3GPP TSG CN Plenary Meeting #16 **NP-020167**
5th - 7th June 2002. Marco Island, USA.

Source:	**TSG_CN WG3**
Title:	**TS29.207, Version 2.0.0**
Agenda item:	**8.5**
Release:	**Rel-5**
Document for:	**APPROVAL**

Abstract of document:

The specification provides the stage 3 specification of the Go interface.

3GPP TS 29.207 V2.0.0 (2002-06)

Technical Specification

3rd Generation Partnership Project;
Technical Specification Group Core Network;
Policy control over Go interface
(Release 5)

Keywords
UMTS, QoS

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 present document provides the stage 3 specification of the Go interface. The functional requirements and the stage 2 specifications of the Go interface are contained in 3GPP TS 23.002 [2] and 3GPP TS 23.207 [3]. The Go interface is the interface between the GGSN and the Policy Control Function (PCF).

The present document defines:

- the protocol to be used between PCF and GGSN over the Go interface;
- the signalling interactions to be performed between PCF and GGSN over the Go interface;
- the information to be exchanged between PCF and GGSN over the Go interface.

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

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

[2] 3GPP TS 23.002: "Network architecture".

[3] 3GPP TS 23.207: "End to end quality of service concept and architecture".

[4] 3GPP TS 23.228: "IP Multimedia Subsystem (IMS); Stage 2".

[5] IETF RFC 2475: "An Architecture for Differentiated Services".

[6] IETF RFC 2753: "A Framework for Policy-based Admission Control".

[7] IETF RFC 2748: "The COPS (Common Open Policy Service) Protocol".

[8] IETF RFC 3084: "COPS Usage for Policy Provisioning (COPS-PR)".

[9] IETF RFC 3159: "Structure of Policy Provisioning Information (SPPI)".

[10] IETF RFC 2205: "Resource ReSerVation Protocol (RSVP) – Version 1 Functional Specification".

[11] IETF RFC tbd: "Session Authorisation for RSVP" (draft-ietf-rap-rsvp-authsession-02.txt).

[12] 3GPP TS 24.008: "Mobile Radio Interface Layer 3 specification; Core network protocols; Stage 3".

[13] 3GPP TS 27.060: "Mobile Station (MS) supporting Packet Switched Services".

3 Definitions and abbreviations

3.1 Definitions

For the purposes of the present document, the terms and definitions given in 3GPP TR 21.905 [1] and the following apply:

Common Open Policy Service (COPS) protocol: is a simple query and response protocol that can be used to exchange policy information between a policy server (Policy Decision Point) and its clients (Policy Enforcement Points)

Differentiated Services (DiffServ): Diffserv networks classify packets into one of a small number of aggregated flows or "classes", based on the DiffServ codepoint (DSCP) in the packet's IP header
This is known as behaviour aggregate (BA) classification. At each DiffServ router, packets are subjected to a "per-hop behaviour" (PHB), which is invoked by the DSCP.

Flow identifier: used for the identification of an IP flow within a media component associated with a SIP session
For example, a single, unidirectional media component may contain one IP flow, or two IP flows in the case of an RTP media stream. In case of a bidirectional flow, the same flow identifier is used for both directions. A flow identifier consists of two parts: 1) Media component number defined in increasing order according to the sequence of the "m=" lines in the SDP session description and 2) IP flow number defined in the order of increasing port numbers within each media component.

Go Interface: interface between PCF and GGSN [2]

IP Bearer Service Manager: uses standard IP mechanisms to manage the IP Bearer Service. It resides in the GGSN and optionally in the UE

Media component: is a part of an SDP session description conveying information about one media stream (e.g. type, format, IP address, port, transport protocol, bandwidth, direction)
The media stream described by a media component can be either bi- or unidirectional. A media stream containing an RTP flow may also contain an associated RTCP flow. An SDP session description can consist of more than one media components. A media component shall not be deleted nor its position changed within the SDP session description.

Policy Control Function (PCF): is a logical policy decision element that uses standard IP mechanisms to implement policy in the IP media layer
The PCF makes decisions in regard to network based IP policy using policy rules, and communicates these decisions to the PEP in the GGSN.

