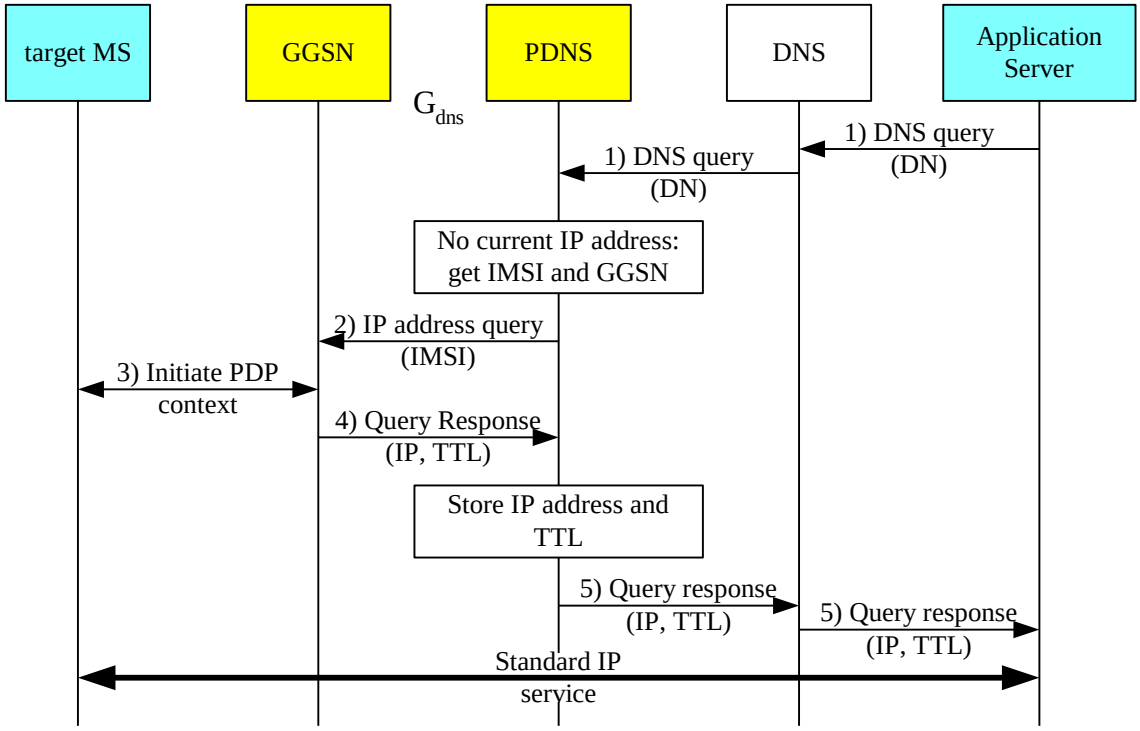

Figure 7.3.2: Pushing data to a target MS

NOTE: The above figure shows that the DNS forwards the query to the PDNS. Alternatively redirection can be used, where the Application Server will directly query the PDNS. This redirection can be persistent.

The scenario consists of the following steps.

1) A module in the Application Server (e.g. DNS client) sends a query to the DNS. This query will be a primitive DNS message as defined in RFC-1035. The request will be routed to the appropriate PDNS.

2) The PDNS server performs a search to fetch the entry corresponding to the target MS DN. In case there is an IP address associated with the DN, and the TTL corresponding to that address is not expired, the PDNS server returns the address together with the remaining TTL.
In the case of this example scenario this is not the case and the PDNS retrieves the MS's IMSI and GGSN and sends a message over the G_{dns} interface to the GGSN. The parameter in the message is the MS's IMSI.

3) The GGSN, upon receipt of message, will look for the target MS's PDP context information. If no information is found it follows the procedure described in GPRS spec to instruct the target MS (mobile) to activate a PDP context (See TS 23.060 clause "Network Requested PDP context Activation Procedure").

4) The assigned IP address will be sent to the PDNS server along with the TTL (This is the time during which the IP address is valid). Procedure for updating DNS is defined in RFC-2136 and is also discussed in IETF draft "Interaction between DHCP and DNS".

5) The PDNS server updates its internal data structures and sends the IP address and TTL in the DNS response to the Application Server. The format of the message is defined in RFC-1035 and is a standard DNS response primitive.

7.3.8 Alternative PDNS Implementation

It is possible to use a fully standard DNS instead of a PDNS. The interface between the GGSN and the PDNS then becomes a standard DNS interface. The DNS still stores the MS's IMSI, keyed by DN. The message flow is slightly modified (Figure 7.3.3). In step 2) the PDNS forwards the Query to the GGSN. This time the parameter is not the IMSI but the DN, which is standard DNS behaviour. The GGSN then queries the PDNS for the IMSI, using the DN as key (step 3). The PDNS returns the IMSI (step 4), which is used by the GGSN to activate the PDP context in step 5). The GGSN responds to the DNS query with the IP address and TTL

The alternative implementation will also work in one to many relationship between PDNS and GGSN provided that the DN contains the anchor GGSN's name.

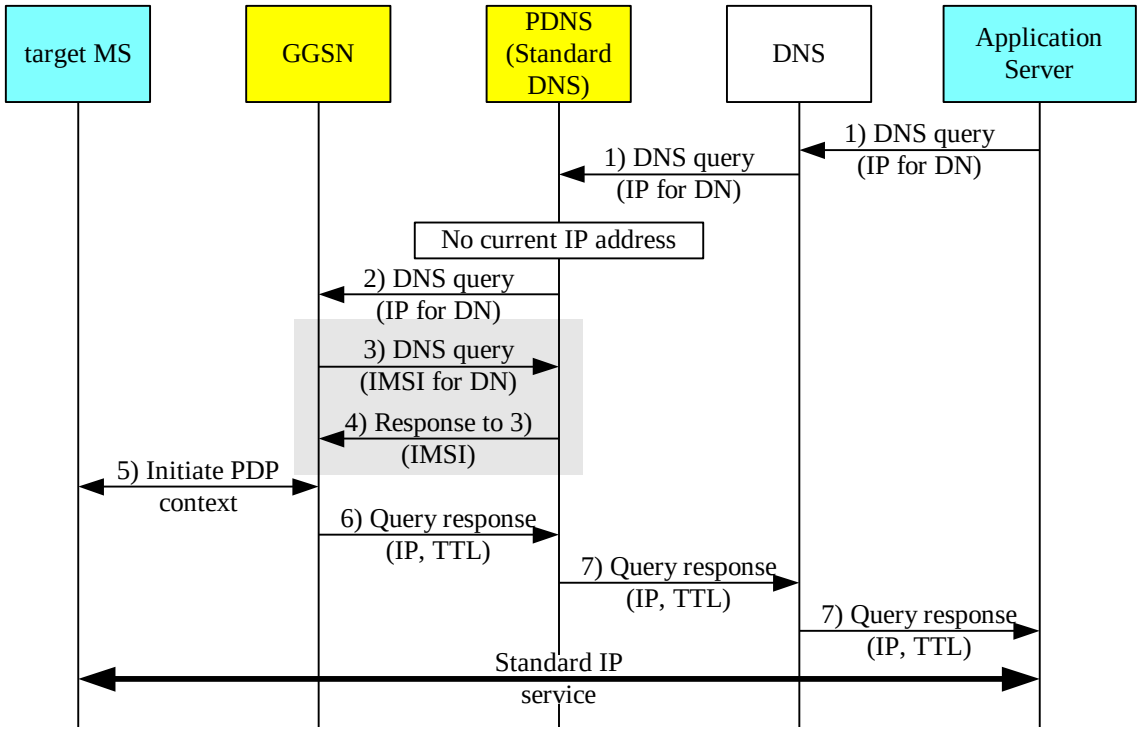

Figure 7.3.3: PDNS implementation using a standard DNS

7.3.9 Alternative GGSN Implementation

In an alternative approach the GGSN defers the creation of a PDP context until it receives an IP packet for the MS. This makes the PDP context activation scenario more like the legacy scenario for a statically assigned IP addresses.

