**3GPP TSG SA WG3 Security — S3#20** **S3-010434**

**16 - 19 October, 2001**

**Sydney, Australia**

**3GPP TSG-SA WG2 Meeting #19** **Tdoc 3GPP S2-01-2423**

**27th – 31st August 2001, Sophia Antipolis**

**Title:** **LS on Security aspects of the 3GPP push service**

**Source:** **TSG SA2**

**To:** **SA 3, CN 1, CN 4**

**CC :** **SA 1**

**Contact Person:**

Name: **Chris Pudney**

E-mail Address: **chris.pudney@vf.vodafone.co.uk**

SA 2 have been discussing the architecture for Push Services and have now received an LS from SA 1 (in S2-01-2062 = S1 -01-0856, attached) containing an early version of their service requirements.

These service requirements include a section on security: unfortunately the LS does not seem to have been copied to SA3.

The latest version of the SA 2's Technical Report 23.974 "Support of Push Service" is also attached (in S2-01-2417). This report identifies several possible architectures, each of which may have different security threats/strengths.

**SA 2 would like SA 3** to provide advice on the security aspects of the different Push architectures which are under consideration. In particular the mechanisms by which a subscriber can (a) avoid being pushed unwanted content and (b) avoid automatic connection by the mobile to Push servers that are not in the (home/subscriber's preferred) operator's domain. This advice is needed by SA 2 to help with the selection of the best architecture(s).

Because CN 1 and CN 4 are responsible for the GPRS (SM and GMM), SMS (PS and CS) and MAP signalling, **SA 2 would like CN 1 and CN 4 to consider ideas for Push security which can then be analysed by SA 3.**

**Example**

One of the possible 3GPP push architectures under consideration is illustrated below for the case that the push proxy is within the home operator's domain.

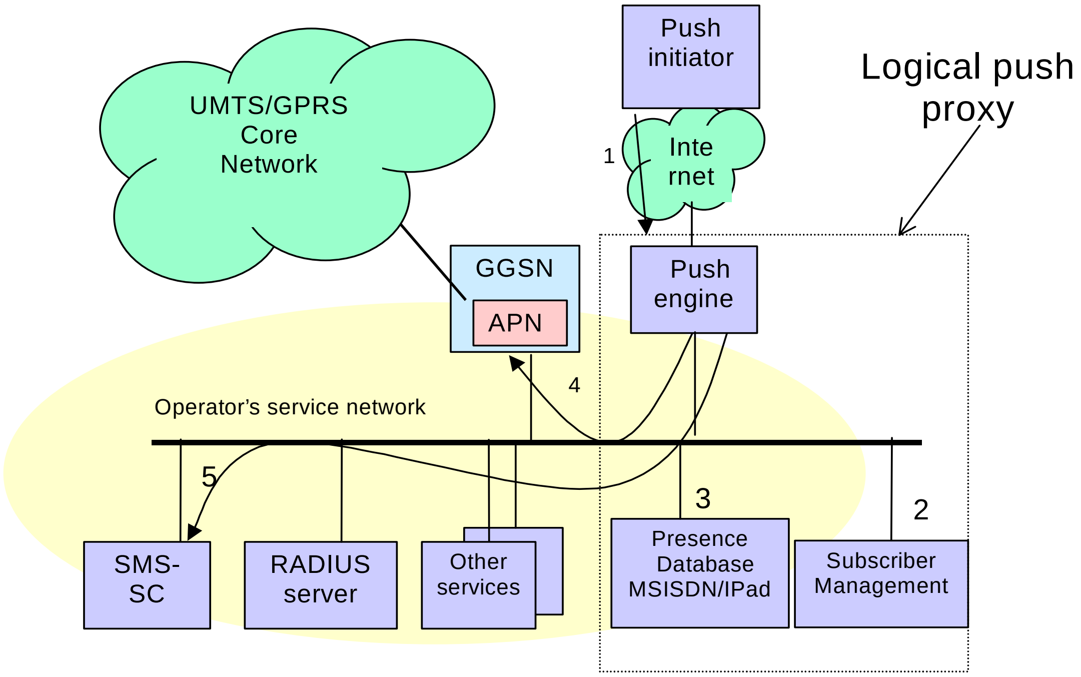


**Figure 1: Delivery of push messages when push proxy is inside operator’s domain**

The push message is delivered through the following steps:

1) The push initiator sends the push message to the operator’s push proxy (PP), e.g. including MSISDN or a list of MSISDN. The link between the push initiator and the push proxy is authenticated.

2) The push proxy checks in the subsciber database if the push message can be sent to the subscriber.

3) If the message is allowed the push proxy checks in the presence database if the destination terminal has an IP address.

4) If an address is returned, the message is sent directly to the terminal through the GGSN

5) If no IP address is returned, the message or an indication to activate the PDP context may be sent via SMS (providing store and forward) depending on the importance of the message and the subscriber security/privacy preferences.

Because the push proxy is in the home operator’s domain, it can be trusted to implement the correct policy according to the subscriber management database. Furthermore, the link between the push initiator and the push proxy can be secured (e.g. using SSL) so that a malicious push initiator cannot masquerade as an authorised push initiator.

## Possible threats

The 3GPP push architectures under consideration also support (or seem to be unable to prevent) the case that the push proxy is outside the home operator’s domain. In this case, trust assumptions cannot be made and there is a danger that, for example, a push SMS could trigger a target user to connect to a malicious third party push proxy that could push unauthorised information towards the target customer.

Threats that may need to be addressed include (but are not limited to):

a) SMS sent through SMSC not in the HPLMN;

b) GGSN in the VPLMN;

c) SMS sent through the SMSC in the HPLMN but with malicious content that causes the mobile to automatically connect to a high cost server.

d) denial of service attacks on the GGSN for the Network Requested Context Activation architecture;

e) disclosure of the IMSI to a third party push proxy;

f) etc

**3GPP TSG-SA WG2 meeting #19** **Tdoc S2-012062**
**Sophia Antipolis, France, 27 – 31 August**

---

TSG-SA WG 1 (Services) meeting #13 TSG S1-010856
Lake Tahoe, USA, 9-13 July 2001 Agenda Item: 6.4

---

|  |  |
|---|---|
| **Title:** | Liaison Statement on Push Service Stage 1 |
| **Source:** | TSG SA1 |
| **To:** | TSG SA2, TSG T2 |
| **cc:** | GSM-A SERG |

**Contact Person:**
**Name:** Nick Sampson, Orange UK
**E-mail Address:** nick.sampson@orange.co.uk
**Tel. Number:** +44 7973 963519

---

### 1. Overall Description:

Following the joint SA1/SA2/T2 ad-hoc meeting on the Push Service (5-6 July 2001), SA1 has discussed further the requirements for the Push Service at SA1#13.

SA1 has produced a draft Stage 1 for the Push Service, which is attached. This stage 1 is an early draft, which will be used by SA1 as a basis for further work. However, SA1 feels it would be useful to get some feedback from SA2 on these requirements.

The attached draft Push Service Stage 1 has been produced from several earlier documents on Push.

### 2. Actions:

TSG SA1 asks SA2 and T2 to review the attached draft stage 1 and provide comments to SA1. In particular, SA1 asks SA2 and T2 to:

1. Detail which of these requirements can be completed in Release 5 and those which are not likely to be completed in Release 5, and
2. Indicate where further clarification of the requirements is needed.

### 3. Date of Next SA1 Meetings:

|  |  |  |
|---|---|---|
| SA1#14 | 5-9 November 2001 | Japan, hosted by NTT DoCoMo |
| SA1#15 | 14-18 Jan 2002, | host offered: AWS Las Vegas |
| SA1#16 | 8-12 Apr 2002, | no host |
| SA1#17 | 8-12 July 2002, | host offered: SBC (North America) |
| SA1#18 | 23-27 Sep 2002, | no host |

### 4. Attachments:

S1-010808 Draft Push Stage 1 TS 22.xxx v0.2.0.

s1-010808.doc

**August 27-31, 2001**
**Sophia Antipolis, France**

**Source**: Rapporteur (NTT Communicationware)

**Title**: TR 23.974Version 0.2.0 (Support of Push Service)

**For**: Approval

*Contact: Nobuyuki UDA
uda.nobuyuki@se.nttcom.co.jp

The attached document, incorporates all agreements in the push drafting meeting in Sophia Antipolis.

The following are proposed to S2:

- Approval of the document as TR23.974 Version 0.2.0 for submission to SA plenary, i.e. as v2.0.0

The following are the changes from TR23.974v0.1.0, incorporating texts from approved contributions: S2-012399, 2400, 2401, 2411, 2412 and 2414.

- Add the conclusion of this TR (S2-012412)
- Replace the chapter 7.2.6 (protocol architecture) with more detailed studies (S2-012399)
- Align SIP based architecture in chapter 7.6 with the agreement of deploying proxy based overall architecture (S2-012401)
- Append SIP MESSAGE Method for push services in chapter 7.6.8 “Connectionless Push” (S2-012411)
- Append “Support of end to end real time multimedia session” into chapter 7.6 (S2-012414)
- editorial modifications based on S2-012400
- Delete the editorial notes

# 3GPP TR 23.974 V0.~~1~~2.0 (2001-~~03~~09)

*Technical Report*

## 3rd Generation Partnership Project;
## Technical Specification Group Services and System Aspects;
## Support of Push service
## (Release 5)

The present document has been developed within the 3rd Generation Partnership Project (3GPP™) and may be further elaborated for the purposes of 3GPP.

The present document has not been subject to any approval process by the 3GPP Organisational Partners and shall not be implemented.
This Specification is provided for future development work within 3GPP only. The Organisational Partners accept no liability for any use of this Specification.
Specifications and reports for implementation of the 3GPP™ system should be obtained via the 3GPP Organisational Partners' Publications Offices.

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

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 ".

[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] 3GPP TS 24.008: "Mobile radio interface layer 3 specification; Core Network Protocols; Stage 3".

[7] 3GPP TS 29.060: "General Packet Radio Service (GPRS); GPRS Tunnelling Protocol (GTP) across the Gn and Gp Interface ".

[8] 3GPP TS 29.061: "Interworking between the Public land Mobile Network (PLMN) supporting Packet Based Services and Packet Data Network (PDN) ".

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

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

[10a] IETF RFC 793: "Transmission Control Protocol"(STD 7).

[11] IETF RFC 821: " Simple Mail Transfer Protocol"(STD 10).

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

[12a] IETF RFC 1631: "The IP Network Address Translator (NAT)".

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

[13a] IETF RFC 2131: "Dynamic Host Configuration Protocol".

[14] IETF RFC 2138: "Remote Authentication Dial In User Service (RADIUS) ".

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

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

[17] IETF RFC 2616: "Hypertext Transfer Protocol – HTTP/1.1".

[18] IEFT RFC 2617: "HTTP Authentication: Basic and Digest Access Authentication".

[19] IETF RFC 2632: "S/MIME Version 3 Certificate Handling".

[20] IETF RFC 2633: "S/MIME Version 3 Message Specification".

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

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

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

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

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

[26] WAP Forum: "WAP Push Architectural Overview"(1999) URL: http://www.wapforum.org/what/technical.htm

[27] WAP Forum: "WAP Push Access Protocol "(1999).

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

[29] W3C Note: "Composite Capability/Preferences Profiles: A user side framework for content negotiation"(1999) URL: http://www.w3.org/TR/1998/NOTE-CCPP-19990727".

# 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:

| | |
|---|---|
| NAT | Network Address Translator |
| NRCA | Network Requested PDP Context Activation |
| OTA | Over The Air delivery protocol |
| PP | Push Proxy |
| 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.

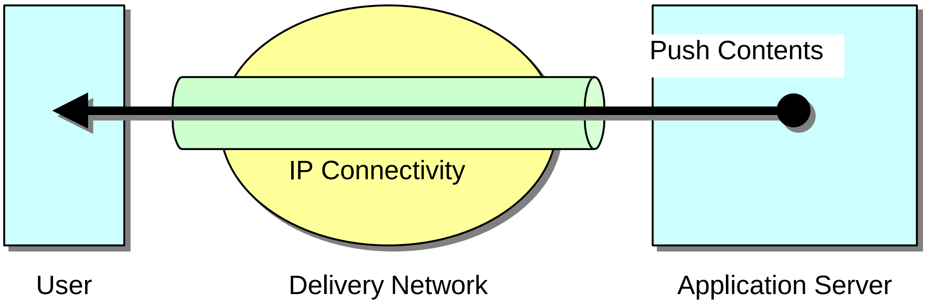


**Figure 6.1: Dedicated Connection Reference Architecture Entities**

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.

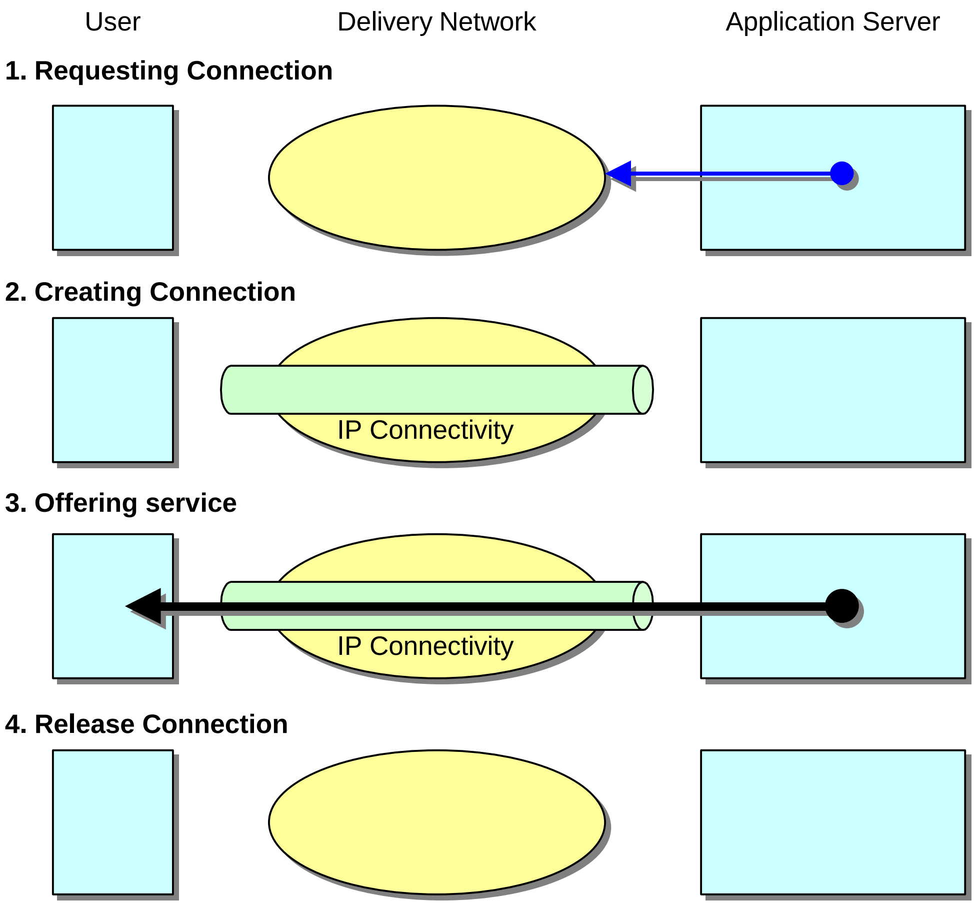


**Figure 6.2: General Service Scenario in Dedicated Connection Approach**

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.

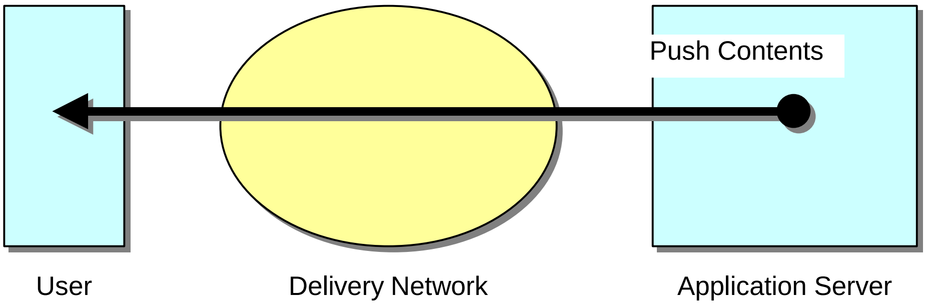


**Figure 6.3: Connectionless Reference Architecture Entities**

Figure 6.4 shows the general service scenario.

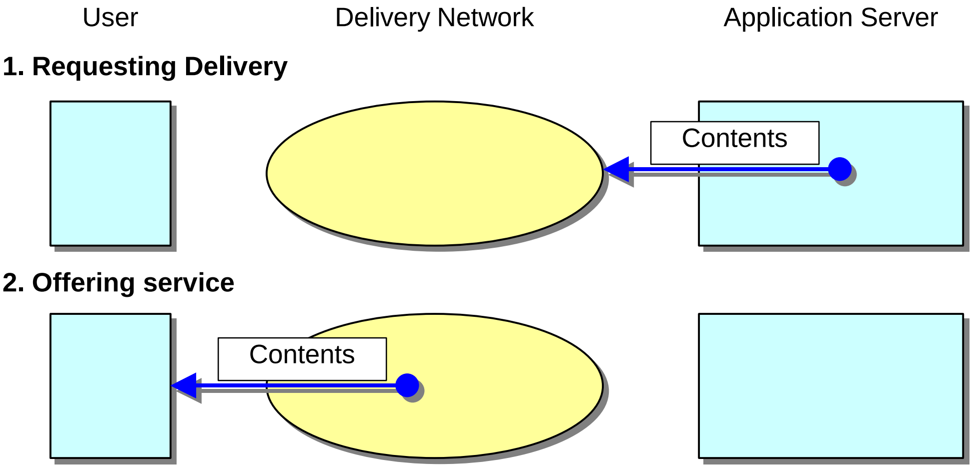


**Figure 6.4: General Service Scenario in Connectionless Approach**

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.

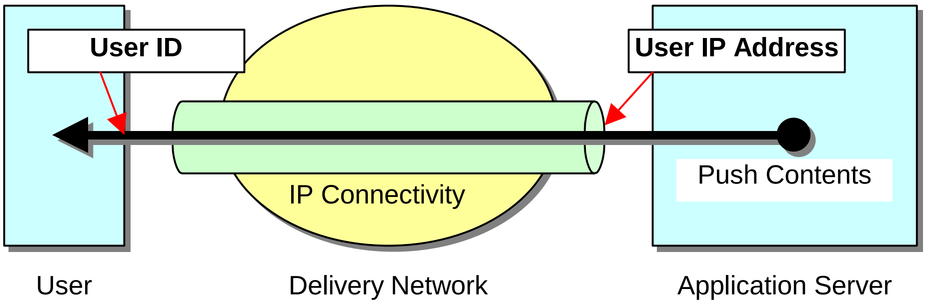


Figure 6.5: User-ID and User IP Address

## 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.

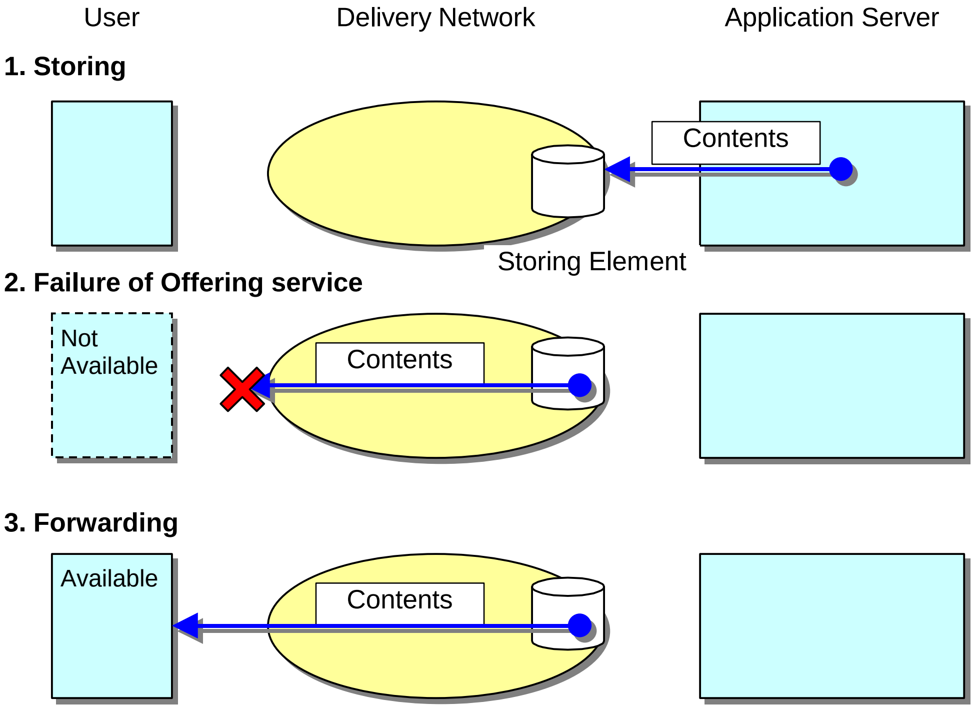


**Figure 6.6: Service Scenario with Store and Forward**

#### 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.

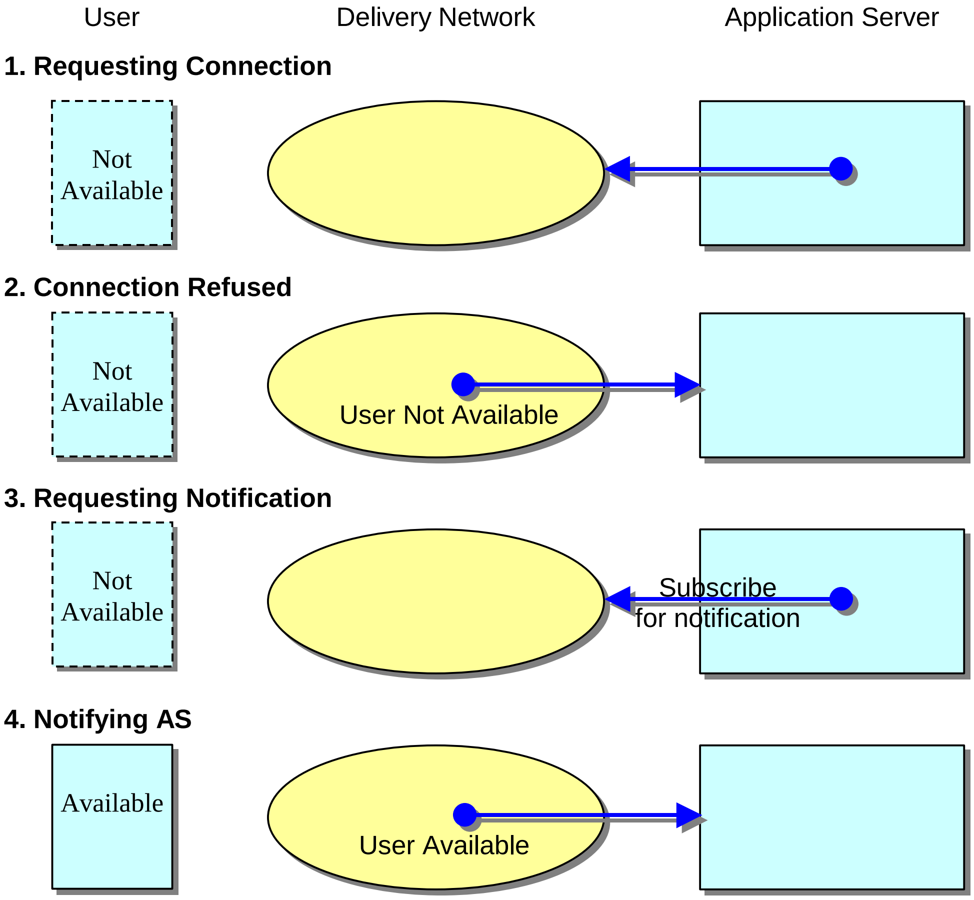


**Figure 6.7: Delivery Network Based Presence Service Scenario**

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.

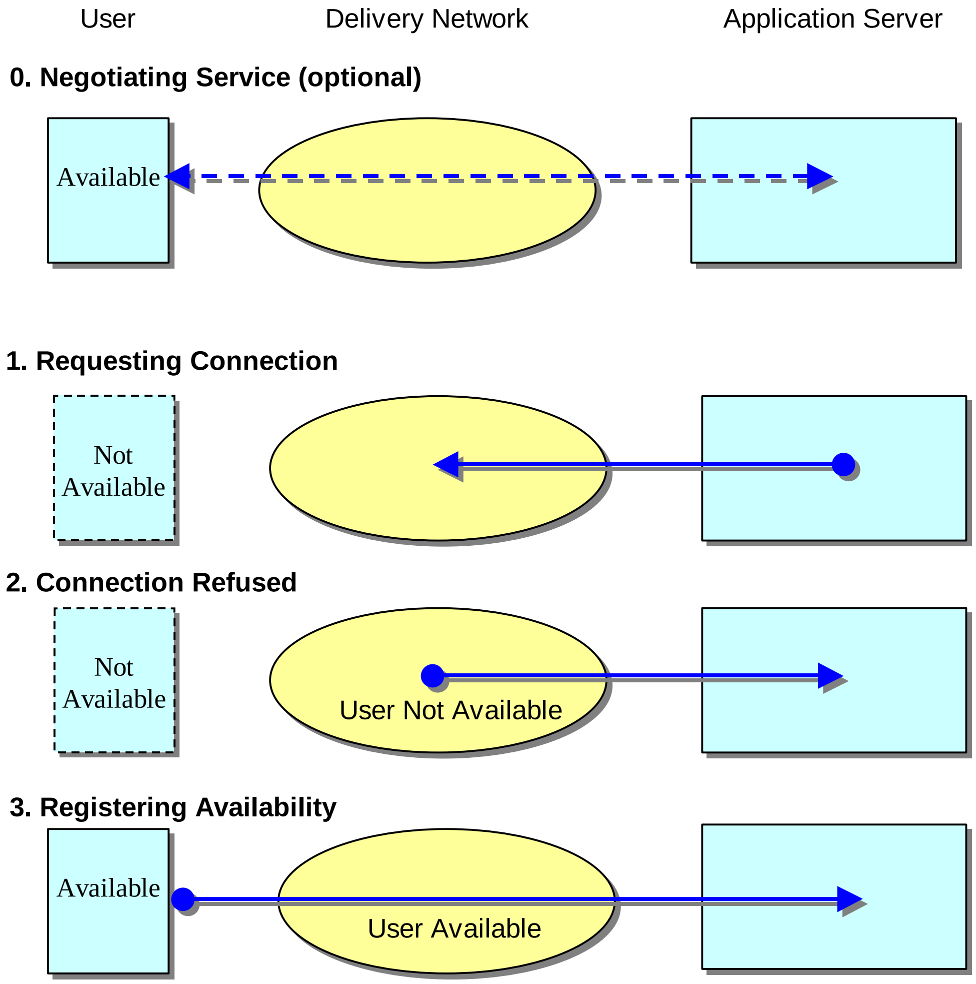


**Figure 6.8: User Based Presence Service Scenario**

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.

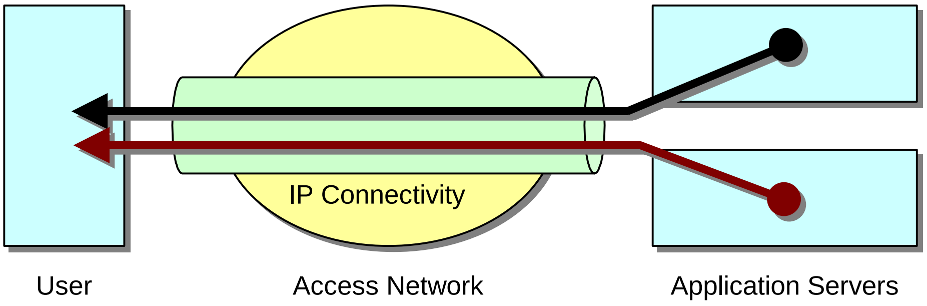


**Figure 6.8: multiple push services over single IP connectivity path**

## 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.

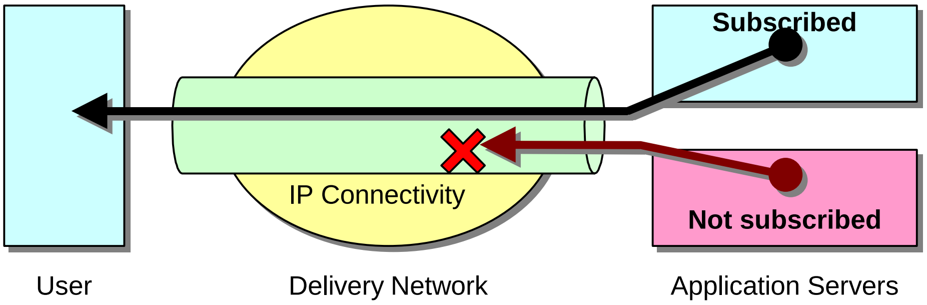


**Figure 6.9: Denial of Service Based on Subscription**

## 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.

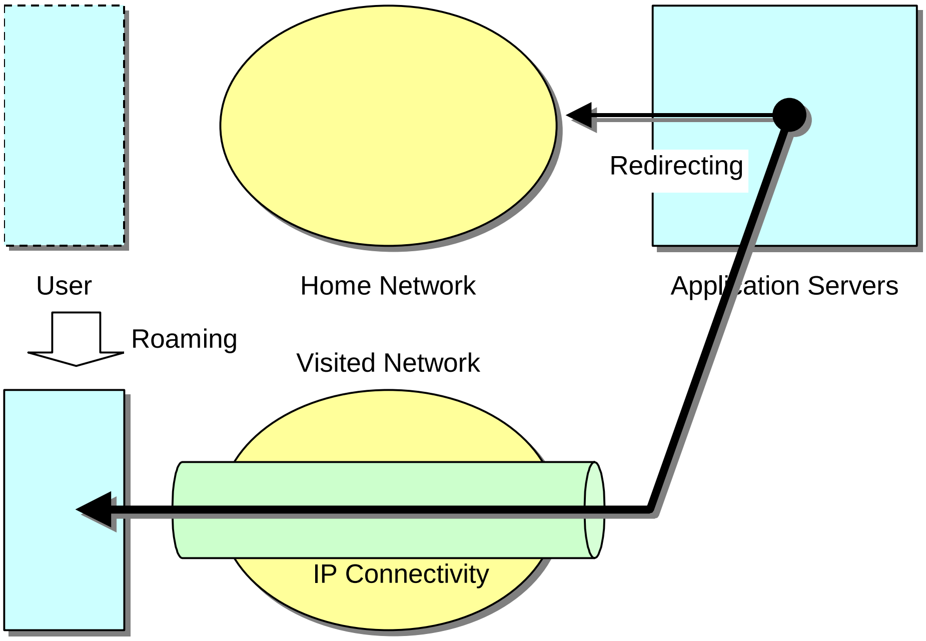


**Figure 6.10: Roaming Support by Redirecting**

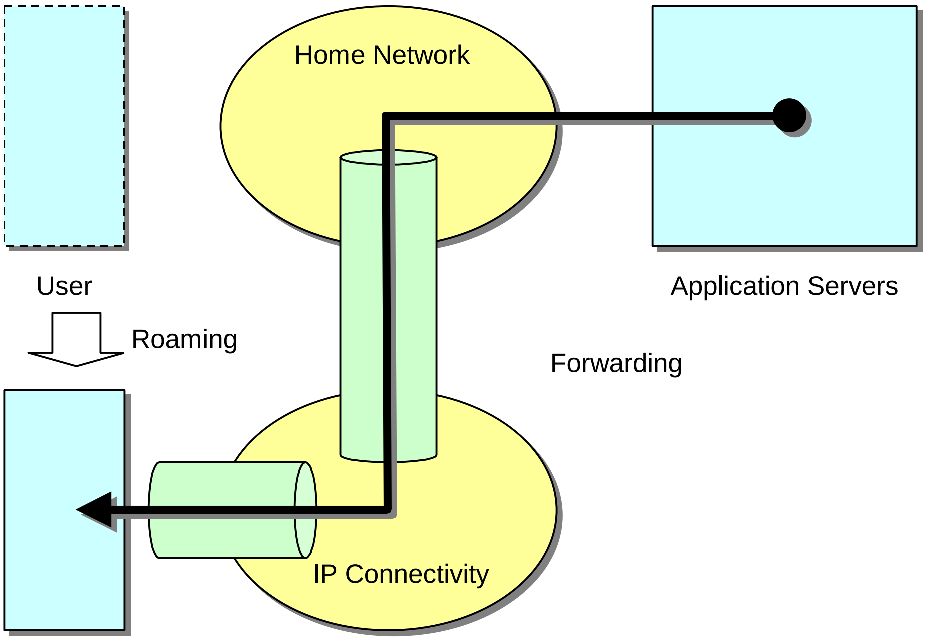


**Figure 6.11: Roaming Support by Forwarding**

# 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 for Push Services

The figure 7.2.1 shows a functional architecture for network requested  PDP context activation with User-ID for Push Services. The User-ID is, for example, MSIDSN or email address (Bell@lucent.com). The blue arrows and the red arrow in the Figure 7.2.1 represents flow of control messages and of push messages, respectively in dedicated connection mode.

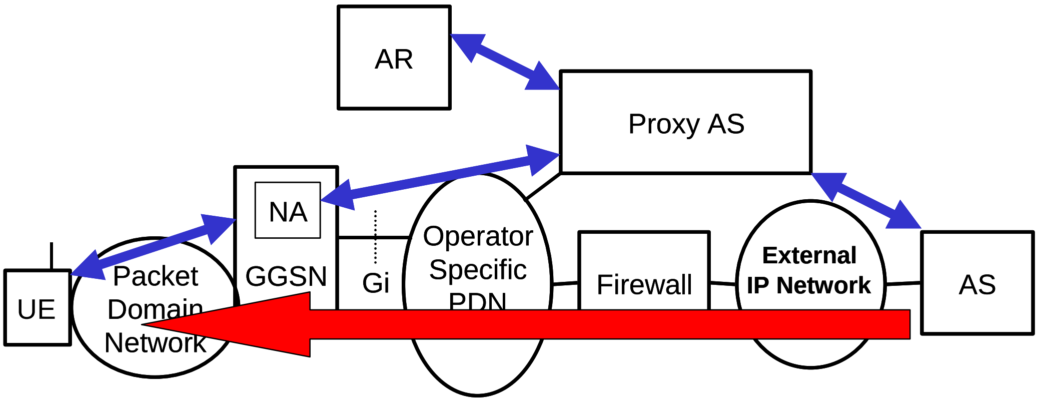


**Figure 7.2.1: Reference Architecture of Network Requested PDP Context Activation for Push Services**

### 7.2.1.1 Application Server (AS)

AS executes push application and resides in operator's network or in the external IP network. When a push message for a recipient identified by User-ID needs to be sent, the AS operating in dedicated connection mode, requests Proxy AS to set up connection to the recipient. When the connection is established by activating PDP context, the Proxy AS provides the AS with the IP address dynamically allocated for the recipient as a response to the request. The AS sends the message to the IP address to deliver it to the recipient. The AS also directs the subsequent push messages for the recipient to the IP address.

The AS operating in connectionless mode, sends a push message containing User-ID to the Proxy AS. To deliver such push message to the recipient identified by the User-ID, the Proxy initiates PDP context activation, obtains a dynamic IP address for the User-ID and sends the message to the IP address. The Proxy directs the subsequent push messages with the User-ID to the IP address.

### 7.2.1.2 Address Resolver (AR)

Address Resolver  is a database which provides the following information for Proxy AS.

- Mapping from User-ID to IMSI
- Provision of subscription information of a push subscriber

The subscription information may be used by the Proxy AS to determine the handing of connection request or a push message. For example, it may discard the request or the message to avoid delivering unwanted messages to the subscriber.

### 7.2.1.3 Proxy AS (Proxy Application Server)

The Proxy AS resides in a operator's network and performs interface function between GPRS packet domain and the AS. When the Proxy receives a request or a push message from the AS, it performs the following procedures.

1) Decide how to handle a connection request or a push message based on subscription information of its recipient identified by User-ID. The subscription information is obtained from the AR.

2) Obtain IMSI associated with User-ID by contacting the AR.

3) Determine if PDP context activation is required by checking the internal cache.

4) Trigger PDP context activation by sending Notification Request to GGSN if required.

5) Update the cache according to the result of the step 4.

6) Provide the IP address assigned to the IMSI for the AS in the dedicated connection mode.

7) Relay a push message to the IP address if the Proxy AS is in the connectionless operation.

To share activated PDP context among the application servers, the Proxy AS maintains the internal cache whose record contains User-ID, dynamic IP address allocated to the User-ID, IP address of GGSN holding active PDP context and a list of application services that are currently using the IP address to deliver messages to the User-ID. How the cache is managed is outlined below in the dedicated connection mode. Please note that the sharing mechanism is for push services requiring best effort delivery and that sharing is optional. (The same procedures apply in the connectionless mode as well.)

When a connection request for User-ID is received by the Proxy for the first time, it initiates PDP context activation and creates a cache record that holds the User-ID, the dynamic IP address allocated for the User-ID in the context activation process, the IP address of GGSN that holds the active PDP context and a list contains the AS which sent the connection request to the Proxy.

Suppose now that another connection request for the same User-ID is received from a different AS (2nd AS), the Proxy checks the cache, finds the record with the User-ID and determines to share the available active PDP context with the 2nd AS. The Proxy adds the 2nd AS in the list of application servers in the cache record and returns connection response containing the IP address to the 2nd AS.

Suppose further that the UE deactivates the PDP context, the deactivation is notified to the Proxy AS and the Proxy then informs the Application Servers in the list contained in the cache record, namely 1st AS and the 2nd AS. The two Application Servers stop using the IP address and mark the address invalid, when notified.

The advantage of having Proxy AS in functional architecture is:

1) Support of interworking with different types of Application Servers.
Dedicated connection or connectionless type of application servers are supported.

2) Provision of security measures
The Proxy can share security association with application servers to allow only messages from authenticated sources to be exchanged through the firewall. This protects the network from denial of services attacks. The Proxy also accesses the subscription information of recipients to prevent unwanted message from being delivered.

3) Sharing of available PDP context information
As explained above, by maintaining cache of active PDP context information, the application servers can share active PDP context to deliver push messages. This reduces the signalling within GPRS network for Push Services. To obtain sharing of a PDP context, the context must be created with APNs that are registered in the table <APN, Proxy_AS> defined in clause 7.2.1.4

4) Dynamic GGSN selection
It is possible for the Proxy AS to provide dynamic GGSN selection which may be based on a local load sharing policy.

### 7.2.1.4 Notification Agent (NA)

The NA in GGSN initiates PDP context activation when Notification Request is received from the Proxy AS. The NA maintains a table of record <APN, Proxy AS>. When the NA is informed of the creation of a new PDP context by the GGSN, the NA searches the table to find a record whose APN is equal to the APN of the created context. If the record is found, the NA sends Update message to the Proxy AS specified in the record. For example, if the table contains <Push, Proxy-Push.abc.com> and a PDP context is created with APN Network Identifier of “Push”, Update message is sent to Proxy.abc.com. ~~Note that only 2 new messages are introduced to support network requested PDP context activation with User-ID for Push Services.~~

#### 7.2.1.5 User Equipments (UE)

In principle, release of the PDP context activated by network should be performed by the UE, because the context may by shared by the other applications on the UE. It may be required to provide a means for coordinating PDP context activation and deactivation among the applications on the UE. Charging aspects related to how to share the cost of the session between the subscriber and the AS are beyond the scope of this document (but it is expected that some form of sharing would be fair if the UE keeps a session active beyond the time a push server is using it).

When the PDP context is deactivated by the UE, the deactivation is notified to the Proxy AS. In the dedicated connection mode, the Proxy also notifies the AS that the IP address for the UE shall be released. The AS stops using the IP address and deletes it on receiving the notification.

#### 7.2.1.6 GPRS network

GRPS network may release a PDP context of the UE for which the radio connection becomes broken, then NA in the GGSN notifies the deactivation to the Proxy AS. The Proxy also informs the AS that the PDP address for the UE shall be released in dedicated connection mode. The AS stops using the IP address and deletes it on receiving the notification.

### 7.2.2 PDP Context Activation with User-ID

In this section, two flow examples are provided to show how PDP context activation is performed with two new messages, Notification Request and Update. The dotted line in the Figures represents a new message or a message which is out of the scope of this contribution.

#### 7.2.2.1 Information Flow Example 1 : Dedicated Connection Approach

The figure 7.2.2 shows network requested PDP context activation in dedicated connection approach.

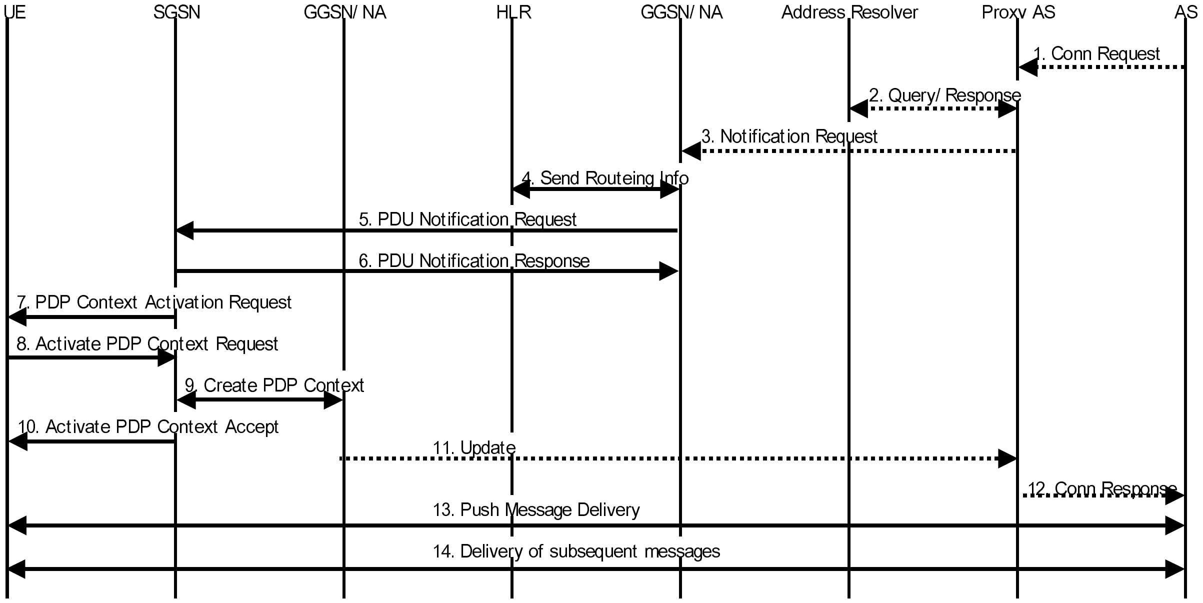

UE
SGSN
GGSN/ NA
HLR
GGSN/ NA
Address Resolver
Proxy AS
AS
1. Conn Request
2. Query/ Response
3. Notification Request
4. Send Routeing Info
5. PDU Notification Request
6. PDU Notification Response
7. PDP Context Activation Request
8. Activate PDP Context Request
9. Create PDP Context
10. Activate PDP Context Accept
11. Update
12. Conn Response
13. Push Message Delivery
14. Delivery of subsequent messages

**Figure 7.2.2: Network Requested PDP Context Activation in dedicated connection approach**

1) The AS sends Connection Request message with User-ID to the Proxy AS. The message is out of the scope of this contribution.

2) The Proxy AS contacts the AR to obtain the IMSI and the subscription information associated with the User-ID. The query and response are out of the scope of the contribution. If the Proxy AS decides to process the request, it checks if an IP address has been allocated to the User-ID by searching the internal cache. In this scenario, there is no IP address is assigned, the Proxy AS, therefore, triggers PDP context activation.

3) The Proxy AS selects one GGSN/NA and sends Notification Request to it. The request may contain IMSI and APN. The criteria for the selection of GGSN may be based on a local load sharing policy.

4) The GGSN obtains routeing information for the UE by issuing Send Routeing Information request to the HLR. The HLR returns the address of SGSN to which the UE is currently attached.

5) The GGSN sends PDU Notification Request to the SGSN identified in procedure 4). Extension to 3GPP TS29.060[6] is required to allow the PDP address in the PDU Notification Request to be set to null to indicate the UE to request dynamic IP address allocation.

6) The GGSN receives a successful PDU Notification Response from the SGSN.

7) The SGSN sends Request PDP Context Activation to UE. The PDP Address is set to null.

8) The UE sends PDP Context Activation Request to the SGSN. The UE requests the assignment of a dynamic IP address by setting the PDP Address to null.

9) The SGSN selects a GGSN and creates PDP context. An IP address is allocated to the UE by the GGSN. The criteria for the selection of GGSN by SGSN, is according to Annex A: APN and GGSN selection in 3GPP TS 23.060[4].

10) On successful creation of the PDP context, the SGSN sends Activate PDP Context Accept to the UE with the assigned IP address.

11) The NA on the GGSN searches the table of <APN, Proxy_AS> for a record whose APN is equal to the APN of the created PDP context. If found, it sends Update message to the Proxy AS specified in the record. The message may contain IMSI and the allocated IP address. The Proxy AS creates the cache record of User-ID, the allocated IP address, the IP address of GGSN/NA and the list containing the IP address of the AS.

12) The Proxy AS sends Connection Response with the User-ID and the allocated IP address to the AS. The message is out of the scope of this contribution.

13) The AS uses the IP address to send the message to the UE. The Proxy AS may not be on the path of dedicated connection from the AS to the UE.

14) The AS can also use the IP address to deliver multiple push messages arrived subsequently for the User-ID.

### 7.2.2.2 Information Flow Example 2: Connectionless Approach

The figure 7.2.3 shows network requested PDP context activation in connectionless approach.

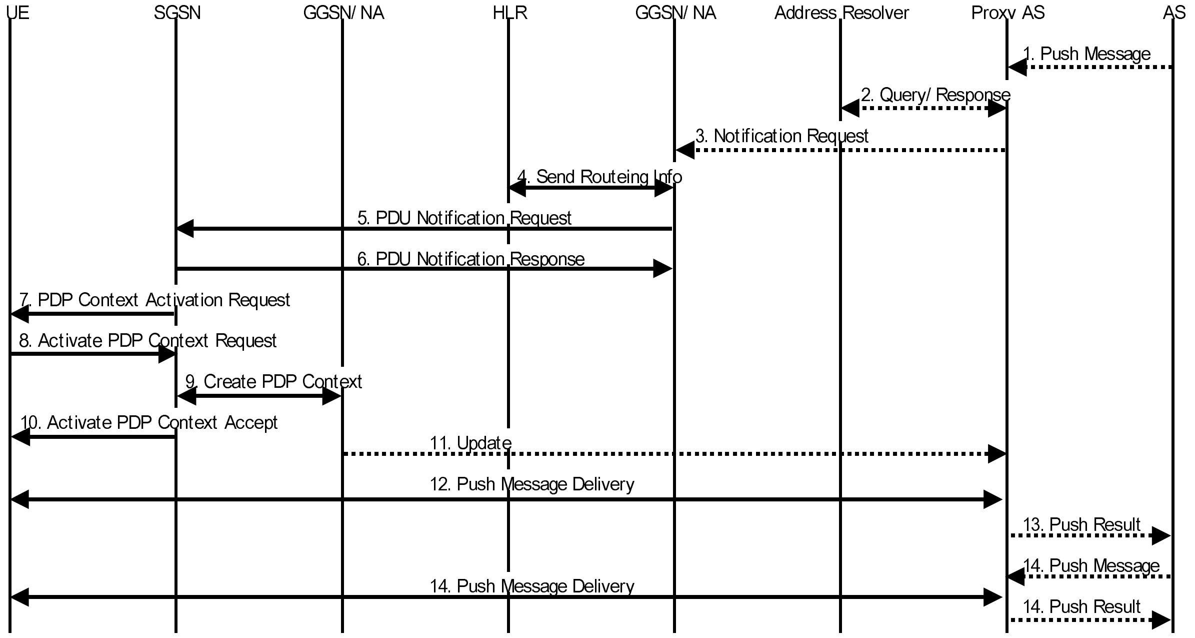


**Figure 7.2.3: Network Requested PDP Context Activation in Connectionless approach**

1) The AS sends a push message with User-ID to the Proxy AS. The message is out of the scope of this contribution.

2) through 11) Same as in the clause 7.2.2.1.

12) The Proxy AS uses the IP address to send the message to the UE.

13) The Proxy AS sends Push Result to the AS. The message is out of the scope of this contribution.

14) The Proxy AS uses the IP address in the cache to deliver the subsequent push messages for the User-ID.

#### 7.2.2.3 Selection of GGSNs in network requested PDP context activation process

As shown in the Figure 7.2.2 and 7.2.3, there are 2 GGSN/NAs involved in PDP context activation. The Proxy AS selects the first GGSN/NA to which Notification Request is sent. Any GGSN/NA in home network can be selected. The selection may be based on a local load sharing policy.