Proxy Call Session Control Function (P-CSCF): is a network element providing session management services (e.g. telephony call control)

Policy Enforcement Point (PEP): is a logical entity that enforces policy decisions made by the PCF. It resides in the IP BS Manager of the GGSN

Policy Information Base (PIB): data carried by COPS-PR is a set of policy data
The protocol assumes a named data structure, known as a Policy Information Base (PIB), to identify the type and purpose of solicited and unsolicited policy information that is sent from the Policy Decision Point to the Policy Enforcement Point for provisioning policy or sent from the Policy Enforcement Point to the Policy Decision Point as a notification.

Provisioning Instance Identifier (PRID): uniquely identifies an instance of a PRC

Resource ReSerVation Protocol (RSVP): is used by a host to request specific qualities of service from the network for particular application data streams or flows
The network responds by explicitly admitting or rejecting RSVP requests.

Translation/mapping function: provides the inter-working between the mechanisms and parameters used within the UMTS Bearer Service and those used within the IP Bearer Service

UMTS Bearer Service Manager: handles resource reservation requests from the UE. It resides in the GGSN and the UE

3.2 Abbreviations

For the purposes of the present document, the abbreviations as specified in 3GPP TR 21.905 [1] and the following abbreviations apply:

COPS	Common Open Policy Service protocol
COPS-PR	COPS for policy PRovisioning
DEC	COPS DECision message
DiffServ	Differentiated Services
DRQ	COPS Delete ReQuest state message
DSCP	DiffServ Code Point
GCID	GPRS Charging IDentifier
ICID	IMS Charging IDentifier
IMS	IP Multimedia core network Subsystem
PCF	Policy Control Function
P-CSCF	Proxy Call Session Control Function
PEP	Policy Enforcement Point
PHB	Per Hop Behaviour
PIB	Policy Information Base
PRC	PRovisioning Class (a type of policy data)
PRI	PRovisioning Instance (an instance of a PRC)
PRID	PRovisioning Instance iDentifier
QoS	Quality of Service
REQ	COPS REQuest message
RPT	COPS RePorT state message
RSVP	resource ReSerVation Protocol
RTCP	RTP Control Protocol

4 Go interface

4.1 Overview

The Go interface allows service-based local policy and QoS inter-working information to be "pushed" to or requested by the Policy Enforcement Point (PEP) in the GGSN from a Policy Control Function (PCF). As defined in the stage 2 specifications [3], this information is used by the GGSN for:

- GPRS bearer authorisation;
- Charging correlation;
- Policy based "gating" function in GGSN;
- Control of DiffServ inter-working;
- Control of RSVP admission control and inter-working.

The Go interface uses IP flow based policies.

The Common Open Policy Service (COPS) protocol has been developed as a protocol for use between a policy server and a network device, as described in [7].

In addition, COPS for Provisioning extensions have been developed as described in [8] with [9] describing a structure for specifying policy information that can then be transmitted to a network device for the purpose of configuring policy at that device. The model underlying this structure is one of well-defined provisioning classes and instances of these classes residing in a virtual information store called the Policy Information Base (PIB).

The Go interface shall conform to the IETF COPS [7] and the extensions of COPS-PR [8]. For the purpose of exchanging the required specific UMTS information, a COPS-PR Policy Information Based (PIB) is defined in the present document.

COPS Usage for Policy Provisioning (COPS-PR) is independent of the type of policy being provisioned (QoS, Security, etc.). In the present document, COPS-PR is used to communicate service-based local policy information between PCF and GGSN. COPS-PR can be extended to provide per-flow policy control along with a 3GPP Go Policy Information Base (PIB). The 3GPP Go PIB may inherit part of the data object definitions from the framework PIB and the DiffServ PIB defined in the IETF.

The minimum functionalities that the Go interface shall cover are introduced below.

1. Media Authorisation request from GGSN:

   The GGSN receives the binding information during the activation of a (Secondary) PDP context or during the modification of an existing PDP context that has been previously authorized by the PCF. To authorise the PDP context activation, the GGSN shall send a media authorisation request to the PCF. To authorise the PDP context modification, the GGSN shall send a media authorisation request to the PCF when the requested QoS exceeds the authorised QoS or new binding information is received.

   This authorisation request shall include the following information:

   - Binding information:

     The binding information is used by the GGSN to identify the correct PCF and subsequently request service-based local policy information from the PCF. The GGSN may receive one or more sets of the binding information during an activation or modification of a PDP context. Each binding information consists of:

     - One Authorisation token;

     - One or more Flow id(s) within the session.