This alternative also increases security under a Denial of Service attack on the PDNS. An unauthorized Application Server may generate a large amount of queries on the P_DNS with different names and cause a large volume of over the air traffic associated with PDP context activations. Preferably the PDNS uses established mechanisms to impose security against such unauthorized queries. This alternative implementation further allows the GGSN to authenticate and authorize the Application Server before contacting the MS Over The Air.

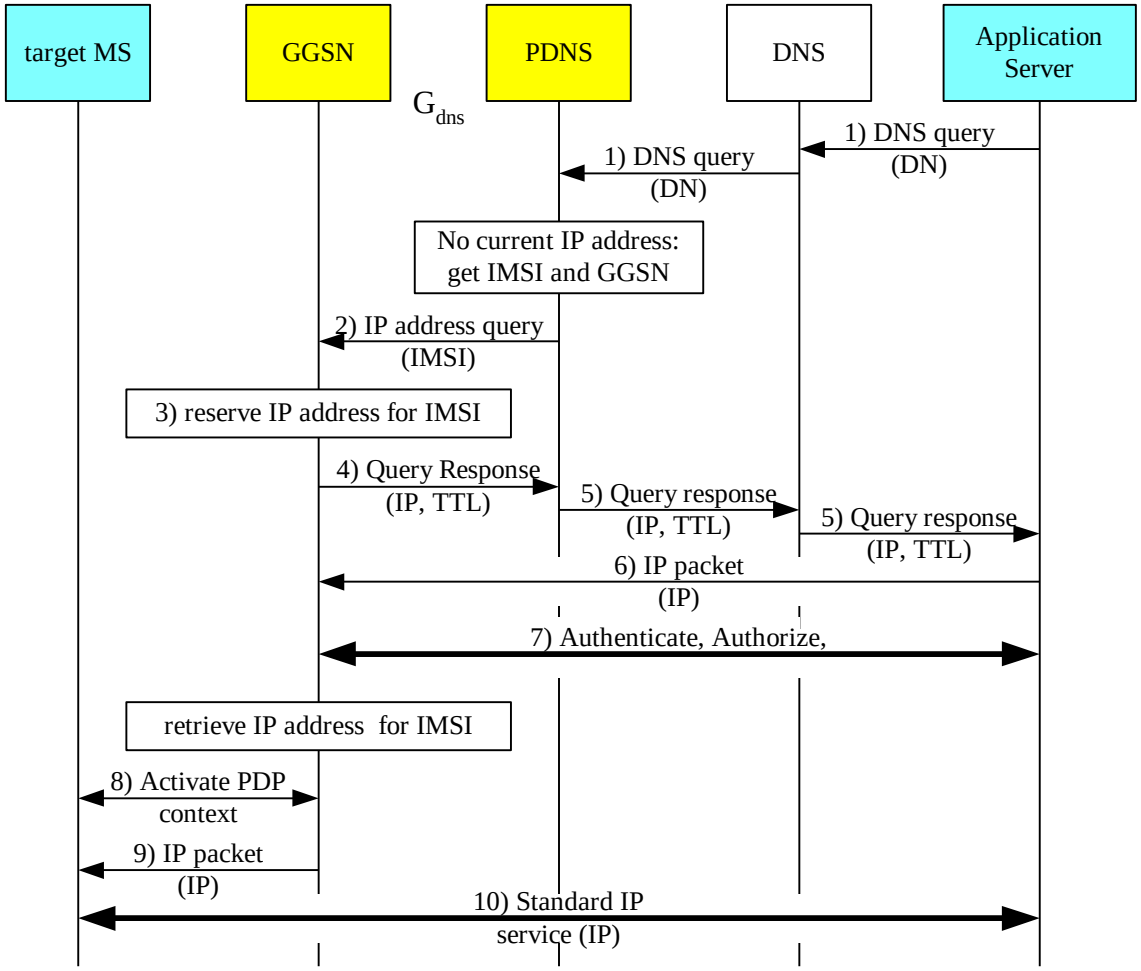

Figure 7.3.4: Deferred activation of the PDP context

This alternative implementation is illustrated in Figure 7.3.4 and is explained in the following steps.

1) The GGSN receives an IP Address Query Message.

2) If the MS has a PDP context that allows for IP push services, the GGSN will return the corresponding IP address together with the TTL. If the address is dynamic, the GGSN returns a TTL of T_{ctxt}; if it is static, the GGSN returns an infinite value.

3) If the MS does not have an active PDP context, and the MS does not have a static address, the GGSN reserves an IP address for that IMSI for a duration of T_{ctxt}.

4,5) The GGSN returns the IP address along with the TTL, as above. The PDNS behaviour after receiving the IP address over G_{dns} interface is similar to that discussed above.

6) After resolving the IP Address, the Application Server starts sending IP packets.

7) Upon receipt of the first packet from the Application Server, the GGSN may perform authorization and authentication (authorization and authentication procedures are outside the scope of this proposal).

   If authorization fails, the GGSN discards the IP packet.

8) If authorization check is successful, the GGSN instructs the target MS to activate a PDP context, using the MS's static address or the IP address reserved for it.
   The GGSN shall return the same IP address in the PDP context notification response to the MS.

This approach will protect the system against creation of PDP contexts for requests from unauthorized application servers. In the default approach a server with access to an MS's DN can force the creation of a PDP context even if it is not allowed to send packets through the GGSN.

This approach will have higher latency when compared to the default approach. The latency for this variation should be in line with that of the network initiated procedure for MS with static IP addresses discussed in GSM document (See TS 23.060 clause "Network Requested PDP context Activation Procedure").

7.3.10 GGSN with embedded PDNS

In a slight twist on the above variation, one can also put the PDNS information inside the GGSN (Figure 7.3.5). The GGSN retains its DNS interface. It has access to a database. The database is provisioned with the domain names and IMSIs of the MSs that use that GGSN as their anchor point. Like a DNS server, the database will contain the MS's IP addresses and TTLs, while relevant. This solution has the drawback of reduced flexibility; the DN of an MS must resolve onto a specific GGSN.

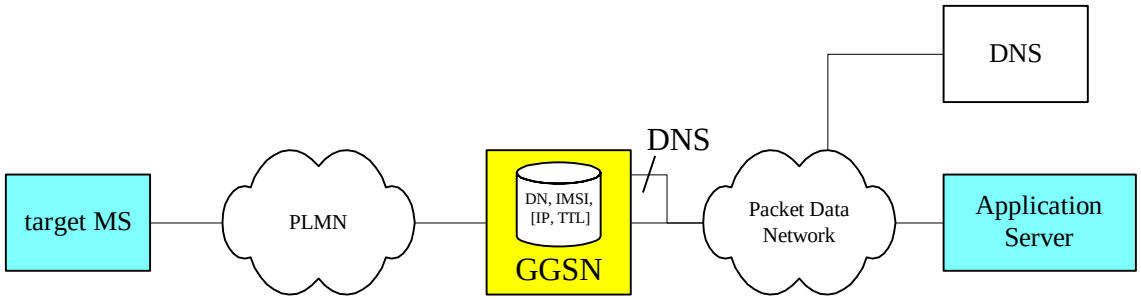

Figure 7.3.5: GGNS with embedded PDNS database.

7.3.11 Avoiding an Application Server Timeout

Under the default implementation, the PDNS will not return an IP address until a PDP context has been successfully activated. An impatient Application Server may time out before the DSN response and hence fails to push its data. The alternative GGSN implementation above defers the PDP context activation and significantly speeds up the DNS response. Unfortunately this is done at the cost of additional latency for the first IP packet. A third variation allows the overlap of the PDP context activation and the DNS query response. It is shown in figure 7.3.6. It has the additional complexity that the first IP packet can arrive before the PDP context is established.