When SGSN receives Activate PDP Context Request, SGSN selects a GGSN to create PDP context. The selected GGSN (the second GGSN) allocates an IP address and the NA on the GGSN sends Update message containing the IP address to the Proxy AS.

The selection of the second GGSN by SGSN should be according to the ANNEX A in 3GPP TS23.060[4]. An UE activating a PDP context with a dynamic address can be connected to any GGSN supporting a given APN. The GGSN may be selected using a round robin mechanism. Therefore, it is expected that the Proxy AS receives Update message from a GGSN/NA which is different from the one to which Notification Request was sent. The NA on the second GGSN uses the table of <APN, Proxy AS> to route Update messages to the Proxy AS that requested the PDP context activation. The NA searches the table for a record whose APN is equal to the APN of created PDP context. If found, it routes the Update message to the Proxy AS specified in the record.

When it is desired that the first and second GGSN/NA is identical, it can be achieved by two approaches. In the first approach, Proxy_AS and APN are pre-configured such a way that a single GGSN/NA can be selected by both the Proxies and SGSN when PDP context is activated. The configuration sacrifices load balancing capability since GGSN is fixed. In the second approach, PDU Notification Request is extended to carry the IP address of the first GGSN so that SGSN can select the first one when creating PDP context.

#### 7.2.2.4 Roaming Support

The proposed information flow supports roaming services. Assuming the roaming agreement between home and visited network operators, the first GGSN sends PDU Notification Request to the SGSN in the visited network when the UE is roaming. When the second GGSN selection by the ANNEX A in 3GPP TS23.060[4] is applied, the second GGSN could be home or visited depending on the APN in use. A SGSN shall select GGSNs in the home network, because there is currently no standardized means to inform the IP address of the Proxy AS to the GGSN/NA in visited network.

#### 7.2.2.5 How to protect HLR from signalling overload

The problem of using Network Requested Context Activation is that the HLR is heavily involved (when finding the SGSN address). Therefore every GGSN/NA should have a mechanism controlling the load of requests sent to the HLR (or a protocol converting GSN).

### 7.2.3 PDP Context Deactivation requested by the UE

The Figure 7.2.4, shows the information flow where PDP context deactivation is requested by the UE to deallocate dynamic IP address. In principle, release of the PDP context activated by network should be performed by the UE, because the context may by shared by the other applications on the UE. It may be required to provide a means for coordinating PDP context activation and deactivation among the applications on the UE.

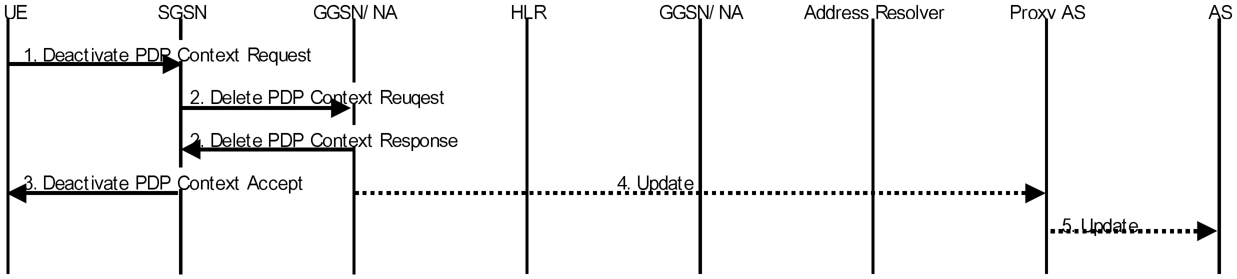

UE
SGSN
GGSN/ NA
HLR
GGSN/ NA
Address Resolver
Proxy AS
AS
1. Deactivate PDP Context Request
2. Delete PDP Context Reuqest
2. Delete PDP Context Response
3. Deactivate PDP Context Accept
4. Update
5. Update

**Figure 7.2.4: PDP Context Deactivation by the UE.**

1) The UE requests PDP context deactivation to the SGSN.

2) The SGSN deletes the PDP context.

3) The SGSN returns Deactivate PDP Context Accept to the UE.

4) The NA on the GGSN searches the table of <APN, Proxy_AS> for a record whose APN is equal to the APN of the deallocated context. If found, the NA sends Update message to the Proxy AS specified in the record to inform that the IP address is no longer valid.

5) The Proxy AS sends Update message to the Application Servers in the list recorded in the cache, and it deletes the cache record. On receiving the Update, the servers stop using the address and delete it.

## 7.2.4 PDP Context Deactivation requested by GPRS network

The information flows in this section show the PDP context deactivation initiated by GPRS network. When the PDP context is deactivated, the GGSN/NA that held the context informs the proxy AS of the deactivation by sending Update message.

### 7.2.4.1 The UE Goes Out of Coverage.

The Figure 7.2.5 shows information flow when the UE goes out of coverage and the SGSN initiates detach procedure after mobile reachable timer expires. In such a case, Delete PDP context request is sent to the GGSN.

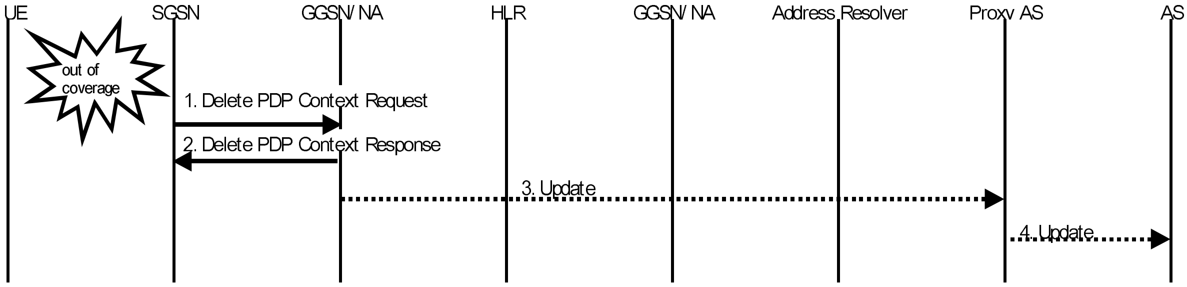


**Figure 7.2.5: PDP Context Deactivation due to Detach Operation**

1) The GGSN receives Delete PDP Context Request.

2) It deletes the PDP context and sends Delete PDP Context Response to the SGSN.

3) The NA in the GGSN searches the table of <APN, Proxy_AS> for a record whose APN is equal to the APN of the deallocated context. If found, the NA sends Update message to the Proxy AS specified in the record to inform that the IP address in no longer valid.

4) If the Proxy is operating in the dedicated connection mode, it informs the application servers that are listed in the cache record containing the IP address that the IP address is no longer valid by sending Update messages to them

and it deletes the cache record. On reception of the Update message, the application servers stop sending push messages and delete the IP address information.

### 7.2.4.2 GGSN initiated PDP context deactivation

The Figure 7.2.6 shows information flow when GGSN initiates PDP context deactivation. GGSN may decide to deactivate PDP context for example, when no traffic to and from the UE is detected within pre-configured time.

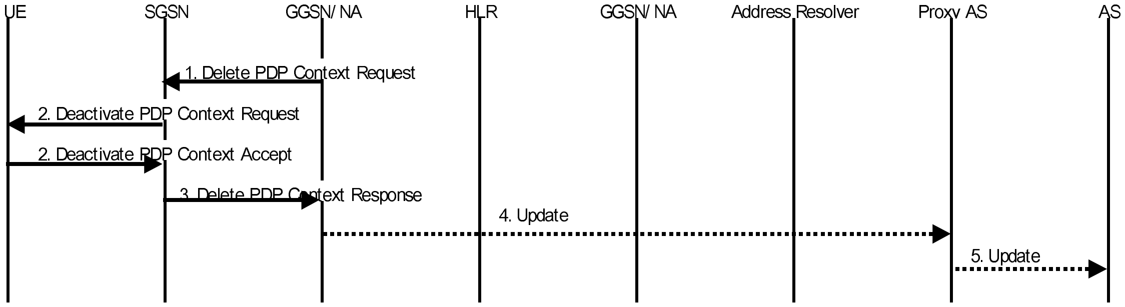

UE
SGSN
GGSN/ NA
HLR
GGSN/ NA
Address Resolver
Proxy AS
AS
1. Delete PDP Context Request
2. Deactivate PDP Context Request
2. Deactivate PDP Context Accept
3. Delete PDP Context Response
4. Update
5. Update

**Figure 7.2.6: GGSN initiated PDP Context Deactivation reported to the Proxy AS**

1) The GGSN sends Delete PDP Context Request to the SGSN.

2) The SGSN sends Deactivate PDP Context Request to the UE and UE accepts.

3) The SGSN sends Delete PDP Context Response back the GGSN.

4) The NA in the GGSN searches the table of <APN, Proxy_AS> for a record whose APN is equal to the APN of the deallocated context. If found, the NA sends Update message to the Proxy AS specified in the record to inform that the IP address is no longer valid.

5) If the Proxy is operating in the dedicated connection mode, it informs the application servers that are listed in the cache record containing the IP address that the IP address is no longer valid by sending Update messages to them, and it deletes the cache record. On reception of the Update message, the application servers stop sending push messages and delete the IP address information.

## 7.2.5 Sharing User activated PDP context with Push Services

This section studies the use of Proxy AS to enable the sharing of PDP context activated by users with Push Services.

When a user activates PDP context with an APN whose value is in a record of the table <APN, Proxy_AS> in GGSN/NA, information including the IP address and the address of GGSN that holds the context, is sent in Update message to the Proxy AS specified in the record and is cached in the Proxy AS so that the context can be shared by the application servers. It is assumed that UE can prohibit creating multiple PDP contexts using the same APN. The diagram 7.2.7 shows user activated PDP context is shared by Push Services in dedicated connection approach.

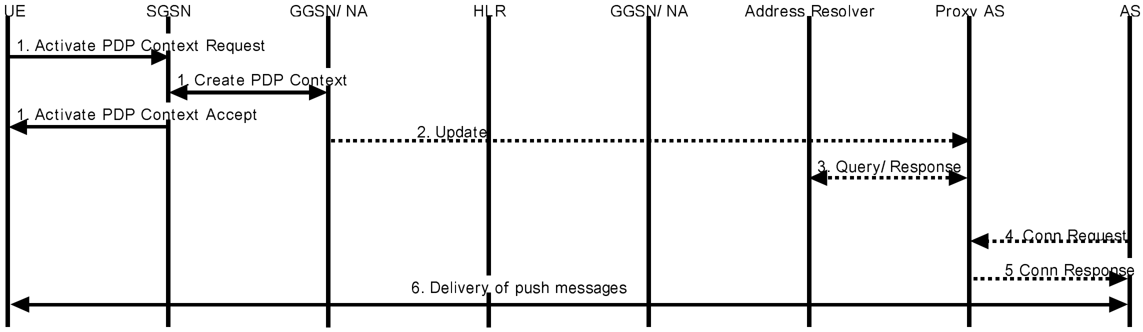


**Figure 7.2.7: Use of user initiated PDP context for Push Services**

1) UE initiates the PDP context activation with an APN whose value is in a record of the table <APN, Proxy_AS> in the GGSN/NA and an IP address is allocated to the UE.

2) The NA on the GGSN searches the table of <APN, Proxy_AS> for a record whose APN is equal to the APN of the created context. If found, the NA sends Update message to the Proxy AS specified in the record. The message may contain IMSI and the allocated IP address.

3) The Proxy AS obtains User-ID associated with the IMSI carried in the Update message, by contacting the AR. The proxy creates a cache record of User-ID, the allocated IP address, and the address of GGSN.

4) Push messages for the User-ID are generated in the AS. The AS sends Connection Request with User-ID to the Proxy AS.

5) The Proxy AS checks if an IP address is allocated to the User-ID by searching the cache. In this case, the Proxy finds a cache record with the User-ID. If the Proxy determines to share the IP address, then it adds the address of the AS in the list of application servers and sends the Connection Response message with the User-ID and the IP address back to the AS.

6) The AS sends push messages for the User-ID to the IP address.

## 7.2.6 Presence Service

Current 3GPP TS23.060[4] defines the Protection and Mobile User Activity procedure in order for the GPRS network to be able to perform some actions to prevent unnecessary enquires to the HLR after an unsuccessful Network-Requested PDP Context Activation procedure. This procedure can be expanded to network requested PDP Context activation with user-ID procedure as follows.

Figure 7.2.8 shows the procedure that the AS to request the presence service to GPRS network.

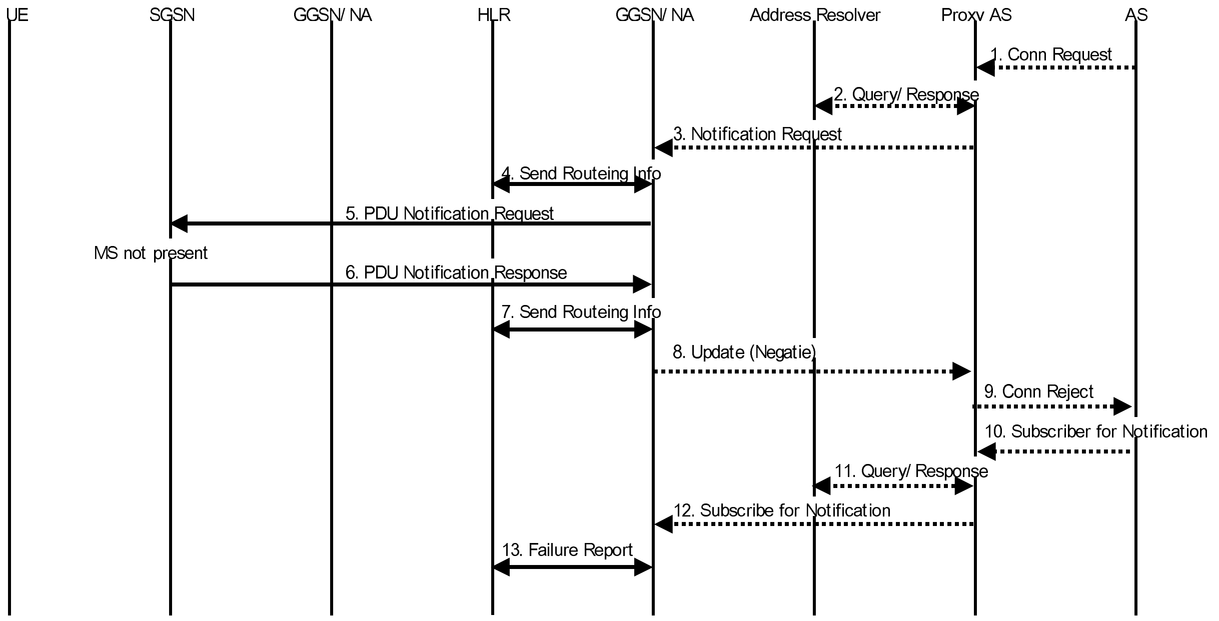


**Figure 7.2.8: Notification Request Procedure**

1) through 5) Same as in the section 7.2.2.1

6) If the MM context of the mobile is IDLE (GSM) or PMM_Detached (UMTS) or if the SGSN has no information about that user, the SGSN returns a PDU Notification Response (Cause) message to the GGSN with Cause equal to "MS GPRS Detached" or "IMSI Not Known". If the SGSN has an MM context for that user, the SGSN sets MNRG to indicate the need to report to the HLR when the next contact with that MS is performed.

7) If Cause equals "IMSI Not Known" the GGSN may send a Send Routeing Information for GPRS message to the HLR. The HLR returns a Send Routeing Information for GPRS Ack message to the GGSN indicating the

address of the SGSN that currently serves the MS. If SGSN Address is different from the one previously stored by the GGSN, then step 5 and 6 in section 7.2.2.1 are followed.

8) If SGSN Address is the same as the one previously stored in the GGSN or if the Cause value returned in step 6 equals "MS GPRS Detached", the NA in the GGSN sends Update message to the Proxy AS to inform the Proxy AS that the IP address cannot be obtained for the user.

9) The Proxy AS informs the AS that the request failed.

10) The AS sends Subscriber for Notification message with User-ID to the Proxy AS. The message is out of the scope of this contribution.

11) The Proxy AS contacts the AR to obtain the IMSI and the subscription information associated with the User-ID.

12) The Proxy AS sends Subscriber for Notification message with ~~User-ID~~IMSI to the ~~Proxy AS~~GGSN in order for ~~GRPS ~~GPRS network to watch the user's activity.

13) The GGSN/NA sends a Failure Report message to the HLR to request MNRG to be set in the HLR. The HLR sets (if not already set) MNRG for the IMSI and adds GGSN Number and GGSN Address to the list of GGSNs to report to when activity from that IMSI is detected.

The Mobile User Activity procedure in 3GPP TS23.060[4] may be invoked when the GPRS network observes user activity. Consequently the GGSN/NA can inform the AS that the MS is available. The figure 7.2.9 shows the procedure.

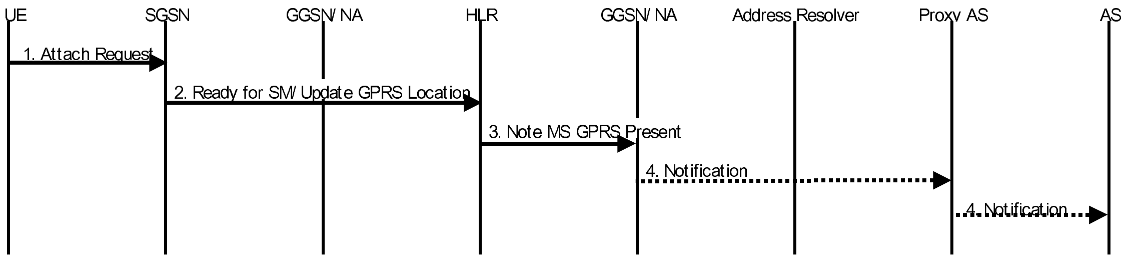


**Figure 7.2.9: notify AS the UE's availability**

1) The SGSN receives an indication that an UE is reachable, e.g., an Attach Request message from the UE.

2) If the SGSN contains an MM context of the MS and MNRG for that UE is set, the SGSN sends a Ready for SM message to the HLR and clears MNRG for that MS. If the SGSN does not keep the MM context of the UE, the SGSN shall send an Update Location message to the HLR.

3) When the HLR receives the Ready for SM message or the Update Location message for an UE that has MNRG set, it clears MNRG for that UE and sends a Note MS GPRS Present message to the GGSN/NA in the list of the subscriber.

4) The ~~SGSN~~GGSN/NA sends Notification message to the Proxy AS and the Proxy AS relays the message to the AS.

## 7.2.~~6~~7 Proposed Protocol Architecture

Figure 7.2.10 and 7.2.11 show the protocol architecture for NRPCA. Notification Protocol is used by the Proxy AS to request GGSN/NA to initiate NRPCA. RADIUS accounting protocol is used to update the mapping between the User-ID and IP address maintained in the Proxy AS.

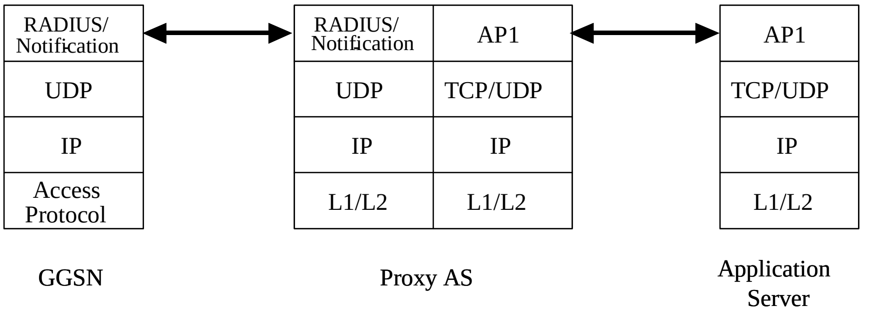


Figure 7.2.10: Control Protocol for NRPCA

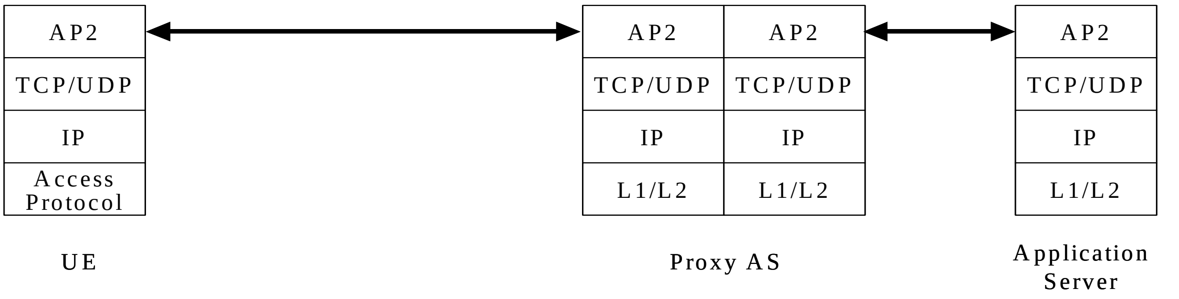


Figure 7.2.11: Data Protocols for Push Services

### 7.2.7.1 Notification Protocol

Requesting NRPCA involves exchange of two messages between the Proxy AS and the GGSN/NA as shown in the Figure 7.2.12.

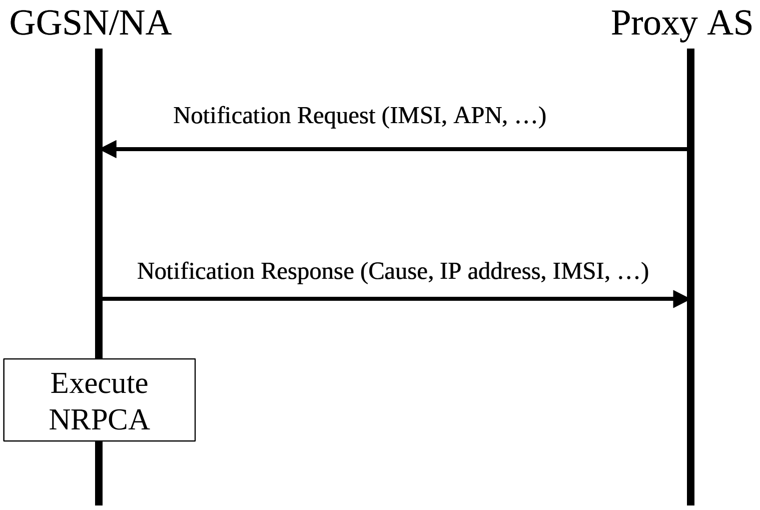

GGSN/NA
Proxy AS
Notification Request (IMSI, APN, …)
Notification Response (Cause, IP address, IMSI, …)
Execute NRPCA

Figure 7.2.12: Message Flow of Notification Request

Figure 7.2.13 depicts the proposed header for the Notification protocol. The value for the message type for the two messages, Notification Request and Notification Response is to be determined. Retransmission mechanism defined for GTP-C can be applied for reliability. The port number for this protocol is to be determined and probably has to be registered with IANA. Further protocol enhancements may include error messages, redirect messages, keep alive messages, but this would be for stage 2 and 3 decisions.

Bits

| Octets | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 |
|---|---|---|---|---|---|---|---|---|
| 1 | Version | | | | Not used | | | |
| 2 | Message Type | | | | | | | |
| 3 | Length (1st Octet) | | | | | | | |
| 4 | Length (2nd Octet) | | | | | | | |
| 5 | Sequence Number (1st Octet) | | | | | | | |
| 6 | Sequence Number (2nd Octet) | | | | | | | |
| 7 | Not used | | | | | | | |
| 8 | Not used | | | | | | | |

**Figure 7.2.13: Header for Notification Protocol**

Table 7.2.2 shows the possible information elements (IEs) included in the Notification Request. The IEs in the table are defined in 29.060[4].

**Table 7.2.2: Information Elements for Notification Request**

| Information element | Reference |
|---|---|
| IMSI | 29.060 7.7.2 |
| Access Point Name | 29.060 7.7.30 |
| MSISDN | 29.060 7.7.33 |

Table 7.2.3 shows the information elements (IE) included in the Notification Response. The value for Cause Values is for further study.

**Table 7.2.3: Information Elements for Notification Response**

| Information element | Reference |
|---|---|
| Cause | 29.060 7.7.1 |
| IMSI | 29.060 7.7.30 |
| MSISDN | 29.060 7.7.33 |

### 7.2.7.2 Update Protocol

Notifying IP address allocated by NRPCA procedure or de-allocated IP address to the Proxy AS involves exchange of two messages as shown in Figure 7.2.14. RADIUS Accounting-Request and Accounting-Response are used to update the status of PDP context for Push services to the Proxy AS. When a PDP context for Push services is created by NRPCA, the allocated IP address is notified. When the PDP context is deleted, the deletion (null IP address) is notified.

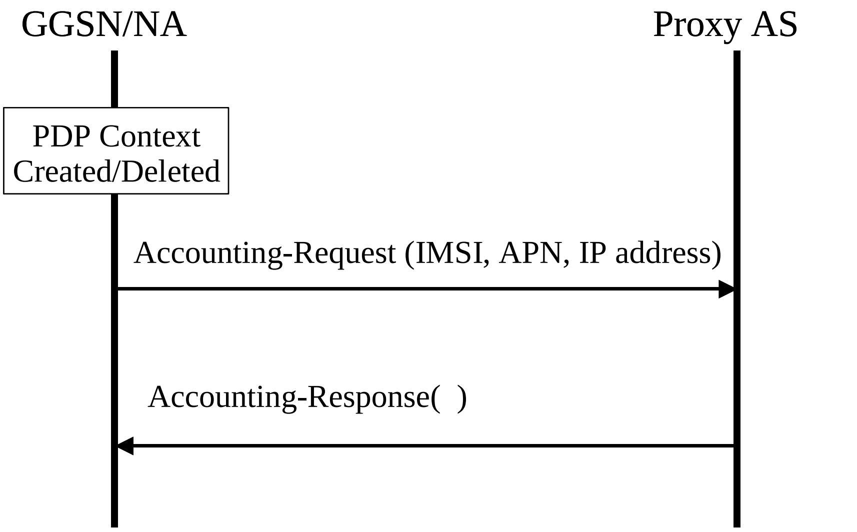
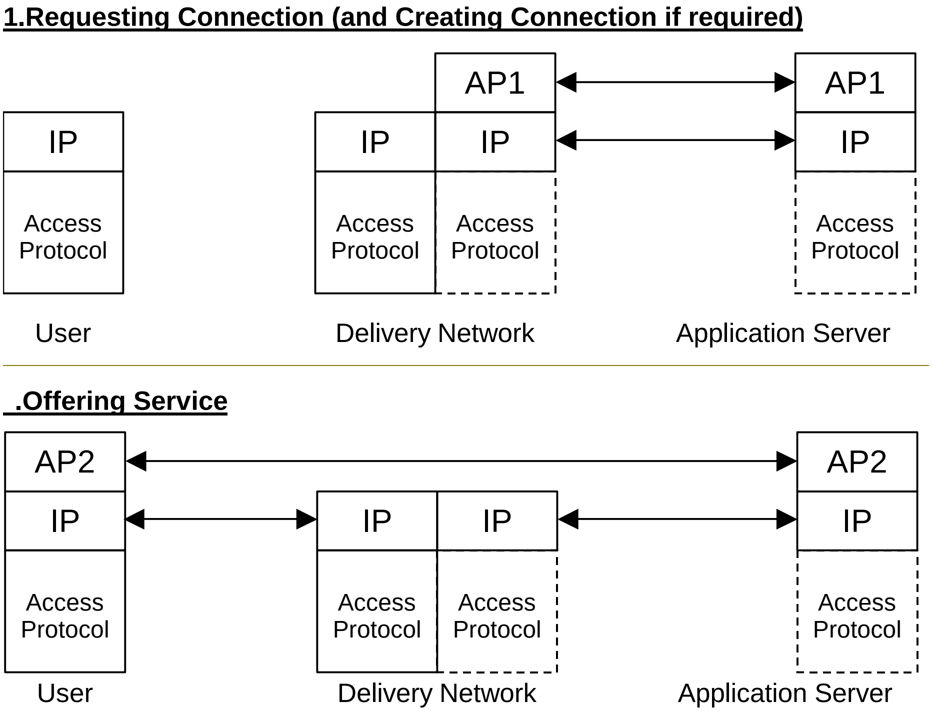

GGSN/NA
Proxy AS
PDP Context Created/Deleted
Accounting-Request (IMSI, APN, IP address)
Accounting-Response( )

**Figure 7.2.14: Message flow of Update**

The attributes that may be carried in the Accounting request are those being defined in 29.061[8]. In particular, the Assigned IP address, IMSI, username and MS-ISDN should be present as appropriate based on the service configuration.

~~In figure 7.2.10, 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. Protocols for AP1 and AP2 shall be identified.~~



~~**Figure 7.2.10: Protocol Stacks**~~

## 7.2.7 Impact to 3G specification

The table below shows impacts to 3GPP specification.

**Table 7.2.1: Impacts to 3GPP specifications.**

| SPEC | Proposed Change |
|---|---|
| 23.060 [4] | Addition of Network Initiated PDP Context Activation with dynamic PDP address<br>Consideration of backward compatibility for UE and SGSN not supporting Network Initiated PDP context Activation with dynamic PDP address. |
| 24.008[6] | Clarification of Network Initiated PDP Context Activation with dynamic PDP address.<br>Consideration of backward compatibility for UE not supporting Network Initiated PDP context Activation with dynamic PDP address. |
| 29.060[7] | Addition of description on usage of End User Address Information Element in PDU Notification Request for requesting dynamic IP address.<br>Consideration of backward compatibility for SGSN not supporting Network Initiated PDP context Activation with dynamic PDP address. |
| 29.061[8] | Addition of the description on Interworking with Proxy AS |

# 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 UE'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 UE'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 UE's IP address if after the addresses TTL expires. If the TTL expires, the application shall perform another DNS lookup to resolve UE's DN to IP address.
- The UE 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 UE 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 UEs.

- 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 UEs 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 UE. For each target UE, the carrier must provision a single PDNS with the UE's DN, its IMSI and the GGSN that is to be used for push services to that UE**.**

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

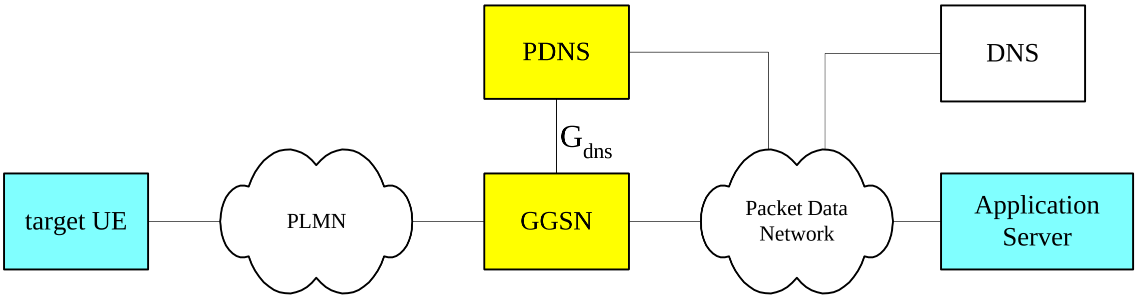


**Figure 7.3.1: Reference Push Service Architecture.**

A DNS is added to the PLMN (the PDNS). Each target UE is represented in the PDNS by a DN. The PDNS is further provisioned with the UE's IMSI and the GGSN to be used for push services to the UE. The Application Server then can query the DNS to resolve the UE'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 UE, 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 UE 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 UE. 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 UE. 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 UE must first query a DNS with the UE's DN.

- When the Application Server queries its local DNS with the UE'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 UE with a statically assigned IP address. For an UE 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 UE'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 UE – if any.

- If the UE 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 UE does not have an active PDP context, the GGSN will use the UE's IMSI to force the UE 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 UE 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 UE 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 UE 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 UE has been resolved.

It may be that the UE 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 UE 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 UE. The PDP context may also be inactivated by the UE, 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 UE 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 UE. 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 UE's IP address, it can push data to the target UE. Data packets will be routed to the associated GGSN.

- If the GGSN receives an IP packet on the IP address of the target UE and the PDP context still exists, the GGSN will forward the packet to the target UE in the legacy way. The UE 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 UE. In this scenario the UE has neither a statically assigned IP address nor an active PDP context.

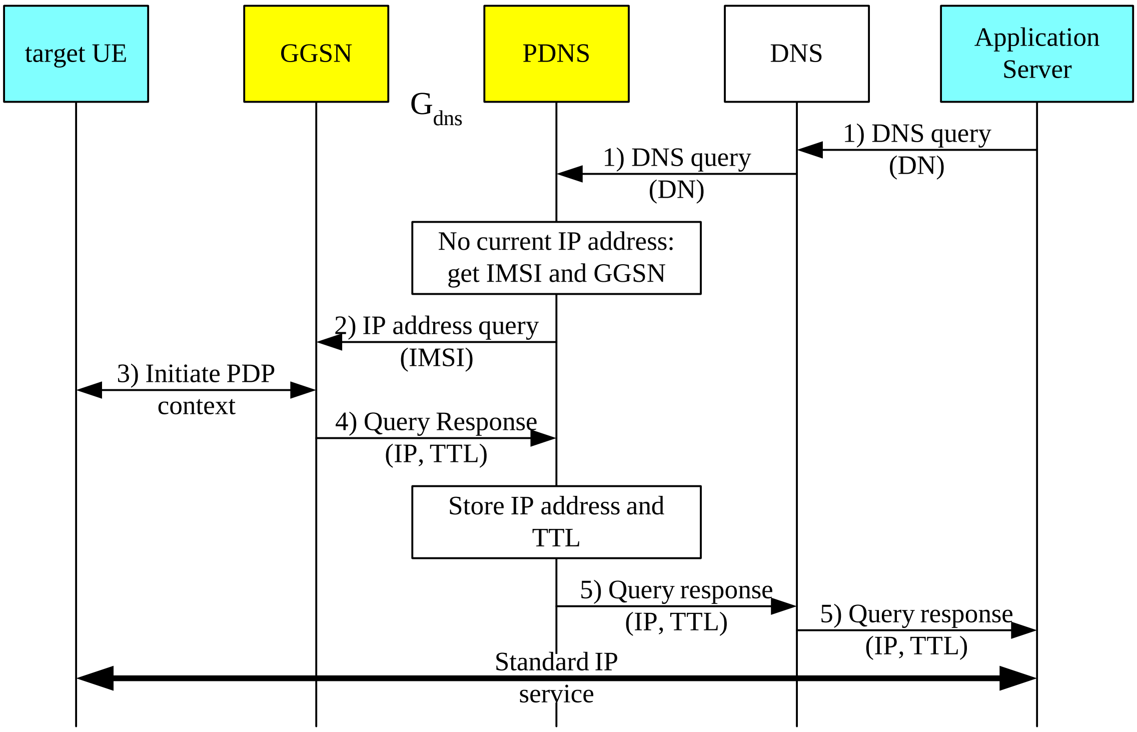

target UE
GGSN
PDNS
DNS
Application Server
G dns
1) DNS query (DN)
1) DNS query (DN)
No current IP address: get IMSI and GGSN
2) IP address query (IMSI)
3) Initiate PDP context
4) Query Response (IP, TTL)
Store IP address and TTL
5) Query response (IP, TTL)
5) Query response (IP, TTL)
Standard IP service

**Figure 7.3.2: Pushing data to a target UE**

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[12]. The request will be routed to the appropriate PDNS.

2) The PDNS server performs a search to fetch the entry corresponding to the target UE 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 UE's IMSI and GGSN and sends a message over the $G_{dns}$ interface to the GGSN. The parameter in the message is the UE's IMSI.

3) The GGSN, upon receipt of message, will look for the target UE's PDP context information. If no information is found it follows the procedure described in GPRS spec to instruct the target UE (mobile) to activate a PDP context (See 3GPP TS 23.060[4]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[13] 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[12] 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 UE'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.

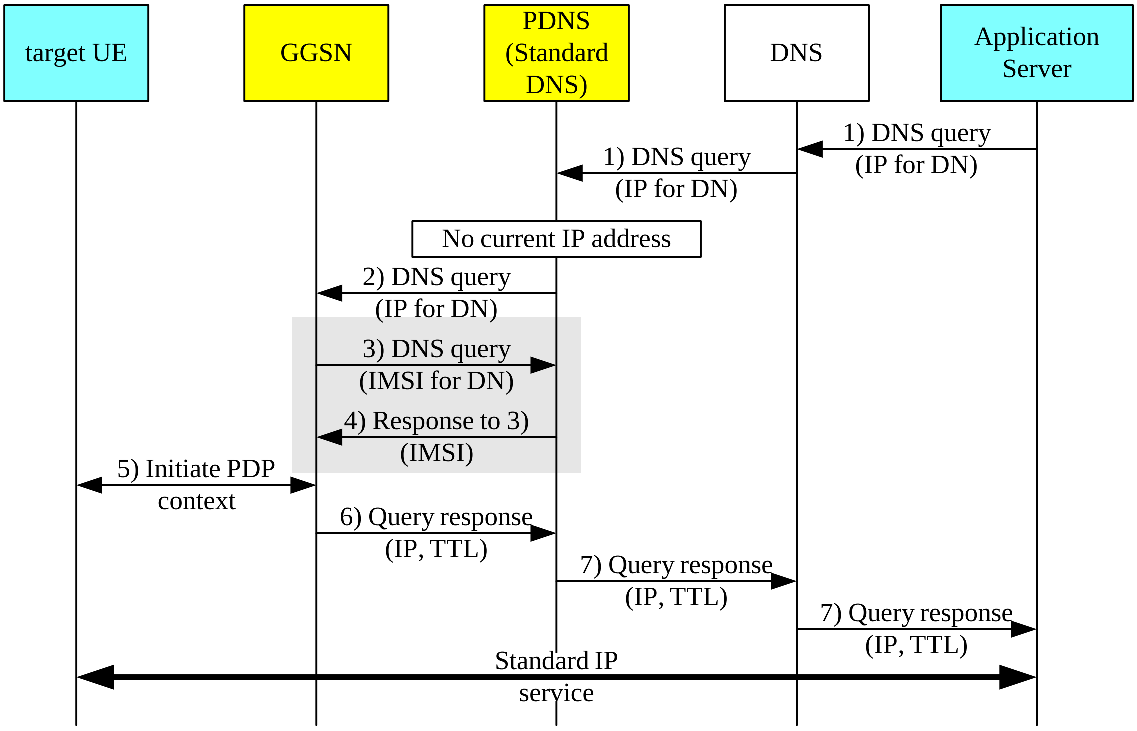


**Figure 7.3.3: PDNS implementation using a standard DNS**

### 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 UE. 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 UE Over The Air.

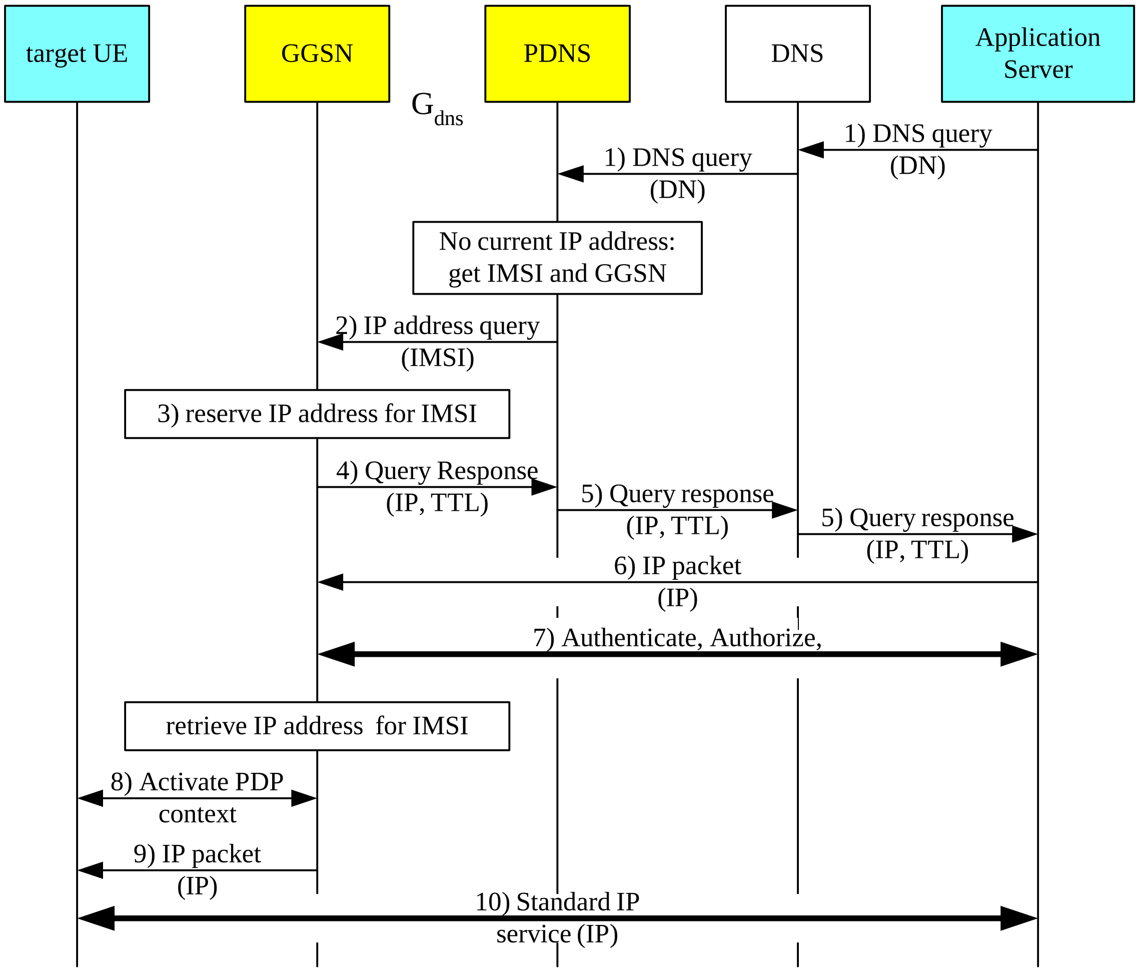

target UE
GGSN
PDNS
DNS
Application Server
$G_{dns}$
1) DNS query (DN)
1) DNS query (DN)
No current IP address: get IMSI and GGSN
2) IP address query (IMSI)
3) reserve IP address for IMSI
4) Query Response (IP, TTL)
5) Query response (IP, TTL)
5) Query response (IP, TTL)
6) IP packet (IP)
7) Authenticate, Authorize,
retrieve IP address for IMSI
8) Activate PDP context
9) IP packet (IP)
10) Standard IP service (IP)

**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 UE 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 UE does not have an active PDP context, and the UE 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 UE to activate a PDP context, using the UE'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 UE.

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 UE'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 UE with static IP addresses discussed in GSM document (See TS 23.060[4] 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 UEs that use that GGSN as their anchor point. Like a DNS server, the database will contain the UE's IP addresses and TTLs, while relevant. This solution has the drawback of reduced flexibility; the DN of an UE must resolve onto a specific GGSN.

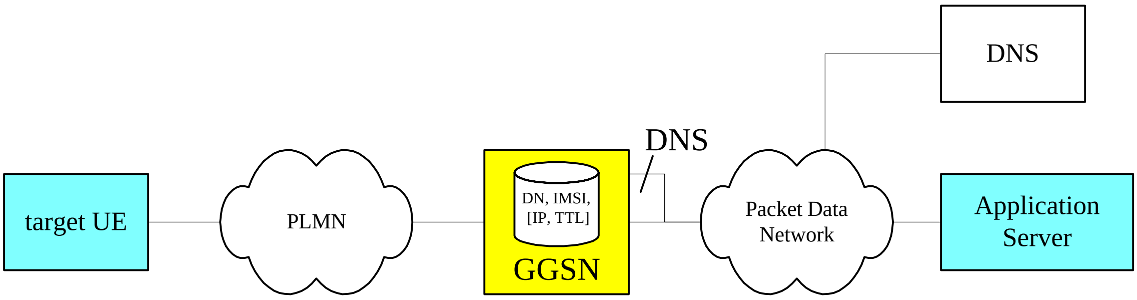


**Figure 7.3.5: GGNS with embedded PDNS database.**

## 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.

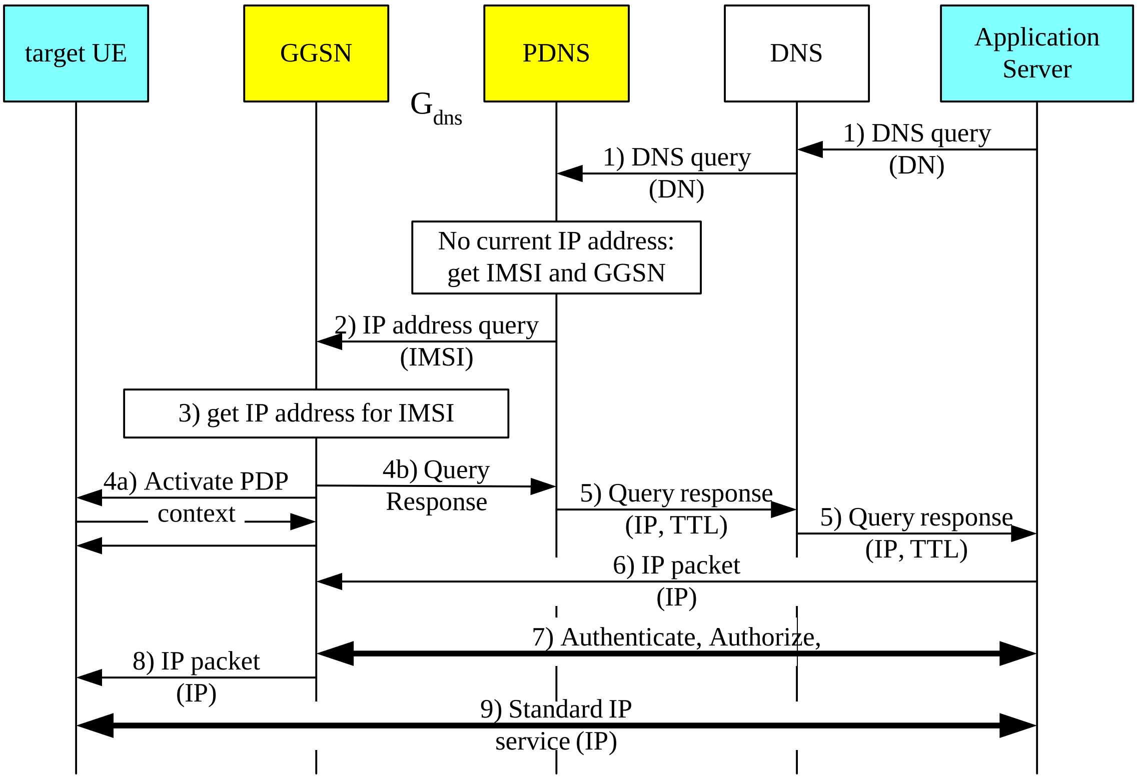

target UE
GGSN
PDNS
DNS
Application Server
$G_{dns}$
1) DNS query (DN)
1) DNS query (DN)
No current IP address: get IMSI and GGSN
2) IP address query (IMSI)
3) get IP address for IMSI
4a) Activate PDP context
4b) Query Response
5) Query response (IP, TTL)
5) Query response (IP, TTL)
6) IP packet (IP)
7) Authenticate, Authorize,
8) IP packet (IP)
9) Standard IP service (IP)

**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.

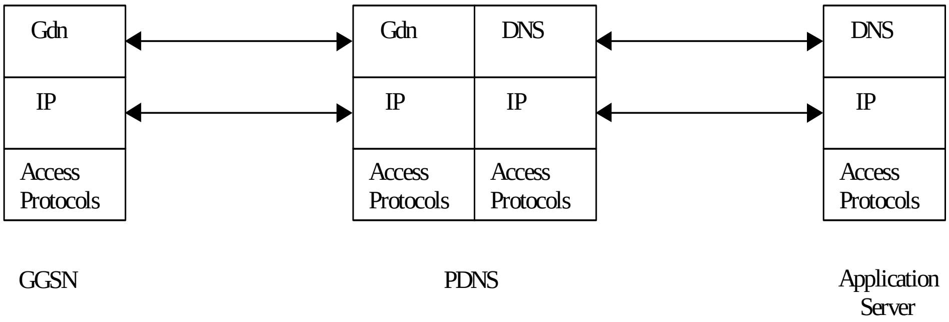


**Figure 7.3.7: Protocol stack for push initialisation**

## 7.3.13 Security

Access network shall protect a target UE 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 UE 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 UE 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 UE is not currently available or failed to establish PDP context)

These error codes are returned to confirm with DNS specification.

### 7.3.16 Impact to 3G specification

The table below shows impacts to 3GPP specifications.