   It is assumed that only one set of binding information is carried within a PDP context in this Release.

2. Media authorisation decision from PCF:

   The media authorisation information sent by the PCF to the GGSN, contains at a minimum the following information:

   - Decision on the binding information.

   The PCF shall respond with an authorisation decision for the binding information. The authorisation decision shall identify that the binding information is validated with an ongoing SIP session. Additionally, the PCF shall verify if the multiple media components are correctly assigned to the PDP Context. If validated, the PCF shall also communicate the following media authorisation details to the GGSN:

   - "Authorised QoS".

     This information is used by the GGSN to authorise the media resources according to the service-based local policy and the requested bearer QoS.

     The "Authorised QoS" for media components signalled over the Go interface is based on the SDP requirements signalled and agreed previously within SIP signalling for this session.

     The "Authorised QoS" specifies the maximum QoS that is authorised for a PDP context for that specific binding information. In case of an aggregation of multiple media components within one PDP context, the combination of the "Authorised QoS" information of the individual media components is provided as the "Authorised QoS" for the bearer.

     The "Authorised QoS" contains the following information:

     - DiffServ class:

       The DiffServ class determines the highest PHB that can be used for the media component. It is derived from the media type information of the SDP media description.

- Data rate:

  The Data rate information is extracted from the SDP bandwidth parameter, more specifically the bandwidth value indicated by the "b=AS:" parameter. The Data rate shall include all the overhead coming from the IP-layer and the layers above, e.g. UDP, RTP. The Data rate shall also include the overhead coming from the possible usage of RTCP. The Data rate within the "Authorized QoS" information for the bearer is the combination of the data rate values of the authorised QoS of the individual media components.

- Packet Classifier.

  The packet classifier for media components is based on the IP-address and port number information in the SDP and shall allow for all IP flows associated with the SDP media component description.

3. Charging correlation:

   The PCF shall send the ICID provided by the P-CSCF as part of the authorisation decision. The GGSN shall send the GCID of the PDP Context and the GGSN address to the PCF as part of the authorisation report.

4. Approval of QoS Commit / Removal of QoS Commit / Revoke Authorisation for GPRS and IP resources:

   The PCF controls media components and may revoke resources at any time. Approval of QoS Commit / Removal of QoS Commit / Revoke Authorisation for GPRS and IP resources is communicated by the PCF to the GGSN.

5. Indication of PDP Context Release / Modification to/from 0 kbit/s:

   The GGSN informs the PCF of bearer changes related to the authorised resources for the IMS session in the following cases:

   - Loss of radio contact (modification to/from 0 kbit/s for conversational and streaming class);

   - Deactivation of PDP context.

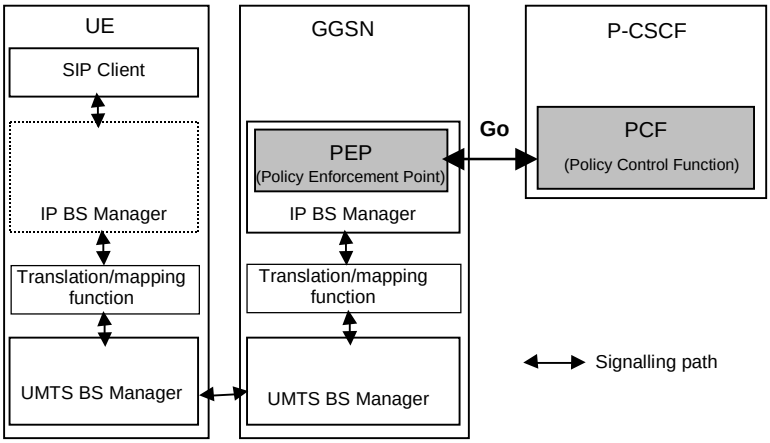
4.2 Go reference model

The Go interface is defined between the PCF and the GGSN [2].

The PCF is a logical entity of the P-CSCF (if the PCF is implemented in a separate physical node, the interface between the PCF and P-CSCF is not standardised).

The P-CSCF(PCF) is in the same PLMN as the GGSN.

The relationships between the different functional entities involved are depicted in figure 4.2.