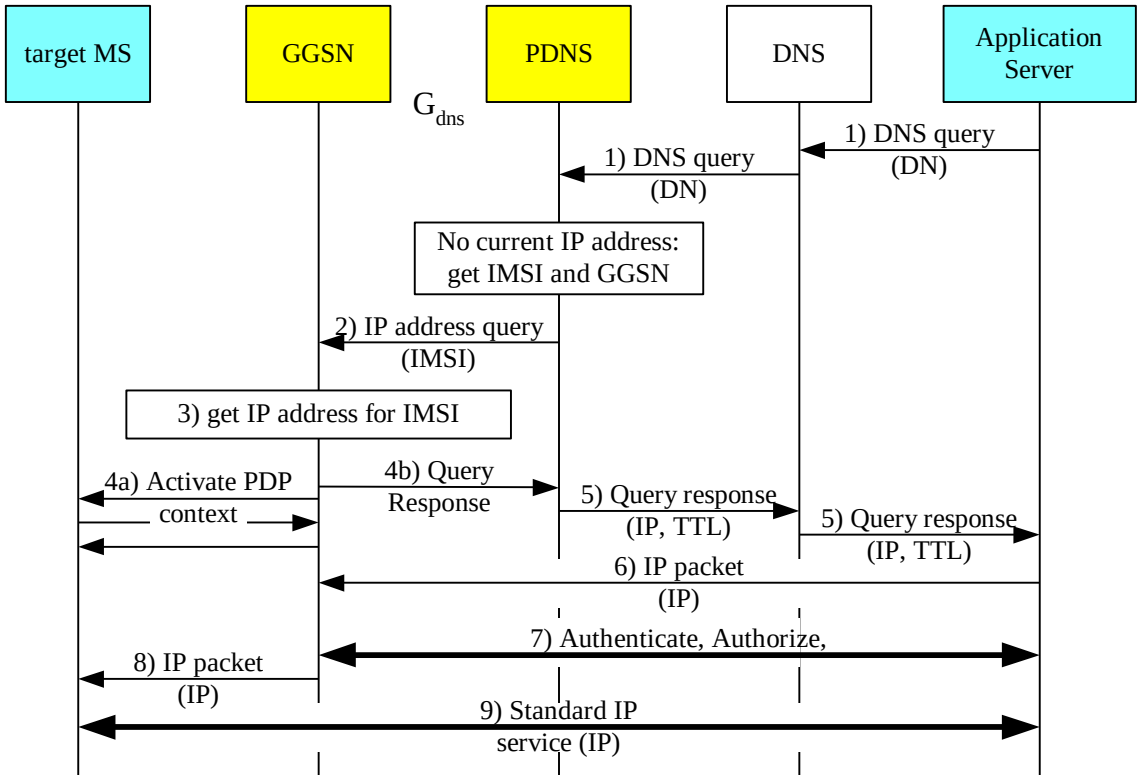

Figure 7.3.6: Simultaneous PDP context activation and DNS query response

7.3.12 Protocol Architecture

The protocol stacks for push service initialisation is shown in Figure 7.3.7. In this figure a new interface - G_{dns} - is introduced to allow a PDNS to communicate with the GGSN to request for the IP address associated with an IMSI. The GGSN uses the G_{dns} interface to update the PDNS with the IMSI - IP address association.

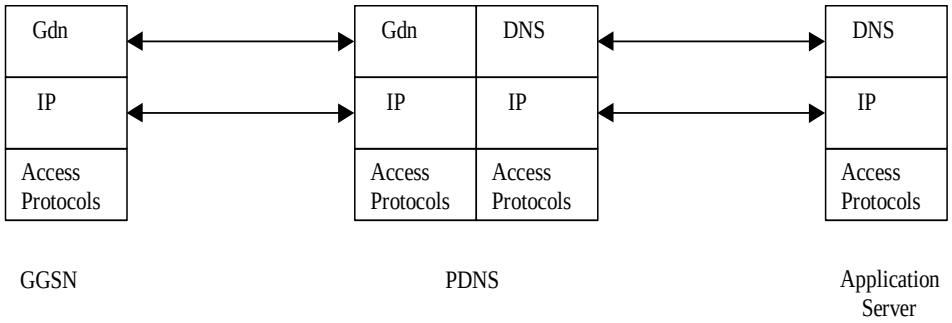

Figure 7.3.7: Protocol stack for push initialisation

7.3.13 Security

Access network shall protect a target MS from attacks by Application Servers. Several approaches can be taken to implement security functionality in the access network.

- The PDNS can implement some of the security features. For example, if it detects an unusual number of resolution requests from a particular source or for a particular DN it can stop giving out the IP address for that DN.

- GGSN shall implement the packet screening criteria specific to the IMSI. Any subscriber-specific screening functions are performed, e.g. verifying the source address, protocol type and port number, enforcing size/volume limits, etc. in GGSN.

- Alternatively one can use a dedicated Gateway. This approach offloads the burden of implementing security features in the access network. Subscriber screening functions like verification of source address, protocol type and port number, and enforcing of size/volume limits, are performed in the gateway.

7.3.14 Roaming Support

The proposed implementation supports roaming service. The PDNS, when needed, will send the IP address query to the same GGSN, independently of whether the Ms is roaming or not. In the preferred implementation the GGSN provides an anchor point for push services. The GGSN retrieves information on the serving SGSN from the HLR before it sets up the PDP context. The PDP context that is activated will terminate on the GGSN. Thus, if a target MS roams to another access network, push service requests coming to the home network will be tunnelled through the anchor GGSN and the serving SGSN. Alternatively one could add new mechanisms to force the creation of a PDP context that terminate on a GGSN in the visited network. This alternative is not described in detail.

7.3.15 Error Responses

- PDNS shall report “Non Existent Domain” if it receives a query with invalid DN.

- PDNS shall report “Query Refused” if a GGSN returns an error (If MS is not currently available or failed to establish PDP context)

These error codes are returned to confirm with DNS specification.

7.4 SMS Push Service

7.4.1 Assumptions

For the SMS push service the following assumptions apply:

- SMS is supported at the user equipment as well as the serving mobile network.
- This SMS approach would be used when the user equipment and/or serving mobile network do not support more advanced push mechanisms such as SIP end-to-end.

7.4.2 Basic Service Scenarios

7.4.2.1 Short Message Push

SMS supports Push of a short message to any mobile handset (2G, 2.5G, 3G). The figure below shows the basic steps involved in an SMS Push service.

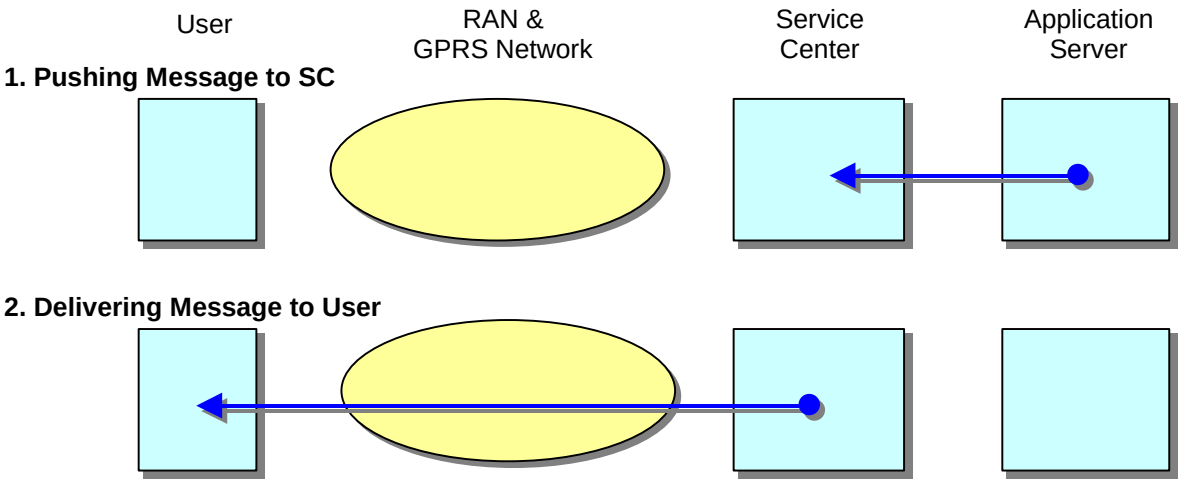

Figure 7.4.1: SMS Push Message Scenario

In the standard SMS Push Service scenario, the SMS Service Center (SC) receives the initial push message from the external application server. The SC delivers the message to the User/UE through the Access Network. Delivery can occur via traditional CS paths or via the PS path (i.e. using the Gd interface).