**Table 7.3.1: Impacts to 3GPP specifications.**

| SPEC | Proposed Change |
|---|---|
| 23.060[4] | Addition of Network Initiated PDP Context Activation with dynamic PDP address<br>GGSN to SGSN: PDU notification request with IP address = 0.<br>Addition of a new message pair on Gi:<br>DNS Proxy to GGSN: IP_address_query( IMSI ), and<br>GGSN to DNS Proxy: IP_address_response( IMSI, IP_address, TTL ) |
| 24.008[6] | Clarification of Network Initiated PDP Context Activation with dynamic PDP address |
| 29.060[7] | Clarification to PDU Notification Request that PDP address may not be provided when dynamic address is used. |

## 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.

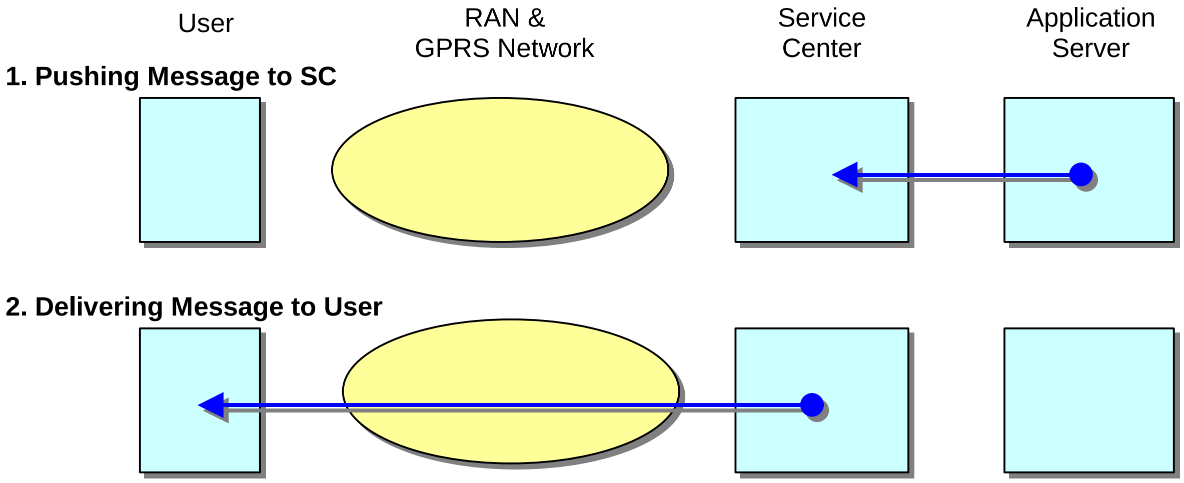


**Figure 7.4.1: SMS Push Message Scenario**

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 3GPP TS 23.040[3]). The interface to the SC from an external Application Server is not standardized by 3GPP (3GPP TR 23.039[2] 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.

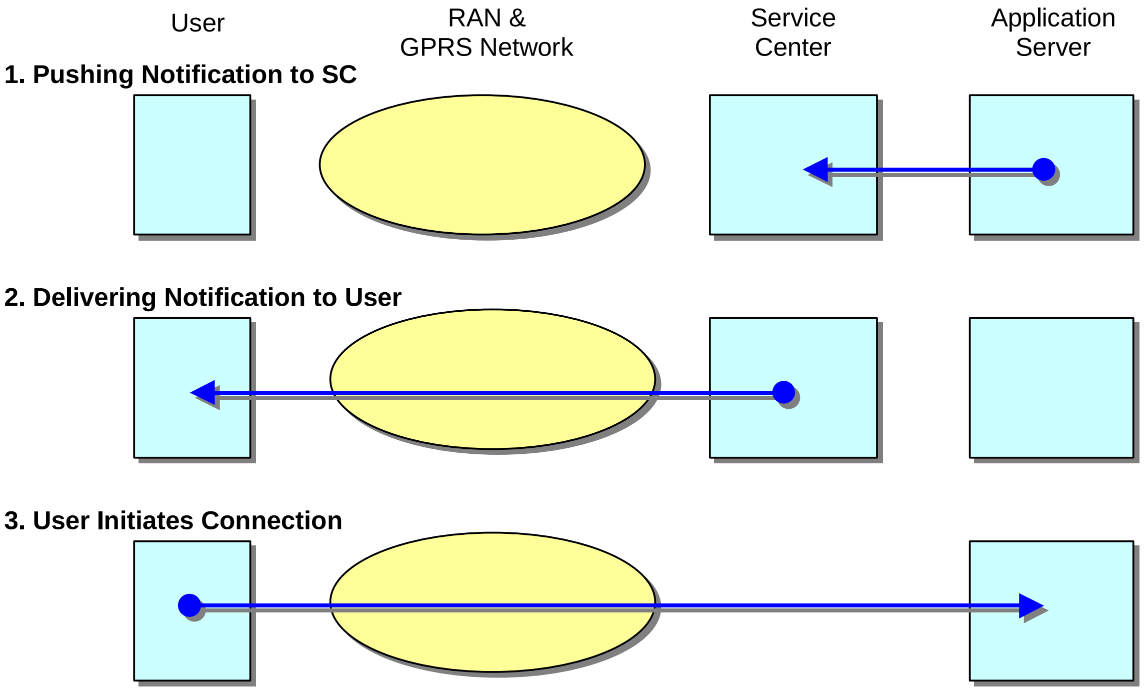

User
RAN &
GPRS Network
Service
Center
Application
Server
1. Pushing Notification to SC
2. Delivering Notification to User
3. User Initiates Connection

**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.

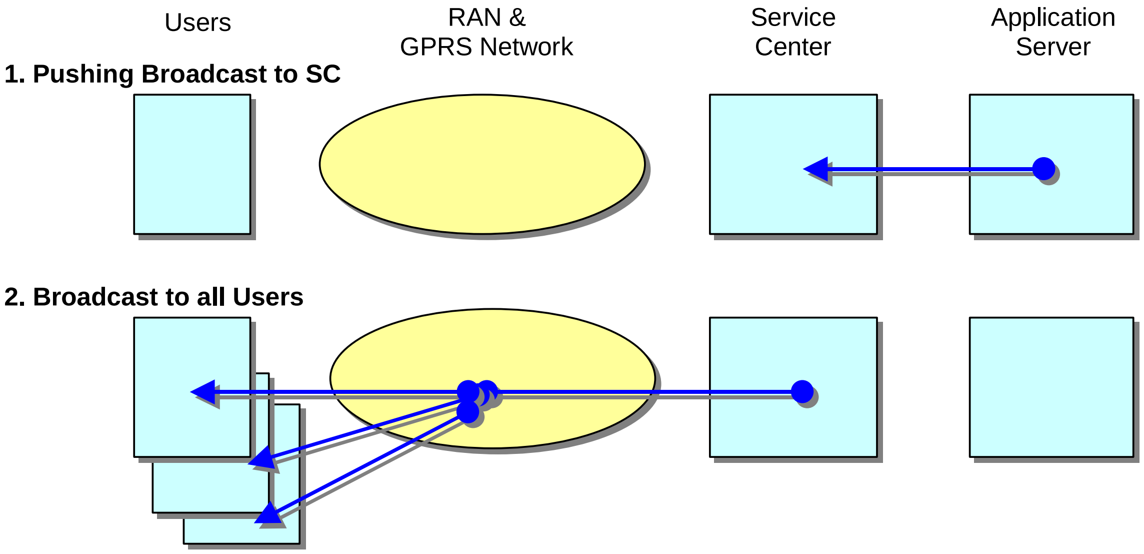


**Figure 7.4.3: Push Broadcast Scenario**

### 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.

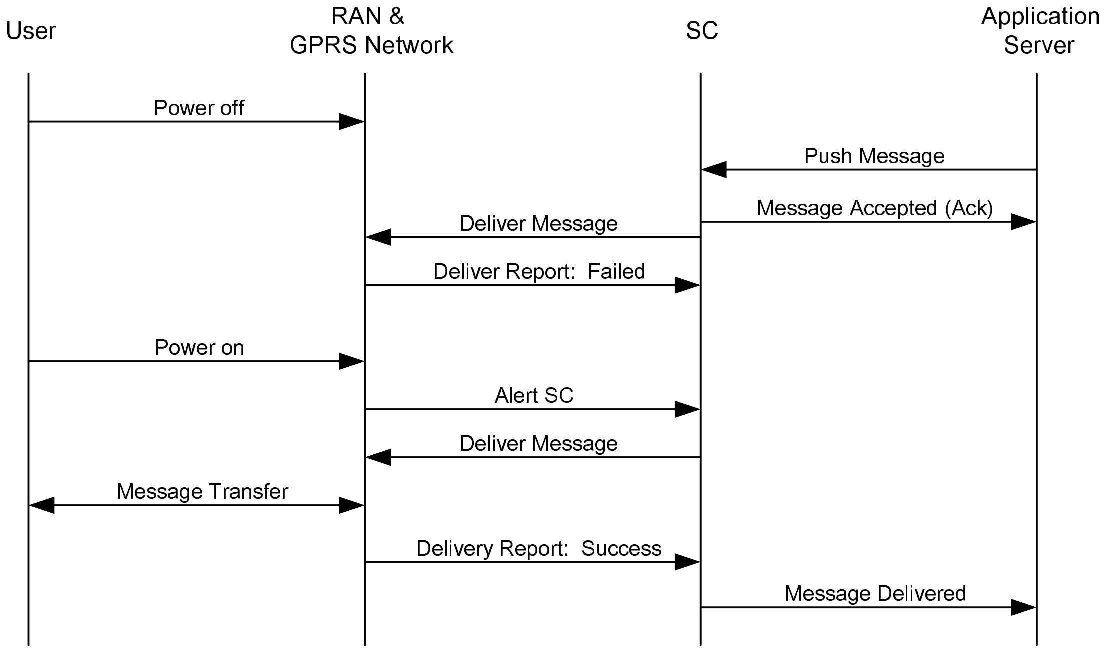


**Figure 7.4.4: SMS Reliable Delivery Sequence**

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[26]) 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 3GPP TS 23.040[3]. SMPP[28] 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).

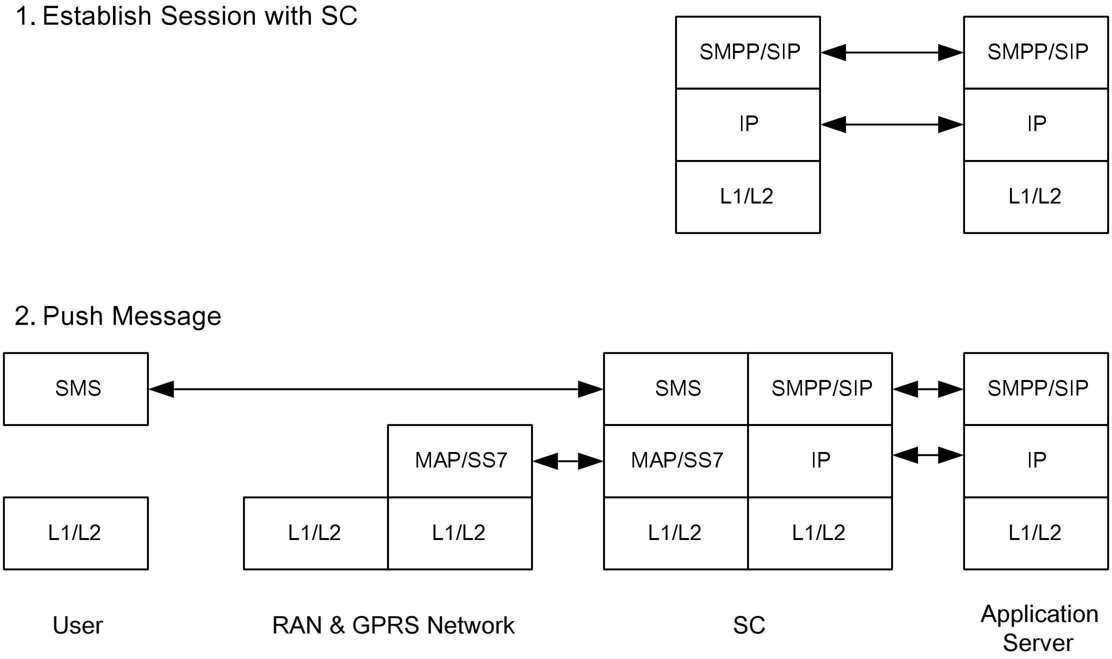


**Figure 7.4.5: SMS Protocol Architecture**

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 solution with dynamic address using always on and SMS

### 7.5.1 Architecture

The push architecture describes how to realize push service in a GPRS/UMTS environment.

The GPRS/UMTS access network is connected to the operator's service network by a GGSN. The APN (requested by UE but authorized by subscriber data) is used to select the operator's service network. This network is most likely an IPv6 or a private IPv4 network, and is managed by the operator. Through this network, the UE can access to a set of services (including push services) or to the Internet. This network may include RADIUS[14] (e.g. for authentication), DNS[12], DHCP[13a], and others services.

This network also includes an SMS Service Center.

For push services, the operator's service network includes a push proxy, while push initiator are typically in external network such as the internet (typically public IPv4 network). The UE trusts the push proxy not to send unwanted push messages. Push initiator is prevented to address the UE directly.

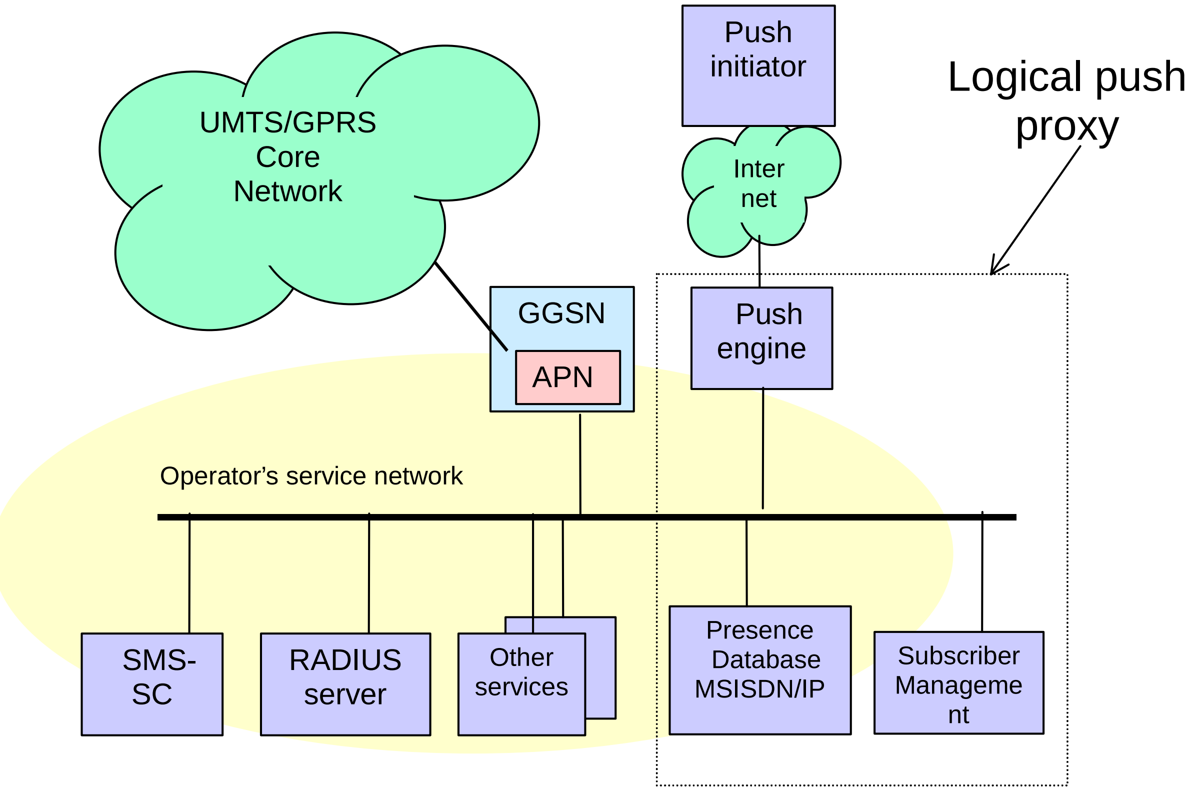


**Figure 7.5.1: Push architecture**

### 7.5.2 Push proxy

The push proxy is logically composed of:

- A presence database indicating if the subscriber (identified by MSISDN) is having a PDP context active and what is its dynamic IP address. Optionally, the presence database may indicate if the subscriber is available through another mean (e.g. fixed IP access).
- A subscriber management database indicating which push services are subscribed for which subscribers. This database is used to find if a received message can be forwarded to an UE or not.
- A push engine part: The push engine receives push messages from push initiator, checks the subscriber management database, checks the presence database, selects the bearer of the push messages and forwards them to the subscribers.

### 7.5.3 PUSH initiator

The content of the push message is received by the push proxy from push initiator servers. The push initiator shall have an agreement with the operator running the push proxy. A security relation shall be established between the push proxy and the push initiator.

Note: As an option, the push proxy might accept push message from unknown push initiator.

The push initiators are therefore not required to know the status of push service subscribers and their dynamic IP address.

Note: The WAP forum has defined a protocol between push initiator and push proxy.

### 7.5.4 Push services subscription

Every user should be able to receive different push services according to their wish.

The push proxy includes a subscription management database and therefore is aware of the service subscribed by each subscriber. The user should be able to modify their push subscription on-line (e.g. Olympic game news).

### 7.5.5 Addressing: Push service using dynamic address

The use of dynamic addresses is preferred as it avoids the configuration of a static IP address in HLR, UE, and push proxy.

The address may be a private IPv4 or an IPv6 address. Both provide virtually unlimited number of IP addresses. Due to restriction in the number of public IPv4 addresses these are unlikely to be used.

Note: In one operator network, different GGSNs may be connected to different private IPv4 operator's service networks. Each of these network can support around 17 millions private IPv4 addresses. This solution provides infinite number of private IPv4 address.

### 7.5.6 Presence description

The presence database needs to know the status of the subscriber (connected or not) and the mapping between its identity (MSISDN) and its dynamic IP address.

The UE application could indicate its presence directly to the presence server (e.g. using SIP). However, in order to optimise the system for wireless (saving one round trip over the radio), the GGSN may also inform the presence database when a PDP context is activated. This solution is described in more detail below.

The Presence database address is configured in the GGSN APN configuration. Each time a PDP context is activated to this APN, the GGSN informs the presence database of the subscriber status and the relation between its dynamic IP address and its MSISDN.

The mechanism is described in more details in the figure 7.5.2.

The GGSN sends a Start message (MSISDN; IP address) when the PDP context is activated, and a stop message (MSISDN; IP address) when the PDP context is deactivated. These start and stop messages may be implemented using RADIUS protocol, which is widely deployed, and to which specific extensions can be easily added.

This solution does not prevent the use of a subscriber logical name if the presence database knows the mapping between MSISDN and logical name. It would then be possible to update a DNS with the proper UE address.

In order to update the presence database of the availability through other accesses, other access router should implement similar behaviour, or the UE application should indicate its presence directly to the presence server (e.g. using SIP).

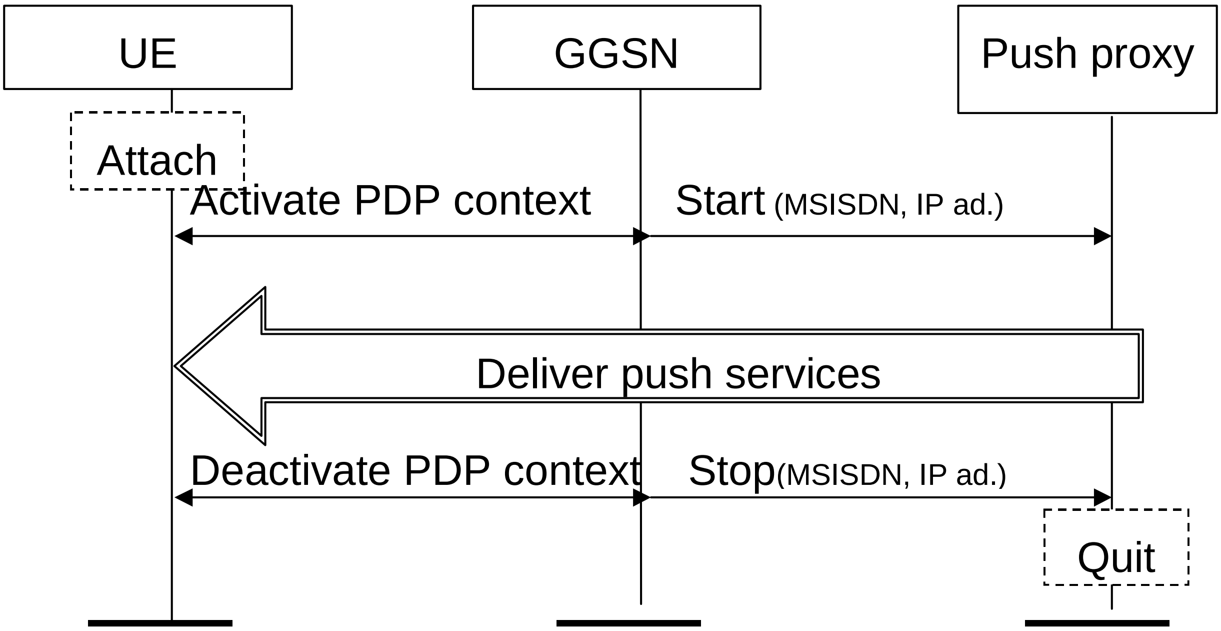


**Figure 7.5.2: Updating presence database from GGSN**

### 7.5.7 Delivery of the push message

In the always-on approach, push content is delivered over the established IP connection (i.e. PDP context or fixed IP connection).

If no IP connection is established, SMS may be used to trigger the terminal to create a PDP context over which the push content is delivered. SMS will trigger the UE as soon as this one becomes reachable (using SMS store and forward and alert capabilities). SMS may in addition deliver a short message to the terminal, and the validity period of the message can be set.

The push message is delivered through the following steps:

1) Push initiator sends push message to Push Proxy (PP), e.g. including MSISDN or a list of MSISDN.

2) PP checks subscription from its subscriber management database

3) If the message is allowed, Push proxy checks from its presence database if the UE has an IP address.

4) If an address is returned, the message is sent directly to UE through GGSN

5) If no IP address is returned, the message or an indication to activate the PDP context may be sent via SMS (providing store and forward) depending on the importance of the message and the subscriber right.

Note: In the case of a fixed IP connection, an access router would replace the GGSN.

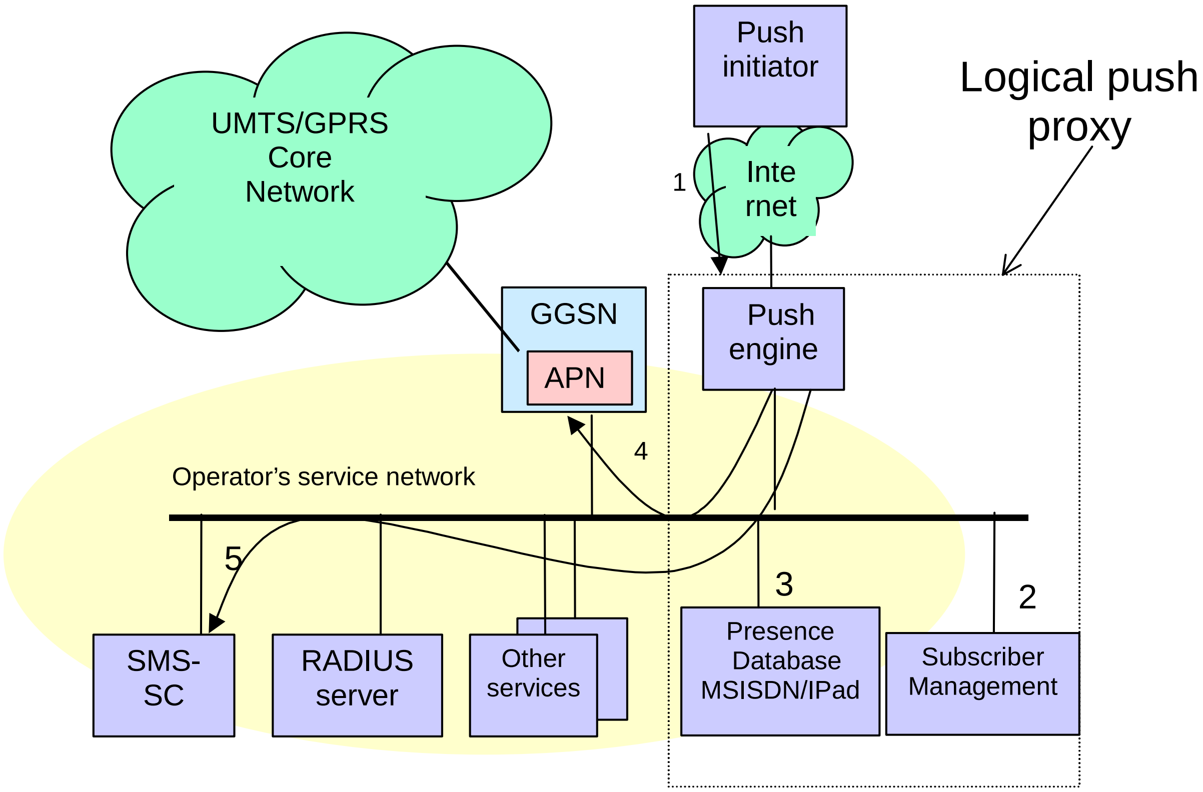


**Figure 7.5.3: Delivery of push messages**

## 7.5.8 Reliability of the delivery of the push message

The push proxy shall provide different level of reliability. It should in particular be capable of guaranteeing the delivery of the message through acknowledgement and retransmission mechanism.

For example, WAP-Push [16] provides Confirmed or non-confirmed delivery of Push messages. The figure 7.5.4 illustrates how Confirmed delivery works, when the UE has an active PDP context.

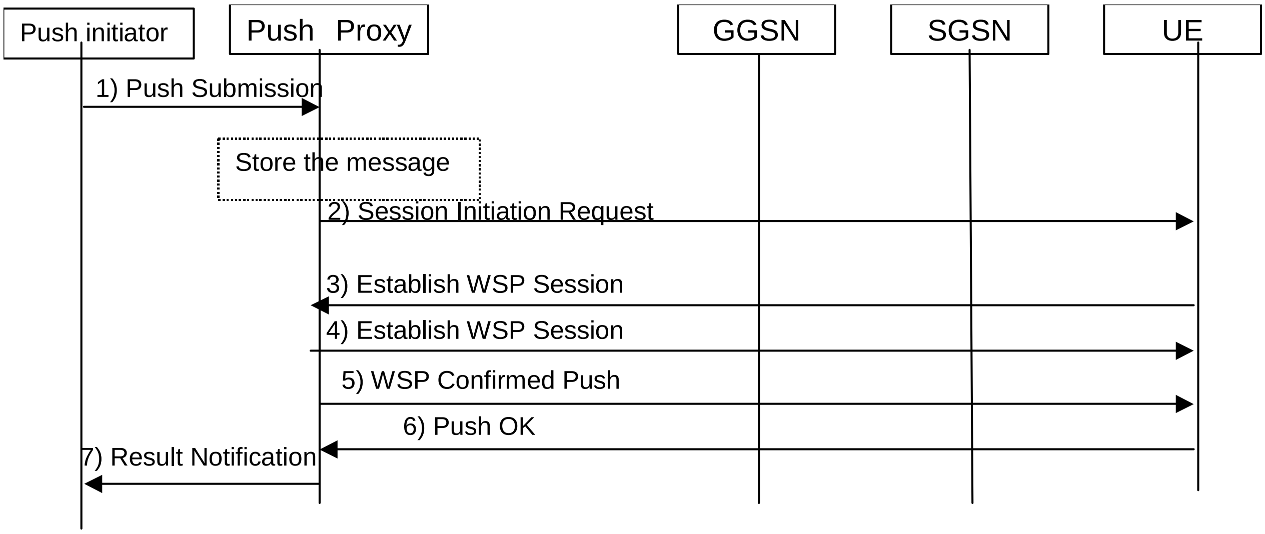


**Figure 7.5.4: WAP-push confirmed delivery to an UE having a PDP context activated.**

### 7.5.9 Store and forward function:

The store and forward functionality consists not only in storing & forwarding a message, but also in receiving an alert as soon as the UE is reachable. Existing SMS services provides such store-and-forward mechanism that may be used for PDP context activation triggers and possibly small push messages (e.g. text + URL). Other push messages are stored in the push proxy itself, and delivered once a PDP context is established.

The push proxy should use the store and forward functionality when requested by the push initiator and either of the following conditions is satisfied:

- If the UE is not having a PDP context activated, or
- if the delivery of the push message failed through the activated PDP context (e.g. no push OK message was received)

If the push message is short, the SMS should carry the push content directly.

If the push message is too long to fit in an SMS, the SMS triggers the UE to automatically activate a PDP context over which the push content is delivered from the PP to the UE.

Note: The SMS will be delivered as soon as the UE becomes IMSI attached or GPRS attached due to existing SMS alert mechanism.

### 7.5.10 Multiple services

An UE may receive push service from multiple push initiator servers. The push proxy supports delivery of push content from multiple sources simultaneously.

### 7.5.11 Security

Allowing directly push initiator from the Internet to push their message to the UE would be a security risk. In addition, if IPv4 private addresses are used by UE, it is doubtful that a push initiator using a public IP address could reach the UE. The PI would not only need to know which of the NAT’s public IP addresses it should target, but also the port. The NAT[12a] device normally assigns both the address and the port dynamically for every request sent by the UE. This implies that the PI needs to be able to request the NAT device to map a certain socket to a specific UE before the push is sent (not standard NAT behaviour), which of course is not feasible.

The push proxy guarantees the security by:

- Authenticating the push initiator as a valid one
- checking from its subscriber management database what should be sent to the UE(s)
- forwarding, if authorized  the push message to the UE over the secure operator service network (private IPv4 or IPv6 network).

Therefore the use of a push proxy protects the user from unwanted attacks. In addition, using a push proxy avoids the use of NAT between UE and push initiator.

The figure 7.5.5 illustrates a possible secure implementation where the push proxy has secure connection to push initiator, and the UE are protected from unwanted request by a firewall. In this example NAT (Network Address Translation) is not needed for push services.

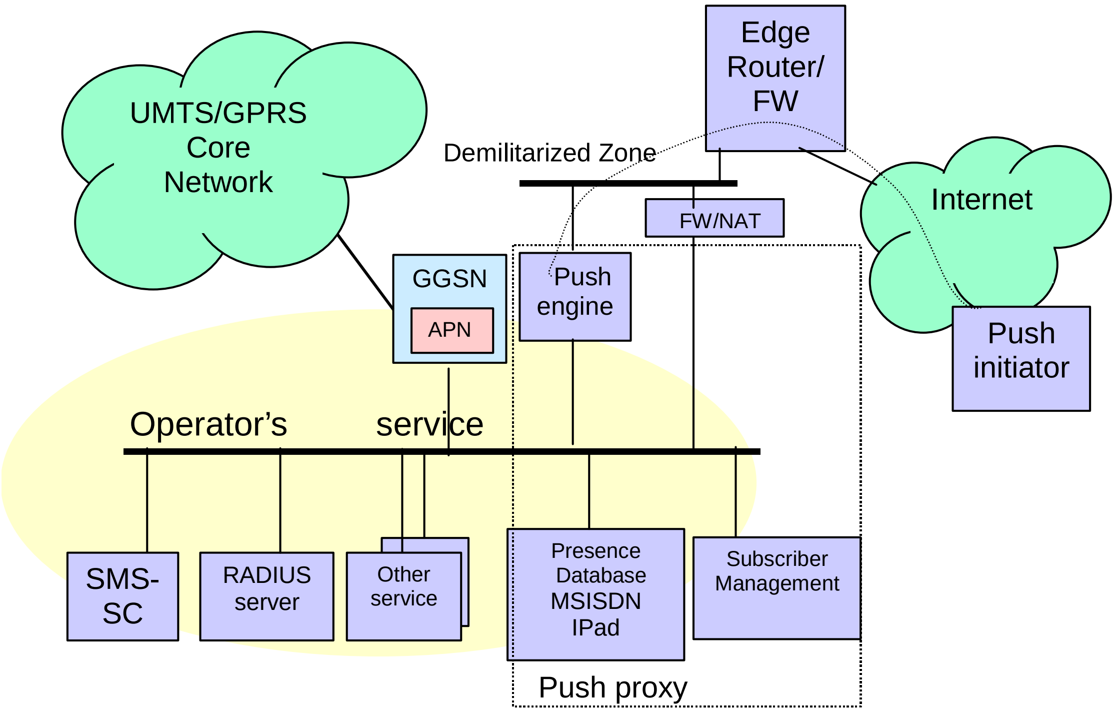


**Figure 7.5.5: A possible secure implementation of push services.**

## 7.5.11 Charging

The network operator may charge based on user subscription to specific services and on the service delivered. Charging should be handled at service level by the push proxy, as:

- The push proxy handles subscription of services to which a monthly charge may be linked.
- The push proxy handles delivery of messages, so that it can create a service charging record per message delivered. This service charging record may have extra information about the service delivered (e.g. delivery was confirmed, push initiator, to which subscribed group it belongs, etc)

## 7.5.12 User terminal

In order to be used for push services, the user terminal is required to support:

- A push application
- IP access (through GPRS/UMTS) and SMS

Note: A release 99 terminal may be used if it supports the proper application, or if the proper application can be added on it (e.g. EPOC terminal).

## 7.5.13 Roaming Support

PLMNs support roaming service. Thus, push service shall be available to subscribed users when they roam. The home GGSN should be used when roaming. The SGSN is connected to the GGSN using the inter-PLMN backbone and all traffic is tunneled from GGSN to SGSN. Therefore the UE access exactly the same services than if it would be in its home network.

This can be enforced by operator in the HLR subscription data.

Note: Using visited GGSN would imply a method to access subscription information from the visited network. This can be made based on agreement between operators.

### 7.5.14 IP address management

Due to the limitation of public IPv4 addresses, a (successful) push service has to use private IPv4 or IPv6 addresses. Both solutions provide unlimited number of addresses.

There are around 17 million private IPv4 addresses for one private IPv4 network. If one operator needs more addresses, different GGSN may be connected to different private IPv4 networks providing infinite number of private IPv4 address. Private network may be differentiated using APN. UE may be allocated to one or the other network based on their subscribed APN. Push proxy (or push initiator) knows upon available information to which network the UE is registered.

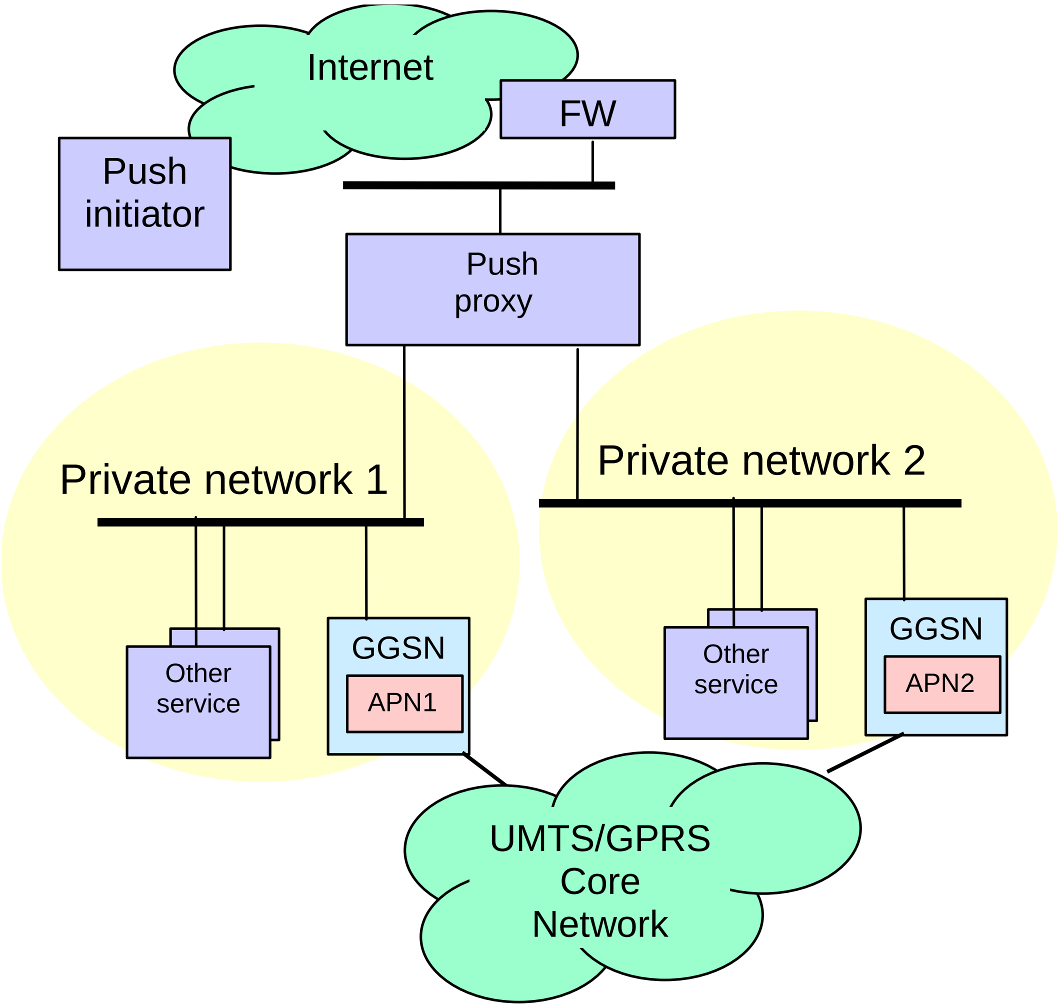


**Figure 7.5.6: A possible implementation of push services with multiple private IPv4 address spaces.**

## 7.6 SIP based Push Service

SIP based push services uses a proxy based architecture where SIP signalling is used to establish a session between the UE and the Push Proxy. An access protocol is used between the Push Initiator and the Push Proxy. The access protocol can be based on the protocol defined in section 7.7 or on SIP in order to perform end to end multimedia session.

~~SIP[16] 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 R5 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.

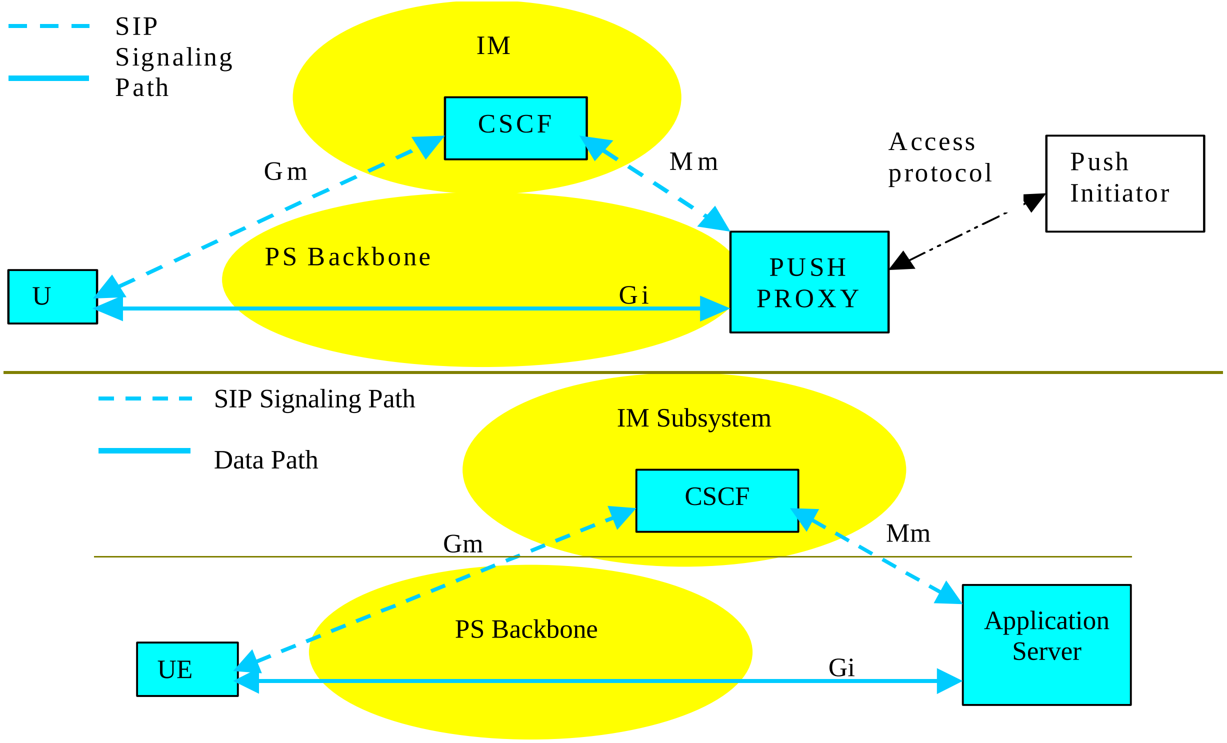


**Figure 7.6.1: SIP based Push via IM Subsystem**

The SIP Signaling Path is a standard ~~bearer~~ IMS Signalling PDP context, across the Gm reference point,  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~~another PDP context in accordance with IMS architecture, see TS23.228. ~~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 signalling PDP context to be used for SIP registration and signaling. The IP address allocated for this PDP context ~~could~~ will be assigned as a dynamic ~~or static~~ IP address(TS23.228). ~~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 3GPP TS 23.228[5]). 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~~ Push Proxy 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~~Push Proxy 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 push service~~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~~ an existing PDP context or establish a ~~separate~~ new PDP context to be used for Data Path traffic. The IP address assigned to the UE for ~~the~~ this PDP context would be provided to the ~~Application Server~~Push Proxy as part of the response to the SIP Invite.

7) Since SIP is a session protocol, the ~~Application Server~~Push Proxy 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~~Push Proxy, 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~~Push Proxy, the PDP context must be active. By using the preservation procedures described in 3GPP TS 23.060[4], 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~~Push Proxy. 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.

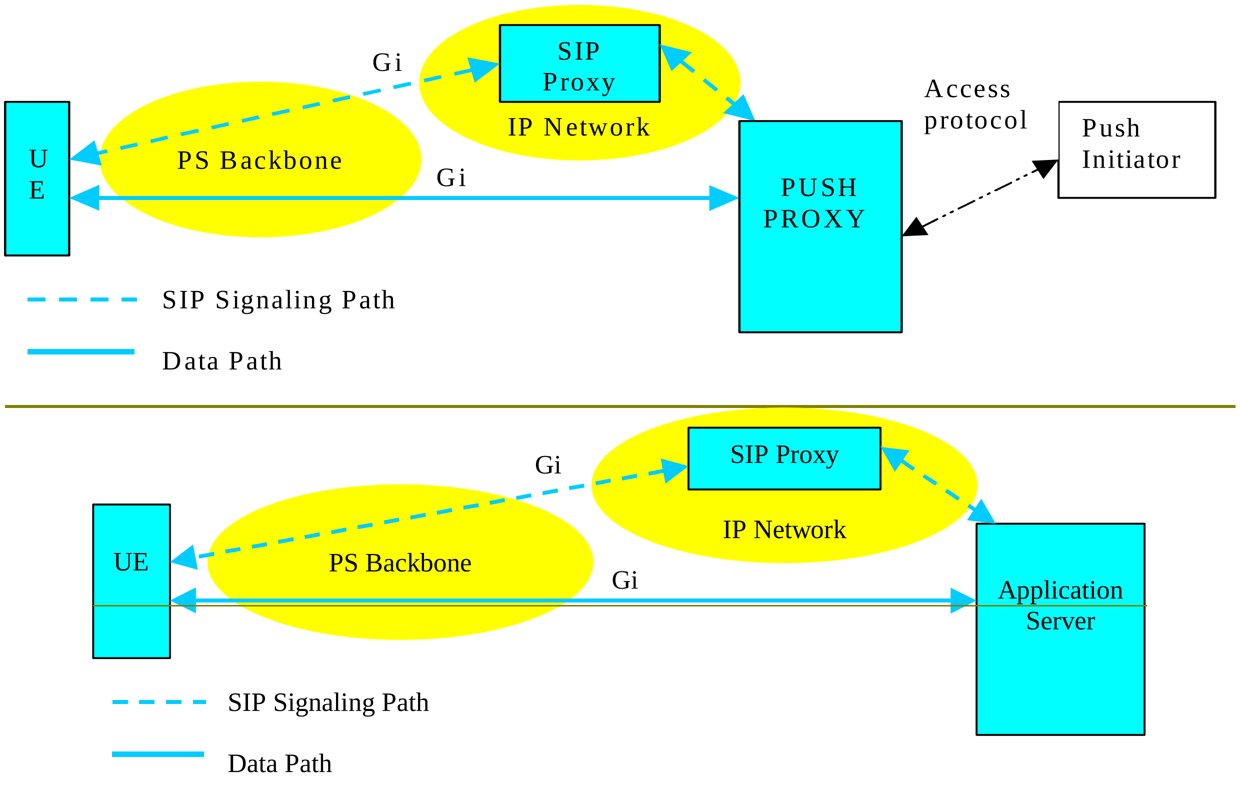


**Figure 7.6.2: SIP based Push via Long-lived PDP Context**

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.

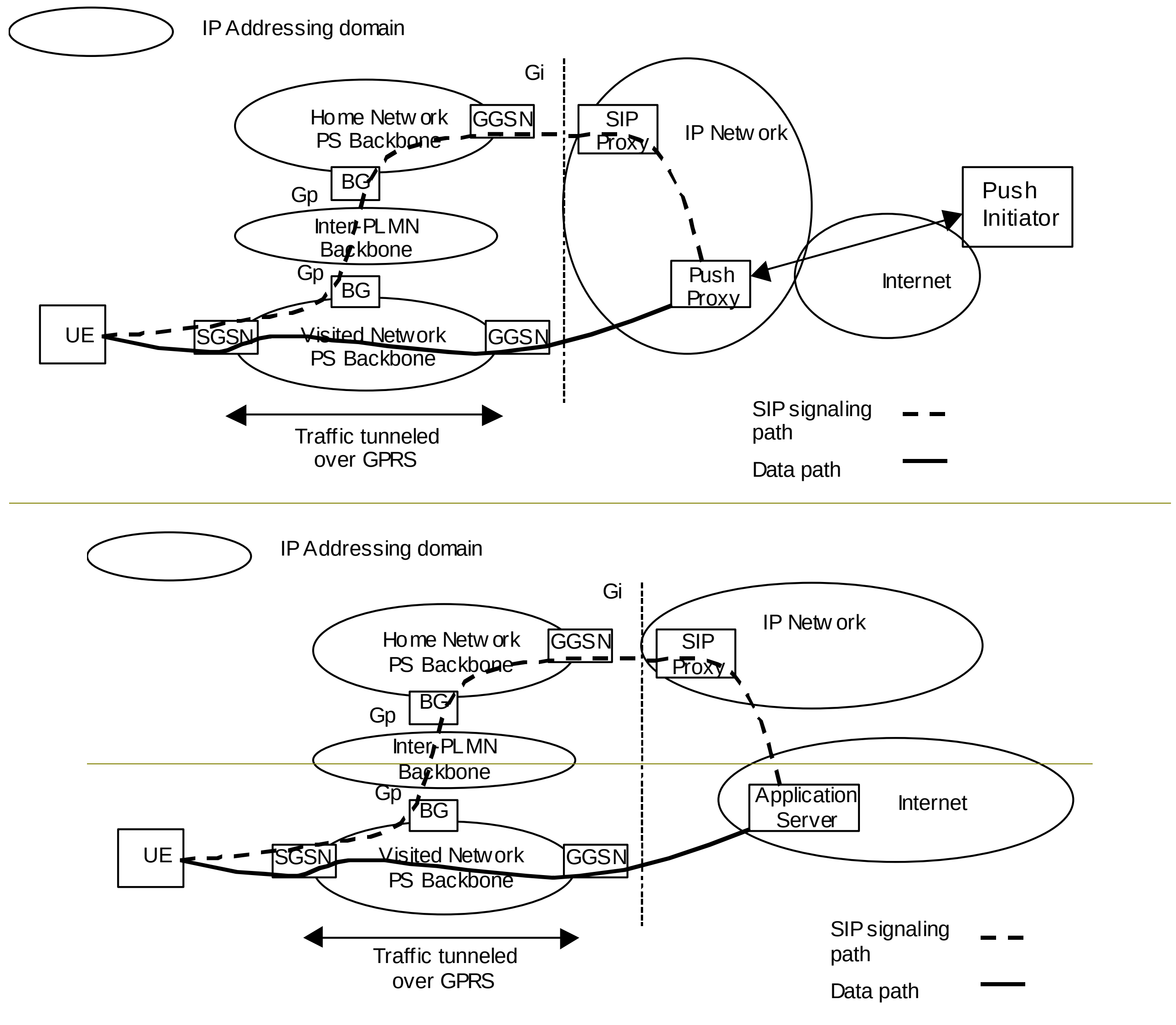


**Figure 7.6.3: Roaming – SIP Proxy via HPLMN**

.