NOTE: For clarity in the diagram, network elements that are not involved in service-based local policy are not presented here (e.g. radio network elements, SGSN, etc).

Figure 4.2: Go interface architecture model

4.3 Functional elements and capabilities

4.3.1 GGSN

4.3.1.1 Service-based local policy enforcement point

The Service-based Local Policy Enforcement Point (PEP) is a logical entity which resides in the GGSN and communicates with the PCF regarding Service-based local policy control. Hereafter in the present document, the GGSN is assumed to contain the PEP implicitly unless otherwise stated. The GGSN sends requests to and receives decisions from the PCF. The GGSN may cache the policy decision data of the PCF decisions. This cached information may be used later for a local policy decision allowing the GGSN to make policy control decision about the QoS authorization for PDP context modifications without requiring additional interaction with the PCF.

The following service-based local policy enforcement point functionalities in the GGSN are identified:

- Authorisation request:

  The GGSN requests authorisation information from PCF for the media components carried by a PDP context. The GGSN enforces the PCF decisions related to the media components carried by a PDP context.

- Authorisation report:

  The GGSN shall also report to the PCF its success or failure in carrying out the PCF decision.

- Policy based admission control:

  The GGSN includes policy-based admission control that is applied to the bearers associated with the media components, and configures the policy based "gating" functionality in the user plane.

  Policy-based admission control ensures that the GPRS bearer carrying media components, which is activated in the GGSN, is authorised by the PCF decision.

  Additionally, policy-based admission control ensures that the resources, which can be used by each particular media component, are within the "Authorised QoS" specified by the PCF. This information is mapped by the Translation/mapping function in the GGSN to give the authorised resources for GPRS bearer admission control.

  To ensure charging correlation, the PEP shall send the GPRS charging identifier and the GGSN address to the PCF.

- Policy based gating functionality:

  Policy based gating functionality represent the control of the GGSN over the Gate Function in the user plane, i.e. the forwarding of IP packets associated with a media component. In the user plane, a "gate" is defined for each direction of a media component. The PCF provides the gate description and the commands to open or close the gate. The gate description is received from the PCF in the authorisation decision. The command to open or close the gate shall be sent either in the authorisation decision or in subsequent decisions from the PCF.

4.3.1.1.1 QoS Information processing

The GGSN is responsible for the policy based admission control, i.e. to ensure that the requested QoS is in-line with the "Authorized QoS".

The GGSN needs the "Authorised QoS" information of the PDP context for the uplink as well as for the downlink direction. Therefore, the "Authorized QoS" information for the combination of all IP flows of each direction associated with the media component as determined by the PCF is used.

In case of an aggregation of multiple media components within one PDP context, the "Authorised QoS" for the bearer is provided by the PCF as the combination of the "Authorised QoS" information of the individual media components.

The GGSN shall perform the proper mapping between the IP QoS information and the UMTS QoS information. This mapping is performed by the Translation/mapping function which maps the "Authorised QoS" information for the PDP context into authorised UMTS QoS information.

It is recommended, the GGSN to derive the highest allowed UMTS Traffic class for the PDP context from the Diffserv PHB in the "Authorized QoS" according to table 4.3.1.1.1.

Table 4.3.1.1.1

Diffserv PHB	Traffic Class	Traffic Handling Priority
EF	Conversational	N/A
$AF4_1$	Streaming	N/A
$AF3_1$	Interactive	1
$AF2_1$		2
$AF1_1$		3
BE	Background	N/A

The Data rate within the "Authorized QoS" information for the bearer is the combination of the data rate values of the "Authorised QoS" of the individual media components and shall be used by the GGSN as the maximum bandwidth value for the PDP context. This bandwidth value shall include all the overhead coming from the IP-layer and the layers above. If RTP is used, then all the overhead coming from the UDP, RTP and RTCP layers shall be included.

In the case of real-time UMTS bearers (conversational and streaming traffic classes), the Data rate value of the "Authorized QoS" information shall be considered as the maximum value of the 'Guaranteed bitrate' UMTS QoS parameter. In the case of non-real-time bearers (interactive and background traffic classes), the Data rate value shall be considered as the maximum value of the 'Maximum bitrate' UMTS QoS parameter.

Editor's note: Mapping the Data rate value for the real time into 'Guaranteed bitrate' or 'Maximum bitrate' parameter is for FFS.