The GSM/3GPP standards do not fully define the SC's interfaces. The interface from the SC to the Access Network is defined within the 3GPP standards (primarily TS 23.040). The interface to the SC from an external Application Server is not standardized by 3GPP (TR 23.039 provides guidance on this interface).

SC implementations today often support an IP network connection for push message access from an Application Server. This existing interface can be used to allow an Application Server in an IP network to push a message or a notification to a mobile user.

7.4.2.2 Push Notification with User Connect Scenario

When the SMS environment is not adequate, the Application Server can push a notification to the User and let the User establish a direct connection to the Application Server. The conditions for this Notification with User Connect Scenario are:

- data to be pushed does not fit within SMS message limits or
- the Application Server needs a directly addressable IP connection to the User.

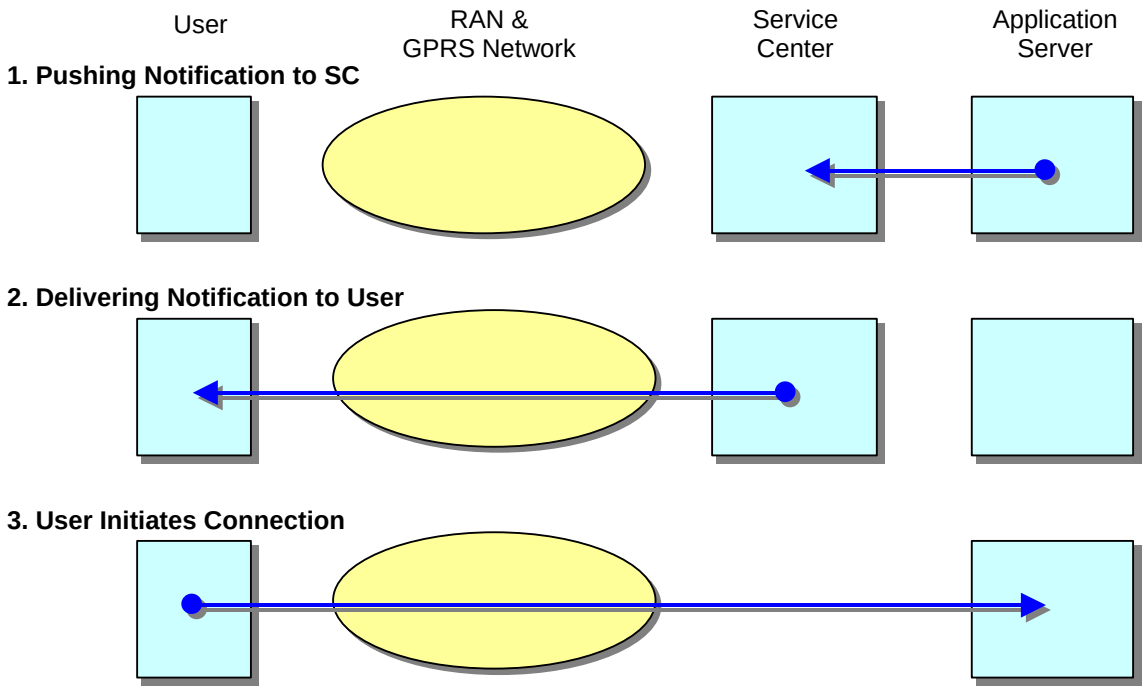

Figure 7.4.2: SMS Push Notification with User Connect Scenario

In this scenario, the notification that is pushed to the user must include the information necessary for the user to initiate retrieval. When the user receives the notification, he can choose to ignore it or he can initiate a connection (e.g. PDP context) to retrieve any additional data.

QoS parameters to be used for the user-initiated connection may be provided by the application as part of the push notification. If they are not provided as part of the notification, they can be re-negotiated, if needed, after the connection is established.

7.4.2.3 Push Broadcast Scenario

The existing standards allow delivery of broadcast messages using SMS formats. This requires support for Cell Broadcast in the Service Center.

Addresses supplied in this case would identify a broadcast area instead of a specific user. This delivery method could be used with either a Push Message or a Push Notification requesting User Connect.

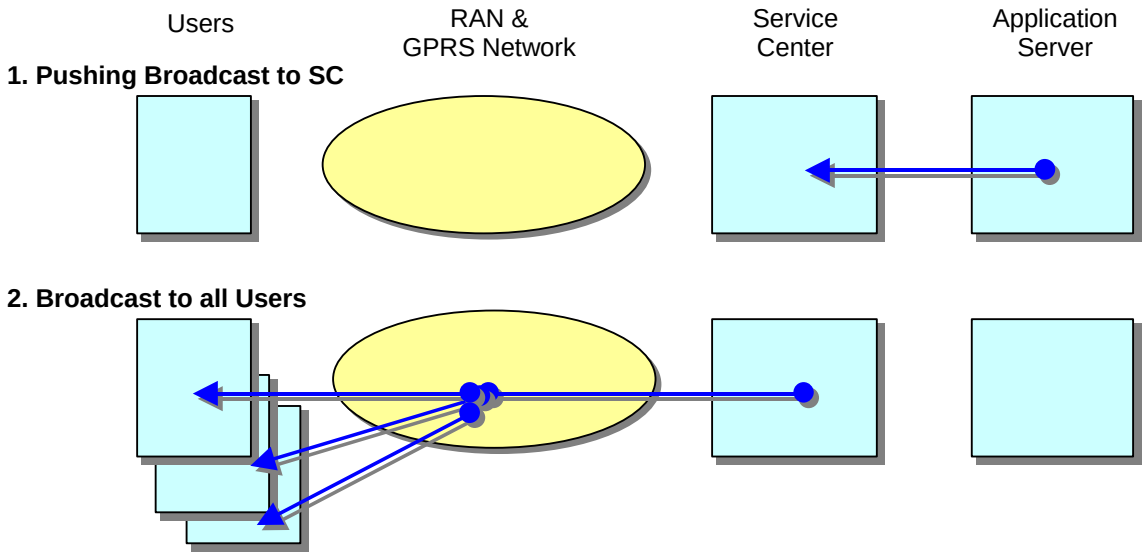

Figure 7.4.3: Push Broadcast Scenario

7.4.3 Addressing

The Application Server will use the existing addressing scheme to the SC. For an IP network interface, the SC will be addressable in a standard network format (e.g. Domain Name, IP address). The IP packet delivered to the SC will contain the destination User (UE) address. This is generally included as the MSISDN or E.164 number. In many cases, the User address delivered to the SC must also include Access Network information.

When the User initiates a connection in response to a Push notification, it may provide its IP address to the Application Server as part of the response. This would be handled at an application level and is outside of the scope of the 3GPP standards.

Delivery of the push message/notification to the destination application within the mobile is dependent on the existing SMS message routing mechanism. As new mobile applications are added that use SMS as a bearer, additional SMS routes may be allocated for these applications (i.e. by defining new SMS "ports").

7.4.4 Subscription, Security, and Charging

The existing security and charging mechanisms for SMS remain unchanged. Network Operators would manage subscription, security, and charging via the SC.

Users would manage retrieval of large messages or connection initiation based on notifications.

7.4.5 Roaming

Roaming would be handled using existing SMS mechanisms.

7.4.6 Delivery Reliability

SMS includes message delivery reliability mechanisms. If a user is not accessible or has some condition that prohibits message delivery, the Access Network will provide an Alert to the SC when the condition has cleared. This allows the SC to attempt delivery again as soon as the user is able to receive the message.

The following figure shows an example sequence with a Push message being delivered while a User's mobile is powered off.