### 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.

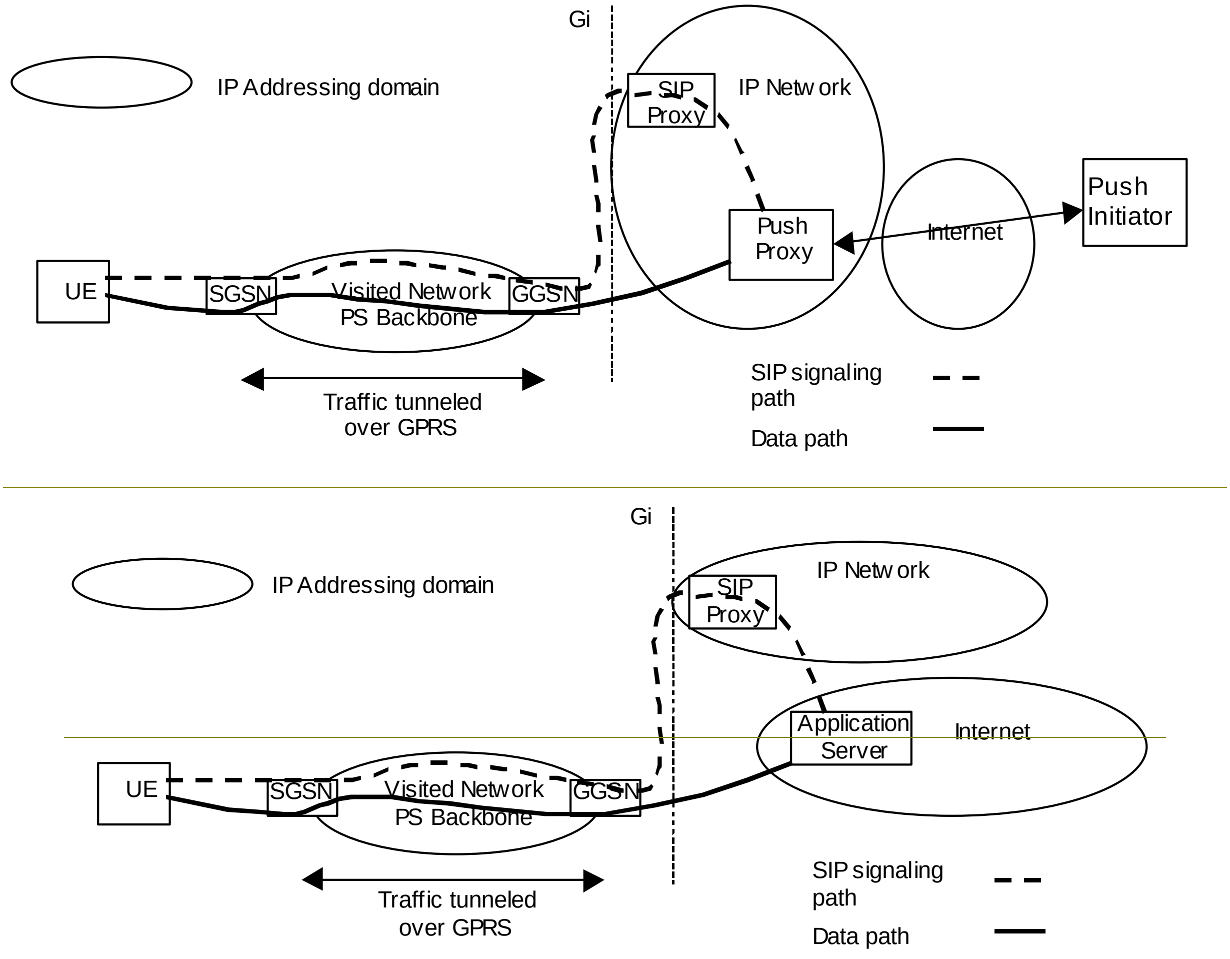


**Figure 7.6.4: Roaming – SIP Proxy via VPLMN**

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.

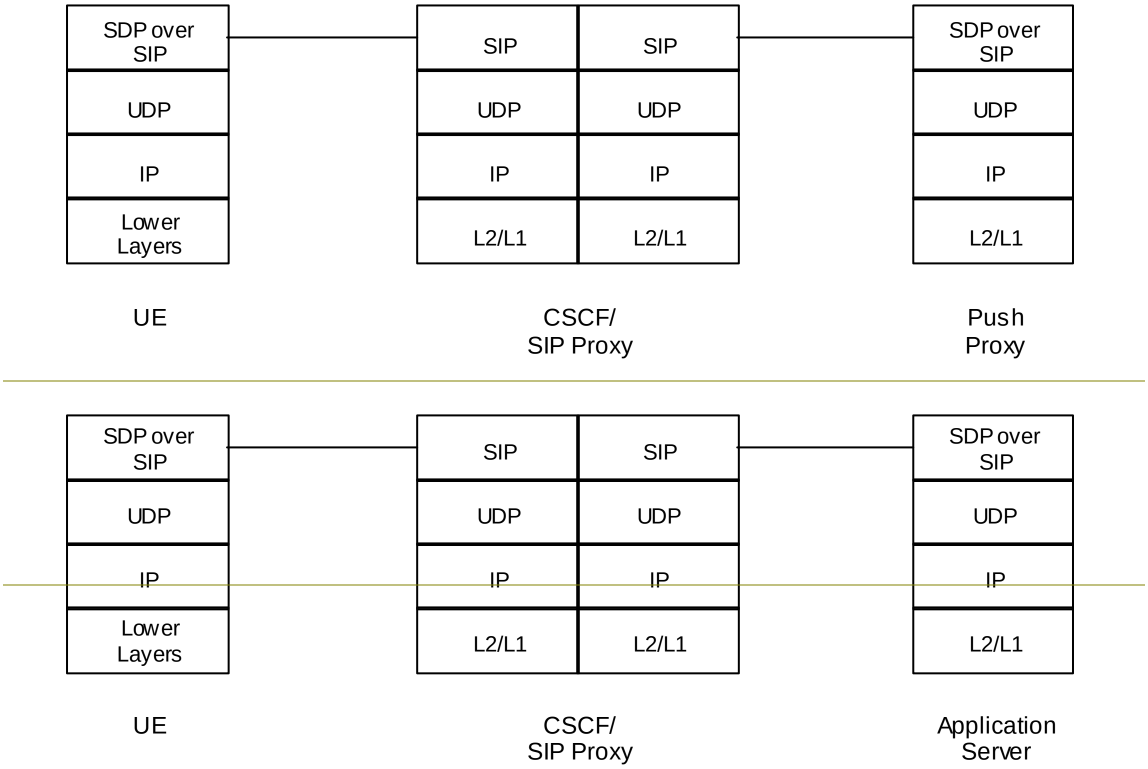


**Figure 7.6.5: SIP Session Management Protocols**

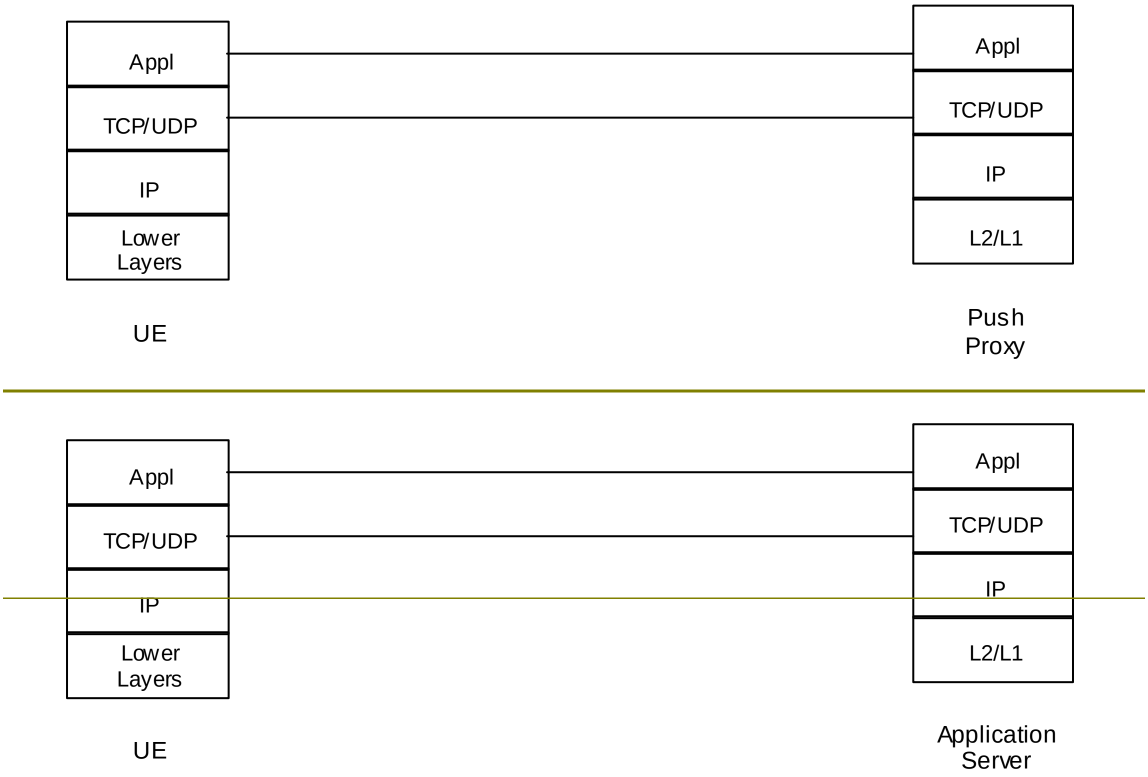


**Figure 7.6.6: Data Protocols**

### 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~~Push Proxy 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~~Push Proxy 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~~Push Proxy 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~~Push Proxy's IP address will be included in the SIP Invite. The UE will provide its Data Path IP address to the ~~Application Server~~Push Proxy 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~~ Push Proxy will be responsible for retrying later.

There are currently IETF draft proposals (draft-rosenberg-impp-lpidf-00.txt[22], draft-rosenberg-impp-presence-00.txt[23], draft-rosenberg-impp-im-00.txt[24]) to include Presence as part of SIP. SIP Presence would allow the ~~Application Server~~Push Proxy 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~~Push Proxy 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~~Push Proxy 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[22], draft-rosenberg-impp-presence-00.txt[23], draft-rosenberg-impp-im-00.txt[24]).

As a further option, SIP MESSAGE method may be used to carry peer-to-peer data.  The push message is carried in the method body. MESSAGE method is specified in draft-ietf-simple-im-01.txt.

Note: Draft-ietf-simple-im-01.txt is due to be sent to the IESG in August 2001.

### 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~~Push Proxy.  The UE will be responsible for establishing the Data Path PDP context using the supplied QoS.

## 7.7 Push Proxy Based Architecture Using HTTP as Delivery Protocol

### 7.7.1 Architecture

A proxy based architecture is made of the three main elements (as shown in figure 7.7.1):

- An access protocol
- A push proxy
- A over-the-air protocol

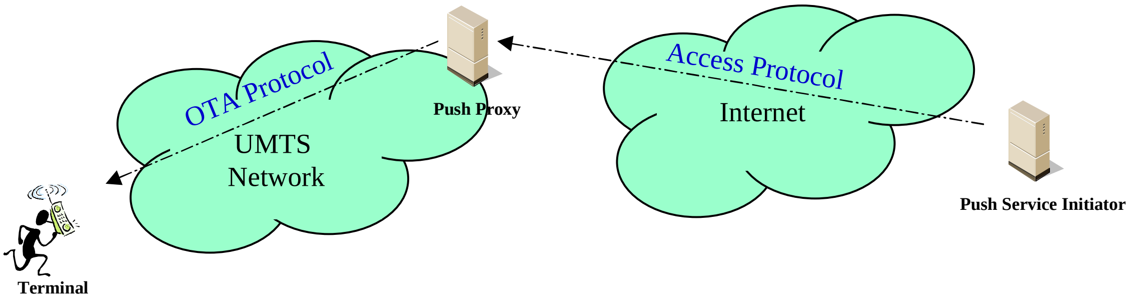


**Figure 7.7.1: Proposed Push Architecture**

The access protocol (between the push service initiator and the push proxy) can be of multiple form. For example it can be:

- HTTP[17] based: the push initiator POST push request to a UE via an HTTP based push proxy
- SMTP[11] based: the push initiator e-mail push request to a UE via an e-mail based push proxy
- WAP PAP[27]: the push initiator uses PAP request to send a push message to a WAP Push proxy

The Over-The-Air (OTA) delivery protocol is HTTP based. The OTA protocol has the following characteristics:

- Optionally uses the HTTP OPTIONS method to authenticate the UE and/or obtain the UE capabilities) on the initial push performed on the TCP connection. UE capabilities are described further in section 7.7.5
- Use the HTTP POST method to deliver content (push message) to an HTTP server based push application.

### 7.7.2 Push Proxy

The Push Proxy (PP) between the push initiator and the UE performs the following functions:

- Asynchronous delivery of messages depending on user availability. The PP manages push messages delivery in accordance with the presence information obtained from other source in the network (Presence Server, HLR, GGSN, SMSC, etc). If the push message intended destination is "present" on the network, the PP then simply forward the message using the OTA protocol specified in section 7.7.4. If the push message intended destination is "not present" on the network, then the PP might:
  - Wait until a presence notification is received and then simply forward the message using the OTA protocol specified in section 7.7.4; or
  - Trigger the intended destination to establish IP connectivity using an SMS message or other available means
- Delivery QoS management: the PP manages the delivery based on QoS indication received from the push initiator. Those are mainly indications on how urgent is it to deliver the message and for how long is the message valid. For example, access protocol such as PAP allows the push initiator to define the delivery timeframe. Similar mechanism can be developed for other access protocol alternatives.
- Service level accounting/charging: The PP support service level billing. This includes but is not limited to:
  - User transaction based billing
  - Push service provider transaction based billing
  - Shared model where the push service provider pays part of the transaction (e.g. advertising)
  - Shared model where the operator shares the access revenue generated by a push service provider with the push service provider (e.g. interest group)
  - Flat billing
- Address resolution: the PP extracts the internal network address of the UE from its external address received in the push message (refer to section 7.7.3 for details).
- Optional push initiator access control: the PP can authenticate the push initiator and filter the incoming push messages.
- Optional content filtering (e.g. malicious content); the PP can filter potentially harmful content (i.e. content that contains viruses). The filter settings can be managed either by the PP owner or by the user himself through the UE capability feature (see section 7.7.5).
- Optional content transformation (based on UE capabilities): the PP can adapt the content to the accepted format (content-type, language, encoding, charset, etc) advertised by the UE in its capability profile.

### 7.7.3 Addressing

The push proxy performs address resolution from the external address to the private network address (if private addressing is used). The external address is not necessarily a network address (depending on the access protocol used). The external address is managed by the PP owner and is known to the push initiator from the push service registration by the user. The external address might contain MSISDN or IP address but should not be required to contain those for privacy reason (e.g. it might not be good, and in some countries it might even be illegal, to distribute MSISDN to push service providers). The PP might resolve the internal address with the support of other systems (Presence server, RADIUS[14], DNS[12], etc). The PP might also trigger the UE to establish a PDP context if the UE is not attached.

## 7.7.4 Push Delivery Mechanism

The push delivery mechanism is based on the well-known HTTP protocol (RFC2616[17]). The push destination application in the UE is an HTTP server based application. Content is sent to that application by POSTing data to the application URI (refer to HTTP).

Once the PP has resolved the internal IP network address of the UE (refer to 7.7.3) the PP:

- Establishes a TCP connection to a well- known port on the UE (port 80 or a new IANA registered port). The TCP connection establishment is a three-way handshake as described in RFC793[10a] and is shown in red in figure 7.7.2 below
- Optionally sends the HTTP OPTIONS method to the UE (including a challenge if UE authentication is required). This is needed if UE authentication and/or UE capability profile query is required. Both proxy authentication (as per RFC2617[18]) and UE authentication (based on RFC2617[18] with slight modifications to allow for server authentication) can be performed during that step. If the UE has a static capability profile known to the PP and if neither the PP or nor the UE authentication are required, that step can be omitted
- Sends the HTTP POST method to the HTTP based push application URI with the push content included in the POST request body
- Recover the HTTP transaction status if required

This sequence described above is illustrated in figure 7.7.2 below.

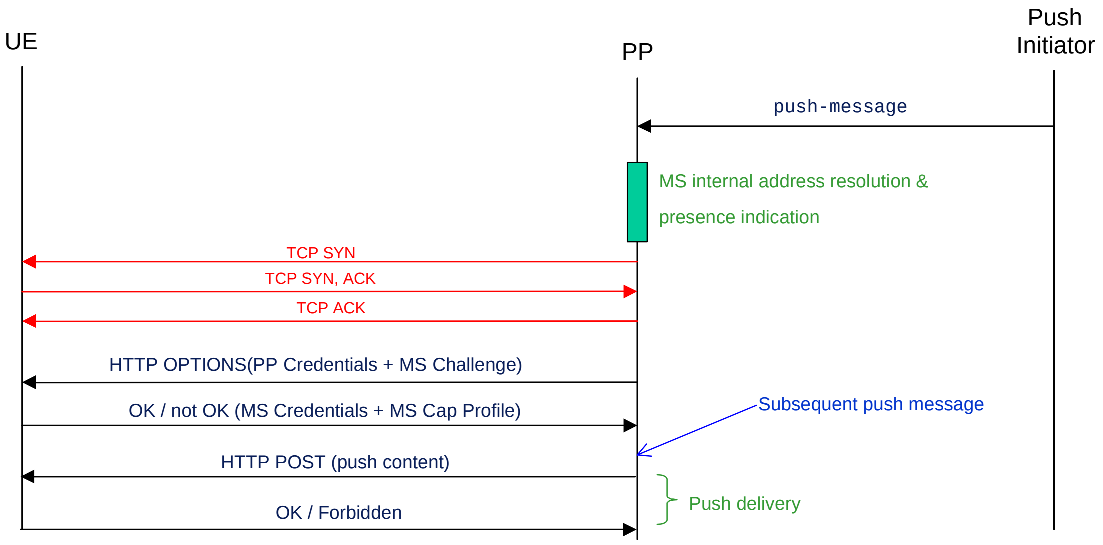


**Figure 7.7.2: Delivery Message Flow**

If the TCP connection is short lived, subsequent push messages require the PP to establish a new TCP connection toward the UE. In that case the authentication/capability query phase (the OPTIONS method) can be omitted only if the PP can assume that its latest authentication/capability data for the UE is still valid (e.g. when capability profile is static and address lease time is long (or permanent)). If the TCP connection is long lived, subsequent push messages are simply forwarded directly using the POST method.

## 7.7.5 UE Capability Profile

The UE capability profile expresses what an UE can do and cannot do. In the wireless world it is very important to know about the UE capabilities before delivering any content due to the broad range of device type (phone, PDA, PC, etc). The capability profile contains various data about the UE such as:

- screen size
- number of pixels supported
- memory size

- receive buffer size
- application contained in the UE
- content-type accepted
- etc

The UE capability profile can also contain the user's preferences to dynamically advertise those preferences to the PP as they are updated by the user (e.g. access control filter setup, malicious content filter setup, etc).

The W3C group have defined a general framework to handle capability profile. This framework is defined in [29]. The CC/PP capability profile is the entity carried in the OPTIONS method response.

### 7.7.6 Roaming Considerations

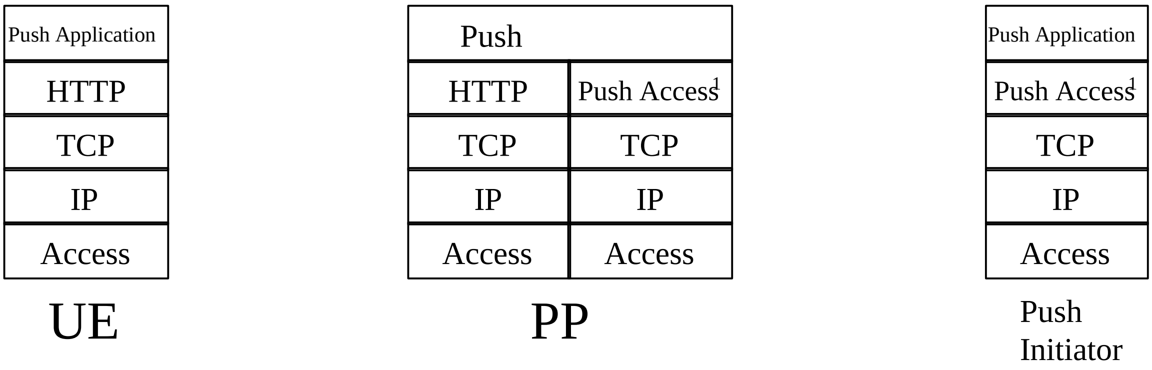
The main roaming model assumes that the UE always reach its home GGSN. The visiting SGSN is connected to the home GGSN using the inter-PLMN backbone. This way roaming is transparent to the push proxy.

### 7.7.7 Delivery Reliability

The Push Proxy manages the reliability of the delivery up to the expiration of the validity period specified for each push message by the push initiator (i.e. the PP will retry message delivery until successful delivery is obtained during that period). Beyond the validity period of the message the push initiator has to resend the message. This offers the flexibility to the application server (Push Initiator) to decide if reliability is controlled by the PP or directly by itself (the architecture supports both).

### 7.7.8 Protocol Architecture

The protocol stack required for push message delivery in this architecture is as shown in figure 7.7.3 below.



1) Push Access: HTTP, SMTP or WAP PAP based

**Figure 7.7.3 Push Message Delivery Protocol Stack**

As described in section 7.7.2 the PP manages push messages delivery in accordance with the presence information obtained from other source in the network (Presence Server, HLR, GGSN, SMSC, etc). If the push message intended destination is "not present" on the network, then the PP might:

- Wait until a presence notification is received and then simply forward the message using the OTA protocol specified in section 7.7.4; or
- Trigger the intended destination to establish IP connectivity using an SMS message or other available means

### 7.7.8 Security Considerations

Due to the nature of a proxy based solution (i.e. the PP functions operate above the transport layer) end-to-end security (between the push service initiator and the UE) is better handle at the application layer. Protocol such as S/MIME (RFC2632 and RFC2633) can be used to achieve this.

Over-the-air protocol security can be achieved at the transport layer by having the UE initiating a TLS handshake upon TCP connection establishment.

The push proxy can perform user protection functionality including malicious content filtering and push initiator access control.

# 8 Conclusion and Recommendations

This technical report has analysed a variety of solutions for the implementation of push services architecture.

The recommended architecture for the push service is a proxy based architecture comprising the following elements:

- Push Access protocol between the Push Initiator (Push Application Server) and the Push Proxy.
- A push transfer protocol handling the push content delivery between the Push Proxy and the UE.
- A Push Proxy that might perform functions such as access control, UE presence handling, store and forward, user profile management, content adaptation, etc.

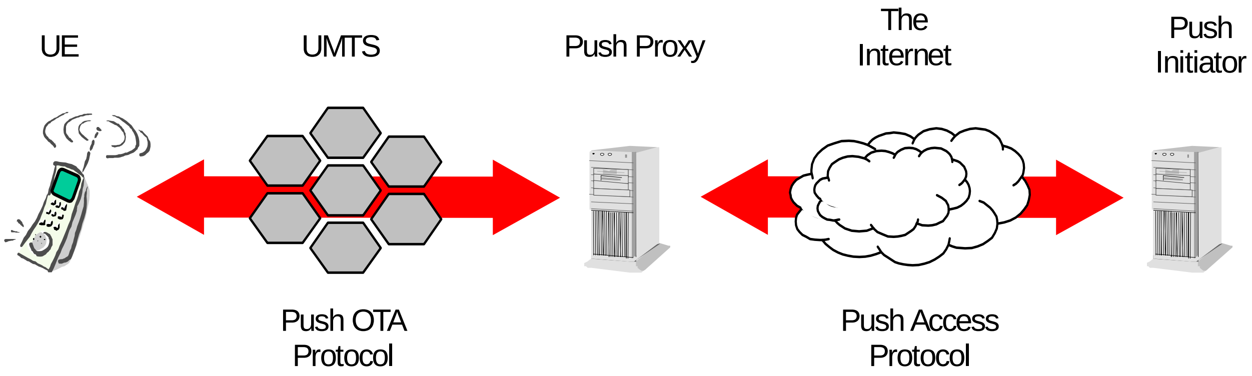


**Figure 8.1: Push Proxy Architecture**

The push service in this architecture is bearer and subsystem independent and available over both the CS and the PS domain.

One solution that has been proposed which satisfies the push proxy architecture is the solution of the WAP Forum. Facilities provided by this solution include those defined in WAP 2.0 specs (WAP-235-pushOTA, WAP-247-PAP, WAP-249-PPGservice, WAP-251-PushMessage, WAP-167-ServiceInd, WAP-168-ServiceLoad, WAP-175-CacheOp), IETF specifications (RFC 2616 – HTTP 1.1, RFC 2617 - HTTP Authentication) and W3-CC/PP – Composite Capability/Preference Profiles.

Additionally the issue of how to establish the bearer has been presented in the document. There are three potential solutions:

1. Long standing PDP context activation – always-on
2. Session initiation using SMS (via the WAP Forum developed Session Initiation Request SIR)
3. Network requested PDP context activation (NRCA) with dynamic IP address allocation

Beyond this, interaction between the architecture above and the architectures/solutions in the IMS will have to be considered, such as SIP signaling as multimedia session establishment for push services.

At this stage it has not been possible to reach a conclusion on which of these three potential solutions for establishment of the bearer should be adopted as stage 1 requirements have not yet been fully defined. This report recommends that the push services work be placed on hold until SA plenary has agreed the stage 1 requirements (expected December 2001). At that stage, work should recommence on evaluation of the architecture and potential solutions against the stage 1 requirements with the objective of agreeing the way forward and, if necessary, producing a stage 2 specification.

Note: Note that collocated SA1 and SA2 meetings are scheduled for January 2002 at which these stage 1 requirements can be discussed in detail.

It must be recognised, however, that there is a strong requirement that one or more viable solutions be found that can be implemented with release 5 timeframe and that meet the business requirements of the operators who wish to deploy push services. It is an open issue as to whether one solution will meet all requirements or whether multiple solutions will need to be standardised. This will be reconsidered once stage 1 requirements are defined.

This report has not considered detailed issues on charging and security; these topics will be referred to the relevant groups within 3GPP for their consideration.

~~[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<br>- No need to be PS attached (less radio signalling e.g. periodic updates, and SGSN capacity needed)<br>- No need for having an active PDP Context (GGSN capacity saved)<br>- Possible during a call<br>- Immediate delivery at switch-on<br>- Reliable<br>- 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)<br>- Supporting 100s of push message per seconds may not be possible<br>- Delays due to signalling<br>- Needs WAP1.2 in the terminal |
| "The Internet way"<br>Push | - Always connected<br>- Minimum delay, i.e. no extra signalling to deliver the push message<br>- Generic, i.e. not bound to a certain access technology<br>- Scaling, the only bottle neck is the radio | - Always PS attached (radio signaling and SGSN capacity)<br>- Always PDP context active (GGSN capacity)<br>- 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)<br>- A lot of traffic makes a lot of HLR interrogations (signalling is comparable to MT SMS)<br>- Delays due to signalling<br>- Needs a new function: GGSN/NA<br>- Needs standardisation work<br>- Mobile capabilities have to be known by the network (i.e. does the terminal support the service)<br>- Needs to support the "Internet Push" when a context is already active<br>- Needs MS supporting NRCA with MSISDN. |

| | | |
|---|---|---|
| | | - Complex |

# Annex B (Informative): A study on how NRCA and "always on" fulfil PUSH service requirements

Note: This annex is a comparison from a single point of view for information. The conclusion given is for the study described in this annex. It is not intended as the conclusion of the whole TR.

## B.1 Introduction

Push service requires support at two levels: a push application (e.g. WAP) and connectivity at UMTS level. This section studies what is the best way to support this connectivity among the two standard ways: "Always on" and NRCA. This section will also discuss where appropriate if NRCA should be used with static IP address (solution standardized) or with MSISDN and dynamic IP addressing (solution under discussion in 3GPP).

Every solution is compared to the requirements, which are affected by the choice of connectivity method.

Note: this section describes only UMTS but the analysis applies also to 2G GPRS.

Note: In 2G GPRS, a different radio signalling is used, but there two a round trip delay over the radio is very time consuming (about 2 seconds), so minimizing the radio signalling is also an issue. 2G GPRS has BSC instead of RNC, GMM standby instead of UMTS MM Idle.

## B.2 Description of the procedures:

### B.2.1 NRCA

The reader is referred to 23.060[4] for details of the different messages. The procedures needed between RNC and MS are depicted on the side. Each of these procedures requires at least two messages over the radio.

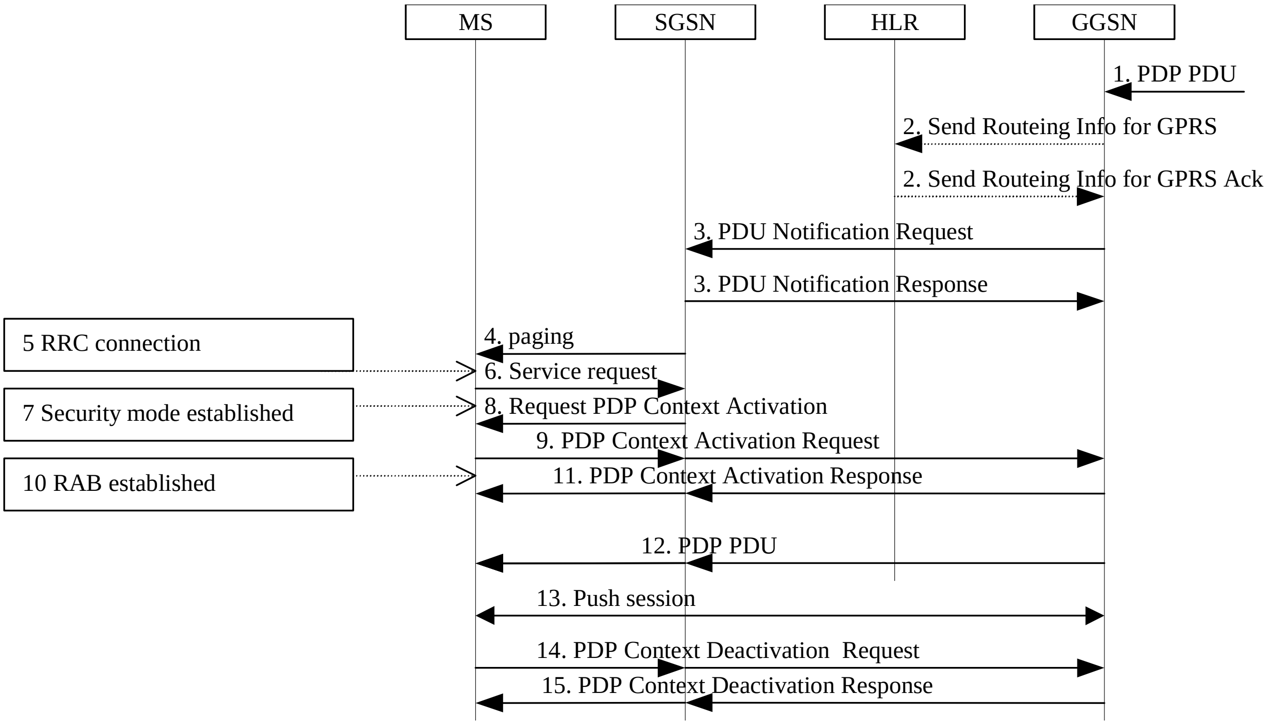

MS
SGSN
HLR
GGSN
1. PDP PDU
2. Send Routeing Info for GPRS
2. Send Routeing Info for GPRS Ack
3. PDU Notification Request
3. PDU Notification Response
5 RRC connection
4. paging
6. Service request
7 Security mode established
8. Request PDP Context Activation
9. PDP Context Activation Request
10 RAB established
11. PDP Context Activation Response
12. PDP PDU
13. Push session
14. PDP Context Deactivation Request
15. PDP Context Deactivation Response

**Figure B.1: delivery of a pushed packet with NRCA**

In this section, The NRCA procedure refers to steps 2,3 and 8. The PDP context activation procedure refers to step 9,10, 11.

The principal of NRCA is to reach an MS who is GPRS attach, but is not having activated a PDP context. This MS must have a static IP address provisioned in HLR. The GGSN must have the mapping of the static IP address to IMSI; i.e. a permanent database needs to be maintained in GGSN.

When the GGSN receive a normal IP packet for one user, it will query HLR to find SGSN address, and then notify the SGSN that one IP packet has been received for this subscriber. SGSN will then ask the MS to activate a PDP context. After the PDP context activation, the GGSN forward the IP packet to the MS.

Note: In one possible implementation, the HLR query can be avoided by storing the old SGSN address and hoping that the MS is still located under the same SGSN. When this is the case the message will be sent straight to the right SGSN, but when the MS has moved, the message will first be sent to a wrong SGSN, and then an HLR query will be performed.

The pushed packet is typically a message with interactive links, which may trigger the user to enter a push session.

Logically, the MS deactivate the PDP context at the end of the push session.

Note: The deactivation is implied by the concept of NRCA. Without a deactivation, the logic would become an always on logic. Note that the network as well may deactivate the PDP context, but the problem is to wait long enough to be sure that the user has finished its session.

In a new proposal being discussed in the 3GPP standard, it is proposed to use NRCA with dynamic address. The mechanism is: an application server query GGSN (or an NRCA node) with a new protocol indicating that it wants to push a message to a certain user identified by MSISDN. The GGSN (or an NRCA node) checks from its permanent database the mapping to IMSI and query the HLR (like in step 2) and then proceed with the NRCA procedure.

It is important to be aware that SGSNs or MS supporting NRCA with static address will not automatically support this new feature as the message do not contain static address.

## B.2.2 Always on

In the always-on concept, the MS is GPRS attached and always has an active PDP context.

Packets received by the GGSN are always forwarded to the MS through the SGSN.

If the MS was MM Idle, a significant amount of signalling is still needed in order to establish secure radio bearer (See figure B.2)

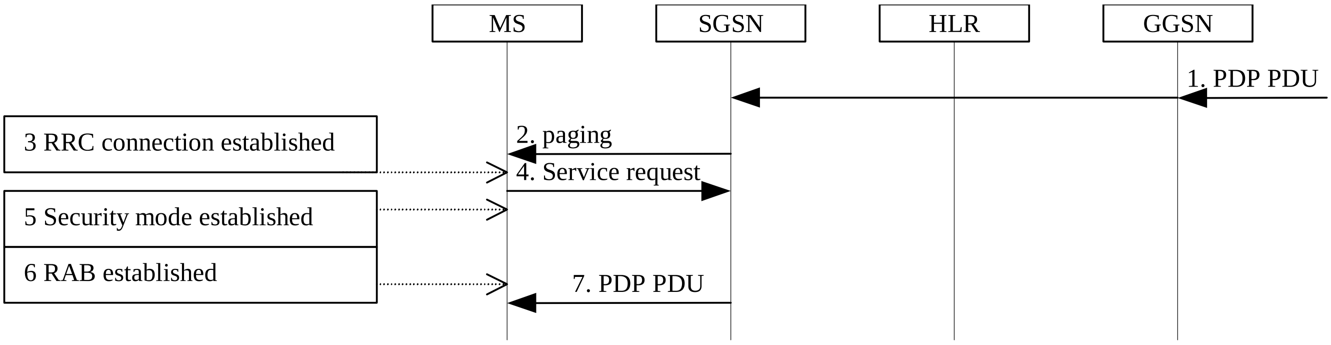


**Figure B.2: delivery of a pushed packet with always-on concept: MS in MM Idle mode**

If the MS was MM Connected, SGSN can forward the packets straight to the RNC that will send it to the MS eventually after paging (See figure B.3).

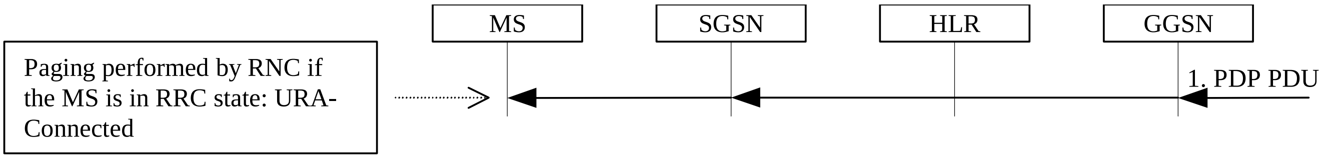

MS
SGSN
HLR
GGSN
Paging performed by RNC if the MS is in RRC state: URA-Connected
1. PDP PDU

**Figure B.3: delivery of a pushed packet with always-on concept: MS in MM Connected mode**

# B.3 Scalability, or supporting burst of push messages during busy hour.

An important requirement is that the network can deliver high peak of push traffic.

## B.3.1 NRCA

Network requested PDP Context Activation is creating a lot of signalling during the traffic peak:

- Query to the HLR to find address of SGSN handling the MS (meaning that HLR signalling capacity may become the bottle neck of the system). Note that this query is also made for detached subscriber.
- SGSN signalling: PDP context activation, NRCA procedure (GTP-C PDU notification message & SM Create PDP context request) and PDP context deactivation. Enough SGSN signalling capacity needs to be installed to take care of normal signalling load during busy hours and the burst of signalling created by NRCA-based push services.

Therefore with NRCA the burst of push messages will generate a simultaneous burst of signalling load putting extra strain on the system. The HLR is the most likely bottleneck. SGSN signalling also risks to be saturated especially if the system is already loaded with normal signalling. However, if the operator has installed enough SGSN and HLR signalling capacity to handle burst of push message, the bottleneck of the system will be the RNC.

Note: if CAMEL is used, NRCA also creates a burst of signalling toward SCP.

Example: HLR of 1200 000 subscribers supporting 300 query per seconds. Pushing football result to 300 000 subscribers at 300 message seconds takes 3000 seconds, i.e. over 15 minutes!

## B.3.2 Always-on

As the MS has already an attached PDP context, no query to HLR and no PDP context activation is needed for the reception of the push message. The bottle neck of the system is probably the number of radio connection that the RNC can established (but always on is less demanding on RNC than NRCA as it saves 5 messages over the RNC and the radio). With Always-on, the scalability is improved if the RNC maintains RRC connection during long time.

Note: With NRCA, the deactivation of the PDP context normally triggers the release of the RRC connection

### B.3.3. Conclusion

NRCA is not a scalable solution as it generates a burst of signalling during the burst of push messages, which increase the signalling load all over the system. To reach the same performance than always-on, it would require a very large installed signalling capacity.

## B.4 Delays

### B.4.1 Always-on:

The delay is coming from the time it takes the first push packet to be sent across the network (backbone and radio interface) and the delay creating by paging the MS, and the delay to activate radio access bearer (if the MS was not having an RRC connection).

### B.4.2 NRCA:

The delay is coming from the HLR query, the query to the MS to activate its PDP context and the PDP context activation itself, and the time it takes the first push packet to be sent across the network (backbone and radio interface) and the delay creating by paging the MS, and the delay to activate radio access bearer. In addition, if one of the signalling messages is lost (probable in congestion situation), it will be recovered through retransmission but this will add a delay of the order of seconds.

Note: In one possible implementation, the HLR query can be avoided by storing the old SGSN address and hoping that the MS is still located under the same SGSN. When this is the case the message will be sent straight to the right SGSN, but when the MS has moved, the message will first be sent to a wrong SGSN, and then an HLR query will be performed.

As has been studied, regarding the situation during burst of push messages, the delay may increase very significantly (i.e. over 15 minutes) due to the saturation of e.g. the HLR.

### B.4.3 Conclusion

In summary, NRCA will always have more delays created mainly by 3 message over the radio and one SS7 query, i.e. an extra delay of roughly 500 ms in an empty network compared to always-on MS is MM-Idle mode. The extra delay may be a few seconds compared to always-on MS is MM-Connected mode.

However during burst of push message, the delay may become very significant due to the poor scalability of NRCA.

## B.5 network resources are used as efficiently as possible;

In order to study the usage of network resources (i.e. radio capacity, signalling links, backbone); let's imagine that a typical scenario will be a push user receiving 5 push message per day (or 1 per busy hour).

### B.5.1 Always-on:

The user when turning on its MS activate its push application, which initiate GPRS attach (3 messages over the radio) and PDP context activation (2 messages over the radio). The push application is deactivated when the user turns off his phone in the evening, i.e. the MS perform GPRS detach (2 messages over the radio). These procedures are typically not performed during busy hour (estimation of 0,5 procedures during busy hours). In addition, the MS shall perform the normal MM signalling (i.e. routing area update) during the time it is GPRS attached. The HLR is queried only at attach and inter SGSN RAU. Maintaining the tunnels between SGSN and GGSN is not creating extra signalling (except to update GGSN when SGSN change) or reserving backbone resources. In addition, if the MS is not having its RAB established when receiving the push message the RAB will be established.

Note1: According to GTP specifications[6], Echo request/response are sent per pair of nodes not per user tunnel.

Note2: The introduction of multipoint Iu will make SGSN changes quite rare.

### B.5.2 NRCA:

The user when turning on its MS activate its push application, which initiate GPRS attach (3 messages over the radio). The push application is deactivated when the user turns off his phone in the evening, i.e. the MS perform GPRS detach (2 messages over the radio). These procedures are typically not performed during busy hour. In addition, the MS shall perform the normal MM signalling (i.e. routing area update) during the time it is GPRS attached. The HLR is queried at attach, inter SGSN RAU, and for every NRCA procedure (i.e. 5 times or 1 query per busy hour). For each push message, one NRCA procedure, one PDP context activation and one PDP context deactivation procedure are performed (i.e. 15 procedures or 3 procedures per busy hour) implying signalling over the radio, over the backbone and over SS7 links. In addition, the RAB will need to be established when receiving the push message. Note that if CAMEL is used every PDP context activation/deactivation may trigger a CAMEL message.

In addition, NRCA also use resources for MS not GPRS attached (HLR query and possibly SGSN query).

### B.5.3 Conclusion

In summary, NRCA increase the amount of signalling in the network and in particular the signalling during busy hour (from 0,5 to 3 procedures). Therefore the "always-on" solution use the network resources more efficiently.

## B.6 Minimum investment

### B.6.1 Always-on:

All push service users are GPRS attached and have a non real time PDP context active. The number of GPRS attached customer will indicate the SGSN capacity needed, and the SGSN needs enough capacity to take care of normal signalling during busy hour (i.e. attach, Routing area update; PDP context activation/modification/deactivation, SMS…). The number of PDP contexts indicate the number of GGSN capacity (and one PDP context is always–on for every connected user).

Supporting a large number of GTP tunnels requires only memory from SGSN and GGSN but no new functions.

In SGSN, GTP tunnels require storing parameter related to active PDP context. However, if the MS would be attached without PDP context, the SGSN is still required to storing inactive PDP context (i.e. subscriber data) and MM context. Therefore, the impact on SGSN capacity of having one GTP tunnel per attached user is marginal (compare to having same number of MS attached without active PDP contexts).

Scaling up the capacity of GGSN is not either a technical challenge. We expect that better platform will dramatically increase the number of PDP context supported per GGSN. And adding more GGSN is straight forward way to add GGSN capacity.

Note: No GGSN or SGSN software upgrade are required as these functions are supported in first phase of GPRS.

### B.6.2 NRCA:

All push service users are GPRS attached. PDP context are only activated on per need basis. The number of GPRS attached customer will indicate the SGSN capacity needed. These SGSNs need more signalling capacity to take care of normal signalling during busy hour and signalling created by peak of NRCA-based push services. The number of push service user that the operator wants to be able to reach during peak of traffic indicates the number of PDP contexts that GGSN should support. In addition the operator needs to support a permanent database where the relation between IMSI and static address (or MSISDN). This database can be implemented inside GGSN (making GGSN more complex and so more expensive) or in a new node (NRCA node). The operator also needs to dimension its signalling support to cope with the signalling created by NRCA during the maximum peak of traffic it plans to support.

In addition, when NRCA is used with static IP address (3GPP release 99), GGSN needs to store the packets received during the duration of the Network Requested Context Activation (between a few seconds and 15 minutes with the example of section B.3.1). If the pushed message "Football result" is 1000 octet, and it is sent to 300 000 subscriber, GGSN needs to have 300 000 Mbytes of extra memory to store all these pushed messages. Note that if the GGSN is planning to use Always-on, this amount of memory could have accommodated around 300 000 extra PDP contexts.

Note: This example assumes one GGSN of 300 000 PDP contexts capacity.

When NRCA is used with dynamic address, software upgrade are needed in release 99 SGSN and GGSN.

### B.6.3 Conclusion

In summary, the "always-on" concept needs more GGSN capacity, while the NRCA require a new permanent database, and more signalling capacity (SGSN signalling cards, SS7 links, HLR&SCP signalling capacity) depending of the peak of push messages that the operator plans to support. The scalability expected by the operator and the pricing need to be known to conclude on this point.

## B.7 Minimum operating cost

### B.7.1 Always-on:

The always-on concept does not require special configuration in the operator network. It is enough to send the MSISDN to the appropriate push proxy with e.g. RADIUS[11]. This is a simple configuration of the Access point name. Note that preferably the same access point is used for all operators' service and Internet access.

### B.7.2 NRCA:

NRCA requires maintaining a new permanent database in the system where every push subscriber need to be added. This requires upgrade to the subscriber management system, and more cost when adding a new customer. In general, NRCA is a new function that requires extra cost to maintain and operate.

### B.7.3 Conclusion

In summary, NRCA adds to the operating cost of the operator

To conclude on the cost issue, NRCA has less efficient network usage and more operating cost than always-on. The initial investment is more difficult to estimate as it depends on the peak of push message than the operator wants to support.

Note that we estimate that to be able to support push services to 25% of the subscriber, SGSN signalling capacity needs to double (HLR and SCP capacity are also affected).

## B.8 interoperability and Service availability when roaming

The MS behaviour will create interoperability problems, as NRCA is an option not expected to be supported by all network or MS. It is important to know that an MS roaming will not know if NRCA is supported in the local network or not. In particular it seems that no network manufacturer support NRCA with static addressing in their first release. If NRCA with dynamic addressing is standardized, it will only work if the visited network has implemented this new standard.

### B.8.1 Always-on:

In a normal GPRS network (without Network requested PDP Context Activation option), the GPRS MS is never only GPRS attached (as only SMS can be used). The MS will either be Internet connected (being both attach and having a PDP context active) or not (i.e. GPRS detach).

### B.8.2 NRCA:

In a GPRS network using Network requested PDP Context Activation option, the GPRS MS is only GPRS attached but is not having a PDP context active. This specific behaviour (being attach but without a PDP context) will lead to problems when the MS is used in a PLMN where Network requested PDP Context Activation is not supported.

Note: The user will not receive its push service, but still the MS will use its battery and radio/SGSN capacity of the visited operator, as it is GPRS attached.

### B.8.3 Conclusion

An MS behaving as if NRCA is supported will not receive its push services in a network not supporting Network requested PDP Context Activation.

On the other hand, "Always-on" will work in any GPRS/UMTS visited network (using GGSN in your home network to make sure your push services are unchanged).

Note: However, there might be problem with roaming agreement if the visited operator wants to bill a high price based on time of connection.

In summary, there is no guarantee that push service based on NRCA will work when roaming, due to different support in different operators.

# B.9 Charging

NRCA has often been presented as required due to time based charging. For a Non real time context charging based on time of connection is more similar to charging the time a GSM MS is turned on than charging the time of a modem call. Contrary to a modem call, an active PDP context do not guarantee any available bit rate.

Time based charging encourages the user to disconnect reducing therefore the overall traffic. For example, the user (or an "optimized" application in its MS) may disconnect before reading the push messages. Connecting again if it wants to read another page (click a link), and disconnect immediately when the page is downloaded.

Time based charging might cause also overcharge the user. This may happen for when the user goes out coverage for a certain amount of time (even in the order of 30 minutes) without the network notices it. The worst case is that the user goes out of coverage, deactivates its PDP context but stay GPRS attached, and come back to coverage. The network may deactivate the context only when the MS will detach (possibly hours latter). However, this problem might be reduced with very high quality coverage. Small "holes" might be compensated by applying some kind of "de fault" reduction in user bills based on some estimates of the minutes in percentage the user might find to be out of coverage. Too long PDP context may be considered as error by the billing system. Still obviously, time based charging will never give a very accurate bill.

Note: We have heard that GSM Association BARG has urged operators to charge on volumes, and not (only) on duration.

Another reason why operators want to use time based charging is for compatibility with their existing billing equipment. On the other hand, CG could perform the conversion from octets (volume) to "equivalent time units".

Note that using time-based charging with NRCA is equivalent to using charging of time of activity of the MS with always-on concept, and this could be supported by an appropriate CDR solution.

# B.10 Type of IP addresses used

Sometimes NRCA has been mistaken as a mean to save IP addresses.

## B.10.1 NRCA with static addresses

This solution requires one IP address per subscriber and is the most address consuming solution. It requires using IPv6 or private IPv4 addresses.

### B.10.2 NRCA with dynamic address and MSISDN addressing

This solution requires one IP address per active user and is the least address consuming solution. However the operator needs to have enough IP addresses to serve burst of push messages. In addition, the address is needed during all the duration of the push session and during a certain guard time (maybe 10 minutes to ensure previous TCP connection is stopped). So to be able to serve significant burst of traffic, it also requires using IPv6 or private IPv4 addresses.