The UMTS BS Manager receives the authorised UMTS QoS information for the PDP context from the Translation/mapping function. If the requested QoS exceeds the authorised QoS it may either reject the activation/modification of the PDP context or downgrade the requested UMTS QoS information to the authorised UMTS QoS information. In case of rejection of the activation/modification, the authorization failure is indicated to UE in the Protocol Configuration Options information element as defined in 3GPP TS 24.008 [12].

The GGSN may store the authorized QoS for the binding information of an active PDP context in order to be able to make local decisions, when the UE requests for a PDP context modification.

4.3.1.2 Initialisation and maintenance

Editor's note: This describes the initialisation and maintenance of the COPS protocol over Go interface. It may be simplified by referring to IETF RFC.

4.3.1.3 Gate function

The Gate Function represents a user plane function enabling or disabling the forwarding of IP packets. A gate is described by a set of packet classifiers that identify IP flows associated to the gate. The packet classifier includes the standard 5-tuple (source IP address, destination IP address, source port, destination port, protocol) explicitly describing a unidirectional IP flow.

The packet classifier is received from the PCF in an authorisation decision. In the packet classifier the source IP address and the source port number are wildcarded by the PCF.

Editor's note: The wildcarding of the source IP address maybe updated depending on the SA2's decision.

The GGSN installs the packet filter applying the packet classifier. After installation of the packet filter the gate shall be closed until the GGSN receives a command to open the gate.

The commands to open or close the gate lead to the enabling or disabling of the passage for IP packets. If the gate is closed all packets of the related IP flows are dropped. If the gate is opened the packets of the related IP flows are allowed to be forwarded. The opening of the gate may be part of the authorisation decision event. The closing of the gate may be part of the revoke authorisation decision event.

IP Packets of a PDP context not matching any packet classifier associated with this PDP context shall be dropped.

If the packet classifier is included as an additional IE in the authorisation information, the GGSN shall check for validity of the TFT in the Create PDP Context Request or Update PDP Context Request. If the TFT proposed will result in packets from the media component being unable to pass through, the PDP context will be rejected with cause value indicating a semantic error in the TFT.

Editor's note: This issue should still be discussed in SA2.

4.3.1.4 DiffServ edge function

Editor's Note: This clause describes the functionality of "DiffServ Edge Function" in GGSN. This is dependent on SA2 decision.

4.3.1.5 Binding mechanism handling

The binding information is used by the GGSN to identify the correct PCF and subsequently request service-based local policy information from the PCF. The binding information associates a PDP context with one or more media components of an IMS session. The GGSN may receive one or more sets of the binding information during an activation or modification of a PDP context. Each binding information consists of an authorisation token and the flow identifier(s) related to the IP flows of the actual media component. If there is more than one media component to be transported within the PDP context the binding information includes the flow identifier(s) for the IP flows of each of the media components.

The GGSN shall store the binding information and apply it to correlate events and actions between the PDP context and the service-based local policy.

The GGSN shall determine the IP address of the PCF from the PCF identifier received as part of the Authorization Token. This identifier shall be in the format of a fully qualified domain name.

The GGSN shall forward the binding information received from the UE to the PCF. If multiple binding information are received by the GSSN, it shall forward them to the PCF. If none of the tokens included in the binding information are of type AUTH_SESSION, or they do not contain an AUTH_ENT_ID attribute to resolve the PCF address, then the GGSN shall reject the PDP context activation request.

When the GGSN receives a PDP context activation/modification to the IMS APN without the binding information the GGSN shall reject the PDP context activation/modification request. The authorization failure is indicated to UE in the Protocol Configuration Options information element as defined in 3GPP TS 24.008 [12].

4.3.2 PCF

4.3.2.1 Service-based local policy decision point

The PCF functions as a Policy Decision Point for the service-based local policy control. The PCF makes policy decisions based on session and media related information obtained from the P-CSCF. The PCF shall exchange the decision information with the GGSN via the Go interface.

The following service-based local policy decision point functionalities are identified:

- Authorisation function:

  The PCF shall be able to provide an authorisation decision upon receiving a bearer authorisation request from the GGSN. The PCF shall authorise the request according to the stored session and media related information received from the P-CSCF.