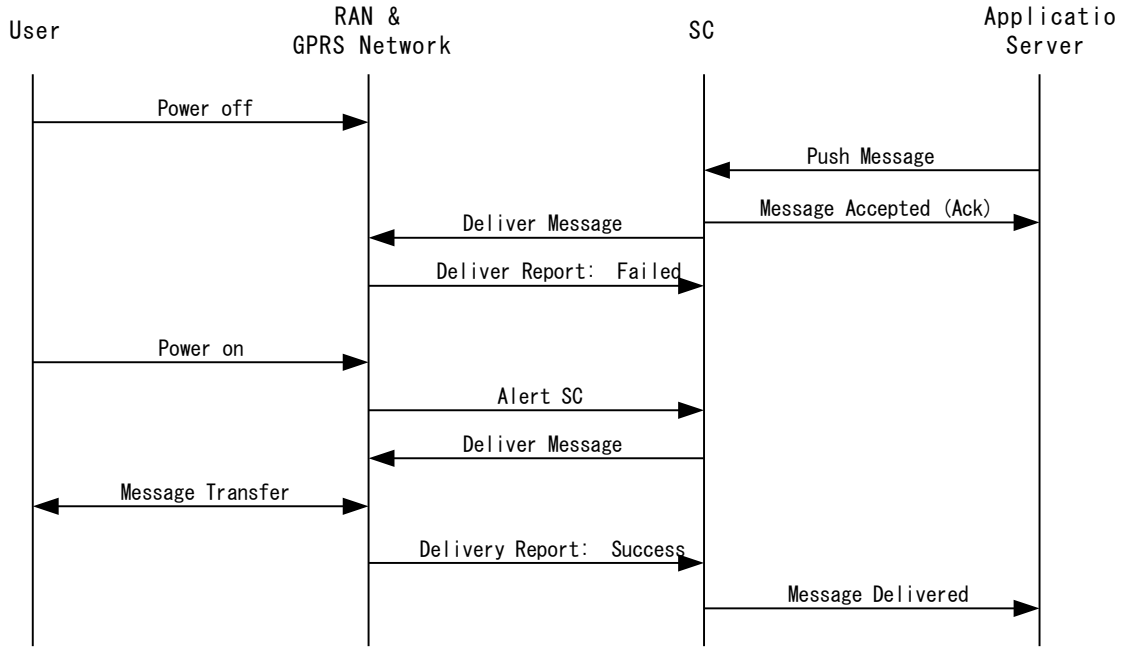

Figure 7.4.4: SMS Reliable Delivery Sequence

As shown in this figure, the SC receives an alert notification when the user becomes accessible. The SC is then able to attempt a second delivery of the message, which now succeeds.

The Alert SC message is provided by the HLR/HSS per the existing SMS service definition.

The reliable delivery feature of SMS would also apply to the "Push notification with user connect" message scenario. In this case, the user may initiate a connection to the Application Server in response to the SMS message that was delivered.

It is also possible for the SC to relay Alert notices to the Application Server. In this case, the Application Server would be responsible for maintaining a copy of the message and re-transmitting when the User becomes available.

The Application Server can represent a push gateway (e.g. PPG as defined in WAP) with a separate push initiator beyond the gateway. Adding an intermediate push gateway simplifies addressing and data formatting for the push initiator.

7.4.7 Protocol Architecture

The protocol used between the SC and the MS is well defined in TS 23.040. SMPP is the most common protocol used today between the SC and the Application Server.

It is possible to adapt the Application Server to SC interface so that it uses SIP instead of SMPP. The Application Server would use SIP to establish a single session with any SC. This session could be used to push messages to any of the mobile users served by that SC.

The figure below shows the protocol architecture involved (from a high level).

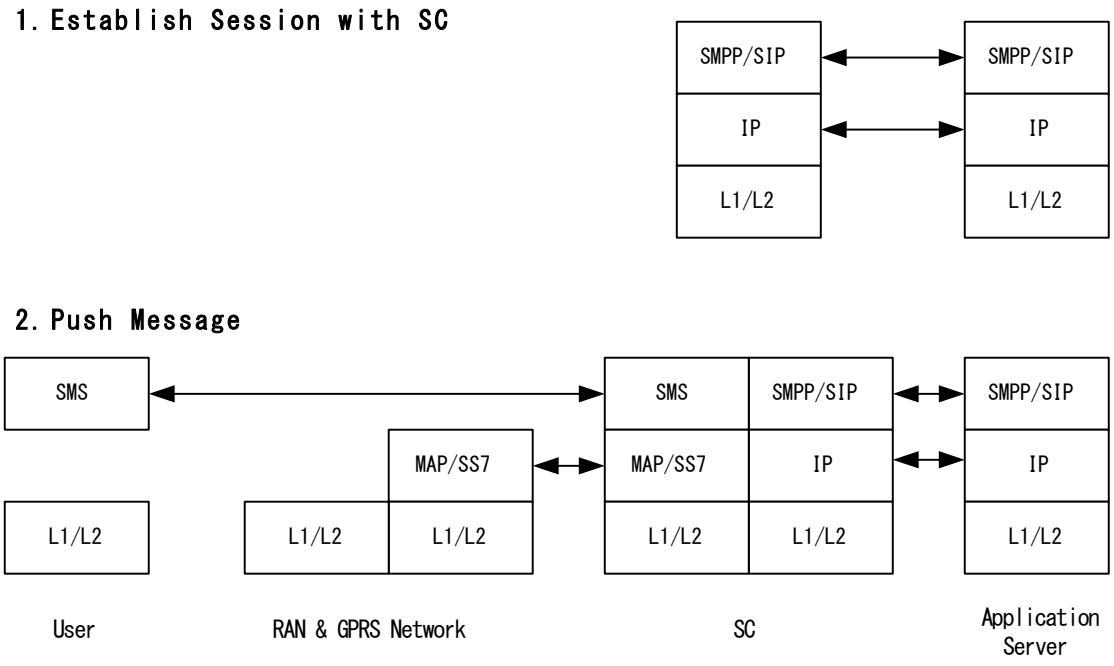

Figure 7.4.5: SMS Protocol Architecture

The Application Server is responsible for providing push content to the SC in a form that is within the SMPP message size limits.

7.5 Push "The internet way"

[Editor's note: The general description for this solution may be required.]

In the Internet, the intelligence is typically at the end points. Therefore, the typical Internet way to provide push service is to have an application in the MS connecting an application in the network to request the activation of push services (see figure 7.5.1).

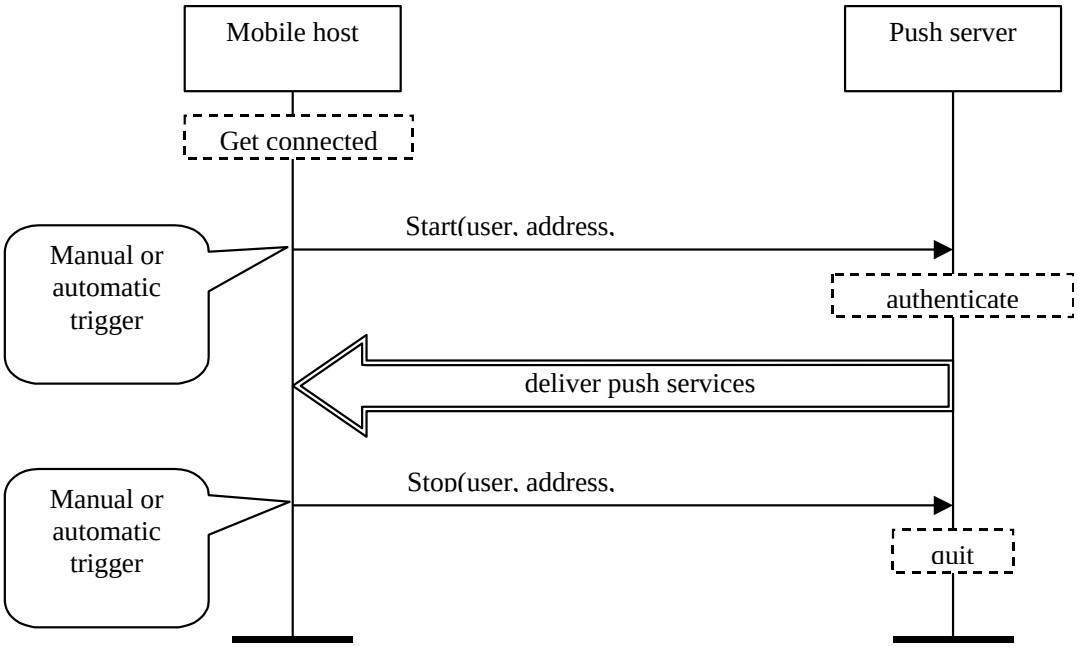

Figure 7.5.1: The "Internet Way" of push service

Such push services are generic, as they are available to any host connected to the Internet. Thus, they are not bound to a particular access technology. In addition, the terminal capabilities do not have to be known by the network in advance, as the terminal would register to the push services only when it is capable of receiving them.