### B.10.3 Always-on:

This solution requires one IP address per connected user. It requires using IPv6 or private IPv4 addresses.

### B.10.4 Conclusion

Any scalable push solution requires using IPv6 or private IPv4 addresses. So the number of addresses is not an issue. One operator may have as many private Ipv4 network (each supporting around 17 millions addresses) as they want. IPv6 is recommended as a future proof solution.

Note: In an operator network, different GGSN may be connected to different private IPv4 network providing infinite number of private IPv4 address. It requires that the operator replicates the same services in these different private networks.

## B.11 FINAL Conclusion

Always on concept and NRCA can provide the same services. However, always-on can provide these services faster, cheaper and in a more scalable way.

NRCA gives the illusion than it can save (in particular GGSN investment). We believe that a serious study of the cost of installing enough signalling capacity to cope with burst of push message and the cost of maintaining more complex system would clear this illusion.

# Annex C (Informative): Comparison of NRCA and SMS as a push solution

Note: The conclusion given is conclusion related to this study, not the conclusion for the whole TR.

## C.1 Introduction

NRCA using MSIDN is being proposed to be used by push services. These two are compared below with SMS, which already uses MSISDN addressing. In addition, SMS combined with WAP[18] already support versatile push services. The operator may use to propose service to the user (WAP Service indication), or push information directly (WAP Service Load can automatically trigger PDP context activation from the MS). In addition, it is possible to define special SMS that always reach the MS (even if MS memory is full). In addition, SMS supports features like Store and Forward and broadcasting inherently.

## C.2 MM signalling required

In order to receive SMS you may be GPRS attached or IMSI attached. So for a typical user who want to receive SMS-based push messages and calls, the MS can be only CS attached and not packet attached. This saves all the Routing Area update signalling for the GPRS Mobility management. It also requires less SGSN capacity.

In order to receive NRCA request, you have to be GPRS attached. So for a typical user who want to receive NRCA-based push messages and calls, the MS must be both CS attached and packet attached. Both Location Area and Routing Area update signalling must then be performed.

In summary, in the case of the MS supports CS and the MS rarely attaches GPRS, an SMS based push solution generates less MM signalling and requires less SGSN capacity than an NRCA based solution.

## C.3 Signalling during push delivery

To compare NRCA and SMS as a push solution, we use the example of a football game result. The message is short (fitting in an SMS) and proposing interactive links to get more information or view a video.

The pushed message may be:

> "France beat Brazil 3.0, and become world champion!
>
> - View game summary (3 minutes video)
>
> - Connect French team WWW
>
> - Play Adidas game to win a picture from Zidane"

If NRCA is used the MS needs to activate a PDP context to receive this message; while with SMS it will be received directly. With SMS a PDP context need to be activated if the user click on any on the link (the signalling is shown on figure C.2 below, PDP context activation (10.) being marked as optional using dotted line).

Another difference is that SMS is acknowledged over the radio (figure C.2 message 8.), while request PDP context activation is not.

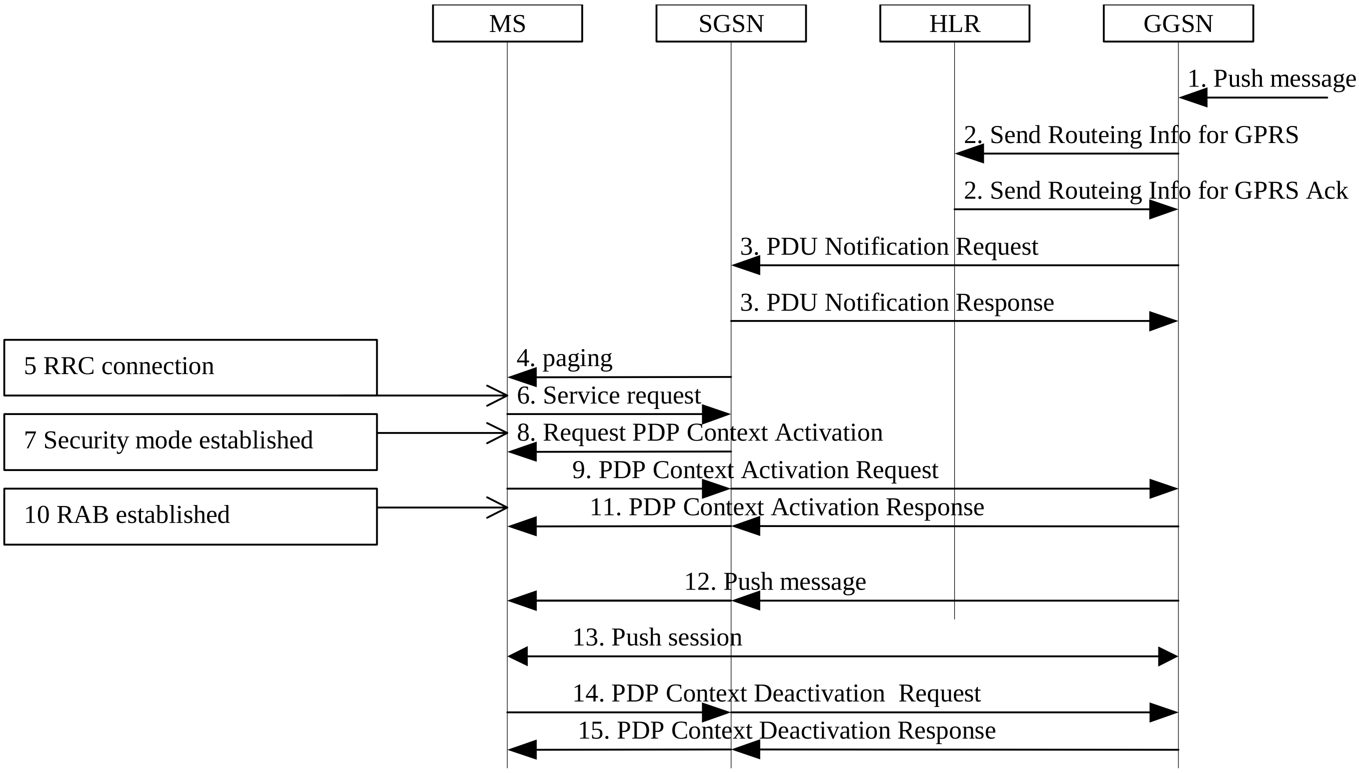

MS
SGSN
HLR
GGSN
1. Push message
2. Send Routeing Info for GPRS
2. Send Routeing Info for GPRS Ack
3. PDU Notification Request
3. PDU Notification Response
5 RRC connection
4. paging
6. Service request
7 Security mode established
8. Request PDP Context Activation
9. PDP Context Activation Request
10 RAB established
11. PDP Context Activation Response
12. Push message
13. Push session
14. PDP Context Deactivation  Request
15. PDP Context Deactivation Response

**Figure C.1: Signalling for push message delivery using NRCA**

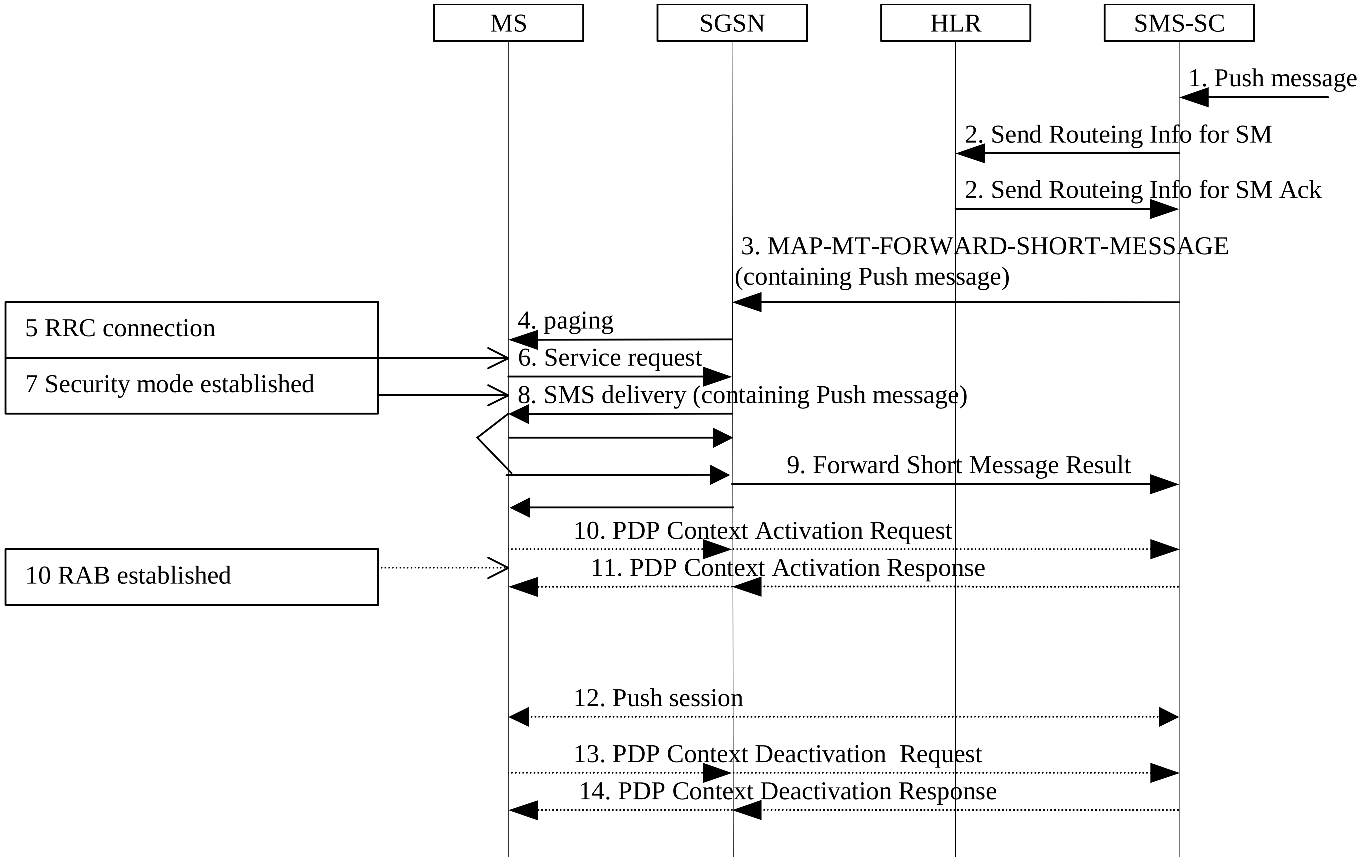

MS
SGSN
HLR
SMS-SC
1. Push message
2. Send Routeing Info for SM
2. Send Routeing Info for SM Ack
3. MAP-MT-FORWARD-SHORT-MESSAGE (containing Push message)
5 RRC connection
7 Security mode established
4. paging
6. Service request
8. SMS delivery (containing Push message)
9. Forward Short Message Result
10. PDP Context Activation Request
10 RAB established
11. PDP Context Activation Response
12. Push session
13. PDP Context Deactivation Request
14. PDP Context Deactivation Response

**Figure C.2: Signalling for push message delivery using SMS**

A comparison of picture 1 and 2 shows that:

- If the user click on one of the links, the amount of signalling over the radio is two messages lessl in NRCA than SMS.
- If the user do not click on any link (i.e. no push session), the amount of signalling is clearly smaller in SMS.

## C.4 Conclusion

Using SMS for push service has following advantages:

- It is already fully standardised (both SMS and WAP1.2 are ready)
- SMS has as an already defined feature store and forward functionality, and may be broadcasted.

# Annex <X>: Change history

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

# 3GPP TR 23.974 V0.2.0 (2001-09)

*Technical Report*

## 3rd Generation Partnership Project;
## Technical Specification Group Services and System Aspects;
## Support of Push service
## (Release 5)

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

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

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 ".

[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] 3GPP TS 24.008: "Mobile radio interface layer 3 specification; Core Network Protocols; Stage 3".

[7] 3GPP TS 29.060: "General Packet Radio Service (GPRS); GPRS Tunnelling Protocol (GTP) across the Gn and Gp Interface ".

[8] 3GPP TS 29.061: "Interworking between the Public land Mobile Network (PLMN) supporting Packet Based Services and Packet Data Network (PDN) ".

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

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

[10a] IETF RFC 793: "Transmission Control Protocol"(STD 7).

[11] IETF RFC 821: " Simple Mail Transfer Protocol"(STD 10).

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

[12a] IETF RFC 1631: "The IP Network Address Translator (NAT)".

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

[13a] IETF RFC 2131: "Dynamic Host Configuration Protocol".

[14] IETF RFC 2138: "Remote Authentication Dial In User Service (RADIUS) ".

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

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

[17] IETF RFC 2616: "Hypertext Transfer Protocol – HTTP/1.1".

[18] IEFT RFC 2617: "HTTP Authentication: Basic and Digest Access Authentication".

[19] IETF RFC 2632: "S/MIME Version 3 Certificate Handling".

[20] IETF RFC 2633: "S/MIME Version 3 Message Specification".

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

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

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

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

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

[26] WAP Forum: "WAP Push Architectural Overview"(1999) URL: http://www.wapforum.org/what/technical.htm

[27] WAP Forum: "WAP Push Access Protocol "(1999).

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

[29] W3C Note: "Composite Capability/Preferences Profiles: A user side framework for content negotiation"(1999) URL: http://www.w3.org/TR/1998/NOTE-CCPP-19990727".

# 3 Definitions, symbols and abbreviations

## 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.

**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:

| | |
|---|---|
| NAT | Network Address Translator |
| NRCA | Network Requested PDP Context Activation |
| OTA | Over The Air delivery protocol |
| PP | Push Proxy |
| 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.

# 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.

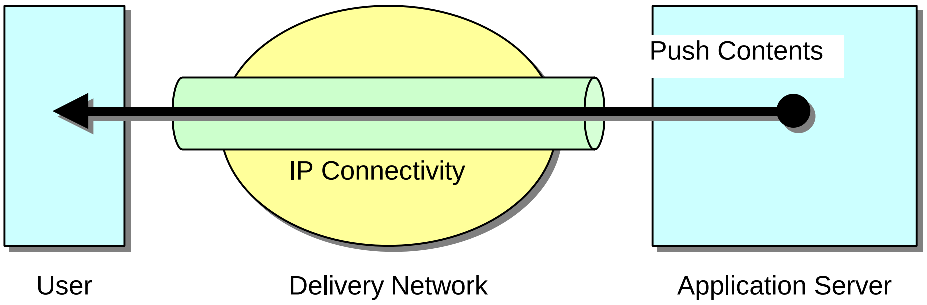


**Figure 6.1: Dedicated Connection Reference Architecture Entities**

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.

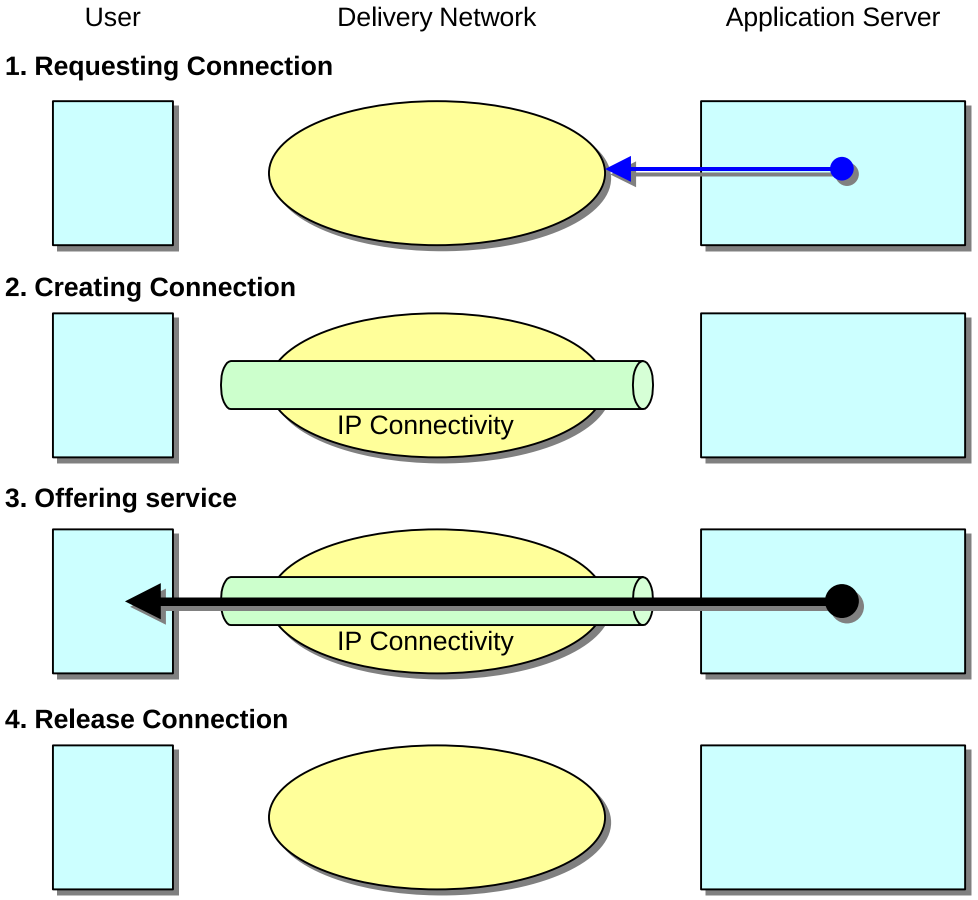


**Figure 6.2: General Service Scenario in Dedicated Connection Approach**

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.

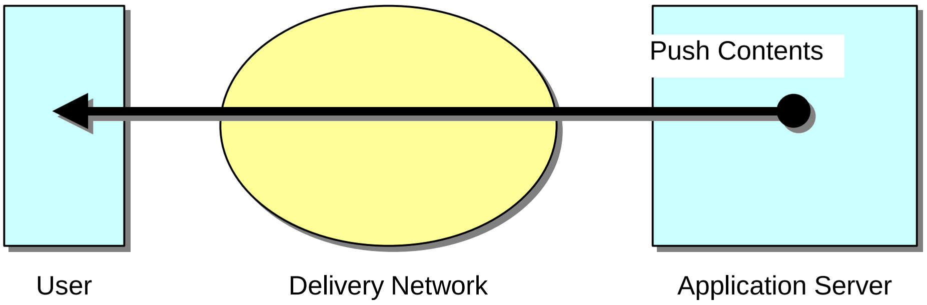


**Figure 6.3: Connectionless Reference Architecture Entities**

Figure 6.4 shows the general service scenario.

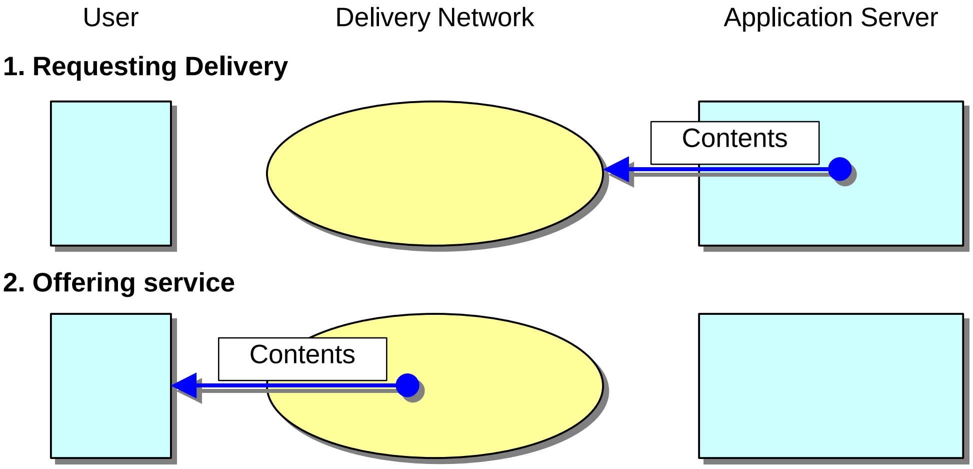


**Figure 6.4: General Service Scenario in Connectionless Approach**

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.

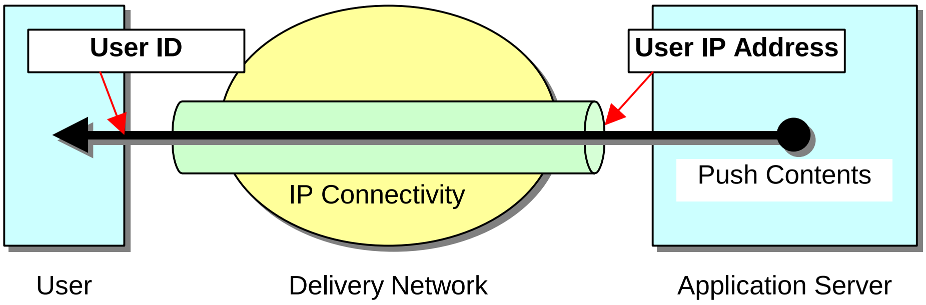


**Figure 6.5: User-ID and User IP Address**

## 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.

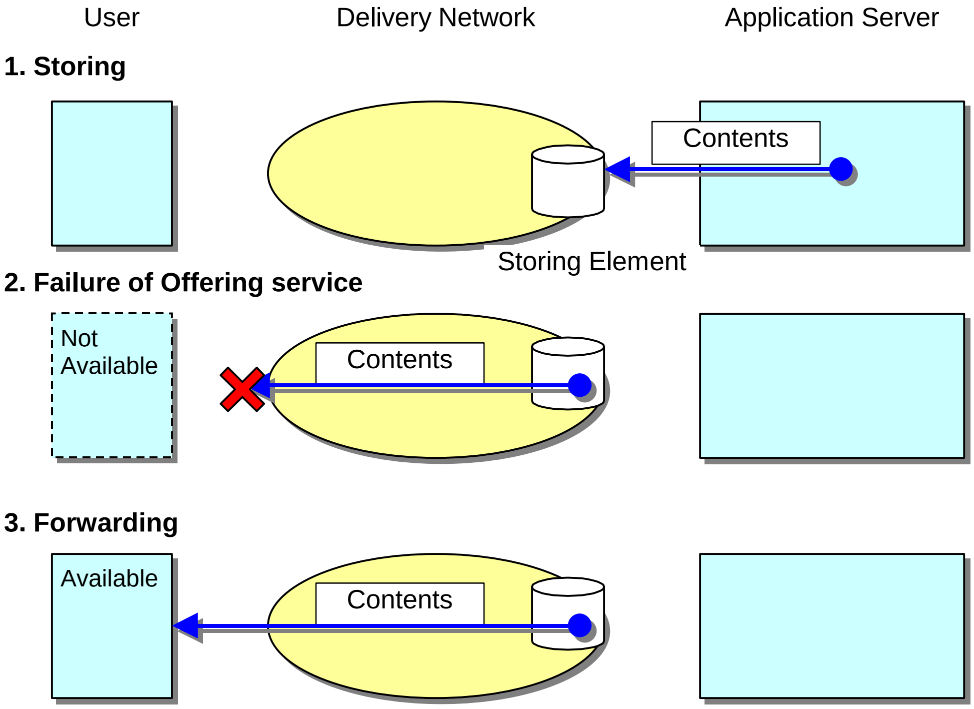


**Figure 6.6: Service Scenario with Store and Forward**

## 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.

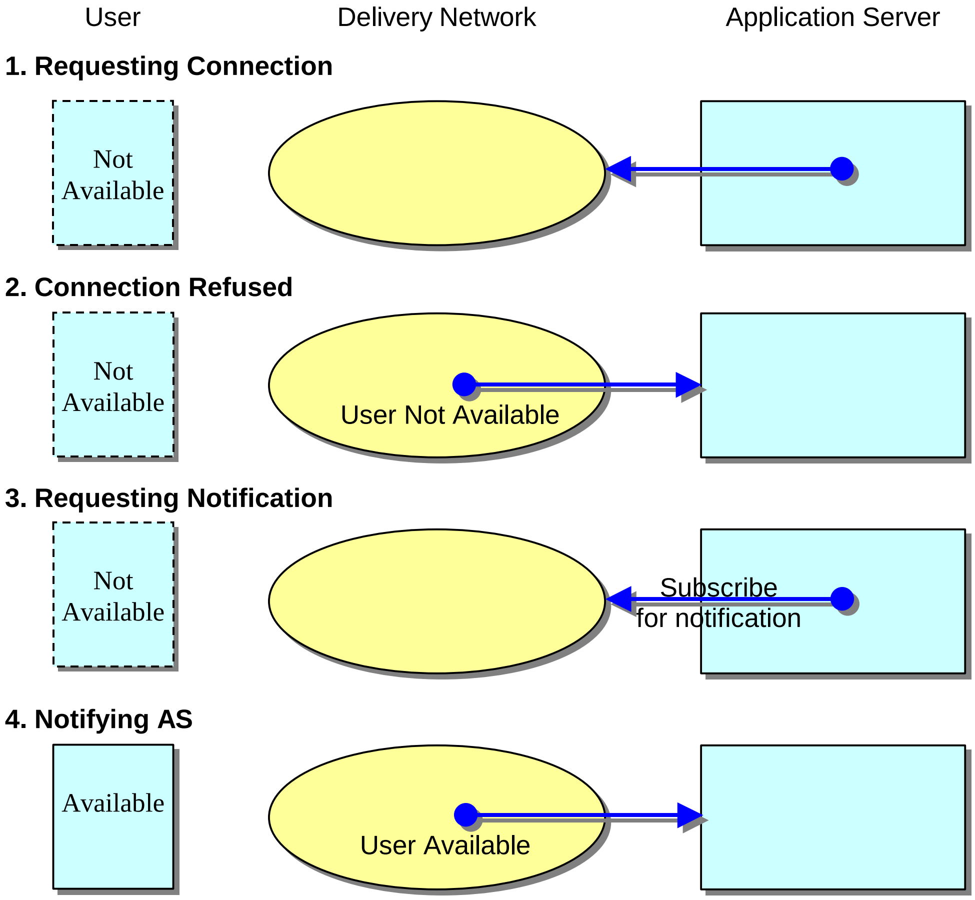


**Figure 6.7: Delivery Network Based Presence Service Scenario**

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.

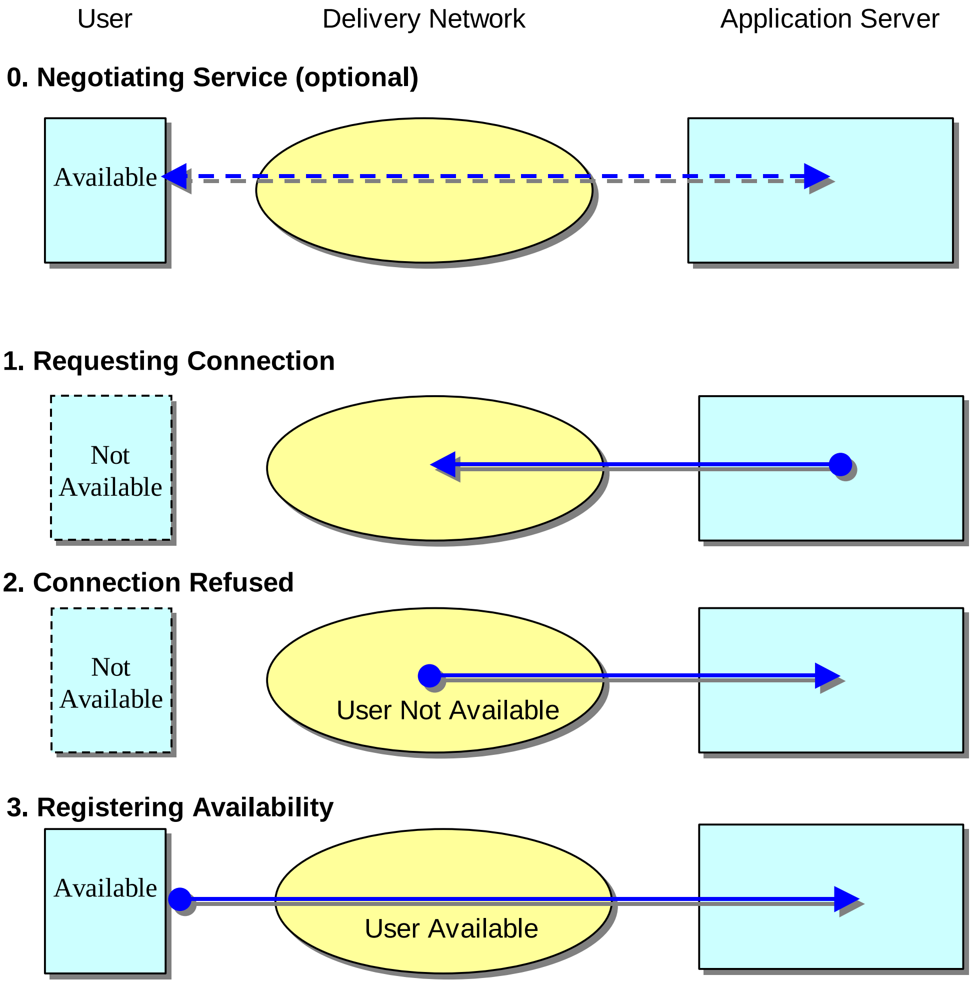


**Figure 6.8: User Based Presence Service Scenario**

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.

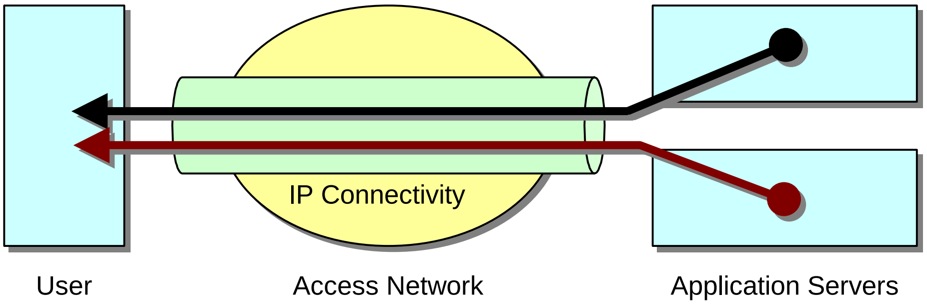


**Figure 6.8: multiple push services over single IP connectivity path**

## 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.

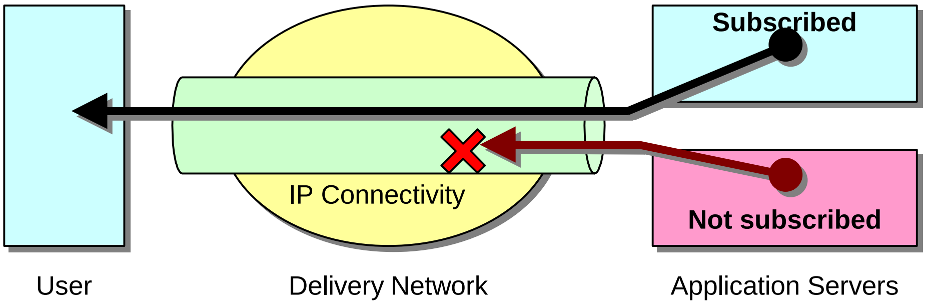


**Figure 6.9: Denial of Service Based on Subscription**

## 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.

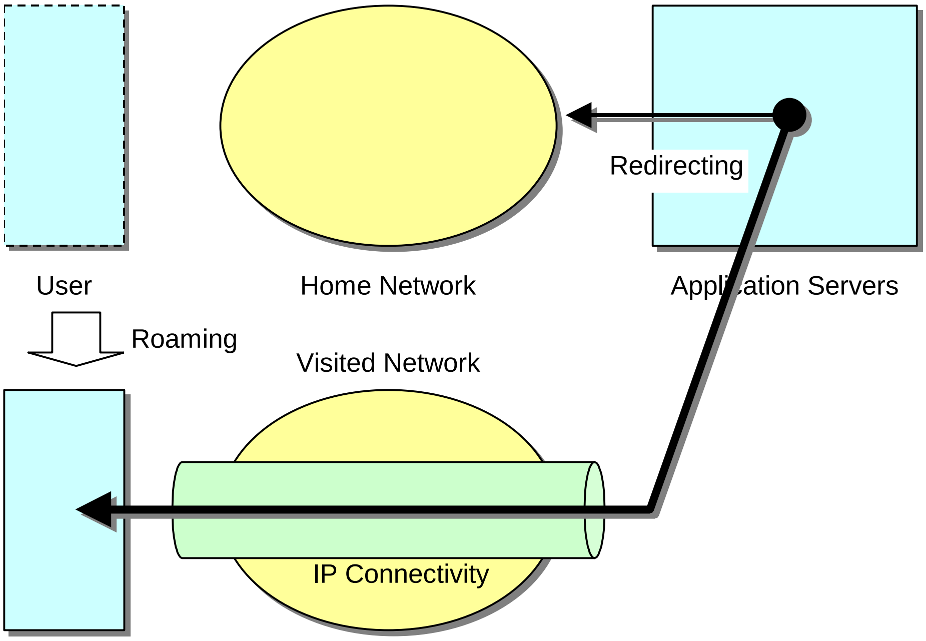


**Figure 6.10: Roaming Support by Redirecting**

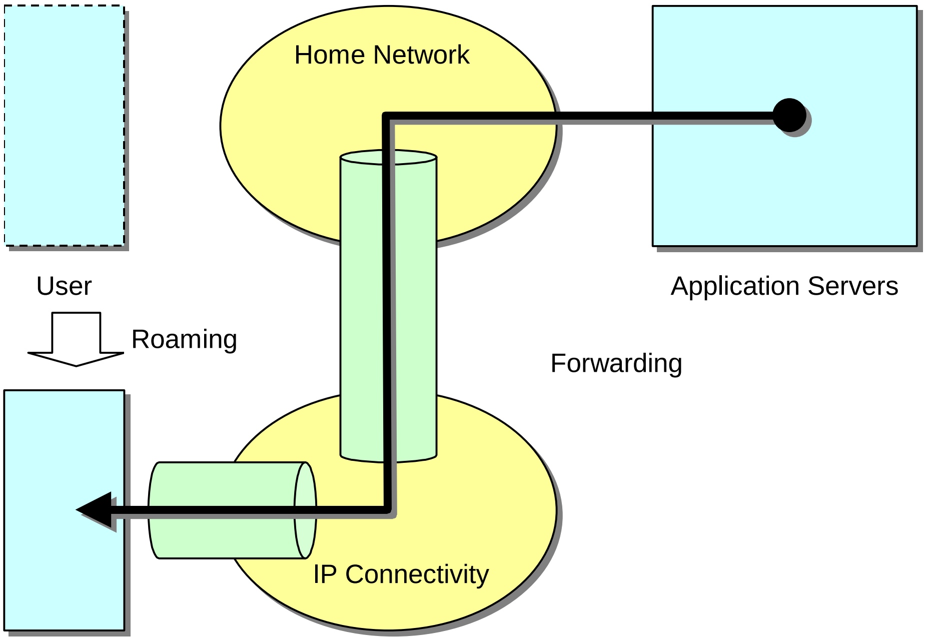


**Figure 6.11: Roaming Support by Forwarding**

# 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

# 7.2 Network requested PDP Context activation with User-ID

## 7.2.1 Functional Architecture for Push Services

The figure 7.2.1 shows a functional architecture for network requested PDP context activation with User-ID for Push Services. The User-ID is, for example, MSIDSN or email address (Bell@lucent.com). The blue arrows and the red arrow in the Figure 7.2.1 represents flow of control messages and of push messages, respectively in dedicated connection mode.

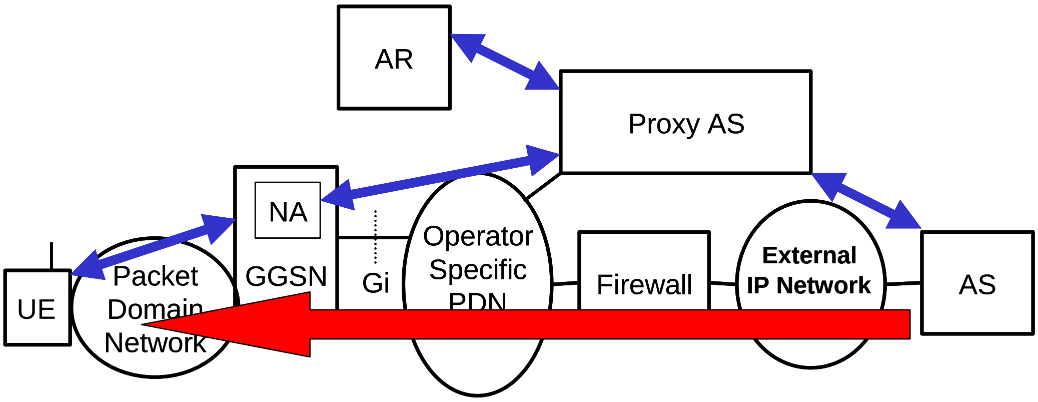


**Figure 7.2.1: Reference Architecture of Network Requested PDP Context Activation for Push Services**

### 7.2.1.1 Application Server (AS)

AS executes push application and resides in operator's network or in the external IP network. When a push message for a recipient identified by User-ID needs to be sent, the AS operating in dedicated connection mode, requests Proxy AS to set up connection to the recipient. When the connection is established by activating PDP context, the Proxy AS provides the AS with the IP address dynamically allocated for the recipient as a response to the request. The AS sends the message to the IP address to deliver it to the recipient. The AS also directs the subsequent push messages for the recipient to the IP address.

The AS operating in connectionless mode, sends a push message containing User-ID to the Proxy AS. To deliver such push message to the recipient identified by the User-ID, the Proxy initiates PDP context activation, obtains a dynamic IP address for the User-ID and sends the message to the IP address. The Proxy directs the subsequent push messages with the User-ID to the IP address.

### 7.2.1.2 Address Resolver (AR)

Address Resolver is a database which provides the following information for Proxy AS.

- Mapping from User-ID to IMSI
- Provision of subscription information of a push subscriber

The subscription information may be used by the Proxy AS to determine the handing of connection request or a push message. For example, it may discard the request or the message to avoid delivering unwanted messages to the subscriber.

### 7.2.1.3 Proxy AS (Proxy Application Server)

The Proxy AS resides in a operator's network and performs interface function between GPRS packet domain and the AS. When the Proxy receives a request or a push message from the AS, it performs the following procedures.

1) Decide how to handle a connection request or a push message based on subscription information of its recipient identified by User-ID. The subscription information is obtained from the AR.
2) Obtain IMSI associated with User-ID by contacting the AR.
3) Determine if PDP context activation is required by checking the internal cache.

4) Trigger PDP context activation by sending Notification Request to GGSN if required.

5) Update the cache according to the result of the step 4.

6) Provide the IP address assigned to the IMSI for the AS in the dedicated connection mode.

7) Relay a push message to the IP address if the Proxy AS is in the connectionless operation.

To share activated PDP context among the application servers, the Proxy AS maintains the internal cache whose record contains User-ID, dynamic IP address allocated to the User-ID, IP address of GGSN holding active PDP context and a list of application services that are currently using the IP address to deliver messages to the User-ID. How the cache is managed is outlined below in the dedicated connection mode. Please note that the sharing mechanism is for push services requiring best effort delivery and that sharing is optional. (The same procedures apply in the connectionless mode as well.)

When a connection request for User-ID is received by the Proxy for the first time, it initiates PDP context activation and creates a cache record that holds the User-ID, the dynamic IP address allocated for the User-ID in the context activation process, the IP address of GGSN that holds the active PDP context and a list contains the AS which sent the connection request to the Proxy.

Suppose now that another connection request for the same User-ID is received from a different AS (2nd AS), the Proxy checks the cache, finds the record with the User-ID and determines to share the available active PDP context with the 2nd AS. The Proxy adds the 2nd AS in the list of application servers in the cache record and returns connection response containing the IP address to the 2nd AS.

Suppose further that the UE deactivates the PDP context, the deactivation is notified to the Proxy AS and the Proxy then informs the Application Servers in the list contained in the cache record, namely 1st AS and the 2nd AS. The two Application Servers stop using the IP address and mark the address invalid, when notified.

The advantage of having Proxy AS in functional architecture is:

1) Support of interworking with different types of Application Servers.
Dedicated connection or connectionless type of application servers are supported.

2) Provision of security measures
The Proxy can share security association with application servers to allow only messages from authenticated sources to be exchanged through the firewall. This protects the network from denial of services attacks. The Proxy also accesses the subscription information of recipients to prevent unwanted message from being delivered.

3) Sharing of available PDP context information
As explained above, by maintaining cache of active PDP context information, the application servers can share active PDP context to deliver push messages. This reduces the signalling within GPRS network for Push Services. To obtain sharing of a PDP context, the context must be created with APNs that are registered in the table <APN, Proxy_AS> defined in clause 7.2.1.4

4) Dynamic GGSN selection
It is possible for the Proxy AS to provide dynamic GGSN selection which may be based on a local load sharing policy.

### 7.2.1.4 Notification Agent (NA)

The NA in GGSN initiates PDP context activation when Notification Request is received from the Proxy AS. The NA maintains a table of record <APN, Proxy AS>. When the NA is informed of the creation of a new PDP context by the GGSN, the NA searches the table to find a record whose APN is equal to the APN of the created context. If the record is found, the NA sends Update message to the Proxy AS specified in the record. For example, if the table contains <Push, Proxy-Push.abc.com> and a PDP context is created with APN Network Identifier of “Push”, Update message is sent to Proxy.abc.com.

#### 7.2.1.5 User Equipments (UE)

In principle, release of the PDP context activated by network should be performed by the UE, because the context may by shared by the other applications on the UE. It may be required to provide a means for coordinating PDP context activation and deactivation among the applications on the UE. Charging aspects related to how to share the cost of the session between the subscriber and the AS are beyond the scope of this document (but it is expected that some form of sharing would be fair if the UE keeps a session active beyond the time a push server is using it).

When the PDP context is deactivated by the UE, the deactivation is notified to the Proxy AS. In the dedicated connection mode, the Proxy also notifies the AS that the IP address for the UE shall be released. The AS stops using the IP address and deletes it on receiving the notification.

#### 7.2.1.6 GPRS network

GRPS network may release a PDP context of the UE for which the radio connection becomes broken, then NA in the GGSN notifies the deactivation to the Proxy AS. The Proxy also informs the AS that the PDP address for the UE shall be released in dedicated connection mode. The AS stops using the IP address and deletes it on receiving the notification.

### 7.2.2 PDP Context Activation with User-ID

In this section, two flow examples are provided to show how PDP context activation is performed with two new messages, Notification Request and Update. The dotted line in the Figures represents a new message or a message which is out of the scope of this contribution.

#### 7.2.2.1 Information Flow Example 1 : Dedicated Connection Approach

The figure 7.2.2 shows network requested PDP context activation in dedicated connection approach.

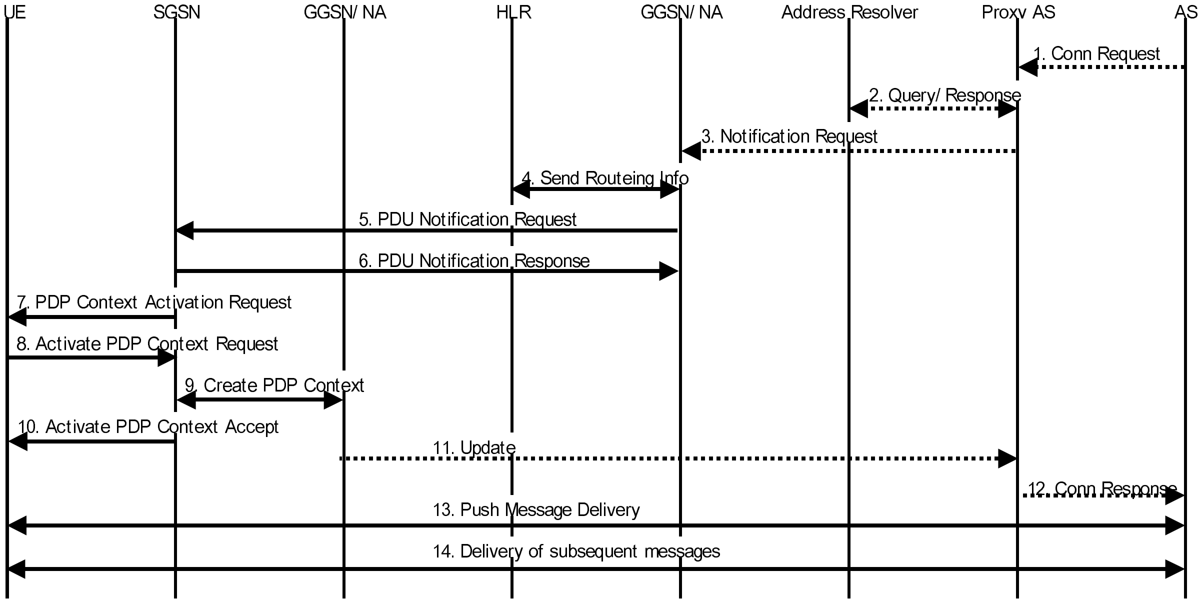

UE
SGSN
GGSN/ NA
HLR
GGSN/ NA
Address Resolver
Proxy AS
AS
1. Conn Request
2. Query/ Response
3. Notification Request
4. Send Routeing Info
5. PDU Notification Request
6. PDU Notification Response
7. PDP Context Activation Request
8. Activate PDP Context Request
9. Create PDP Context
10. Activate PDP Context Accept
11. Update
12. Conn Response
13. Push Message Delivery
14. Delivery of subsequent messages

**Figure 7.2.2: Network Requested PDP Context Activation in dedicated connection approach**

1) The AS sends Connection Request message with User-ID to the Proxy AS. The message is out of the scope of this contribution.

2) The Proxy AS contacts the AR to obtain the IMSI and the subscription information associated with the User-ID. The query and response are out of the scope of the contribution. If the Proxy AS decides to process the request, it checks if an IP address has been allocated to the User-ID by searching the internal cache. In this scenario, there is no IP address is assigned, the Proxy AS, therefore, triggers PDP context activation.

3) The Proxy AS selects one GGSN/NA and sends Notification Request to it. The request may contain IMSI and APN. The criteria for the selection of GGSN may be based on a local load sharing policy.

4) The GGSN obtains routeing information for the UE by issuing Send Routeing Information request to the HLR. The HLR returns the address of SGSN to which the UE is currently attached.

5) The GGSN sends PDU Notification Request to the SGSN identified in procedure 4). Extension to 3GPP TS29.060[6] is required to allow the PDP address in the PDU Notification Request to be set to null to indicate the UE to request dynamic IP address allocation.

6) The GGSN receives a successful PDU Notification Response from the SGSN.

7) The SGSN sends Request PDP Context Activation to UE. The PDP Address is set to null.

8) The UE sends PDP Context Activation Request to the SGSN. The UE requests the assignment of a dynamic IP address by setting the PDP Address to null.

9) The SGSN selects a GGSN and creates PDP context. An IP address is allocated to the UE by the GGSN. The criteria for the selection of GGSN by SGSN, is according to Annex A: APN and GGSN selection in 3GPP TS 23.060[4].

10) On successful creation of the PDP context, the SGSN sends Activate PDP Context Accept to the UE with the assigned IP address.

11) The NA on the GGSN searches the table of <APN, Proxy_AS> for a record whose APN is equal to the APN of the created PDP context. If found, it sends Update message to the Proxy AS specified in the record. The message may contain IMSI and the allocated IP address. The Proxy AS creates the cache record of User-ID, the allocated IP address, the IP address of GGSN/NA and the list containing the IP address of the AS.

12) The Proxy AS sends Connection Response with the User-ID and the allocated IP address to the AS. The message is out of the scope of this contribution.

13) The AS uses the IP address to send the message to the UE. The Proxy AS may not be on the path of dedicated connection from the AS to the UE.

14) The AS can also use the IP address to deliver multiple push messages arrived subsequently for the User-ID.

### 7.2.2.2 Information Flow Example 2: Connectionless Approach

The figure 7.2.3 shows network requested PDP context activation in connectionless approach.