Editor's Note: a potential for theft of service scenario has been identified with the current mechanism for authorisation. Extensions to the authorisation mechanisms to close potential theft of service scenarios are currently under investigation, and will be specified when determined.

- Revoke function:

  The PCF may revoke the authorisation of resources at any time. Revoke Authorisation for GPRS and IP resources is communicated by the PCF to the GGSN.

- Approval of QoS Commit / Removal of QoS Commit:

  The PCF may allow or deny for the media component(s) the usage of the PDP context by controlling the correlated gate(s).

  The "Approval of QoS Commit" command may either be part of the authorisation decision, or the PCF may provide a separate decision with the "Approval of QoS Commit" command to open the gate.

  The "Removal of QoS Commit" command may either be part of the revoke authorisation decision, or the PCF may provide a separate decision with the "Removal of QoS Commit" command to close the gate.

- Actions due to Indication of bearer release:

  When the GGSN informs the PCF of bearer deactivation, the PCF shall remove the corresponding authorisation request state. Additionally, the PCF shall inform the P-CSCF about this deletion event.

- Actions due to Indication of bearer modification:

  When the PCF receives an indication of bearer modification of the maximum bitrate to or from 0 kbits/s, the PCF shall inform the P-CSCF about this modification event.

- Generation of authorisation token:

  During the session set-up the PCF generates an authorisation token for the IMS session.

- Mapping SDP parameters to "Authorized QoS" parameters:

  To perform proper authorisation, the PCF shall map the necessary SDP parameters containing session and media related information to "Authorized QoS" parameters.

- Charging identifiers exchange:

  The PCF shall send the ICID provided by the P-CSCF as part of the initial authorisation decision of all the bearer authorization requests that correspond to the respective SIP session.

  When the PCF receives the GCID together with the GGSN address from the GGSN, it shall forward these information to the P-CSCF to ensure charging correlation.

4.3.2.2 Initialisation and maintenance

Editor's note: This describes the initialisation and maintenance of the COPS protocol over Go interface. It may be simplified by referring to IETF RFC.

4.3.2.3 Binding mechanism handling

The binding information is used by the GGSN to identify the correct PCF and subsequently request service-based local policy information from the PCF. Each set of binding information consists of an authorisation token and one of more flow identifier(s).

During the session set-up the PCF generates an Authorisation Token for the IMS session. The Authorisation token is forwarded to the UE by SIP signalling. The PCF shall allocate its PCF identifier as part of the Authorization Token. This identifier shall be in the format of a fully qualified domain name.

The PCF receives the binding information and a Client Handle as part of a REQ from the GGSN. The PCF shall store the Client Handle for each media component identified by the binding information for subsequent message exchanges.

The authorisation token is applied by the PCF to identify the IMS session. If no IMS session can be found for an authorisation token, or if the PCF is otherwise unable to authorise the binding information, the PCF shall send a COPS decision message carrying both an INSTALL and REMOVE decision. The INSTALL decision shall identify an authorisation failure to the GGSN, and may include further details identifying the cause. The REMOVE decision shall subsequently remove this state from the GGSN. For an initial authorisation, the PCF shall then initiate a remove for the authorisation request.

For a valid authorisation token the flow identifier is used to select the available information on the media component of this IMS session. The PCF sends the available information on the media component back to the GGSN.

If the binding information consists of more than one flow identifier, the PCF shall also verify that the media components identified by the flow identifiers are allowed to be transferred in the same PDP context. If any of these media components was mandated to be carried in a separate PDP Context, the PCF shall send a COPS decision message carrying both an INSTALL and REMOVE decision. The INSTALL decision shall identify an authorisation failure to the GGSN, and may include further details identifying the cause. The REMOVE decision shall subsequently remove this state from the GGSN. For an initial authorisation, the PCF shall then initiate a remove for the authorisation request.

For a valid binding information consisting of more than one flow identifier, the information sent back to the GGSN shall include the aggregated QoS for all the flows and a packet filter for each flow. The flow identifiers within the binding information can span one or more media components.

5 Policy control procedures

5.1 GGSN

5.1.1 Initial authorization at PDP context activation

The GGSN receives binding information during the activation of a PDP context by the UE. To perform initial authorization at the PDP context activation the GGSN shall send an authorisation request to the PCF including the binding information received from the UE.