When optimisation of the services is required, the access provider could automatically activate the push service when the host connects (see figure 7.5.2). This solution is particularly applicable to cellular technology such as the GPRS as it saves radio resources. Another aspect of this solution is that it tights the delivery of push services to the access provider push proxy.

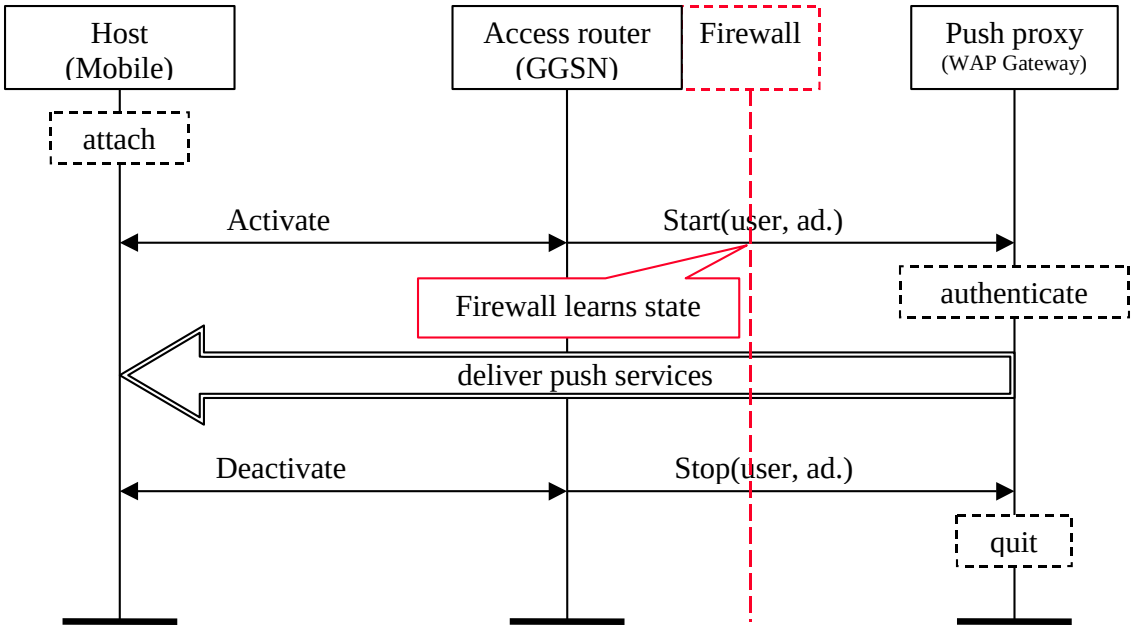

Figure 7.5.2: The "Internet Way" of push service over GPRS

In order to use this solution with MSISDN, a push proxy can host a database maintaining the relation between MSISDN and IP address. The push server can then query this database to get the IP address.

One of the advantages of this push method is that there is no limit to the number of messages sent to the user. The disadvantage is, however, that the PDP Context has to be open to receive push messages. This requires a considerable

number of IP addresses allocated. This is not a problem with IPv6 but will set limitations for the usage of IPv4 terminals.

A particularly relevant application for push services for MS is WAPpush as defined in WAP1.2. Also, many Internet push applications already exist, e.g. netcaster (http://www.netscape.com/newsref/pr/newsrelease407.html)

7.6 SIP Push Service

SIP Push operates as an application that uses SIP to establish and manage a session between the client (push initiator) and server (UE). The SIP Invite may be delivered using an active PDP context or the Invite may trigger a network initiated PDP context. When the UE receives the Invite, it may establish a separate PDP context dedicated for the session's data path or allow reuse of an existing PDP context for the data path.

Session Description Protocol (SDP) is one possible content type that can be used with certain SIP messages for Push service. In general, a unique content type may be used for session presentation. However, this document does not specify the content type or the application protocol.

7.6.1 IM Subsystem Scenario

This scenario applies to an R00 network with an IM Subsystem. Rather than define the specific CSCF names (e.g. I-CSCF, P-CSCF) and providing detailed interactions between them, the diagram and scenario steps are kept at a more abstract level. For details on UMTS CSCF registration and various CSCF types, see TS 23.228.

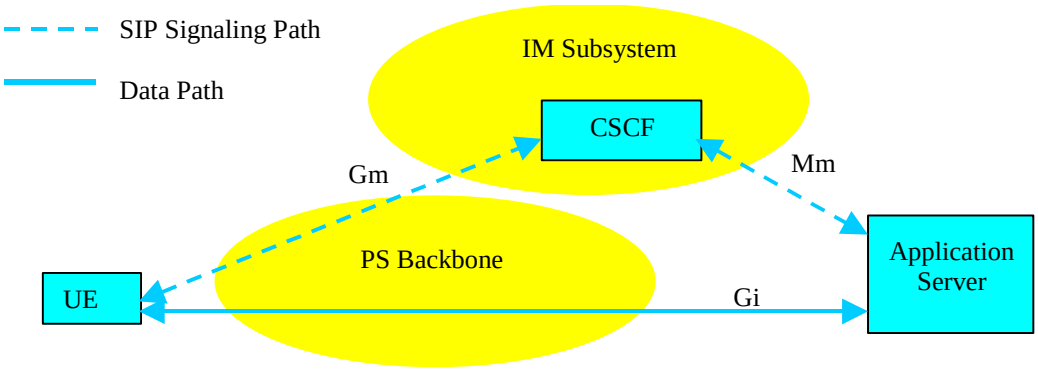

Figure 7.6.1: SIP Push via IM Subsystem

The SIP Signaling Path is a standard bearer PDP context used to transfer SIP control messages (e.g. SIP Register, SIP Invite). The Data Path could use the same PDP context or it could be a separate PDP context established on demand to support the SIP Push data.

In this scenario, the following steps are required:

1) When a UE attaches to the network, it will establish a PDP context to be used for SIP registration and signaling. The IP address allocated for this PDP context could be assigned as a dynamic or static IP address. If it is a dynamic address, SIP registration and the expiration time must be managed as described in section 2.6.2.

2) The UE will register with the CSCF using the SIP Signaling PDP context. The CSCF records the UE's SIP identity (e.g. user@domain) and IP address (provided with the registration). The SIP registration is the same registration (and identity) that is used to register for other IM services (as identified in TS 23.228). As part of the SIP Register message, the UE identifies that it supports Push Services (via the SIP SUPPORTED extension). The SIP Register message may include multiple services supported via the same identity (e.g. VoIP and Push).

3) If the UE is not in his SIP home network (network containing UE's SIP registrar) when it registers, his current contact identity will be registered with his home identity through the via mechanism.

4) When the Application Server is ready to initiate a push to the user, it does so by sending a SIP Invite to the user's SIP identity. The SIP identity known to the Application Server will generally be based on the user's home identity. The SIP Invite will be redirected or forwarded to the SIP Proxy with the same domain name that is in the current contact identity. This would be the CSCF identified in step 1 above.

5) When the CSCF receives the SIP Invite for the UE, it checks to see if it has a valid registration for this identity (i.e. if the UE's registration has not expired). The CSCF can also filter the Invite and reject it if this application/server is not supported on this UE. If there is a valid registration, the CSCF relays the Invite to the UE using the IP address associated with the UE's registration. When the IP packet containing the SIP Invite is received by the GGSN associated with the UE's IP address, the GGSN sends the packet over the associated GTP tunnel for this IP address.

6) If the UE accepts the Invite, it may reuse the existing PDP context or establish a separate PDP context to be used for Data Path traffic. The IP address assigned to the UE for the PDP context would be provided to the Application Server as part of the response to the SIP Invite.

7) Since SIP is a session protocol, the Application Server is granted use of the IP address for Data Path traffic as long as the SIP session is active.