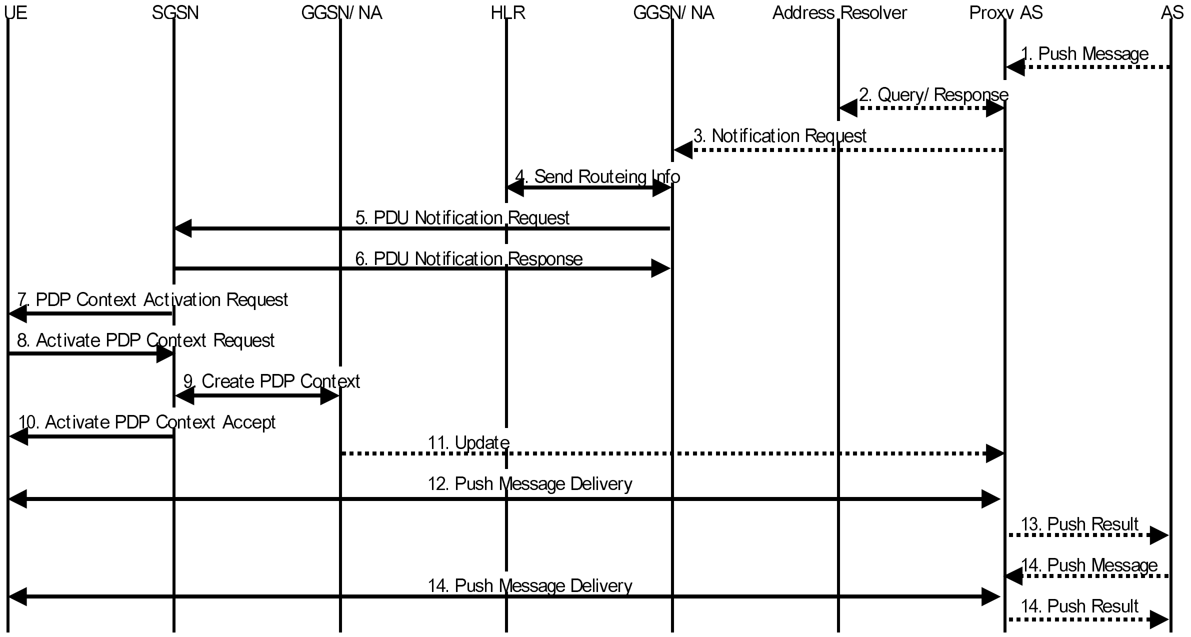


Figure 7.2.3: Network Requested PDP Context Activation in Connectionless approach

1) The AS sends a push message with User-ID to the Proxy AS. The message is out of the scope of this contribution.

2) through 11) Same as in the clause 7.2.2.1.

12) The Proxy AS uses the IP address to send the message to the UE.

13) The Proxy AS sends Push Result to the AS. The message is out of the scope of this contribution.

14) The Proxy AS uses the IP address in the cache to deliver the subsequent push messages for the User-ID.

#### 7.2.2.3 Selection of GGSNs in network requested PDP context activation process

As shown in the Figure 7.2.2 and 7.2.3, there are 2 GGSN/NAs involved in PDP context activation. The Proxy AS selects the first GGSN/NA to which Notification Request is sent. Any GGSN/NA in home network can be selected. The selection may be based on a local load sharing policy.

When SGSN receives Activate PDP Context Request, SGSN selects a GGSN to create PDP context. The selected GGSN (the second GGSN) allocates an IP address and the NA on the GGSN sends Update message containing the IP address to the Proxy AS.

The selection of the second GGSN by SGSN should be according to the ANNEX A in 3GPP TS23.060[4]. An UE activating a PDP context with a dynamic address can be connected to any GGSN supporting a given APN. The GGSN may be selected using a round robin mechanism. Therefore, it is expected that the Proxy AS receives Update message from a GGSN/NA which is different from the one to which Notification Request was sent. The NA on the second GGSN uses the table of <APN, Proxy AS> to route Update messages to the Proxy AS that requested the PDP context activation. The NA searches the table for a record whose APN is equal to the APN of created PDP context. If found, it routes the Update message to the Proxy AS specified in the record.

When it is desired that the first and second GGSN/NA is identical, it can be achieved by two approaches. In the first approach, Proxy_AS and APN are pre-configured such a way that a single GGSN/NA can be selected by both the Proxies and SGSN when PDP context is activated. The configuration sacrifices load balancing capability since GGSN is fixed. In the second approach, PDU Notification Request is extended to carry the IP address of the first GGSN so that SGSN can select the first one when creating PDP context.

#### 7.2.2.4 Roaming Support

The proposed information flow supports roaming services. Assuming the roaming agreement between home and visited network operators, the first GGSN sends PDU Notification Request to the SGSN in the visited network when the UE is roaming. When the second GGSN selection by the ANNEX A in 3GPP TS23.060[4] is applied, the second GGSN could be home or visited depending on the APN in use. A SGSN shall select GGSNs in the home network, because there is currently no standardized means to inform the IP address of the Proxy AS to the GGSN/NA in visited network.

#### 7.2.2.5 How to protect HLR from signalling overload

The problem of using Network Requested Context Activation is that the HLR is heavily involved (when finding the SGSN address). Therefore every GGSN/NA should have a mechanism controlling the load of requests sent to the HLR (or a protocol converting GSN).

### 7.2.3 PDP Context Deactivation requested by the UE

The Figure 7.2.4, shows the information flow where PDP context deactivation is requested by the UE to deallocate dynamic IP address. In principle, release of the PDP context activated by network should be performed by the UE, because the context may by shared by the other applications on the UE. It may be required to provide a means for coordinating PDP context activation and deactivation among the applications on the UE.

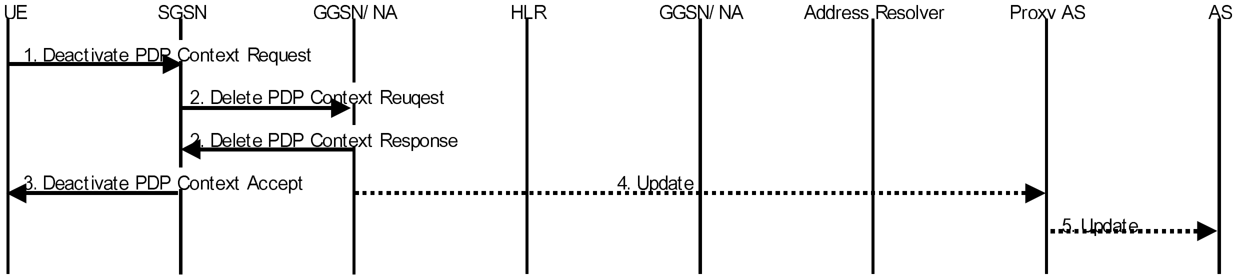

UE
SGSN
GGSN/ NA
HLR
GGSN/ NA
Address Resolver
Proxy AS
AS
1. Deactivate PDP Context Request
2. Delete PDP Context Reuqest
2. Delete PDP Context Response
3. Deactivate PDP Context Accept
4. Update
5. Update

**Figure 7.2.4: PDP Context Deactivation by the UE.**

1) The UE requests PDP context deactivation to the SGSN.

2) The SGSN deletes the PDP context.

3) The SGSN returns Deactivate PDP Context Accept to the UE.

4) The NA on the GGSN searches the table of <APN, Proxy_AS> for a record whose APN is equal to the APN of the deallocated context. If found, the NA sends Update message to the Proxy AS specified in the record to inform that the IP address is no longer valid.

5) The Proxy AS sends Update message to the Application Servers in the list recorded in the cache, and it deletes the cache record. On receiving the Update, the servers stop using the address and delete it.

## 7.2.4 PDP Context Deactivation requested by GPRS network

The information flows in this section show the PDP context deactivation initiated by GPRS network. When the PDP context is deactivated, the GGSN/NA that held the context informs the proxy AS of the deactivation by sending Update message.

### 7.2.4.1 The UE Goes Out of Coverage.

The Figure 7.2.5 shows information flow when the UE goes out of coverage and the SGSN initiates detach procedure after mobile reachable timer expires. In such a case, Delete PDP context request is sent to the GGSN.

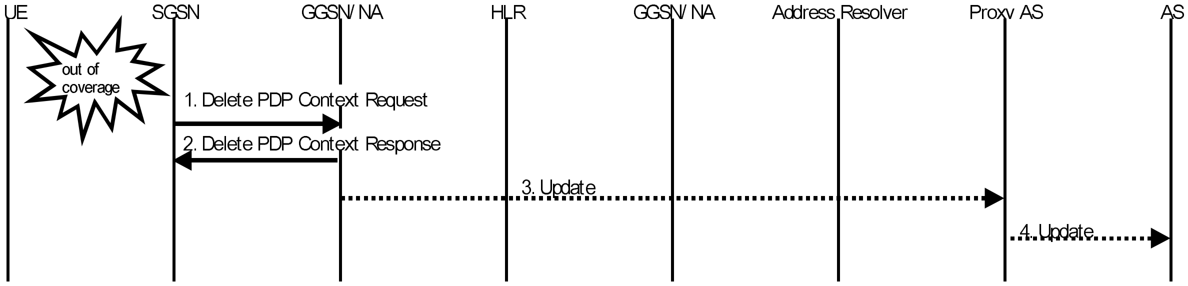


**Figure 7.2.5: PDP Context Deactivation due to Detach Operation**

1) The GGSN receives Delete PDP Context Request.

2) It deletes the PDP context and sends Delete PDP Context Response to the SGSN.

3) The NA in the GGSN searches the table of <APN, Proxy_AS> for a record whose APN is equal to the APN of the deallocated context. If found, the NA sends Update message to the Proxy AS specified in the record to inform that the IP address in no longer valid.

4) If the Proxy is operating in the dedicated connection mode, it informs the application servers that are listed in the cache record containing the IP address that the IP address is no longer valid by sending Update messages to them

and it deletes the cache record. On reception of the Update message, the application servers stop sending push messages and delete the IP address information.

### 7.2.4.2 GGSN initiated PDP context deactivation

The Figure 7.2.6 shows information flow when GGSN initiates PDP context deactivation. GGSN may decide to deactivate PDP context for example, when no traffic to and from the UE is detected within pre-configured time.

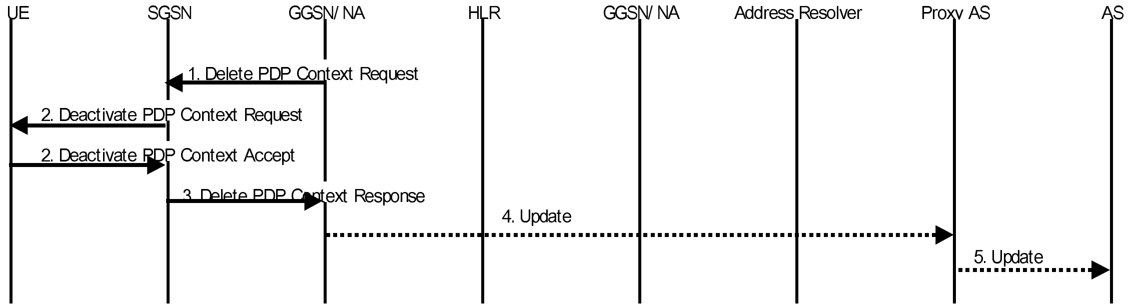

UE
SGSN
GGSN/ NA
HLR
GGSN/ NA
Address Resolver
Proxy AS
AS
1. Delete PDP Context Request
2. Deactivate PDP Context Request
2. Deactivate PDP Context Accept
3. Delete PDP Context Response
4. Update
5. Update

**Figure 7.2.6: GGSN initiated PDP Context Deactivation reported to the Proxy AS**

1) The GGSN sends Delete PDP Context Request to the SGSN.

2) The SGSN sends Deactivate PDP Context Request to the UE and UE accepts.

3) The SGSN sends Delete PDP Context Response back the GGSN.

4) The NA in the GGSN searches the table of <APN, Proxy_AS> for a record whose APN is equal to the APN of the deallocated context. If found, the NA sends Update message to the Proxy AS specified in the record to inform that the IP address is no longer valid.

5) If the Proxy is operating in the dedicated connection mode, it informs the application servers that are listed in the cache record containing the IP address that the IP address is no longer valid by sending Update messages to them, and it deletes the cache record. On reception of the Update message, the application servers stop sending push messages and delete the IP address information.

## 7.2.5 Sharing User activated PDP context with Push Services

This section studies the use of Proxy AS to enable the sharing of PDP context activated by users with Push Services.

When a user activates PDP context with an APN whose value is in a record of the table <APN, Proxy_AS> in GGSN/NA, information including the IP address and the address of GGSN that holds the context, is sent in Update message to the Proxy AS specified in the record and is cached in the Proxy AS so that the context can be shared by the application servers. It is assumed that UE can prohibit creating multiple PDP contexts using the same APN. The diagram 7.2.7 shows user activated PDP context is shared by Push Services in dedicated connection approach.

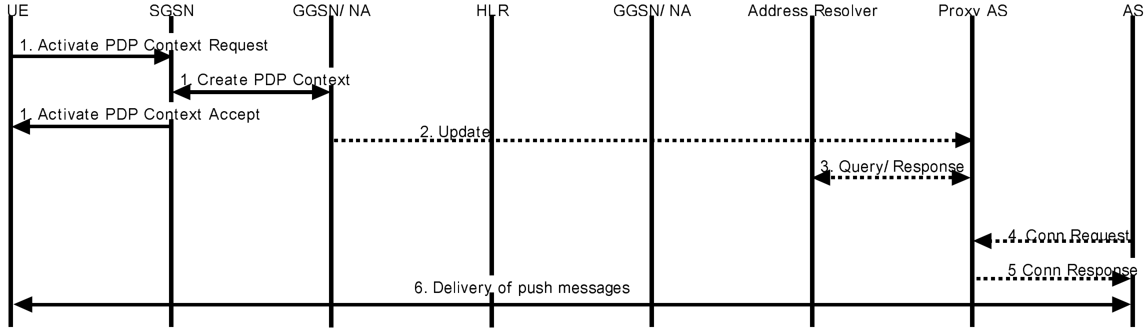


**Figure 7.2.7: Use of user initiated PDP context for Push Services**

1) UE initiates the PDP context activation with an APN whose value is in a record of the table <APN, Proxy_AS> in the GGSN/NA and an IP address is allocated to the UE.

2) The NA on the GGSN searches the table of <APN, Proxy_AS> for a record whose APN is equal to the APN of the created context. If found, the NA sends Update message to the Proxy AS specified in the record. The message may contain IMSI and the allocated IP address.

3) The Proxy AS obtains User-ID associated with the IMSI carried in the Update message, by contacting the AR. The proxy creates a cache record of User-ID, the allocated IP address, and the address of GGSN.

4) Push messages for the User-ID are generated in the AS. The AS sends Connection Request with User-ID to the Proxy AS.

5) The Proxy AS checks if an IP address is allocated to the User-ID by searching the cache. In this case, the Proxy finds a cache record with the User-ID. If the Proxy determines to share the IP address, then it adds the address of the AS in the list of application servers and sends the Connection Response message with the User-ID and the IP address back to the AS.

6) The AS sends push messages for the User-ID to the IP address.

### 7.2.6 Presence Service

Current 3GPP TS23.060[4] defines the Protection and Mobile User Activity procedure in order for the GPRS network to be able to perform some actions to prevent unnecessary enquires to the HLR after an unsuccessful Network-Requested PDP Context Activation procedure. This procedure can be expanded to network requested PDP Context activation with user-ID procedure as follows.

Figure 7.2.8 shows the procedure that the AS to request the presence service to GPRS network.

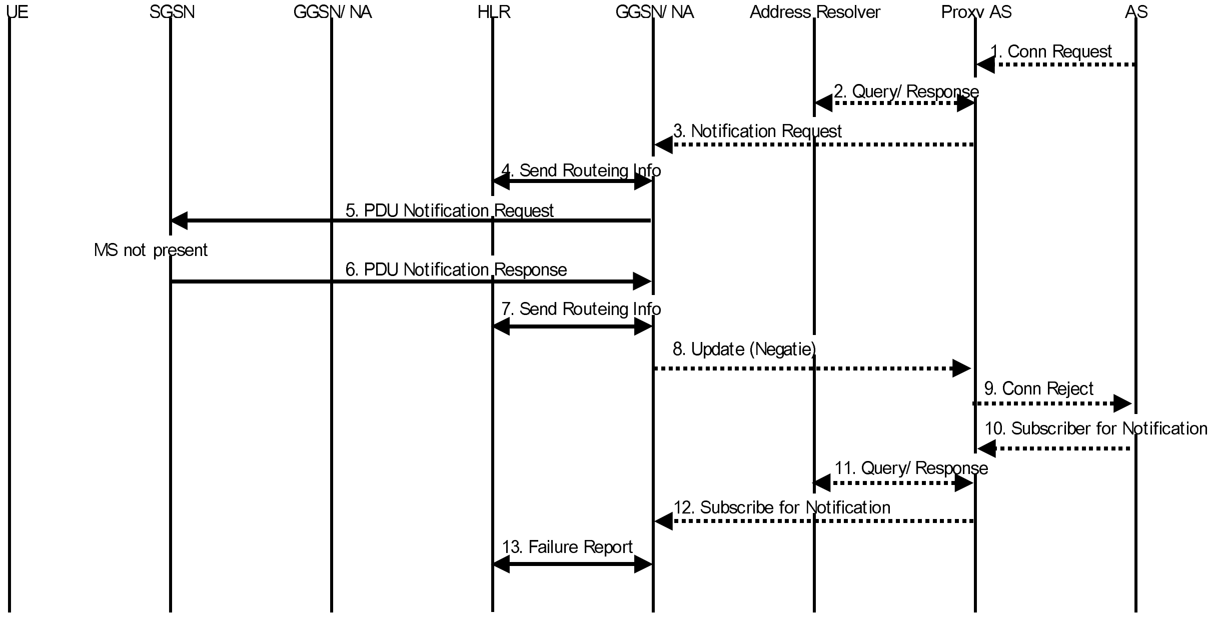


**Figure 7.2.8: Notification Request Procedure**

1) through 5) Same as in the section 7.2.2.1

6) If the MM context of the mobile is IDLE (GSM) or PMM_Detached (UMTS) or if the SGSN has no information about that user, the SGSN returns a PDU Notification Response (Cause) message to the GGSN with Cause equal to "MS GPRS Detached" or "IMSI Not Known". If the SGSN has an MM context for that user, the SGSN sets MNRG to indicate the need to report to the HLR when the next contact with that MS is performed.

7) If Cause equals "IMSI Not Known" the GGSN may send a Send Routeing Information for GPRS message to the HLR. The HLR returns a Send Routeing Information for GPRS Ack message to the GGSN indicating the

address of the SGSN that currently serves the MS. If SGSN Address is different from the one previously stored by the GGSN, then step 5 and 6 in section 7.2.2.1 are followed.

8) If SGSN Address is the same as the one previously stored in the GGSN or if the Cause value returned in step 6 equals "MS GPRS Detached", the NA in the GGSN sends Update message to the Proxy AS to inform the Proxy AS that the IP address cannot be obtained for the user.

9) The Proxy AS informs the AS that the request failed.

10) The AS sends Subscriber for Notification message with User-ID to the Proxy AS. The message is out of the scope of this contribution.

11) The Proxy AS contacts the AR to obtain the IMSI and the subscription information associated with the User-ID.

12) The Proxy AS sends Subscriber for Notification message with IMSI to the GGSN in order for GPRS network to watch the user’s activity.

13) The GGSN/NA sends a Failure Report message to the HLR to request MNRG to be set in the HLR. The HLR sets (if not already set) MNRG for the IMSI and adds GGSN Number and GGSN Address to the list of GGSNs to report to when activity from that IMSI is detected.

The Mobile User Activity procedure in 3GPP TS23.060[4] may be invoked when the GPRS network observes user activity. Consequently the GGSN/NA can inform the AS that the MS is available. The figure 7.2.9 shows the procedure.

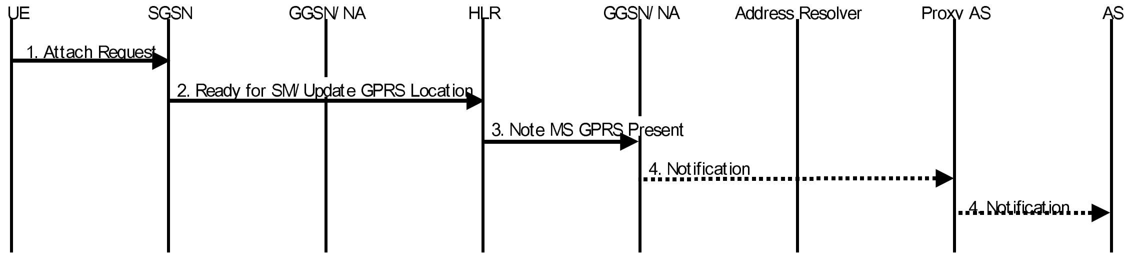


**Figure 7.2.9: notify AS the UE’s availability**

1) The SGSN receives an indication that an UE is reachable, e.g., an Attach Request message from the UE.

2) If the SGSN contains an MM context of the MS and MNRG for that UE is set, the SGSN sends a Ready for SM message to the HLR and clears MNRG for that MS. If the SGSN does not keep the MM context of the UE, the SGSN shall send an Update Location message to the HLR.

3) When the HLR receives the Ready for SM message or the Update Location message for an UE that has MNRG set, it clears MNRG for that UE and sends a Note MS GPRS Present message to the GGSN/NA in the list of the subscriber.

4) The GGSN/NA sends Notification message to the Proxy AS and the Proxy AS relays the message to the AS.

## 7.2.7 Proposed Protocol Architecture

Figure 7.2.10 and 7.2.11 show the protocol architecture for NRPCA. Notification Protocol is used by the Proxy AS to request GGSN/NA to initiate NRPCA. RADIUS accounting protocol is used to update the mapping between the User-ID and IP address maintained in the Proxy AS.

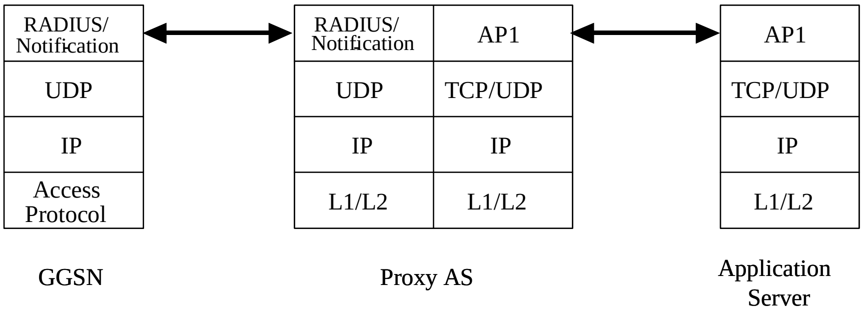


**Figure 7.2.10: Control Protocol for NRPCA**

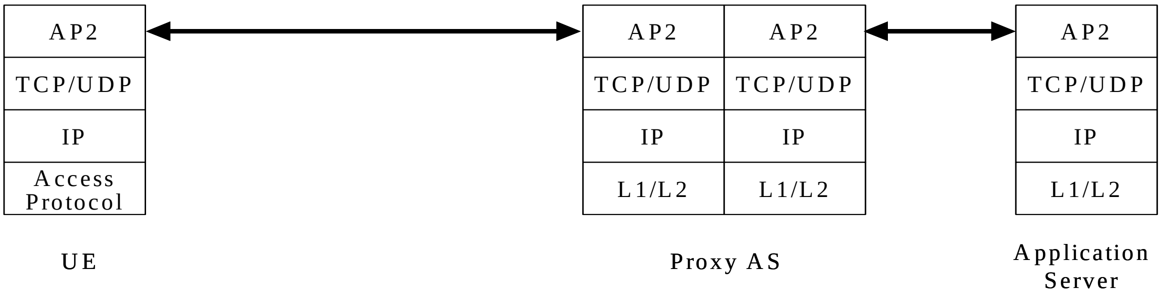


**Figure 7.2.11: Data Protocols for Push Services**

### 7.2.7.1 Notification Protocol

Requesting NRPCA involves exchange of two messages between the Proxy AS and the GGSN/NA as shown in the Figure 7.2.12.

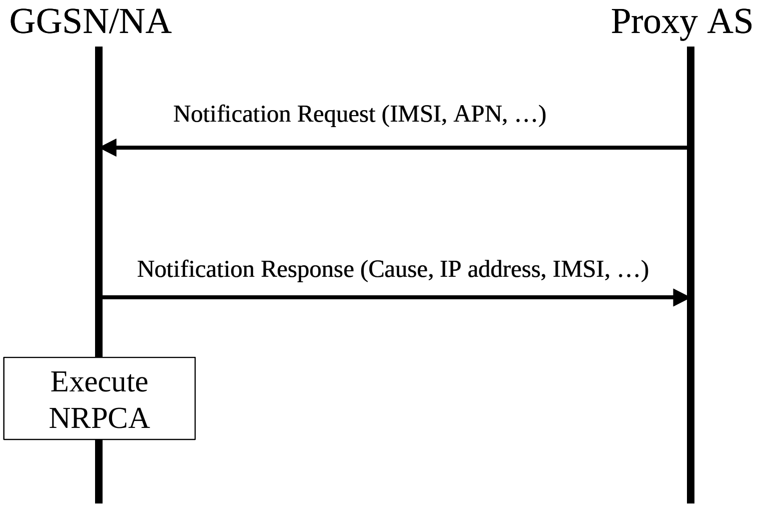

GGSN/NA
Proxy AS
Notification Request (IMSI, APN, …)
Notification Response (Cause, IP address, IMSI, …)
Execute NRPCA

**Figure 7.2.12: Message Flow of Notification Request**

Figure 7.2.13 depicts the proposed header for the Notification protocol. The value for the message type for the two messages, Notification Request and Notification Response is to be determined. Retransmission mechanism defined for GTP-C can be applied for reliability. The port number for this protocol is to be determined and probably has to be registered with IANA. Further protocol enhancements may include error messages, redirect messages, keep alive messages, but this would be for stage 2 and 3 decisions.

Bits

| Octets | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 |
|---|---|---|---|---|---|---|---|---|
| 1 | Version | | | Not used | | | | |
| 2 | Message Type | | | | | | | |
| 3 | Length (1st Octet) | | | | | | | |
| 4 | Length (2nd Octet) | | | | | | | |
| 5 | Sequence Number (1st Octet) | | | | | | | |
| 6 | Sequence Number (2nd Octet) | | | | | | | |
| 7 | Not used | | | | | | | |
| 8 | Not used | | | | | | | |

**Figure 7.2.13: Header for Notification Protocol**

Table 7.2.2 shows the possible information elements (IEs) included in the Notification Request. The IEs in the table are defined in 29.060[4].

**Table 7.2.2: Information Elements for Notification Request**

| Information element | Reference |
|---|---|
| IMSI | 29.060 7.7.2 |
| Access Point Name | 29.060 7.7.30 |
| MSISDN | 29.060 7.7.33 |

Table 7.2.3 shows the information elements (IE) included in the Notification Response. The value for Cause Values is for further study.

**Table 7.2.3: Information Elements for Notification Response**

| Information element | Reference |
|---|---|
| Cause | 29.060 7.7.1 |
| IMSI | 29.060 7.7.30 |
| MSISDN | 29.060 7.7.33 |

### 7.2.7.2 Update Protocol

Notifying IP address allocated by NRPCA procedure or de-allocated IP address to the Proxy AS involves exchange of two messages as shown in Figure 7.2.14. RADIUS Accounting-Request and Accounting-Response are used to update the status of PDP context for Push services to the Proxy AS. When a PDP context for Push services is created by NRPCA, the allocated IP address is notified. When the PDP context is deleted, the deletion (null IP address) is notified.

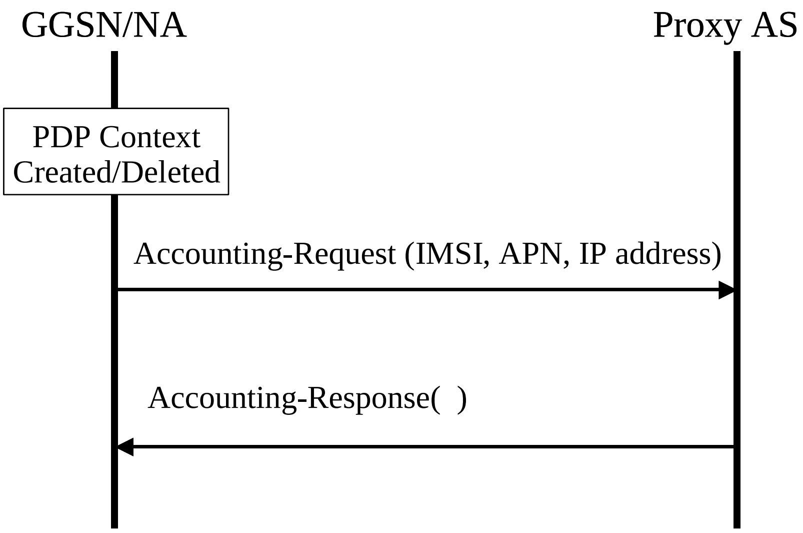

GGSN/NA
Proxy AS
PDP Context Created/Deleted
Accounting-Request (IMSI, APN, IP address)
Accounting-Response( )

**Figure 7.2.14: Message flow of Update**

The attributes that may be carried in the Accounting request are those being defined in 29.061[8]. In particular, the Assigned IP address, IMSI, username and MS-ISDN should be present as appropriate based on the service configuration.

### 7.2.7 Impact to 3G specification

The table below shows impacts to 3GPP specification.

**Table 7.2.1: Impacts to 3GPP specifications.**

| SPEC | Proposed Change |
|---|---|
| 23.060 [4] | Addition of Network Initiated PDP Context Activation with dynamic PDP address<br>Consideration of backward compatibility for UE and SGSN not supporting Network Initiated PDP context Activation with dynamic PDP address. |
| 24.008[6] | Clarification of Network Initiated PDP Context Activation with dynamic PDP address.<br>Consideration of backward compatibility for UE not supporting Network Initiated PDP context Activation with dynamic PDP address. |
| 29.060[7] | Addition of description on usage of End User Address Information Element in PDU Notification Request for requesting dynamic IP address.<br>Consideration of backward compatibility for SGSN not supporting Network Initiated PDP context Activation with dynamic PDP address. |
| 29.061[8] | Addition of the description on Interworking with Proxy AS |

## 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 UE'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 UE'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 UE's IP address if after the addresses TTL expires. If the TTL expires, the application shall perform another DNS lookup to resolve UE's DN to IP address.
- The UE 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 UE 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 UEs.
- 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 UEs 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 UE. For each target UE, the carrier must provision a single PDNS with the UE's DN, its IMSI and the GGSN that is to be used for push services to that UE**.**

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

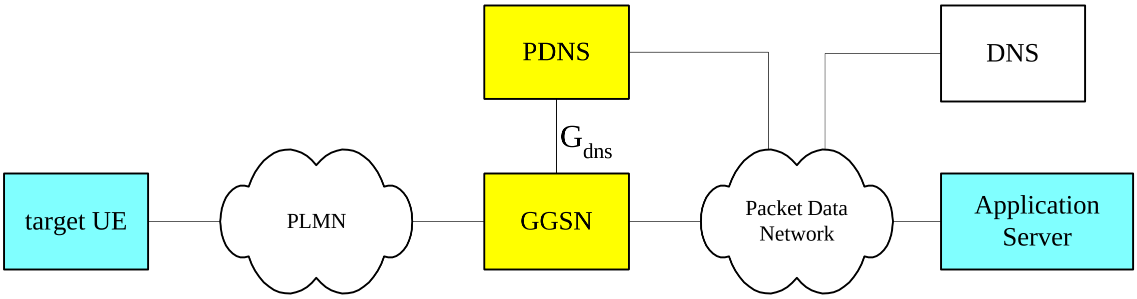


**Figure 7.3.1: Reference Push Service Architecture.**

A DNS is added to the PLMN (the PDNS). Each target UE is represented in the PDNS by a DN. The PDNS is further provisioned with the UE's IMSI and the GGSN to be used for push services to the UE. The Application Server then can query the DNS to resolve the UE'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 UE, 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 UE 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 UE. 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 UE. 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 UE must first query a DNS with the UE's DN.

- When the Application Server queries its local DNS with the UE'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 UE with a statically assigned IP address. For an UE 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 UE'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 UE – if any.

- If the UE 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 UE does not have an active PDP context, the GGSN will use the UE's IMSI to force the UE 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 UE 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 UE 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 UE 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 UE has been resolved.

It may be that the UE 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 UE 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 UE. The PDP context may also be inactivated by the UE, 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 UE 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 UE. 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 UE’s IP address, it can push data to the target UE. Data packets will be routed to the associated GGSN.

- If the GGSN receives an IP packet on the IP address of the target UE and the PDP context still exists, the GGSN will forward the packet to the target UE in the legacy way. The UE 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 UE. In this scenario the UE has neither a statically assigned IP address nor an active PDP context.

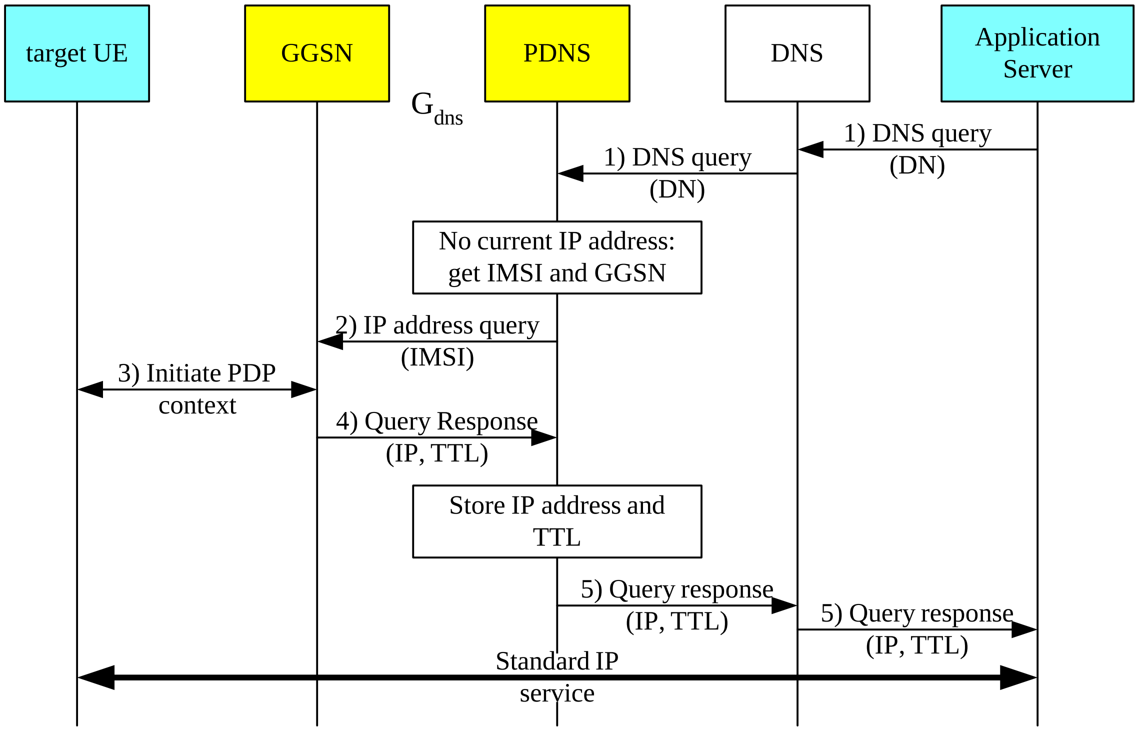

target UE
GGSN
PDNS
DNS
Application Server
G_dns
1) DNS query (DN)
1) DNS query (DN)
No current IP address: get IMSI and GGSN
2) IP address query (IMSI)
3) Initiate PDP context
4) Query Response (IP, TTL)
Store IP address and TTL
5) Query response (IP, TTL)
5) Query response (IP, TTL)
Standard IP service

**Figure 7.3.2: Pushing data to a target UE**

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[12]. The request will be routed to the appropriate PDNS.

2) The PDNS server performs a search to fetch the entry corresponding to the target UE 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 UE's IMSI and GGSN and sends a message over the $G_{dns}$ interface to the GGSN. The parameter in the message is the UE's IMSI.

3) The GGSN, upon receipt of message, will look for the target UE's PDP context information. If no information is found it follows the procedure described in GPRS spec to instruct the target UE (mobile) to activate a PDP context (See 3GPP TS 23.060[4]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[13] 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[12] 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 UE'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.

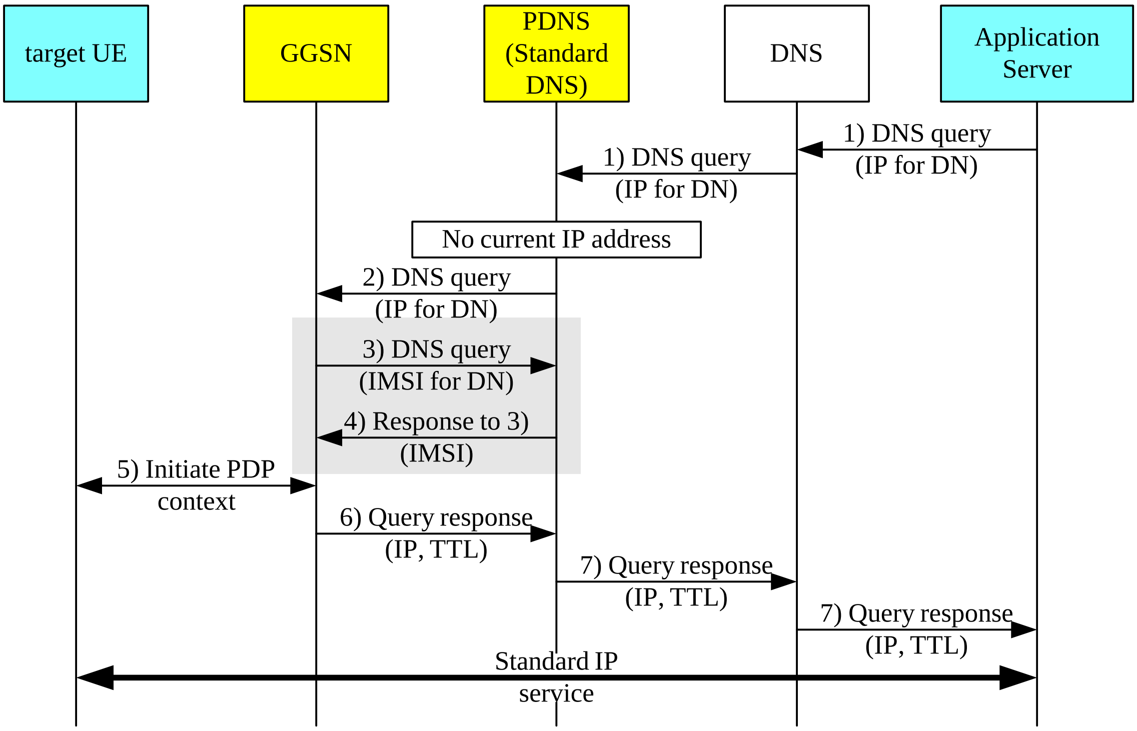


**Figure 7.3.3: PDNS implementation using a standard DNS**

## 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 UE. 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 UE Over The Air.

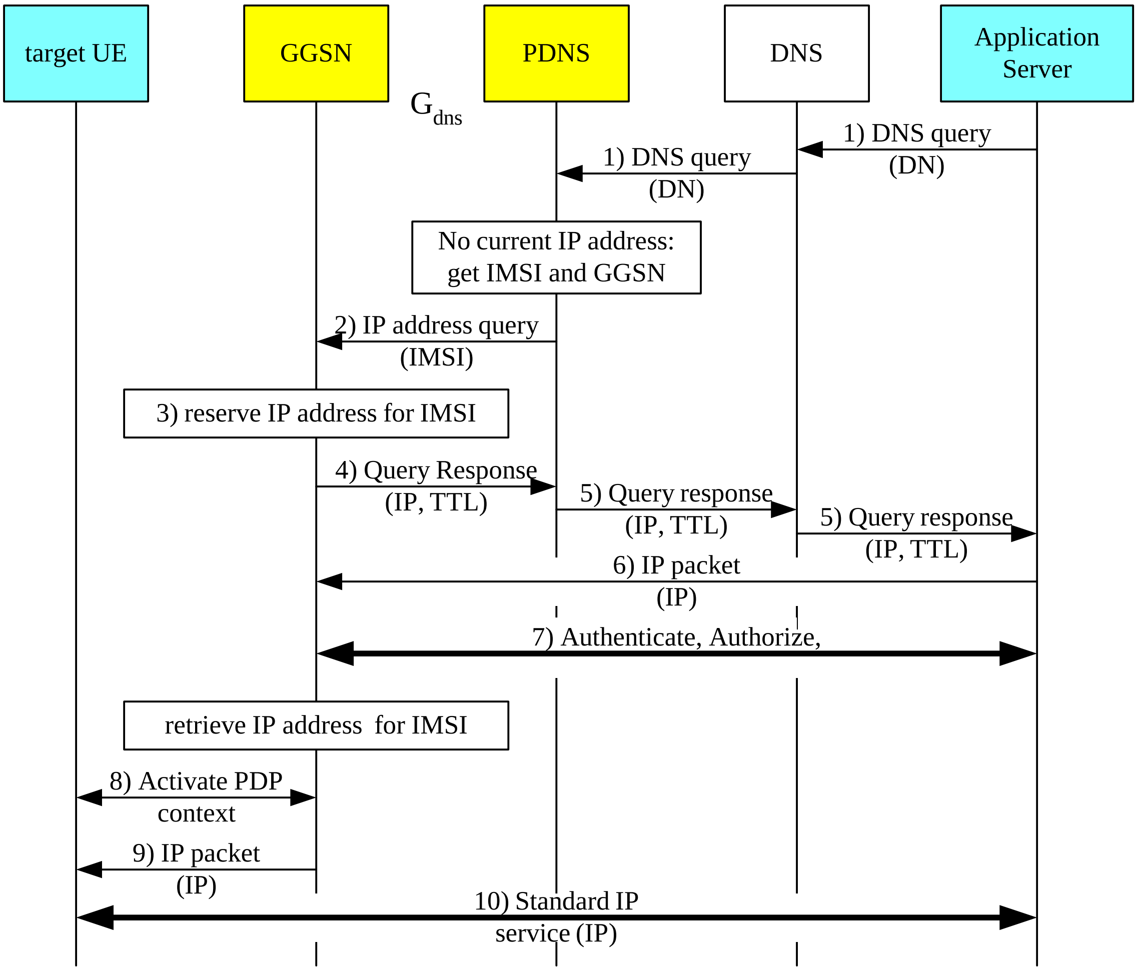

target UE
GGSN
PDNS
DNS
Application Server
G dns
1) DNS query (DN)
1) DNS query (DN)
No current IP address: get IMSI and GGSN
2) IP address query (IMSI)
3) reserve IP address for IMSI
4) Query Response (IP, TTL)
5) Query response (IP, TTL)
5) Query response (IP, TTL)
6) IP packet (IP)
7) Authenticate, Authorize,
retrieve IP address for IMSI
8) Activate PDP context
9) IP packet (IP)
10) Standard IP service (IP)

**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 UE 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 UE does not have an active PDP context, and the UE 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 UE to activate a PDP context, using the UE'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 UE.

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 UE'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 UE with static IP addresses discussed in GSM document (See TS 23.060[4] 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 UEs that use that GGSN as their anchor point. Like a DNS server, the database will contain the UE’s IP addresses and TTLs, while relevant. This solution has the drawback of reduced flexibility; the DN of an UE must resolve onto a specific GGSN.

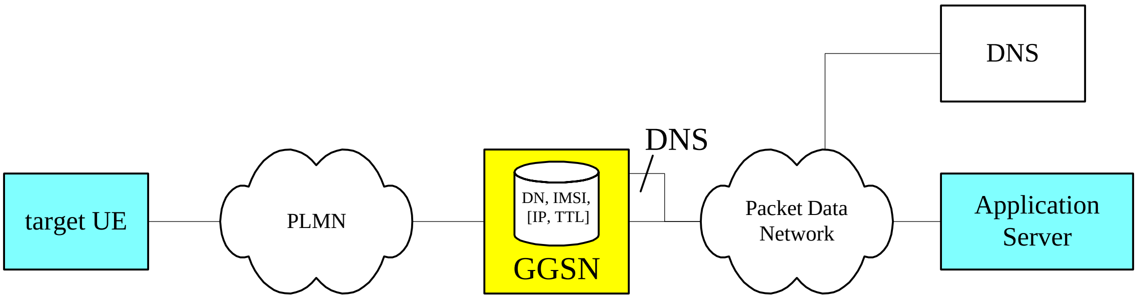


**Figure 7.3.5: GGNS with embedded PDNS database.**

### 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.

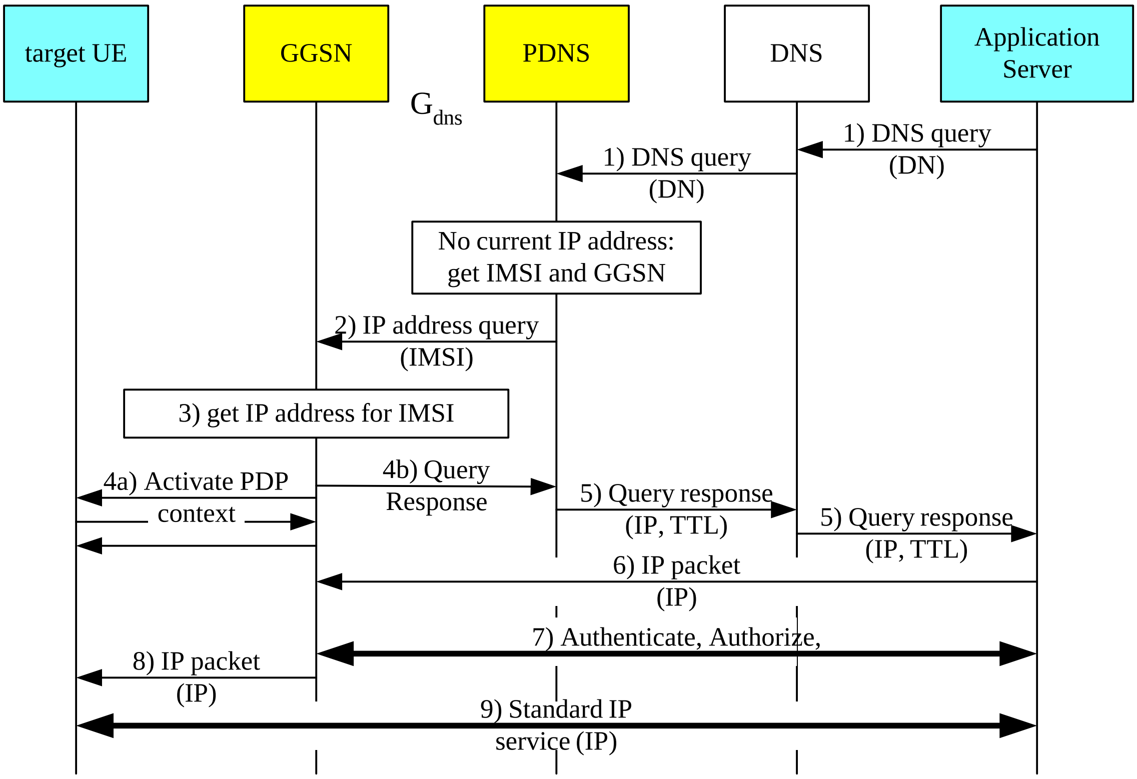

target UE
GGSN
PDNS
DNS
Application Server
$G_{dns}$
1) DNS query (DN)
1) DNS query (DN)
No current IP address: get IMSI and GGSN
2) IP address query (IMSI)
3) get IP address for IMSI
4a) Activate PDP context
4b) Query Response
5) Query response (IP, TTL)
5) Query response (IP, TTL)
6) IP packet (IP)
7) Authenticate, Authorize,
8) IP packet (IP)
9) Standard IP service (IP)

**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.

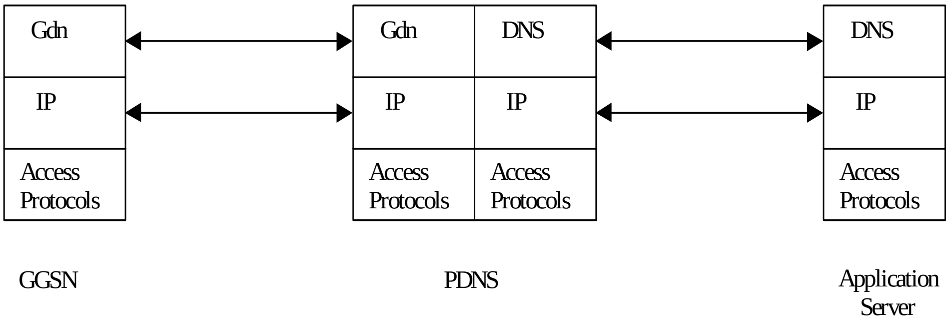


**Figure 7.3.7: Protocol stack for push initialisation**

## 7.3.13 Security

Access network shall protect a target UE 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 UE 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 UE 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 UE is not currently available or failed to establish PDP context)

These error codes are returned to confirm with DNS specification.

## 7.3.16 Impact to 3G specification

The table below shows impacts to 3GPP specifications.