7.6.2 No IM Subsystem Scenario

When the serving network does not include an IM Subsystem, an optional interim SIP solution may be provided. Whether this solution is available or not is dependent on the capabilities of the operator's network and the UE. The interim solution relies on maintaining a long-lived PDP context with the SIP Proxy. This solution may be used in an IP version 4 or version 6 environment. To an Application Server, there is no difference between this scenario and the IM Scenario.

When there is no IM, the UE can connect to a SIP Proxy via a provisioned APN. If a long-lived PDP context to the SIP Proxy will be used, the APN could provide access to:

- A private network within this operator's network (i.e. the SIP Proxy is a locally provided service);
- A private network outside of the operator's network (e.g. a private third party network); or
- The Internet where users can reach a globally available SIP Proxy service.

In order for the UE to be reachable by the Application Server, the PDP context must be active. By using the preservation procedures described in TS 23.060, it is possible for the RABs to be released while maintaining an active PDP context. Since the PDP contexts are not modified in the Core Network, the RABs can then be re-established at a later stage. For the SIP Push Service, as long as the UE is attached to the network, it can activate a PDP context, register with the SIP Server and maintain the active PDP context to receive Push messages from the Application Server. Therefore it is assumed that the PDP context used for SIP signaling is long-lived.

The figure below shows a SIP Proxy in a generic IP Network. The IP Network could be within the operator's network or it could be an external Internet. This figure shows the simplest case. It does not include roaming or use of different GGSNs for the SIP and Data paths.

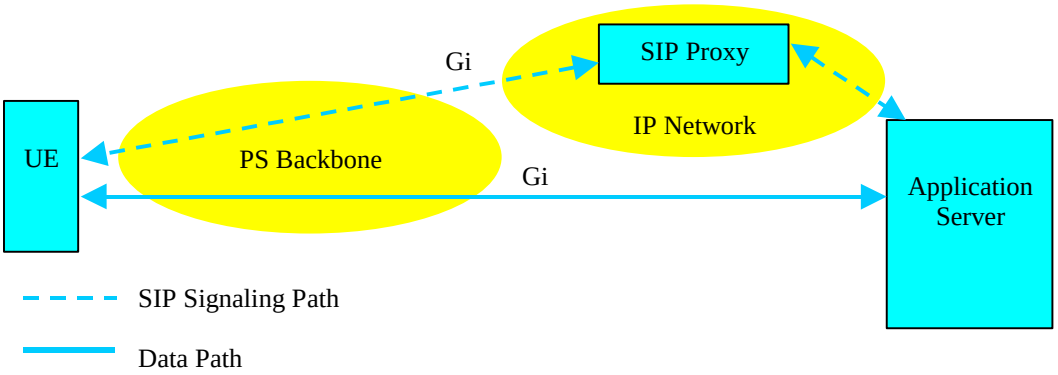

Figure 7.6.2: SIP Push via Long-lived PDP Context

The SIP Signaling Path is a standard bearer PDP context used to transfer SIP control messages (e.g. SIP Register, SIP Invite). The Data Path could use the same PDP context or it could be a separate PDP context established on demand to support the SIP Push data.

In this scenario, the steps for SIP service are the same as the previous scenario with the following differences:

- The SIP Proxy takes the role of the CSCF.
- The UE's SIP registration is sent to the SIP multicast address. If needed, the UE will also register its mobile contact identity with its primary home contact (i.e. if that home is not in this mobile network).

7.6.3 Roaming

7.6.3.1 IM Roaming

Roaming for SIP Push via the IM Subsystem follows the standard being developed for IM.

7.6.3.2 Roaming with SIP Proxy in Home Network

This case applies when the APN defined in the UE for SIP Push service is available in the home network but not in the visited network.

In general, the UE is responsible for registering with the SIP Proxy and maintaining an active registration. The first step for registering is establishing a PDP context for the provisioned APN.

If the subscriber has roamed and the APN takes the subscriber to his home network, the PDP context will be established from the visited SGSN to the home GGSN. In this case, the IP address assigned to the UE for the SIP path PDP context will be from the home network. This context could be used only for SIP session management or it could also be used for any Data context created based on a SIP Invite.

The UE could choose to use a separate PDP context provided in the visited network for the Data context. This is shown in the figure below.

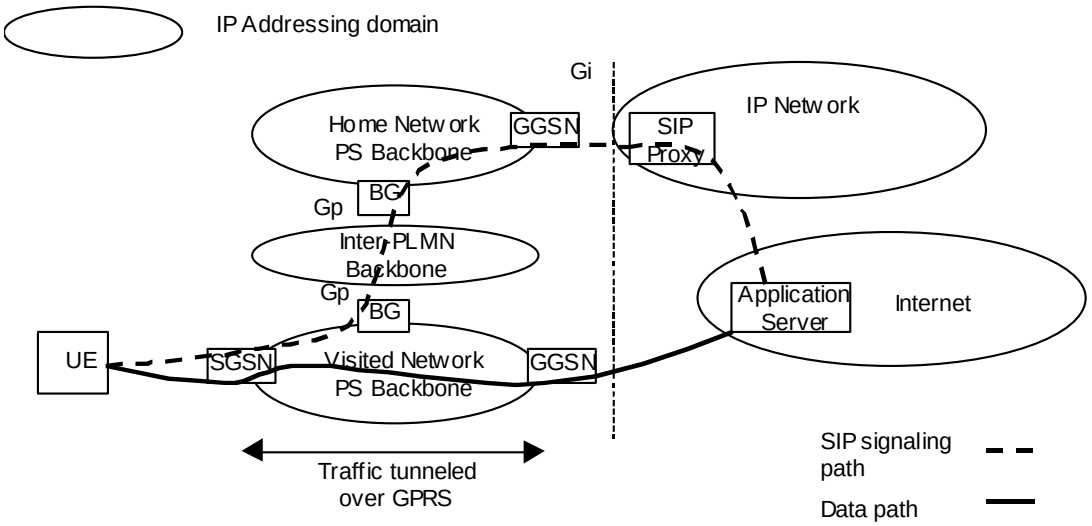

Figure 7.6.3: Roaming – SIP Proxy via HPLMN

.

7.6.3.3 Roaming with SIP Proxy in Visited Network

If the subscriber has roamed and the APN is available in the visited network, the PDP context will be established locally. Since the UE is only visiting in this network, any PDP context activated in this network is given a dynamic IP address from the visited network.

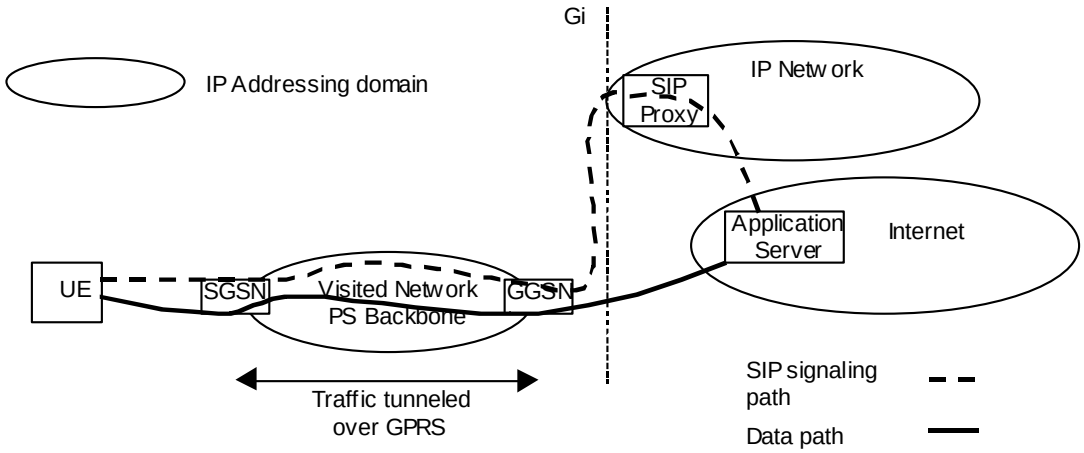

Figure 7.6.4: Roaming – SIP Proxy via VPLMN

Again, the PDP context established for the SIP Proxy connection can be reused for the Data Path if the UE chooses to do so.