**Table 7.3.1: Impacts to 3GPP specifications.**

| SPEC | Proposed Change |
|---|---|
| 23.060[4] | Addition of Network Initiated PDP Context Activation with dynamic PDP address<br>GGSN to SGSN: PDU notification request with IP address = 0.<br>Addition of a new message pair on Gi:<br>DNS Proxy to GGSN: IP_address_query( IMSI ), and<br>GGSN to DNS Proxy: IP_address_response( IMSI, IP_address, TTL ) |
| 24.008[6] | Clarification of Network Initiated PDP Context Activation with dynamic PDP address |
| 29.060[7] | Clarification to PDU Notification Request that PDP address may not be provided when dynamic address is used. |

# 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.

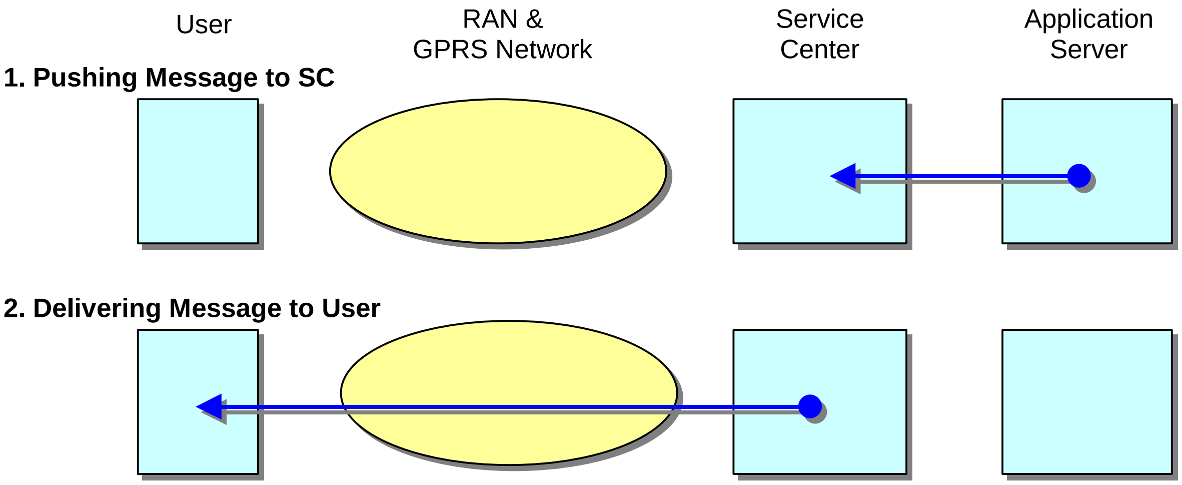


**Figure 7.4.1: SMS Push Message Scenario**

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 3GPP TS 23.040[3]). The interface to the SC from an external Application Server is not standardized by 3GPP (3GPP TR 23.039[2] 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.

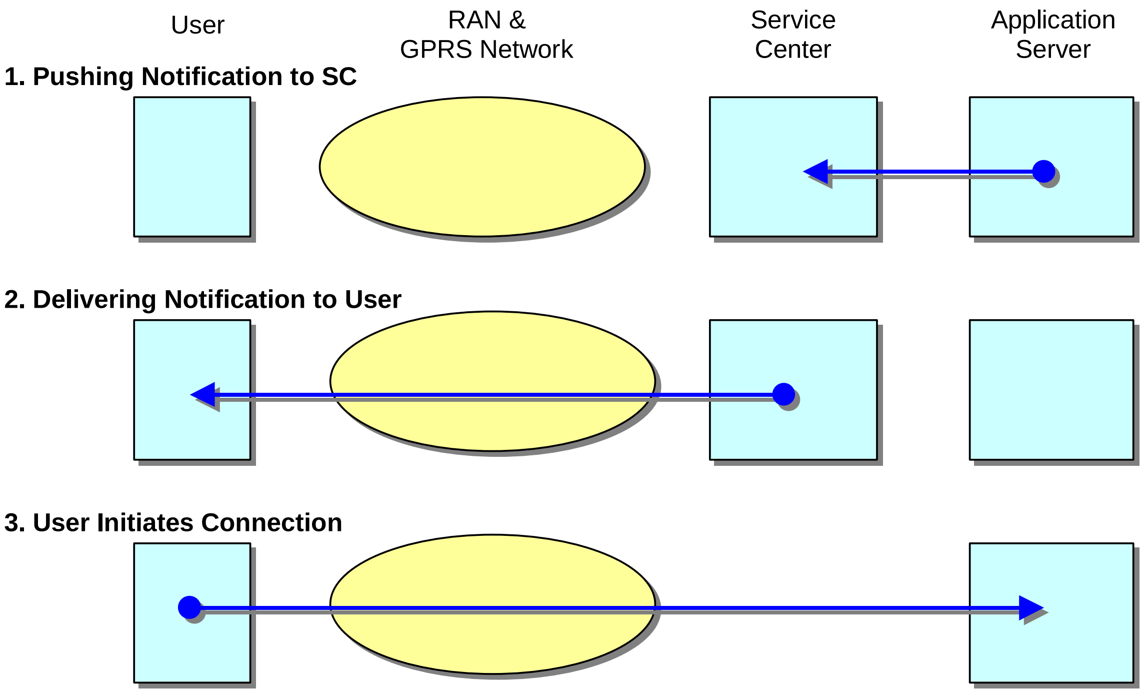

User
RAN &
GPRS Network
Service
Center
Application
Server
1. Pushing Notification to SC
2. Delivering Notification to User
3. User Initiates Connection

**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.

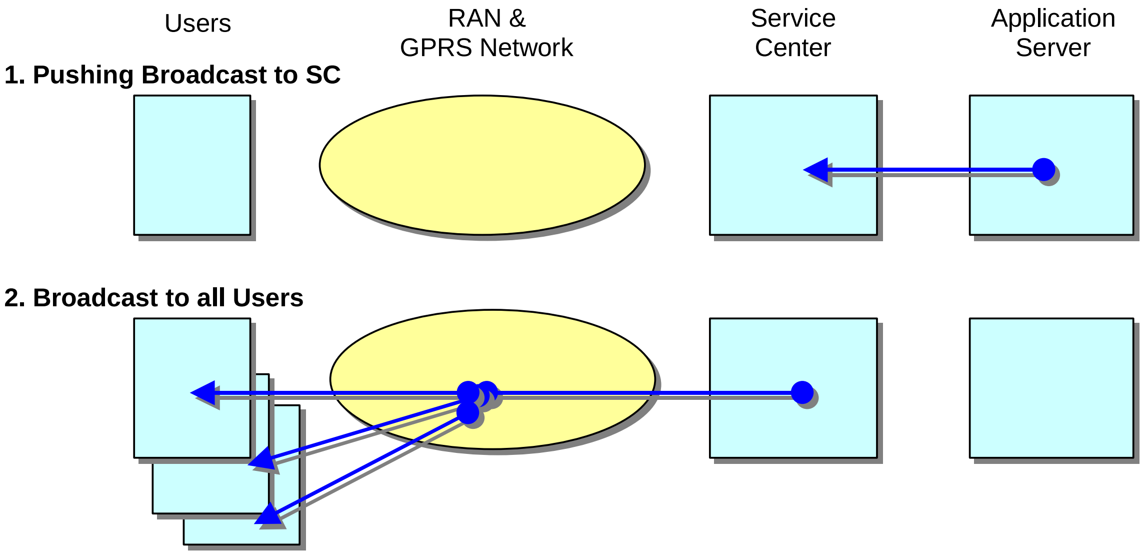


**Figure 7.4.3: Push Broadcast Scenario**

### 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.

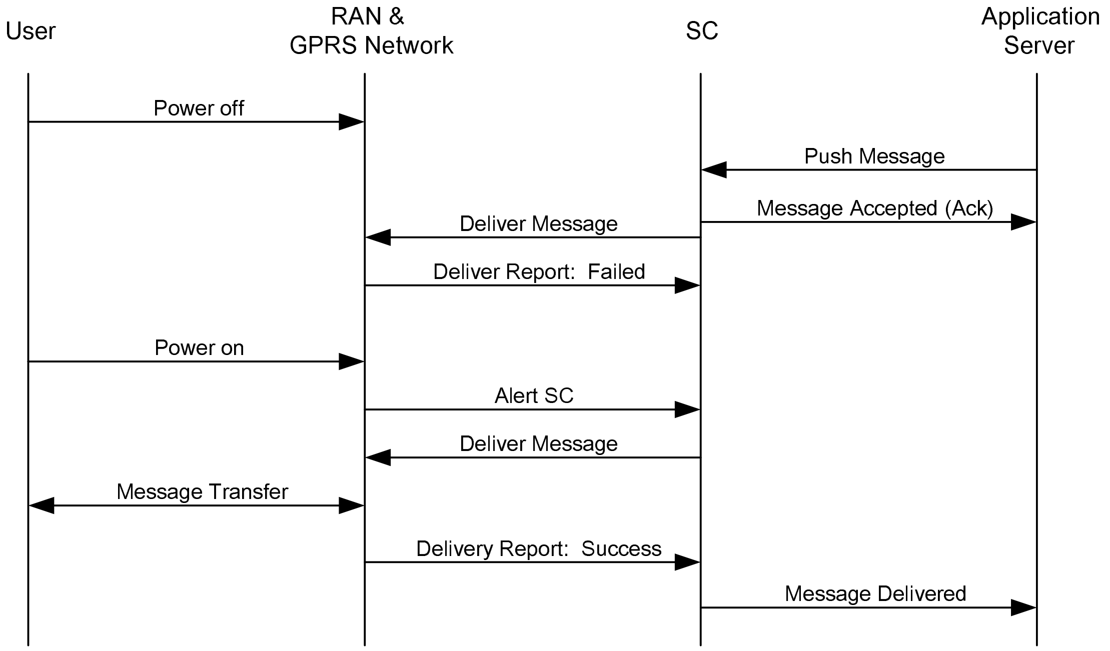


**Figure 7.4.4: SMS Reliable Delivery Sequence**

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[26]) 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 3GPP TS 23.040[3]. SMPP[28] 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).

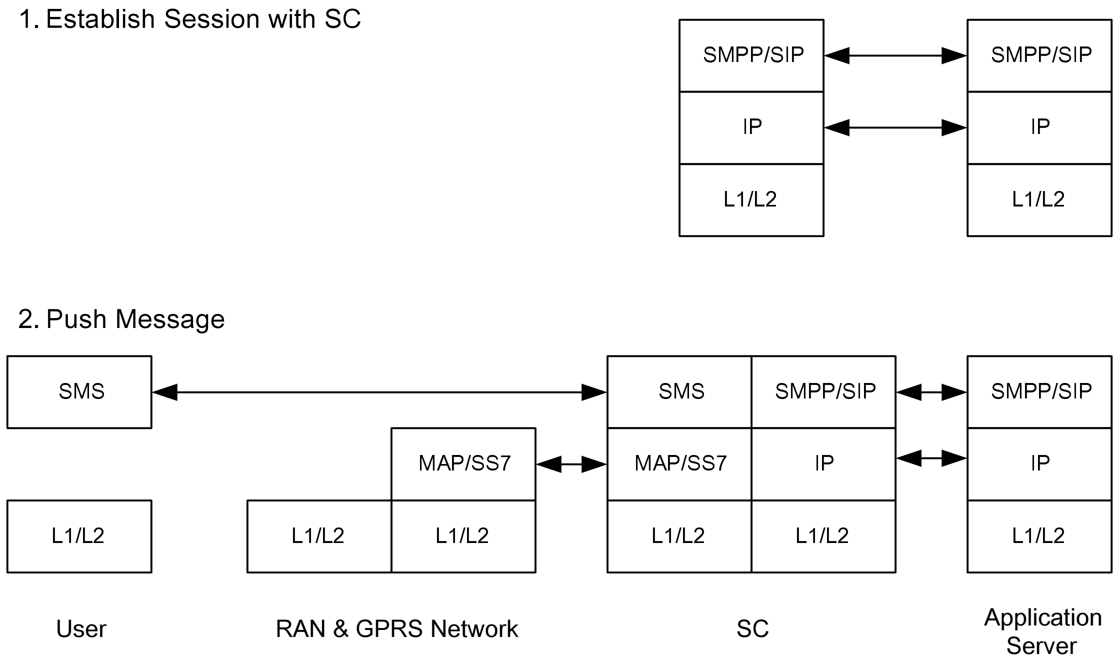


**Figure 7.4.5: SMS Protocol Architecture**

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 solution with dynamic address using always on and SMS

## 7.5.1 Architecture

The push architecture describes how to realize push service in a GPRS/UMTS environment.

The GPRS/UMTS access network is connected to the operator's service network by a GGSN. The APN (requested by UE but authorized by subscriber data) is used to select the operator's service network. This network is most likely an IPv6 or a private IPv4 network, and is managed by the operator. Through this network, the UE can access to a set of services (including push services) or to the Internet. This network may include RADIUS[14] (e.g. for authentication), DNS[12], DHCP[13a], and others services.

This network also includes an SMS Service Center.

For push services, the operator's service network includes a push proxy, while push initiator are typically in external network such as the internet (typically public IPv4 network). The UE trusts the push proxy not to send unwanted push messages. Push initiator is prevented to address the UE directly.

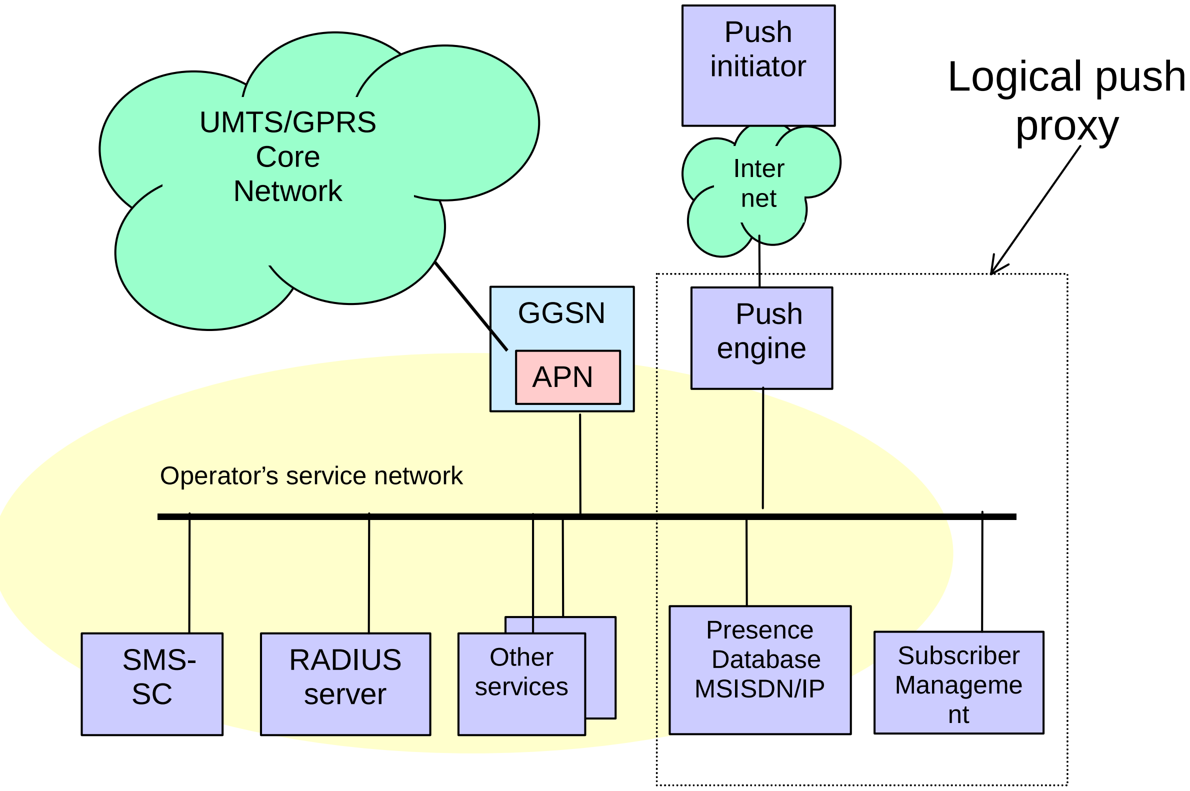


**Figure 7.5.1: Push architecture**

## 7.5.2 Push proxy

The push proxy is logically composed of:

- A presence database indicating if the subscriber (identified by MSISDN) is having a PDP context active and what is its dynamic IP address. Optionally, the presence database may indicate if the subscriber is available through another mean (e.g. fixed IP access).
- A subscriber management database indicating which push services are subscribed for which subscribers. This database is used to find if a received message can be forwarded to an UE or not.
- A push engine part: The push engine receives push messages from push initiator, checks the subscriber management database, checks the presence database, selects the bearer of the push messages and forwards them to the subscribers.

## 7.5.3 PUSH initiator

The content of the push message is received by the push proxy from push initiator servers. The push initiator shall have an agreement with the operator running the push proxy. A security relation shall be established between the push proxy and the push initiator.

Note: As an option, the push proxy might accept push message from unknown push initiator.

The push initiators are therefore not required to know the status of push service subscribers and their dynamic IP address.

Note: The WAP forum has defined a protocol between push initiator and push proxy.

## 7.5.4 Push services subscription

Every user should be able to receive different push services according to their wish.

The push proxy includes a subscription management database and therefore is aware of the service subscribed by each subscriber. The user should be able to modify their push subscription on-line (e.g. Olympic game news).

### 7.5.5 Addressing: Push service using dynamic address

The use of dynamic addresses is preferred as it avoids the configuration of a static IP address in HLR, UE, and push proxy.

The address may be a private IPv4 or an IPv6 address. Both provide virtually unlimited number of IP addresses. Due to restriction in the number of public IPv4 addresses these are unlikely to be used.

Note: In one operator network, different GGSNs may be connected to different private IPv4 operator's service networks. Each of these network can support around 17 millions private IPv4 addresses. This solution provides infinite number of private IPv4 address.

### 7.5.6 Presence description

The presence database needs to know the status of the subscriber (connected or not) and the mapping between its identity (MSISDN) and its dynamic IP address.

The UE application could indicate its presence directly to the presence server (e.g. using SIP). However, in order to optimise the system for wireless (saving one round trip over the radio), the GGSN may also inform the presence database when a PDP context is activated. This solution is described in more detail below.

The Presence database address is configured in the GGSN APN configuration. Each time a PDP context is activated to this APN, the GGSN informs the presence database of the subscriber status and the relation between its dynamic IP address and its MSISDN.

The mechanism is described in more details in the figure 7.5.2.

The GGSN sends a Start message (MSISDN; IP address) when the PDP context is activated, and a stop message (MSISDN; IP address) when the PDP context is deactivated. These start and stop messages may be implemented using RADIUS protocol, which is widely deployed, and to which specific extensions can be easily added.

This solution does not prevent the use of a subscriber logical name if the presence database knows the mapping between MSISDN and logical name. It would then be possible to update a DNS with the proper UE address.

In order to update the presence database of the availability through other accesses, other access router should implement similar behaviour, or the UE application should indicate its presence directly to the presence server (e.g. using SIP).

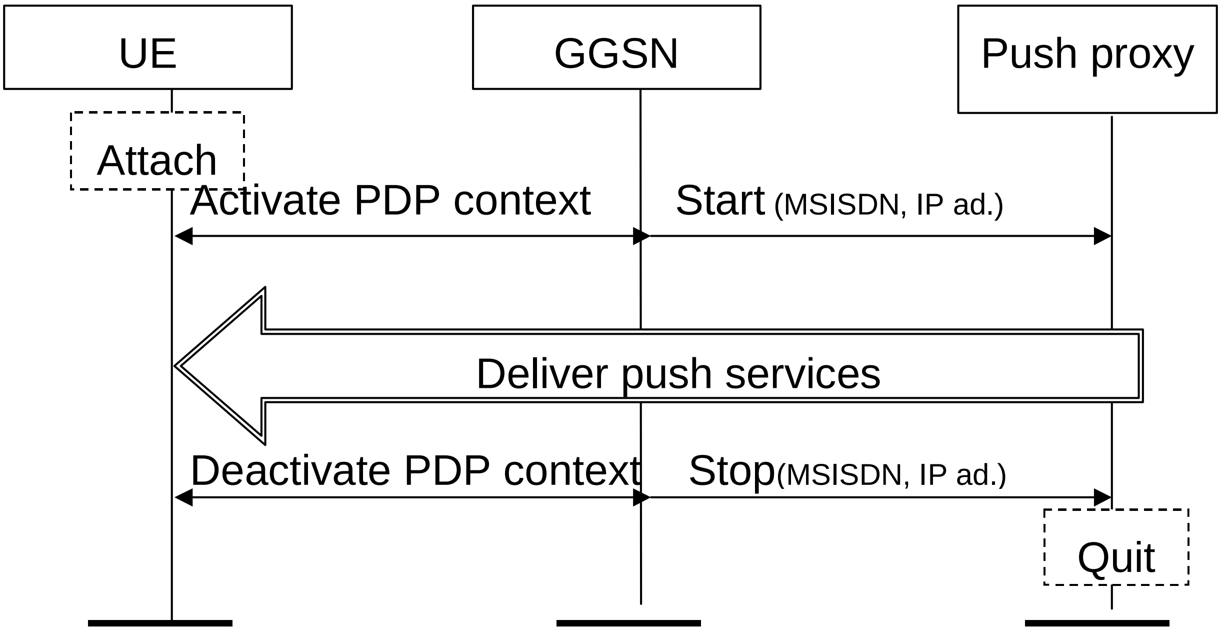


**Figure 7.5.2: Updating presence database from GGSN**

### 7.5.7 Delivery of the push message

In the always-on approach, push content is delivered over the established IP connection (i.e. PDP context or fixed IP connection).

If no IP connection is established, SMS may be used to trigger the terminal to create a PDP context over which the push content is delivered. SMS will trigger the UE as soon as this one becomes reachable (using SMS store and forward and alert capabilities). SMS may in addition deliver a short message to the terminal, and the validity period of the message can be set.

The push message is delivered through the following steps:

1) Push initiator sends push message to Push Proxy (PP), e.g. including MSISDN or a list of MSISDN.

2) PP checks subscription from its subscriber management database

3) If the message is allowed, Push proxy checks from its presence database if the UE has an IP address.

4) If an address is returned, the message is sent directly to UE through GGSN

5) If no IP address is returned, the message or an indication to activate the PDP context may be sent via SMS (providing store and forward) depending on the importance of the message and the subscriber right.

Note: In the case of a fixed IP connection, an access router would replace the GGSN.

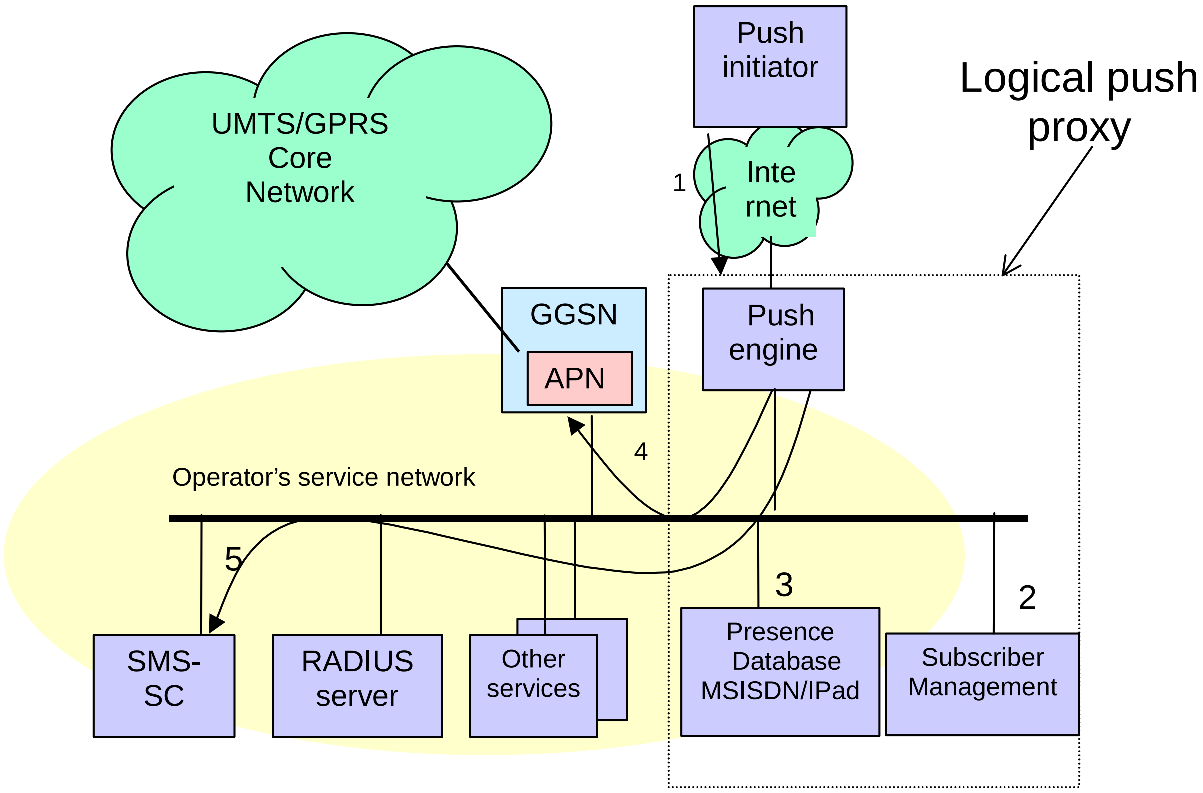


**Figure 7.5.3: Delivery of push messages**

### 7.5.8 Reliability of the delivery of the push message

The push proxy shall provide different level of reliability. It should in particular be capable of guaranteeing the delivery of the message through acknowledgement and retransmission mechanism.

For example, WAP-Push [16] provides Confirmed or non-confirmed delivery of Push messages. The figure 7.5.4 illustrates how Confirmed delivery works, when the UE has an active PDP context.

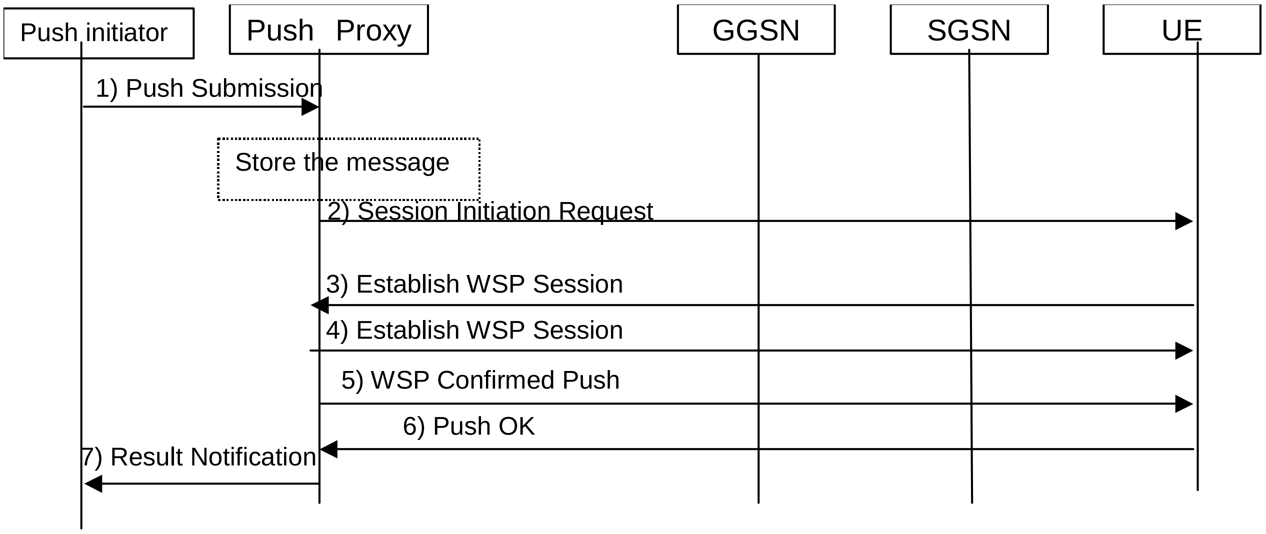


**Figure 7.5.4: WAP-push confirmed delivery to an UE having a PDP context activated.**

## 7.5.9 Store and forward function:

The store and forward functionality consists not only in storing & forwarding a message, but also in receiving an alert as soon as the UE is reachable. Existing SMS services provides such store-and-forward mechanism that may be used for PDP context activation triggers and possibly small push messages (e.g. text + URL). Other push messages are stored in the push proxy itself, and delivered once a PDP context is established.

The push proxy should use the store and forward functionality when requested by the push initiator and either of the following conditions is satisfied:

- If the UE is not having a PDP context activated, or
- if the delivery of the push message failed through the activated PDP context (e.g. no push OK message was received)

If the push message is short, the SMS should carry the push content directly.

If the push message is too long to fit in an SMS, the SMS triggers the UE to automatically activate a PDP context over which the push content is delivered from the PP to the UE.

Note: The SMS will be delivered as soon as the UE becomes IMSI attached or GPRS attached due to existing SMS alert mechanism.

## 7.5.10 Multiple services

An UE may receive push service from multiple push initiator servers. The push proxy supports delivery of push content from multiple sources simultaneously.

## 7.5.11 Security

Allowing directly push initiator from the Internet to push their message to the UE would be a security risk. In addition, if IPv4 private addresses are used by UE, it is doubtful that a push initiator using a public IP address could reach the UE. The PI would not only need to know which of the NAT’s public IP addresses it should target, but also the port. The NAT[12a] device normally assigns both the address and the port dynamically for every request sent by the UE. This implies that the PI needs to be able to request the NAT device to map a certain socket to a specific UE before the push is sent (not standard NAT behaviour), which of course is not feasible.

The push proxy guarantees the security by:

- Authenticating the push initiator as a valid one
- checking from its subscriber management database what should be sent to the UE(s)

- forwarding, if authorized  the push message to the UE over the secure operator service network (private IPv4 or IPv6 network).

Therefore the use of a push proxy protects the user from unwanted attacks. In addition, using a push proxy avoids the use of NAT between UE and push initiator.

The figure 7.5.5 illustrates a possible secure implementation where the push proxy has secure connection to push initiator, and the UE are protected from unwanted request by a firewall. In this example NAT (Network Address Translation) is not needed for push services.

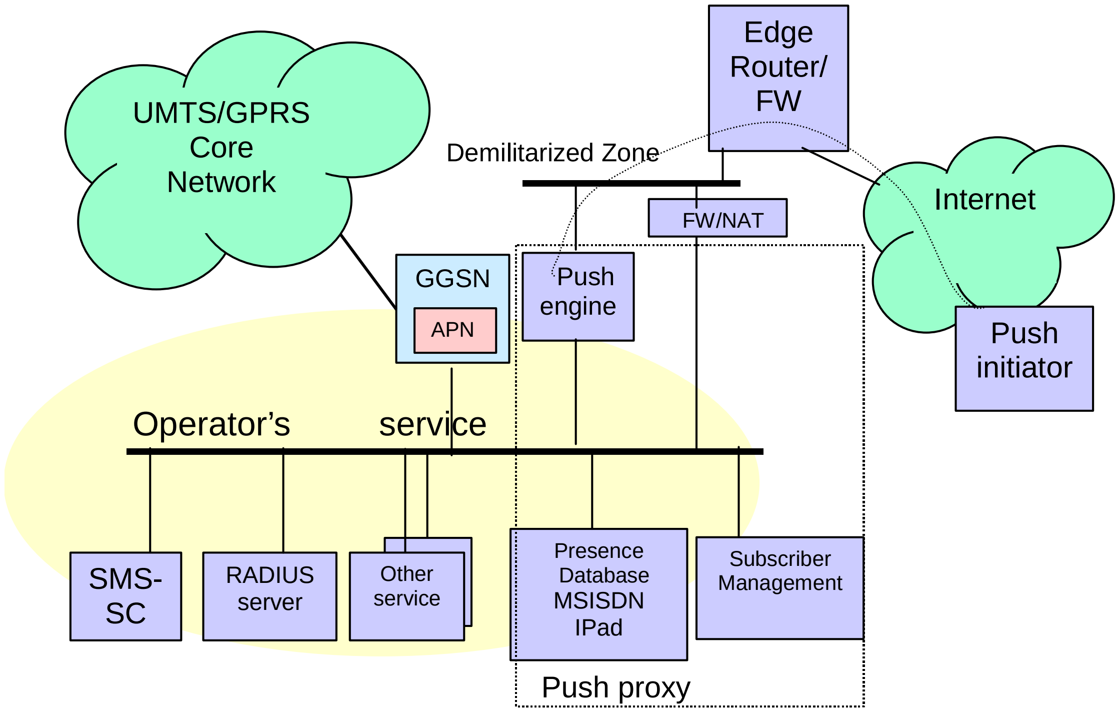


**Figure 7.5.5: A possible secure implementation of push services.**

## 7.5.11 Charging

The network operator may charge based on user subscription to specific services and on the service delivered. Charging should be handled at service level by the push proxy, as:

- The push proxy handles subscription of services to which a monthly charge may be linked.
- The push proxy handles delivery of messages, so that it can create a service charging record per message delivered.  This service charging record may have extra information about the service delivered (e.g. delivery was confirmed, push initiator, to which subscribed group it belongs, etc)

## 7.5.12 User terminal

In order to be used for push services, the user terminal is required to support:

- A push application
- IP access (through GPRS/UMTS) and SMS

Note: A release 99 terminal may be used if it supports the proper application, or if the proper application can be added on it (e.g. EPOC terminal).

### 7.5.13 Roaming Support

PLMNs support roaming service.  Thus, push service shall be available to subscribed users when they roam. The home GGSN should be used when roaming. The SGSN is connected to the GGSN using the inter-PLMN backbone and all traffic is tunneled from GGSN to SGSN. Therefore the UE access exactly the same services than if it would be in its home network.

This can be enforced by operator in the HLR subscription data.

Note: Using visited GGSN would imply a method to access subscription information from the visited network. This can be made based on agreement between operators.

### 7.5.14 IP address management

Due to the limitation of public IPv4 addresses, a (successful) push service has to use private IPv4 or IPv6 addresses. Both solutions provide unlimited number of addresses.

There are around 17 million private IPv4 addresses for one private IPv4 network. If one operator needs more addresses, different GGSN may be connected to different private IPv4 networks providing infinite number of private IPv4 address. Private network may be differentiated using APN. UE may be allocated to one or the other network based on their subscribed APN. Push proxy (or push initiator) knows upon available information to which network the UE is registered.

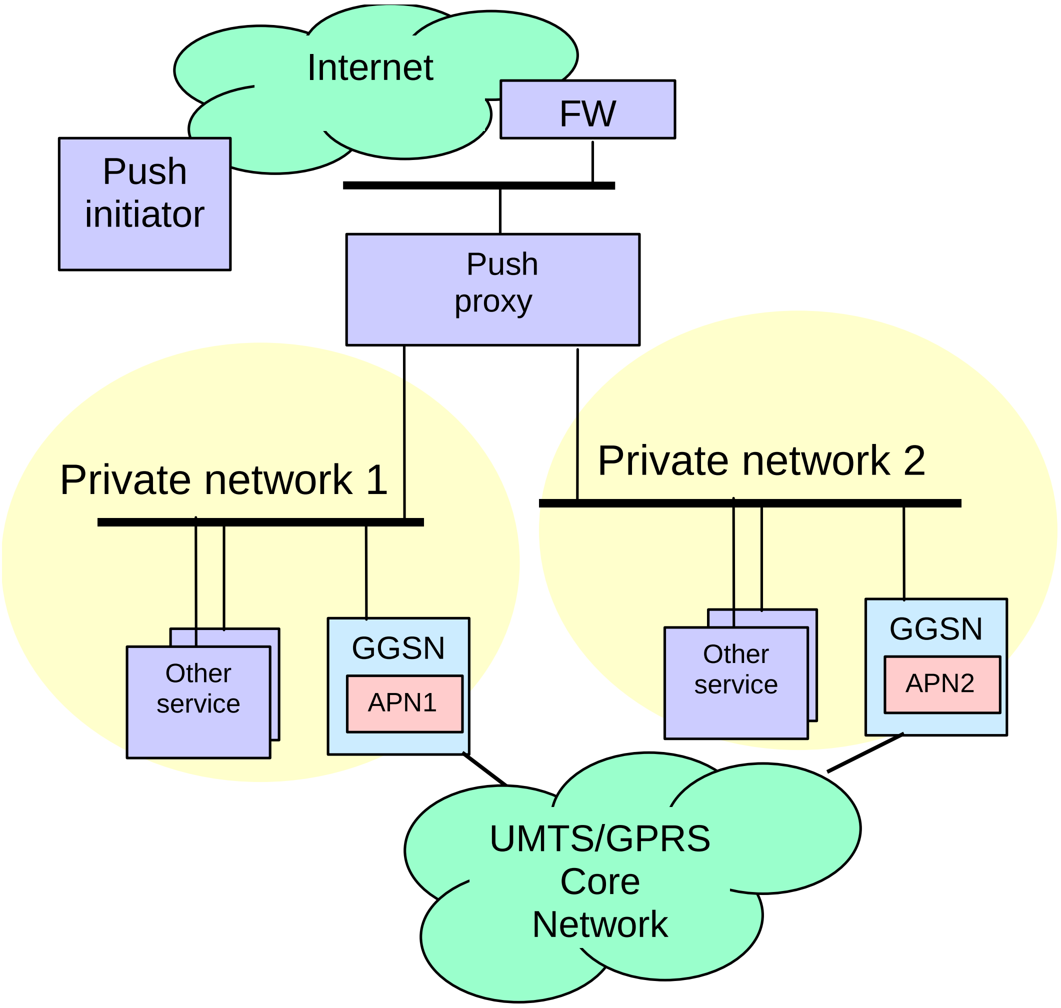


**Figure 7.5.6: A possible implementation of push services with multiple private IPv4 address spaces.**

## 7.6 SIP based Push Service

SIP based push services uses a proxy based architecture where SIP signalling is used to establish a session between the UE and the Push Proxy. An access protocol is used between the Push Initiator and the Push Proxy. The access protocol can be based on the protocol defined in section 7.7 or on SIP in order to perform end to end multimedia session.

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 R5 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.

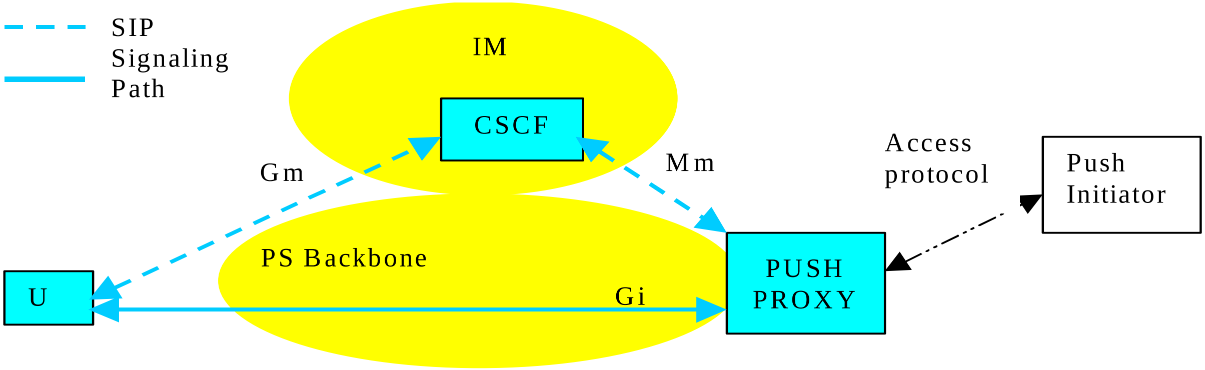


**Figure 7.6.1: SIP based Push via IM Subsystem**

The SIP Signaling Path is a standard IMS Signalling PDP context, across the Gm reference point, used to transfer SIP control messages (e.g. SIP Register, SIP Invite). The Data Path use another PDP context in accordance with IMS architecture, see TS23.228.

In this scenario, the following steps are required:

1) When a UE attaches to the network, it will establish a signalling PDP context to be used for SIP registration and signaling. The IP address allocated for this PDP context will be assigned as a dynamic IP address(TS23.228).

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 3GPP TS 23.228[5]). 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 Push Proxy 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 Push Proxy 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 push service 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 an existing PDP context or establish a new PDP context to be used for Data Path traffic. The IP address assigned to the UE for this PDP context would be provided to the Push Proxy as part of the response to the SIP Invite.

7) Since SIP is a session protocol, the Push Proxy 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 Push Proxy, 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 Push Proxy, the PDP context must be active. By using the preservation procedures described in 3GPP TS 23.060[4], 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 Push Proxy. 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.

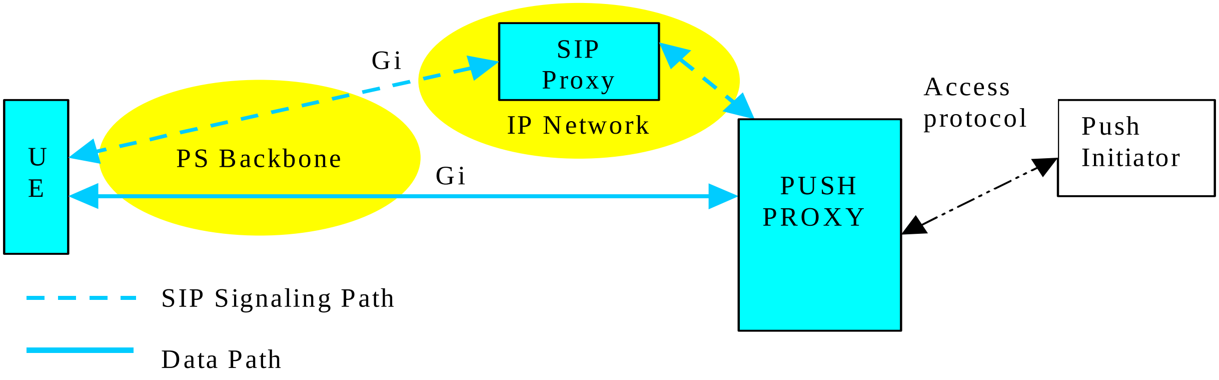


**Figure 7.6.2: SIP based Push via Long-lived PDP Context**

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.

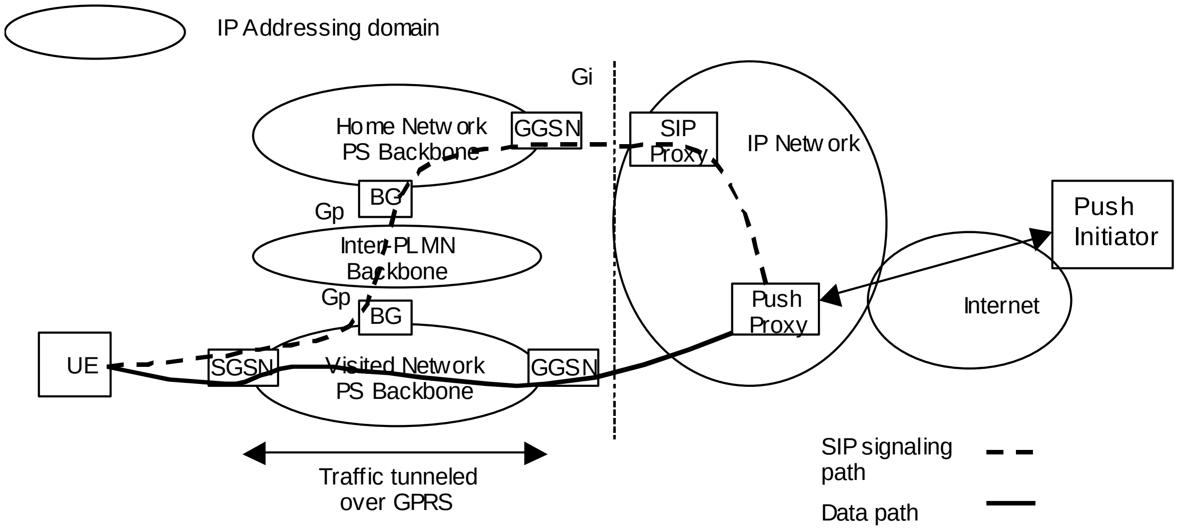


**Figure 7.6.3: Roaming – SIP Proxy via HPLMN**

.

### 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.

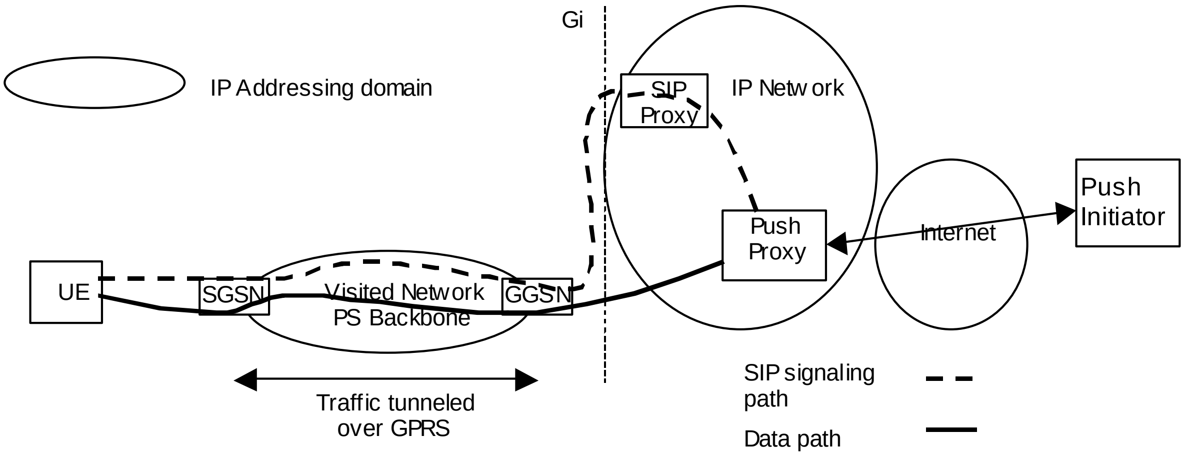


**Figure 7.6.4: Roaming – SIP Proxy via VPLMN**

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.

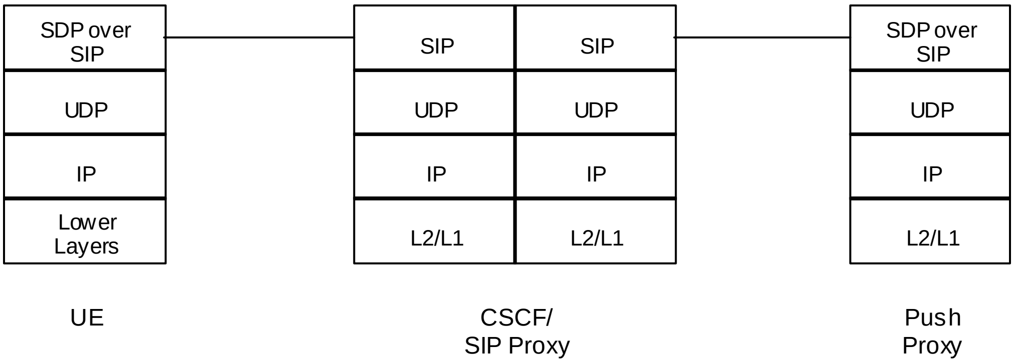


**Figure 7.6.5: SIP Session Management Protocols**

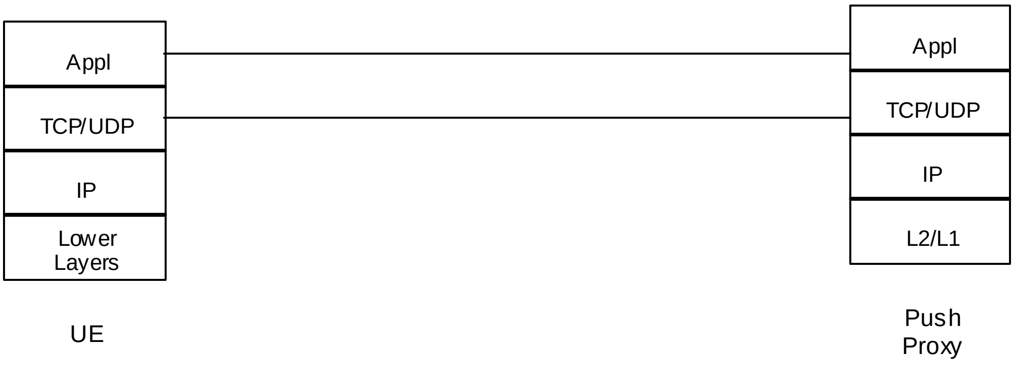


**Figure 7.6.6: Data Protocols**

## 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 Push Proxy 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 Push Proxy 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 Push Proxy 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 Push Proxy's IP address will be included in the SIP Invite. The UE will provide its Data Path IP address to the Push Proxy 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 Push Proxy will be responsible for retrying later.

There are currently IETF draft proposals (draft-rosenberg-impp-lpidf-00.txt[22], draft-rosenberg-impp-presence-00.txt[23], draft-rosenberg-impp-im-00.txt[24]) to include Presence as part of SIP. SIP Presence would allow the Push Proxy 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 Push Proxy 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 Push Proxy 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[22], draft-rosenberg-impp-presence-00.txt[23], draft-rosenberg-impp-im-00.txt[24]).

As a further option, SIP MESSAGE method may be used to carry peer-to-peer data. The push message is carried in the method body. MESSAGE method is specified in draft-ietf-simple-im-01.txt.

Note: Draft-ietf-simple-im-01.txt is due to be sent to the IESG in August 2001.

### 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 Push Proxy. The UE will be responsible for establishing the Data Path PDP context using the supplied QoS.