7.6.4 Protocol Architecture

In each scenario, the protocol stack architecture is the same.

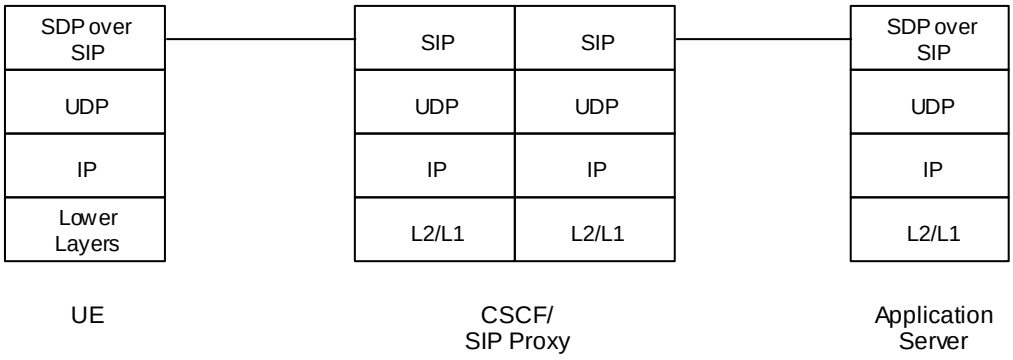

Figure 7.6.5: SIP Session Management Protocols

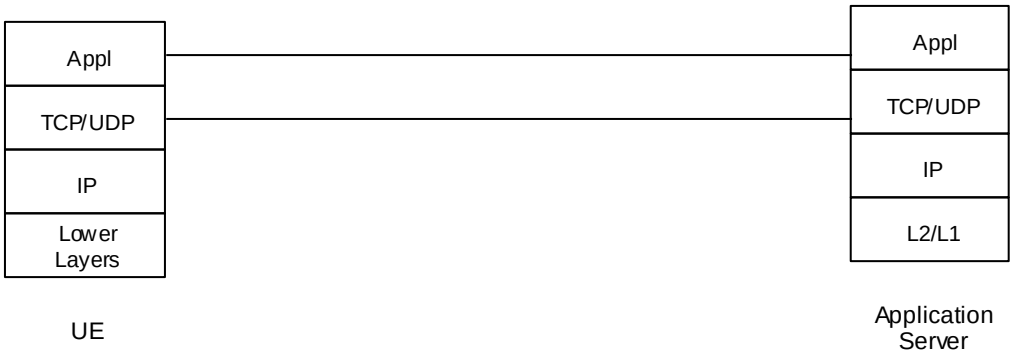

Figure 7.6.6: Data Protocols

7.6.5 Addressing

With the end-to-end SIP approach there are two forms of addressing involved: SIP identity (e.g. user@domain) and IP address.

7.6.5.1 SIP Identity

The SIP identity must be known to the UE (i.e. provisioned). The UE will provide its identity to the SIP Proxy or CSCF when it registers.

The Application Server may know the specific SIP identity for this UE or it may know the user by a different SIP identity that is relayed or redirected to the UE. This characteristic of SIP allows a user to SIP register from any location (e.g. mobile phone, or PC on the internet) and receive notifications at his current location independent of the device and network.

The Application Server may receive the user's SIP identity or the UE SIP identity from multiple sources including from the UE via a previous session initiated by the UE.

7.6.5.2 IP Address

The Application Server will generally query the DNS using the domain name from the user's SIP identity to get a public IP address that can be used to deliver the SIP Invite to the SIP Proxy or CSCF.

The Application Server's IP address will be included in the SIP Invite. The UE will provide its Data Path IP address to the Application Server as part of the response to the SIP Invite.

The SIP Proxy or CSCF will receive the UE's IP address with the SIP Register. The SIP registration will also carry an expiration timeout.

When the UE is using a dynamically assigned IP address for the SIP Proxy registration, the expiration timeout will be based on the lease time for the dynamic address. The UE may also "de-register" with the CSCF or SIP Proxy whenever it no longer wishes to receive SIP push service via the IP address provided. De-registration is accomplished by sending a SIP Register with the expiration time set to 0.

7.6.6 Subscription, Security, and Charging

Network operators will manage subscription and charging for push services. When the CSCF is used, charging for push services will be managed through defined mechanisms. Some tailoring of the charging parameters may be needed to support simple data transfer via this method.

When a non-CSCF SIP Proxy is used that is within the operator's network, the network operator may enhance the Proxy to include a method for collection of charging information for push services.

SIP is designed to support user managed subscription to services. As SIP is deployed, extensions to SIP related services will become widely available. Push services would become just another SIP service where users can either manage their subscription directly or allow network operators to establish basic controls (via SIP) on their behalf.

Since the SIP Invite message is delivered to the end user prior to establishing a session capable of transporting large amounts of data, the end user will also have the ability to refuse any large SIP traffic.

Note: communication of the size of the data to be delivered would be dependent on the application level protocol selected/designed for Push service.

7.6.7 Delivery Reliability

If a user is not accessible (e.g. registration has expired) when the SIP Invite is received at the SIP Proxy, the Application Server will be responsible for retrying later.

There are currently IETF draft proposals (draft-rosenberg-impp-lpidf-00.txt, draft-rosenberg-impp-presence-00.txt, draft-rosenberg-impp-im-00.txt) to include Presence as part of SIP. SIP Presence would allow the Application Server to request a SIP Notify message from the CSCF or SIP Proxy when the user becomes available. The next time the UE

sends the SIP Register message to the CSCF or SIP Proxy, the Application Server would receive a SIP Notify to let it know that the user is now available to receive the SIP Invite.

7.6.8 Connectionless Push

As an option, a SIP Notify can be delivered in place of the SIP Invite. The SIP Notify message can carry peer-to-peer data. The Application Server could deliver the entire push message inside of a single SIP Notify when the message is small enough to fit in the Notify message body. The Notify message is part of the new Presence IETF drafts (draft-rosenberg-impp-lpidf-00.txt, draft-rosenberg-impp-presence-00.txt, draft-rosenberg-impp-im-00.txt).

7.6.9 Quality of Service

QoS requirements can be included in the application portion (i.e. message body) of the SIP Invite from the Application Server. The UE will be responsible for establishing the Data Path PDP context using the supplied QoS.

8 Conclusion and Recommendations

[Editor's note: Chapter to be completed. See also Annex A.]

Annex A (Informative): Comparison of the Push Techniques comparison

[Editors note: This section is going to be moved to the main part of the body when agreed upon.]

	Pros	Cons
SMS based push	- SMS deliverable over CS or GPRS - No need to be PS attached (less radio signalling e.g. periodic updates, and SGSN capacity needed) - No need for having an active PDP Context (GGSN capacity saved) - Possible during a call - Immediate delivery at switch-on - Reliable - After the push message is received, further information or service can be pulled from the network using standard GPRS or CSD procedures	- A lot of traffic makes a lot of MT SMS (i.e. HLR interrogation) - Supporting 100s of push message per seconds may not be possible - Delays due to signalling - Needs WAP1.2 in the terminal
"The Internet way" Push	- Always connected - Minimum delay, i.e. no extra signalling to deliver the push message - Generic, i.e. not bound to a certain access technology - Scaling, the only bottle neck is the radio	- Always PS attached (radio signaling and SGSN capacity) - Always PDP context active (GGSN capacity) - Requires a considerable amounts of IP addresses
NRCA based on MSISDN for push	No need for all subscriber to be PDP context active (GGSN capacity)	- Always PS attached (radio signalling and SGSN capacity) - A lot of traffic makes a lot of HLR interrogations (signalling is comparable to MT SMS) - Delays due to signalling - Needs a new function: GGSN/NA - Needs standardisation work - Mobile capabilities have to be known by the network (i.e. does the terminal support the service) - Needs to support the "Internet Push" when a context is already active - Needs MS supporting NRCA with MSISDN.

		- Complex

Annex <X>: Change history

Change history							
Date	TSG #	TSG Doc.	CR	Rev	Subject/Comment	Old	New