## 7.7 Push Proxy Based Architecture Using HTTP as Delivery Protocol

### 7.7.1 Architecture

A proxy based architecture is made of the three main elements (as shown in figure 7.7.1):

- An access protocol
- A push proxy
- A over-the-air protocol

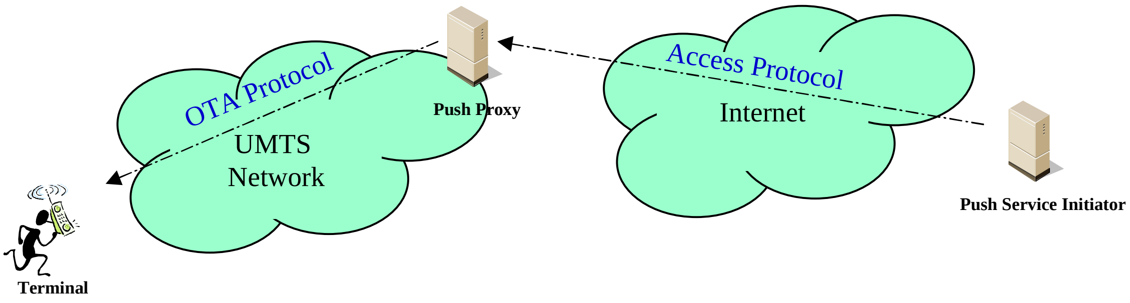


**Figure 7.7.1: Proposed Push Architecture**

The access protocol (between the push service initiator and the push proxy) can be of multiple form. For example it can be:

- HTTP[17] based: the push initiator POST push request to a UE via an HTTP based push proxy
- SMTP[11] based: the push initiator e-mail push request to a UE via an e-mail based push proxy
- WAP PAP[27]: the push initiator uses PAP request to send a push message to a WAP Push proxy

The Over-The-Air (OTA) delivery protocol is HTTP based. The OTA protocol has the following characteristics:

- Optionally uses the HTTP OPTIONS method to authenticate the UE and/or obtain the UE capabilities) on the initial push performed on the TCP connection. UE capabilities are described further in section 7.7.5
- Use the HTTP POST method to deliver content (push message) to an HTTP server based push application.

### 7.7.2 Push Proxy

The Push Proxy (PP) between the push initiator and the UE performs the following functions:

- Asynchronous delivery of messages depending on user availability. The PP manages push messages delivery in accordance with the presence information obtained from other source in the network (Presence Server, HLR, GGSN, SMSC, etc). If the push message intended destination is "present" on the network, the PP then simply forward the message using the OTA protocol specified in section 7.7.4. If the push message intended destination is "not present" on the network, then the PP might:
    - Wait until a presence notification is received and then simply forward the message using the OTA protocol specified in section 7.7.4; or
    - Trigger the intended destination to establish IP connectivity using an SMS message or other available means
- Delivery QoS management: the PP manages the delivery based on QoS indication received from the push initiator. Those are mainly indications on how urgent is it to deliver the message and for how long is the message valid. For example, access protocol such as PAP allows the push initiator to define the delivery timeframe. Similar mechanism can be developed for other access protocol alternatives.
- Service level accounting/charging: The PP support service level billing. This includes but is not limited to:
    - User transaction based billing
    - Push service provider transaction based billing
    - Shared model where the push service provider pays part of the transaction (e.g. advertising)
    - Shared model where the operator shares the access revenue generated by a push service provider with the push service provider (e.g. interest group)
    - Flat billing
- Address resolution: the PP extracts the internal network address of the UE from its external address received in the push message (refer to section 7.7.3 for details).
- Optional push initiator access control: the PP can authenticate the push initiator and filter the incoming push messages.
- Optional content filtering (e.g. malicious content); the PP can filter potentially harmful content (i.e. content that contains viruses). The filter settings can be managed either by the PP owner or by the user himself through the UE capability feature (see section 7.7.5).
- Optional content transformation (based on UE capabilities): the PP can adapt the content to the accepted format (content-type, language, encoding, charset, etc) advertised by the UE in its capability profile.

### 7.7.3 Addressing

The push proxy performs address resolution from the external address to the private network address (if private addressing is used). The external address is not necessarily a network address (depending on the access protocol used). The external address is managed by the PP owner and is known to the push initiator from the push service registration by the user. The external address might contain MSISDN or IP address but should not be required to contain those for privacy reason (e.g. it might not be good, and in some countries it might even be illegal, to distribute MSISDN to push service providers). The PP might resolve the internal address with the support of other systems (Presence server, RADIUS[14], DNS[12], etc). The PP might also trigger the UE to establish a PDP context if the UE is not attached.

### 7.7.4 Push Delivery Mechanism

The push delivery mechanism is based on the well-known HTTP protocol (RFC2616[17]). The push destination application in the UE is an HTTP server based application. Content is sent to that application by POSTing data to the application URI (refer to HTTP).

Once the PP has resolved the internal IP network address of the UE (refer to 7.7.3) the PP:

- Establishes a TCP connection to a well- known port on the UE (port 80 or a new IANA registered port). The TCP connection establishment is a three-way handshake as described in RFC793[10a] and is shown in red in figure 7.7.2 below

- Optionally sends the HTTP OPTIONS method to the UE (including a challenge if UE authentication is required). This is needed if UE authentication and/or UE capability profile query is required. Both proxy authentication (as per RFC2617[18]) and UE authentication (based on RFC2617[18] with slight modifications to allow for server authentication) can be performed during that step. If the UE has a static capability profile known to the PP and if neither the PP or nor the UE authentication are required, that step can be omitted

- Sends the HTTP POST method to the HTTP based push application URI with the push content included in the POST request body

- Recover the HTTP transaction status if required

This sequence described above is illustrated in figure 7.7.2 below.

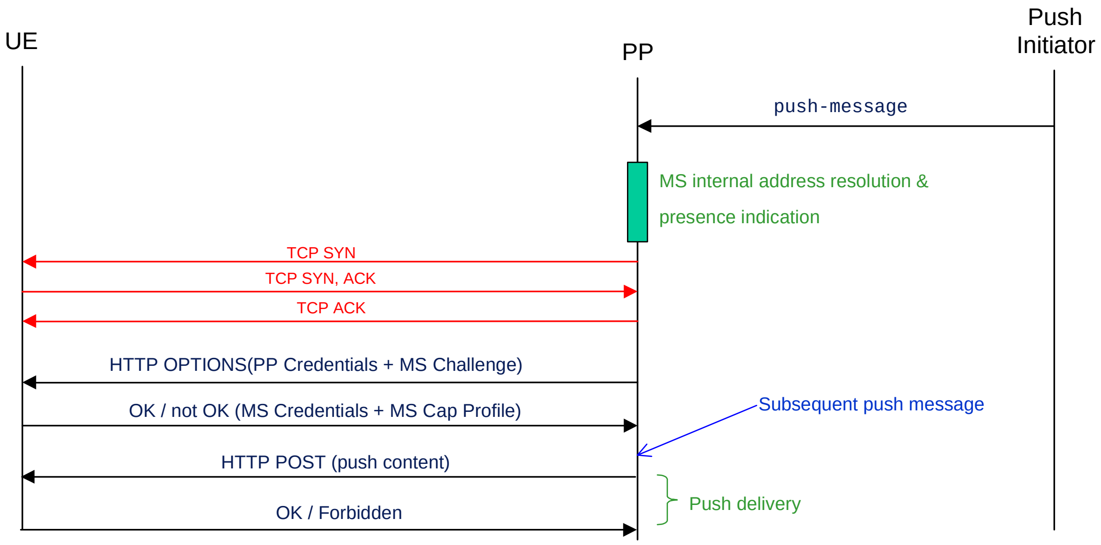


**Figure 7.7.2: Delivery Message Flow**

If the TCP connection is short lived, subsequent push messages require the PP to establish a new TCP connection toward the UE. In that case the authentication/capability query phase (the OPTIONS method) can be omitted only if the PP can assume that its latest authentication/capability data for the UE is still valid (e.g. when capability profile is static and address lease time is long (or permanent)). If the TCP connection is long lived, subsequent push messages are simply forwarded directly using the POST method.

## 7.7.5 UE Capability Profile

The UE capability profile expresses what an UE can do and cannot do. In the wireless world it is very important to know about the UE capabilities before delivering any content due to the broad range of device type (phone, PDA, PC, etc). The capability profile contains various data about the UE such as:

- screen size
- number of pixels supported
- memory size
- receive buffer size
- application contained in the UE
- content-type accepted
- etc

The UE capability profile can also contain the user's preferences to dynamically advertise those preferences to the PP as they are updated by the user (e.g. access control filter setup, malicious content filter setup, etc).

The W3C group have defined a general framework to handle capability profile. This framework is defined in [29]. The CC/PP capability profile is the entity carried in the OPTIONS method response.

### 7.7.6 Roaming Considerations

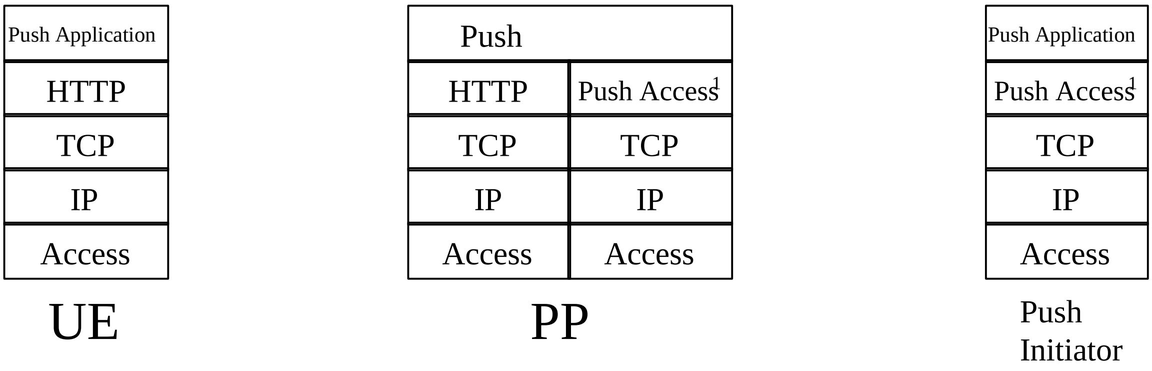
The main roaming model assumes that the UE always reach its home GGSN. The visiting SGSN is connected to the home GGSN using the inter-PLMN backbone. This way roaming is transparent to the push proxy.

### 7.7.7 Delivery Reliability

The Push Proxy manages the reliability of the delivery up to the expiration of the validity period specified for each push message by the push initiator (i.e. the PP will retry message delivery until successful delivery is obtained during that period). Beyond the validity period of the message the push initiator has to resend the message. This offers the flexibility to the application server (Push Initiator) to decide if reliability is controlled by the PP or directly by itself (the architecture supports both).

### 7.7.8 Protocol Architecture

The protocol stack required for push message delivery in this architecture is as shown in figure 7.7.3 below.



1) Push Access: HTTP, SMTP or WAP PAP based

**Figure 7.7.3 Push Message Delivery Protocol Stack**

As described in section 7.7.2 the PP manages push messages delivery in accordance with the presence information obtained from other source in the network (Presence Server, HLR, GGSN, SMSC, etc). If the push message intended destination is "not present" on the network, then the PP might:

- Wait until a presence notification is received and then simply forward the message using the OTA protocol specified in section 7.7.4; or
- Trigger the intended destination to establish IP connectivity using an SMS message or other available means

### 7.7.8 Security Considerations

Due to the nature of a proxy based solution (i.e. the PP functions operate above the transport layer) end-to-end security (between the push service initiator and the UE) is better handle at the application layer. Protocol such as S/MIME (RFC2632 and RFC2633) can be used to achieve this.

Over-the-air protocol security can be achieved at the transport layer by having the UE initiating a TLS handshake upon TCP connection establishment.

The push proxy can perform user protection functionality including malicious content filtering and push initiator access control.

# 8 Conclusion and Recommendations

This technical report has analysed a variety of solutions for the implementation of push services architecture.

The recommended architecture for the push service is a proxy based architecture comprising the following elements:

- Push Access protocol between the Push Initiator (Push Application Server) and the Push Proxy.
- A push transfer protocol handling the push content delivery between the Push Proxy and the UE.
- A Push Proxy  that might perform functions such as access control, UE presence handling, store and forward, user profile management, content adaptation, etc.

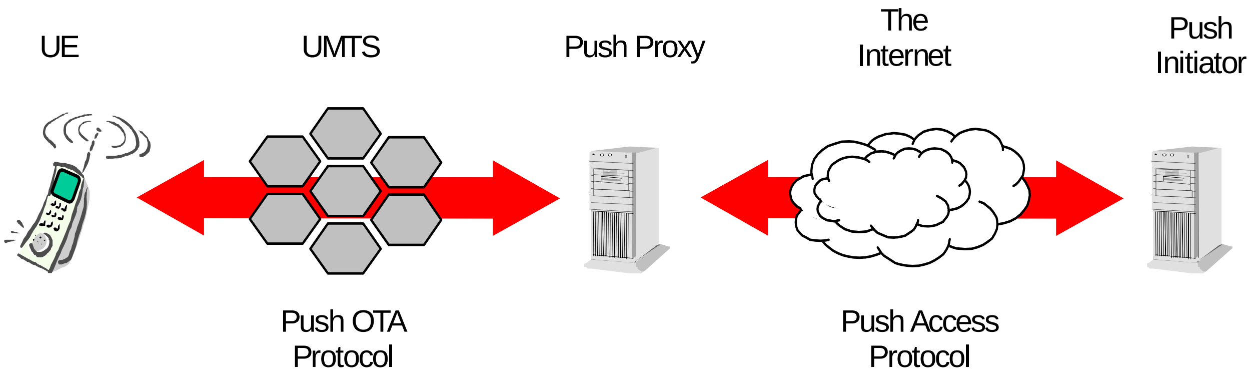


**Figure 8.1: Push Proxy Architecture**

The push service in this architecture is bearer and subsystem independent and available over both the CS and the PS domain.

One solution that has been proposed which satisfies the push proxy architecture is the solution of the WAP Forum. Facilities provided by this solution include those defined in WAP 2.0 specs (WAP-235-pushOTA, WAP-247-PAP, WAP-249-PPGservice, WAP-251-PushMessage, WAP-167-ServiceInd, WAP-168-ServiceLoad, WAP-175-CacheOp), IETF specifications (RFC 2616 – HTTP 1.1, RFC 2617 - HTTP Authentication) and W3-CC/PP – Composite Capability/Preference Profiles.

Additionally the issue of how to establish the bearer has been presented in the document.  There are three potential solutions:

1. Long standing PDP context activation – always-on
2. Session initiation using SMS (via the WAP Forum developed Session Initiation Request SIR)
3. Network requested PDP context activation (NRCA) with dynamic IP address allocation

Beyond this, interaction between the architecture above and the architectures/solutions in the IMS will have to be considered, such as SIP signaling as multimedia session establishment for push services.

At this stage it has not been possible to reach a conclusion on which of these three potential solutions for establishment of the bearer should be adopted as stage 1 requirements have not yet been fully defined. This report recommends that the push services work be placed on hold until SA plenary has agreed the stage 1 requirements (expected December 2001). At that stage, work should recommence on evaluation of the architecture and potential solutions against the stage 1 requirements with the objective of agreeing the way forward and, if necessary, producing a stage 2 specification.

Note: Note that collocated SA1 and SA2 meetings are scheduled for January 2002 at which these stage 1 requirements can be discussed in detail.

It must be recognised, however, that there is a strong requirement that one or more viable solutions be found that can be implemented with release 5 timeframe and that meet the business requirements of the operators who wish to deploy push services. It is an open issue as to whether one solution will meet all requirements or whether multiple solutions will need to be standardised. This will be reconsidered once stage 1 requirements are defined.

This report has not considered detailed issues on charging and security; these topics will be referred to the relevant groups within 3GPP for their consideration.

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

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

# Annex B (Informative): A study on how NRCA and "always on" fulfil PUSH service requirements

Note: This annex is a comparison from a single point of view for information. The conclusion given is for the study described in this annex. It is not intended as the conclusion of the whole TR.

## B.1 Introduction

Push service requires support at two levels: a push application (e.g. WAP) and connectivity at UMTS level. This section studies what is the best way to support this connectivity among the two standard ways: "Always on" and NRCA. This section will also discuss where appropriate if NRCA should be used with static IP address (solution standardized) or with MSISDN and dynamic IP addressing (solution under discussion in 3GPP).

Every solution is compared to the requirements, which are affected by the choice of connectivity method.

Note: this section describes only UMTS but the analysis applies also to 2G GPRS.

Note: In 2G GPRS, a different radio signalling is used, but there two a round trip delay over the radio is very time consuming (about 2 seconds), so minimizing the radio signalling is also an issue. 2G GPRS has BSC instead of RNC, GMM standby instead of UMTS MM Idle.

## B.2 Description of the procedures:

### B.2.1 NRCA

The reader is referred to 23.060[4] for details of the different messages. The procedures needed between RNC and MS are depicted on the side. Each of these procedures requires at least two messages over the radio.

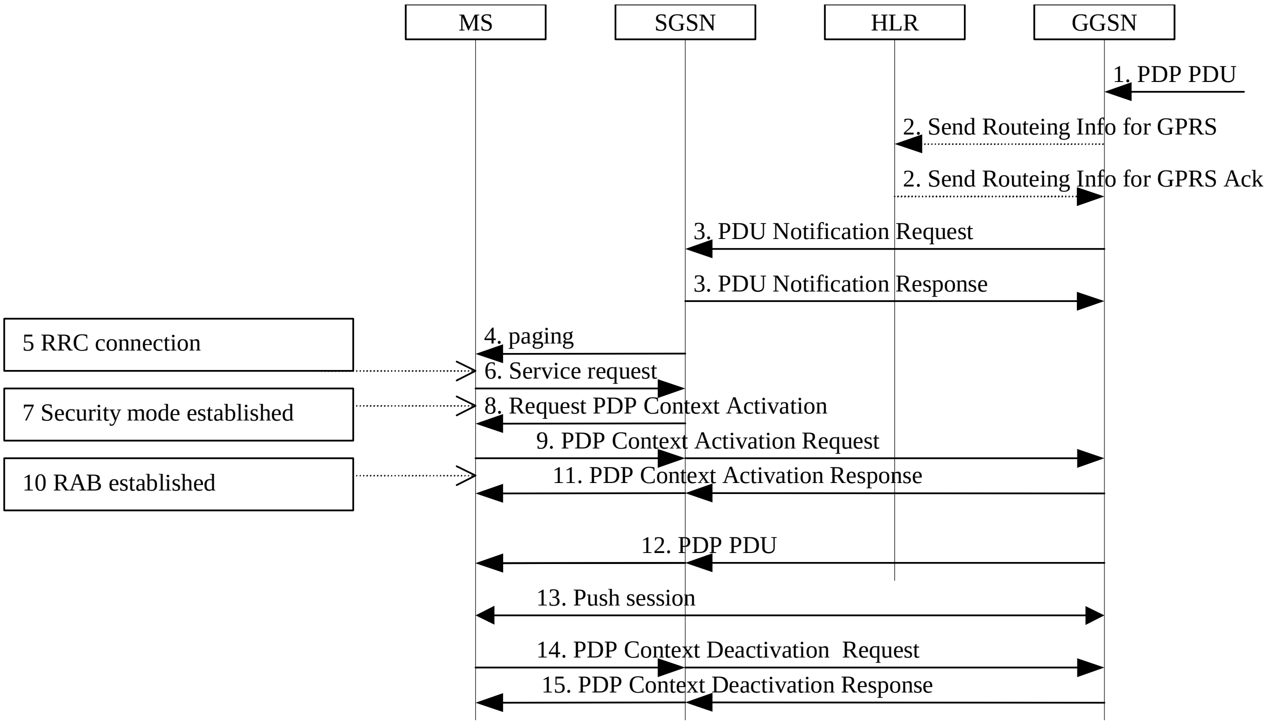

MS
SGSN
HLR
GGSN
1. PDP PDU
2. Send Routeing Info for GPRS
2. Send Routeing Info for GPRS Ack
3. PDU Notification Request
3. PDU Notification Response
5 RRC connection
4. paging
6. Service request
7 Security mode established
8. Request PDP Context Activation
9. PDP Context Activation Request
10 RAB established
11. PDP Context Activation Response
12. PDP PDU
13. Push session
14. PDP Context Deactivation Request
15. PDP Context Deactivation Response

Figure B.1: delivery of a pushed packet with NRCA

In this section, The NRCA procedure refers to steps 2,3 and 8. The PDP context activation procedure refers to step 9,10, 11.

The principal of NRCA is to reach an MS who is GPRS attach, but is not having activated a PDP context. This MS must have a static IP address provisioned in HLR. The GGSN must have the mapping of the static IP address to IMSI; i.e. a permanent database needs to be maintained in GGSN.

When the GGSN receive a normal IP packet for one user, it will query HLR to find SGSN address, and then notify the SGSN that one IP packet has been received for this subscriber. SGSN will then ask the MS to activate a PDP context. After the PDP context activation, the GGSN forward the IP packet to the MS.

Note: In one possible implementation, the HLR query can be avoided by storing the old SGSN address and hoping that the MS is still located under the same SGSN. When this is the case the message will be sent straight to the right SGSN, but when the MS has moved, the message will first be sent to a wrong SGSN, and then an HLR query will be performed.

The pushed packet is typically a message with interactive links, which may trigger the user to enter a push session.

Logically, the MS deactivate the PDP context at the end of the push session.

Note: The deactivation is implied by the concept of NRCA. Without a deactivation, the logic would become an always on logic. Note that the network as well may deactivate the PDP context, but the problem is to wait long enough to be sure that the user has finished its session.

In a new proposal being discussed in the 3GPP standard, it is proposed to use NRCA with dynamic address. The mechanism is: an application server query GGSN (or an NRCA node) with a new protocol indicating that it wants to push a message to a certain user identified by MSISDN. The GGSN (or an NRCA node) checks from its permanent database the mapping to IMSI and query the HLR (like in step 2) and then proceed with the NRCA procedure.

It is important to be aware that SGSNs or MS supporting NRCA with static address will not automatically support this new feature as the message do not contain static address.

## B.2.2 Always on

In the always-on concept, the MS is GPRS attached and always has an active PDP context.

Packets received by the GGSN are always forwarded to the MS through the SGSN.

If the MS was MM Idle, a significant amount of signalling is still needed in order to establish secure radio bearer (See figure B.2)

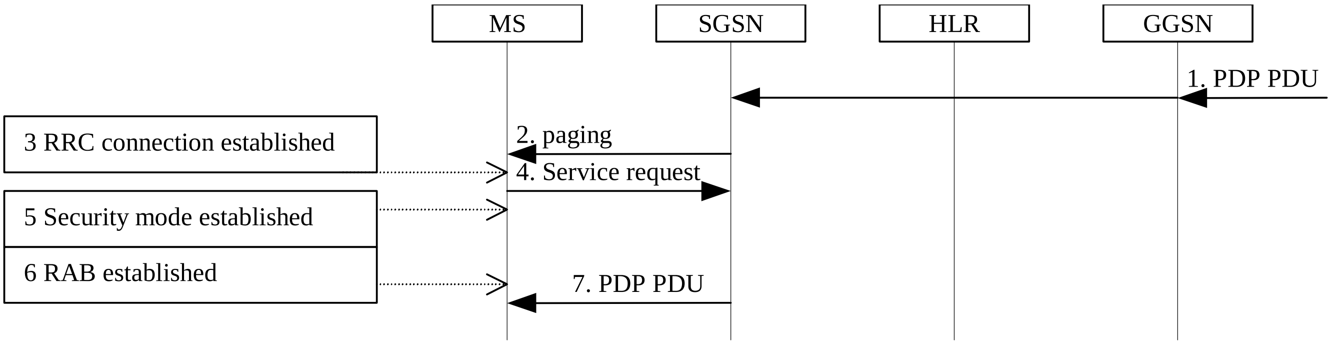


**Figure B.2: delivery of a pushed packet with always-on concept: MS in MM Idle mode**

If the MS was MM Connected, SGSN can forward the packets straight to the RNC that will send it to the MS eventually after paging (See figure B.3).

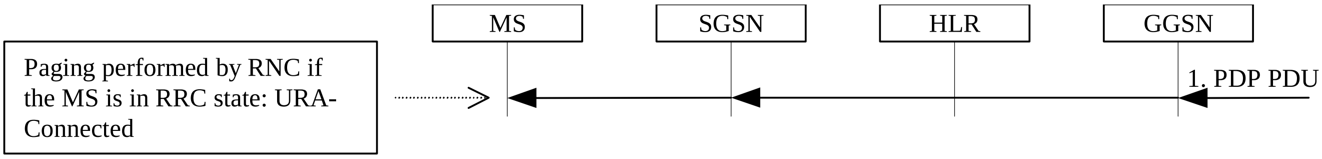

MS
SGSN
HLR
GGSN
Paging performed by RNC if the MS is in RRC state: URA-Connected
1. PDP PDU

**Figure B.3: delivery of a pushed packet with always-on concept: MS in MM Connected mode**

# B.3 Scalability, or supporting burst of push messages during busy hour.

An important requirement is that the network can deliver high peak of push traffic.

## B.3.1 NRCA

Network requested PDP Context Activation is creating a lot of signalling during the traffic peak:

- Query to the HLR to find address of SGSN handling the MS (meaning that HLR signalling capacity may become the bottle neck of the system). Note that this query is also made for detached subscriber.
- SGSN signalling: PDP context activation, NRCA procedure (GTP-C PDU notification message & SM Create PDP context request) and PDP context deactivation. Enough SGSN signalling capacity needs to be installed to take care of normal signalling load during busy hours and the burst of signalling created by NRCA-based push services.

Therefore with NRCA the burst of push messages will generate a simultaneous burst of signalling load putting extra strain on the system. The HLR is the most likely bottleneck. SGSN signalling also risks to be saturated especially if the system is already loaded with normal signalling. However, if the operator has installed enough SGSN and HLR signalling capacity to handle burst of push message, the bottleneck of the system will be the RNC.

Note: if CAMEL is used, NRCA also creates a burst of signalling toward SCP.

Example: HLR of 1200 000 subscribers supporting 300 query per seconds. Pushing football result to 300 000 subscribers at 300 message seconds takes 3000 seconds, i.e. over 15 minutes!

## B.3.2 Always-on

As the MS has already an attached PDP context, no query to HLR and no PDP context activation is needed for the reception of the push message. The bottle neck of the system is probably the number of radio connection that the RNC can established (but always on is less demanding on RNC than NRCA as it saves 5 messages over the RNC and the radio). With Always-on, the scalability is improved if the RNC maintains RRC connection during long time.

Note: With NRCA, the deactivation of the PDP context normally triggers the release of the RRC connection

## B.3.3. Conclusion

NRCA is not a scalable solution as it generates a burst of signalling during the burst of push messages, which increase the signalling load all over the system. To reach the same performance than always-on, it would require a very large installed signalling capacity.

# B.4 Delays

## B.4.1 Always-on:

The delay is coming from the time it takes the first push packet to be sent across the network (backbone and radio interface) and the delay creating by paging the MS, and the delay to activate radio access bearer (if the MS was not having an RRC connection).

### B.4.2 NRCA:

The delay is coming from the HLR query, the query to the MS to activate its PDP context and the PDP context activation itself, and the time it takes the first push packet to be sent across the network (backbone and radio interface) and the delay creating by paging the MS, and the delay to activate radio access bearer. In addition, if one of the signalling messages is lost (probable in congestion situation), it will be recovered through retransmission but this will add a delay of the order of seconds.

Note: In one possible implementation, the HLR query can be avoided by storing the old SGSN address and hoping that the MS is still located under the same SGSN. When this is the case the message will be sent straight to the right SGSN, but when the MS has moved, the message will first be sent to a wrong SGSN, and then an HLR query will be performed.

As has been studied, regarding the situation during burst of push messages, the delay may increase very significantly (i.e. over 15 minutes) due to the saturation of e.g. the HLR.

### B.4.3 Conclusion

In summary, NRCA will always have more delays created mainly by 3 message over the radio and one SS7 query, i.e. an extra delay of roughly 500 ms in an empty network compared to always-on MS is MM-Idle mode. The extra delay may be a few seconds compared to always-on MS is MM-Connected mode.

However during burst of push message, the delay may become very significant due to the poor scalability of NRCA.

## B.5 network resources are used as efficiently as possible;

In order to study the usage of network resources (i.e. radio capacity, signalling links, backbone); let’s imagine that a typical scenario will be a push user receiving 5 push message per day (or 1 per busy hour).

### B.5.1 Always-on:

The user when turning on its MS activate its push application, which initiate GPRS attach (3 messages over the radio) and PDP context activation (2 messages over the radio). The push application is deactivated when the user turns off his phone in the evening, i.e. the MS perform GPRS detach (2 messages over the radio). These procedures are typically not performed during busy hour (estimation of 0,5 procedures during busy hours). In addition, the MS shall perform the normal MM signalling (i.e. routing area update) during the time it is GPRS attached. The HLR is queried only at attach and inter SGSN RAU. Maintaining the tunnels between SGSN and GGSN is not creating extra signalling (except to update GGSN when SGSN change) or reserving backbone resources. In addition, if the MS is not having its RAB established when receiving the push message the RAB will be established.

Note1: According to GTP specifications[6], Echo request/response are sent per pair of nodes not per user tunnel.

Note2: The introduction of multipoint Iu will make SGSN changes quite rare.

### B.5.2 NRCA:

The user when turning on its MS activate its push application, which initiate GPRS attach (3 messages over the radio). The push application is deactivated when the user turns off his phone in the evening, i.e. the MS perform GPRS detach (2 messages over the radio). These procedures are typically not performed during busy hour. In addition, the MS shall perform the normal MM signalling (i.e. routing area update) during the time it is GPRS attached. The HLR is queried at attach, inter SGSN RAU, and for every NRCA procedure (i.e. 5 times or 1 query per busy hour). For each push message, one NRCA procedure, one PDP context activation and one PDP context deactivation procedure are performed (i.e. 15 procedures or 3 procedures per busy hour) implying signalling over the radio, over the backbone and over SS7 links. In addition, the RAB will need to be established when receiving the push message. Note that if CAMEL is used every PDP context activation/deactivation may trigger a CAMEL message.

In addition, NRCA also use resources for MS not GPRS attached (HLR query and possibly SGSN query).

### B.5.3 Conclusion

In summary, NRCA increase the amount of signalling in the network and in particular the signalling during busy hour (from 0,5 to 3 procedures). Therefore the "always-on" solution use the network resources more efficiently.

## B.6 Minimum investment

### B.6.1 Always-on:

All push service users are GPRS attached and have a non real time PDP context active. The number of GPRS attached customer will indicate the SGSN capacity needed, and the SGSN needs enough capacity to take care of normal signalling during busy hour (i.e. attach, Routing area update; PDP context activation/modification/deactivation, SMS…). The number of PDP contexts indicate the number of GGSN capacity (and one PDP context is always–on for every connected user).

Supporting a large number of GTP tunnels requires only memory from SGSN and GGSN but no new functions.

In SGSN, GTP tunnels require storing parameter related to active PDP context. However, if the MS would be attached without PDP context, the SGSN is still required to storing inactive PDP context (i.e. subscriber data) and MM context. Therefore, the impact on SGSN capacity of having one GTP tunnel per attached user is marginal (compare to having same number of MS attached without active PDP contexts).

Scaling up the capacity of GGSN is not either a technical challenge. We expect that better platform will dramatically increase the number of PDP context supported per GGSN. And adding more GGSN is straight forward way to add GGSN capacity.

Note: No GGSN or SGSN software upgrade are required as these functions are supported in first phase of GPRS.

### B.6.2 NRCA:

All push service users are GPRS attached. PDP context are only activated on per need basis. The number of GPRS attached customer will indicate the SGSN capacity needed. These SGSNs need more signalling capacity to take care of normal signalling during busy hour and signalling created by peak of NRCA-based push services. The number of push service user that the operator wants to be able to reach during peak of traffic indicates the number of PDP contexts that GGSN should support. In addition the operator needs to support a permanent database where the relation between IMSI and static address (or MSISDN). This database can be implemented inside GGSN (making GGSN more complex and so more expensive) or in a new node (NRCA node). The operator also needs to dimension its signalling support to cope with the signalling created by NRCA during the maximum peak of traffic it plans to support.

In addition, when NRCA is used with static IP address (3GPP release 99), GGSN needs to store the packets received during the duration of the Network Requested Context Activation (between a few seconds and 15 minutes with the example of section B.3.1). If the pushed message "Football result" is 1000 octet, and it is sent to 300 000 subscriber, GGSN needs to have 300 000 Mbytes of extra memory to store all these pushed messages. Note that if the GGSN is planning to use Always-on, this amount of memory could have accommodated around 300 000 extra PDP contexts.

Note: This example assumes one GGSN of 300 000 PDP contexts capacity.

When NRCA is used with dynamic address, software upgrade are needed in release 99 SGSN and GGSN.

### B.6.3 Conclusion

In summary, the "always-on" concept needs more GGSN capacity, while the NRCA require a new permanent database, and more signalling capacity (SGSN signalling cards, SS7 links, HLR&SCP signalling capacity) depending of the peak of push messages that the operator plans to support. The scalability expected by the operator and the pricing need to be known to conclude on this point.

# B.7 Minimum operating cost

## B.7.1 Always-on:

The always-on concept does not require special configuration in the operator network. It is enough to send the MSISDN to the appropriate push proxy with e.g. RADIUS[11]. This is a simple configuration of the Access point name. Note that preferably the same access point is used for all operators' service and Internet access.

## B.7.2 NRCA:

NRCA requires maintaining a new permanent database in the system where every push subscriber need to be added. This requires upgrade to the subscriber management system, and more cost when adding a new customer. In general, NRCA is a new function that requires extra cost to maintain and operate.

## B.7.3 Conclusion

In summary, NRCA adds to the operating cost of the operator

To conclude on the cost issue, NRCA has less efficient network usage and more operating cost than always-on. The initial investment is more difficult to estimate as it depends on the peak of push message than the operator wants to support.

Note that we estimate that to be able to support push services to 25% of the subscriber, SGSN signalling capacity needs to double (HLR and SCP capacity are also affected).

# B.8 interoperability and Service availability when roaming

The MS behaviour will create interoperability problems, as NRCA is an option not expected to be supported by all network or MS. It is important to know that an MS roaming will not know if NRCA is supported in the local network or not. In particular it seems that no network manufacturer support NRCA with static addressing in their first release. If NRCA with dynamic addressing is standardized, it will only work if the visited network has implemented this new standard.

## B.8.1 Always-on:

In a normal GPRS network (without Network requested PDP Context Activation option), the GPRS MS is never only GPRS attached (as only SMS can be used). The MS will either be Internet connected (being both attach and having a PDP context active) or not (i.e. GPRS detach).

## B.8.2 NRCA:

In a GPRS network using Network requested PDP Context Activation option, the GPRS MS is only GPRS attached but is not having a PDP context active. This specific behaviour (being attach but without a PDP context) will lead to problems when the MS is used in a PLMN where Network requested PDP Context Activation is not supported.

Note: The user will not receive its push service, but still the MS will use its battery and radio/SGSN capacity of the visited operator, as it is GPRS attached.

## B.8.3 Conclusion

An MS behaving as if NRCA is supported will not receive its push services in a network not supporting Network requested PDP Context Activation.

On the other hand, "Always-on" will work in any GPRS/UMTS visited network (using GGSN in your home network to make sure your push services are unchanged).

Note: However, there might be problem with roaming agreement if the visited operator wants to bill a high price based on time of connection.

In summary, there is no guarantee that push service based on NRCA will work when roaming, due to different support in different operators.

# B.9 Charging

NRCA has often been presented as required due to time based charging. For a Non real time context charging based on time of connection is more similar to charging the time a GSM MS is turned on than charging the time of a modem call. Contrary to a modem call, an active PDP context do not guarantee any available bit rate.

Time based charging encourages the user to disconnect reducing therefore the overall traffic. For example, the user (or an "optimized" application in its MS) may disconnect before reading the push messages. Connecting again if it wants to read another page (click a link), and disconnect immediately when the page is downloaded.

Time based charging might cause also overcharge the user. This may happen for when the user goes out coverage for a certain amount of time (even in the order of 30 minutes) without the network notices it. The worst case is that the user goes out of coverage, deactivates its PDP context but stay GPRS attached, and come back to coverage. The network may deactivate the context only when the MS will detach (possibly hours latter). However, this problem might be reduced with very high quality coverage. Small "holes" might be compensated by applying some kind of "de fault" reduction in user bills based on some estimates of the minutes in percentage the user might find to be out of coverage. Too long PDP context may be considered as error by the billing system. Still obviously, time based charging will never give a very accurate bill.

Note: We have heard that GSM Association BARG has urged operators to charge on volumes, and not (only) on duration.

Another reason why operators want to use time based charging is for compatibility with their existing billing equipment. On the other hand, CG could perform the conversion from octets (volume) to "equivalent time units".

Note that using time-based charging with NRCA is equivalent to using charging of time of activity of the MS with always-on concept, and this could be supported by an appropriate CDR solution.

# B.10 Type of IP addresses used

Sometimes NRCA has been mistaken as a mean to save IP addresses.

## B.10.1 NRCA with static addresses

This solution requires one IP address per subscriber and is the most address consuming solution. It requires using IPv6 or private IPv4 addresses.

## B.10.2 NRCA with dynamic address and MSISDN addressing

This solution requires one IP address per active user and is the least address consuming solution. However the operator needs to have enough IP addresses to serve burst of push messages. In addition, the address is needed during all the duration of the push session and during a certain guard time (maybe 10 minutes to ensure previous TCP connection is stopped). So to be able to serve significant burst of traffic, it also requires using IPv6 or private IPv4 addresses.

## B.10.3 Always-on:

This solution requires one IP address per connected user. It requires using IPv6 or private IPv4 addresses.

## B.10.4 Conclusion

Any scalable push solution requires using IPv6 or private IPv4 addresses. So the number of addresses is not an issue. One operator may have as many private Ipv4 network (each supporting around 17 millions addresses) as they want. IPv6 is recommended as a future proof solution.

Note: In an operator network, different GGSN may be connected to different private IPv4 network providing infinite number of private IPv4 address. It requires that the operator replicates the same services in these different private networks.

## B.11 FINAL Conclusion

Always on concept and NRCA can provide the same services. However, always-on can provide these services faster, cheaper and in a more scalable way.

NRCA gives the illusion than it can save (in particular GGSN investment). We believe that a serious study of the cost of installing enough signalling capacity to cope with burst of push message and the cost of maintaining more complex system would clear this illusion.

# Annex C (Informative): Comparison of NRCA and SMS as a push solution

Note: The conclusion given is conclusion related to this study, not the conclusion for the whole TR.

## C.1 Introduction

NRCA using MSIDN is being proposed to be used by push services. These two are compared below with SMS, which already uses MSISDN addressing. In addition, SMS combined with WAP[18] already support versatile push services. The operator may use to propose service to the user (WAP Service indication), or push information directly (WAP Service Load can automatically trigger PDP context activation from the MS). In addition, it is possible to define special SMS that always reach the MS (even if MS memory is full). In addition, SMS supports features like Store and Forward and broadcasting inherently.

## C.2 MM signalling required

In order to receive SMS you may be GPRS attached or IMSI attached. So for a typical user who want to receive SMS-based push messages and calls, the MS can be only CS attached and not packet attached. This saves all the Routing Area update signalling for the GPRS Mobility management. It also requires less SGSN capacity.

In order to receive NRCA request, you have to be GPRS attached. So for a typical user who want to receive NRCA-based push messages and calls, the MS must be both CS attached and packet attached. Both Location Area and Routing Area update signalling must then be performed.

In summary, in the case of the MS supports CS and the MS rarely attaches GPRS, an SMS based push solution generates less MM signalling and requires less SGSN capacity than an NRCA based solution.

## C.3 Signalling during push delivery

To compare NRCA and SMS as a push solution, we use the example of a football game result. The message is short (fitting in an SMS) and proposing interactive links to get more information or view a video.

The pushed message may be:

> "France beat Brazil 3.0, and become world champion!
>
> - View game summary (3 minutes video)
>
> - Connect French team WWW
>
> - Play Adidas game to win a picture from Zidane"

If NRCA is used the MS needs to activate a PDP context to receive this message; while with SMS it will be received directly. With SMS a PDP context need to be activated if the user click on any on the link (the signalling is shown on figure C.2 below, PDP context activation (10.) being marked as optional using dotted line).

Another difference is that SMS is acknowledged over the radio (figure C.2 message 8.), while request PDP context activation is not.

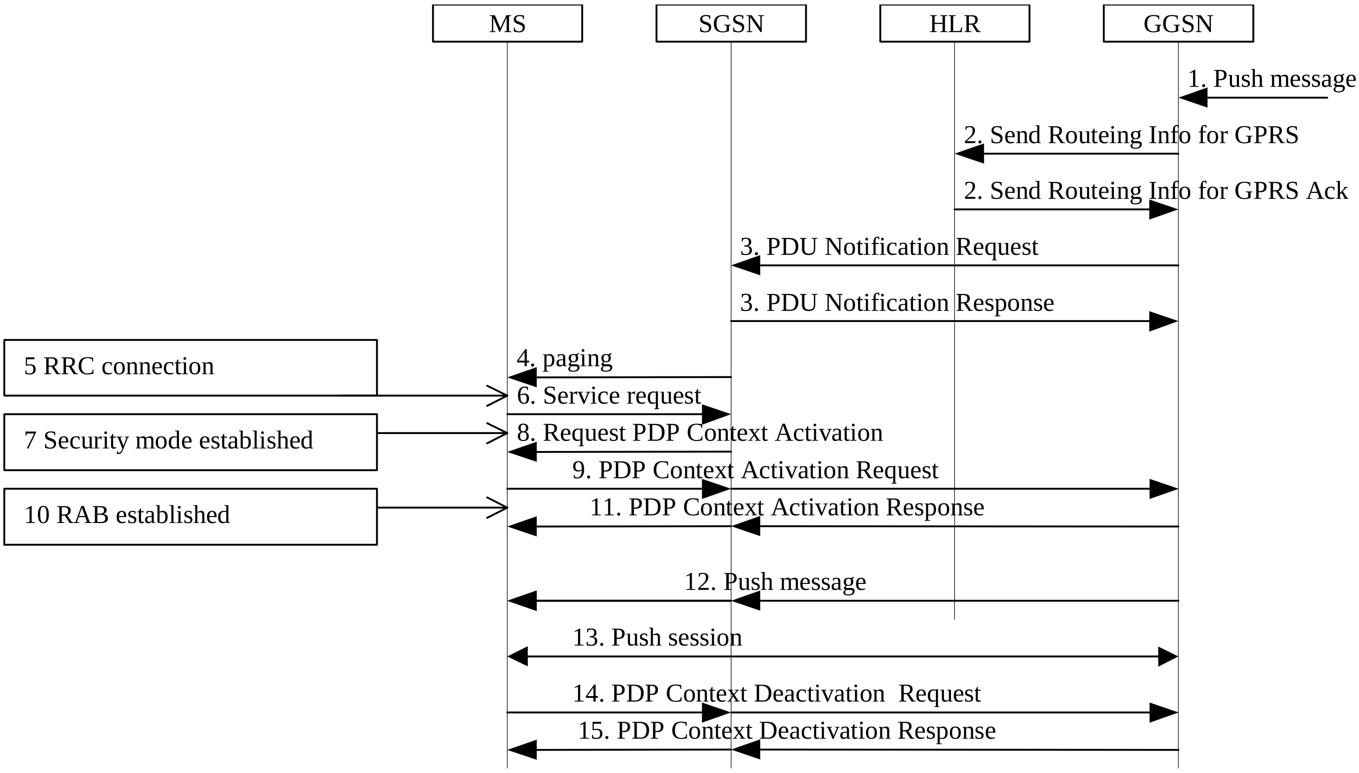

MS
SGSN
HLR
GGSN
1. Push message
2. Send Routeing Info for GPRS
2. Send Routeing Info for GPRS Ack
3. PDU Notification Request
3. PDU Notification Response
5 RRC connection
4. paging
6. Service request
7 Security mode established
8. Request PDP Context Activation
9. PDP Context Activation Request
10 RAB established
11. PDP Context Activation Response
12. Push message
13. Push session
14. PDP Context Deactivation  Request
15. PDP Context Deactivation Response

Figure C.1: Signalling for push message delivery using NRCA

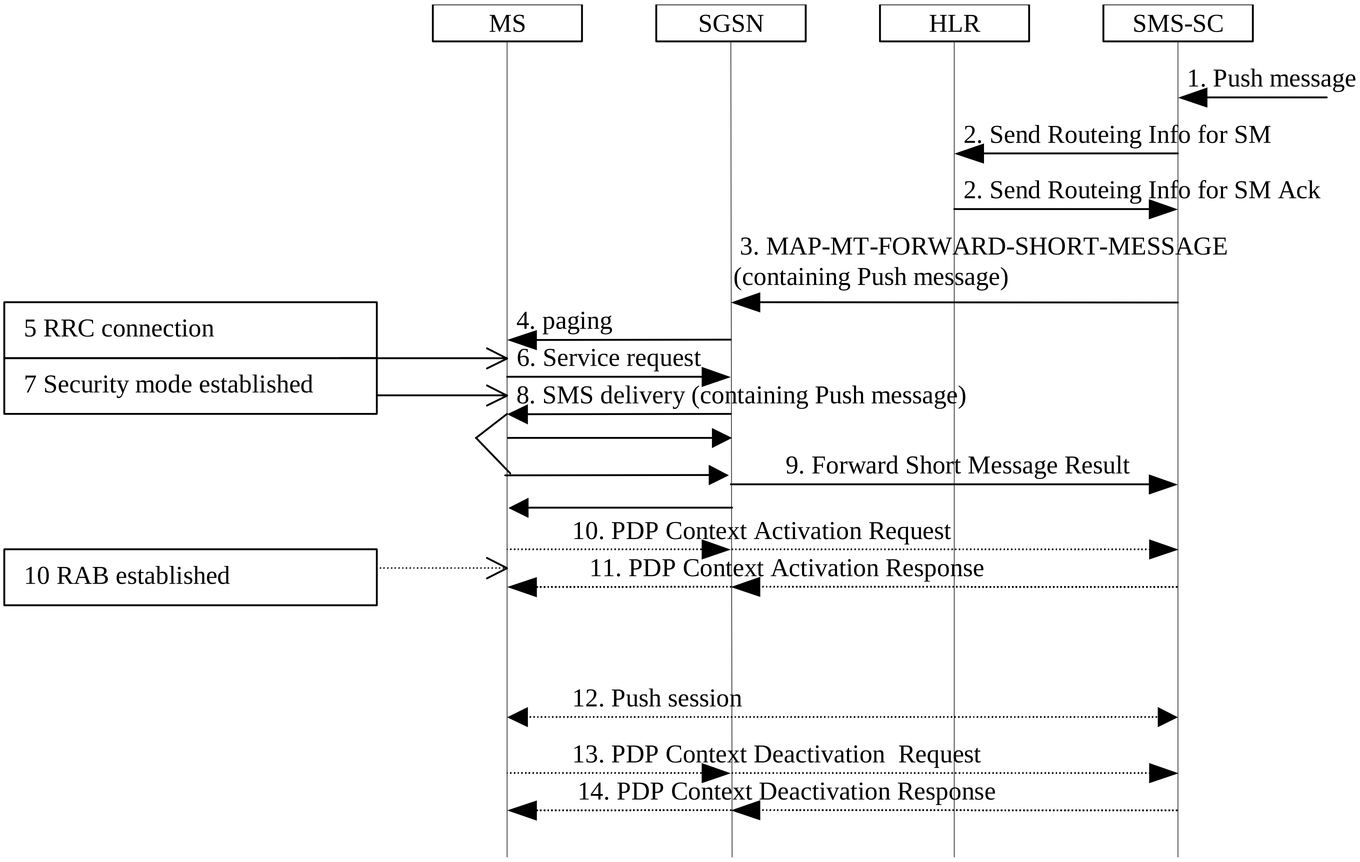

MS
SGSN
HLR
SMS-SC
1. Push message
2. Send Routeing Info for SM
2. Send Routeing Info for SM Ack
3. MAP-MT-FORWARD-SHORT-MESSAGE (containing Push message)
5 RRC connection
7 Security mode established
4. paging
6. Service request
8. SMS delivery (containing Push message)
9. Forward Short Message Result
10. PDP Context Activation Request
10 RAB established
11. PDP Context Activation Response
12. Push session
13. PDP Context Deactivation Request
14. PDP Context Deactivation Response

**Figure C.2: Signalling for push message delivery using SMS**

A comparison of picture 1 and 2 shows that:

- If the user click on one of the links, the amount of signalling over the radio is two messages lessl in NRCA than SMS.
- If the user do not click on any link (i.e. no push session), the amount of signalling is clearly smaller in SMS.

## C.4 Conclusion

Using SMS for push service has following advantages:

- It is already fully standardised (both SMS and WAP1.2 are ready)
- SMS has as an already defined feature store and forward functionality, and may be broadcasted.

# Annex <X>:
Change history

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