The GGSN identifies the required PCF from the binding information. The binding information is formatted according to the structure of the policy element defined in [11] and shall include the AUTH_ENT_ID and the SESSION_ID attributes. The GGSN checks for a Policy Element of type AUTH_SESSION ([11]) and retrieves the AUTH_ENT_ID attribute from this. If this is in the form of a Fully Qualified Domain Name, then this is used to identity the correct PCF.

The GGSN authorisation request message to the PCF shall allow the GGSN to request policy information for authorisation of the media components carried by a PDP context identified by binding information.

When the GGSN receives the PCF decision regarding authorisation of the media components, the GGSN shall enforce the policy decision.

The PCF shall verify the binding information by checking if the authorization token is associated with an ongoing SIP session at IMS level and by checking if the media components are allowed to be grouped.

If the PCF decision information indicates that the binding information provided by the GGSN is associated with an ongoing SIP session at IMS level, the GGSN shall proceed with activation of the PDP context. The GGSN shall map the authorized QoS resources into authorized resources for the bearer admission control.

To ensure charging correlation, the GGSN shall send the GPRS charging identifier and GGSN address information to the PCF after the successful establishment of the PDP context, i.e. with the report following the initial authorization decision.

When the PCF detects that the binding information provided by the GGSN is not associated with an ongoing SIP session at application layer, or is otherwise unable to authorise the binding information, the GGSN will receive a COPS decision message from the PCF carrying both an INSTALL and REMOVE decision. The GGSN shall reject the PDP context activation, using any received decision information from the PCF to identify the error reason. The GGSN shall subsequently remove this state according to the REMOVE decision. For an initial authorisation request, the GGSN shall then send a COPS Delete Request State (DRQ) message to the PCF to remove the state in the GGSN and the PCF. The authorization failure is indicated to UE in the Protocol Configuration Options information element as defined in 3GPP TS 24.008 [12].

Upon receiving a Remove decision from the PCF for the PDP context authorisation, the GGSN shall reject the PDP context and shall delete the Request-state that has been established in the PCF and the GGSN by sending the COPS Delete Request State (DRQ) message to the PCF. The authorization failure is indicated to UE in the Protocol Configuration Options information element as defined in 3GPP TS 24.008 [12].

When the GGSN sends an authorization request to the PCF but the PCF doesn't respond with the decision message, the GGSN's action is according to the local policy in the GGSN. The local policy may be configured by the operator.

If the GGSN supports a local policy decision point (LPDP) configuration it may make local policy decisions in the absence of the PCF. The local policy decisions may be used to accept new PDP context activations while the connection to the PCF is lost. The synchronization behaviour between the GGSN and the PCF is based on the local policy configured by operators.

5.1.2 Modification of previously authorized PDP context

The GGSN is responsible for notifying the PCF when a procedure of PDP context modification of a previously authorized PDP context is performed. To authorise the PDP context modification the GGSN shall send an authorisation request to the PCF including the binding information received from the UE in the following cases:

- Requested QoS exceeds "Authorised QoS";
- New binding information is received.

The GGSN on receiving the PDP context modification request from the UE will verify the authorisation. If the GGSN does not have sufficient information to authorize the PDP context modification request then the GGSN shall interrogate the PCF for modification request authorisation.

If the requested QoS is within the already "Authorized QoS" and the binding information is not changed, the GGSN need not send an authorization request to the PCF.

The GGSN is responsible for notifying to the PCF when the procedure of the PDP context modification is performed in the following cases:

- Requested QoS maximum bit rate is 0 kbit/s;
- Requested QoS maximum bit rate changes from 0 kbit/s.

5.1.3 PDP context deactivation

The GGSN is responsible for notifying the PCF when a procedure of a PDP context deactivation is performed. In case of a PDP context deactivation, the GGSN shall inform the PCF of the bearer release related to the SIP session.

When a revoke authorisation for the set of media components on that PDP context is performed, the GGSN receives a decision message from the PCF for disabling the use of the "Authorised QoS" resources and deactivation of the PDP context associated with the binding information. The GGSN shall disable the use of the "Authorized QoS" resources. The GGSN shall initiate deactivation of the PDP context used for carrying these media components, in case that the UE has not performed it within an operator specific time.

5.1.4 User plane operation

The GGSN shall enforce the configuration of the policy based "gating" functionality according to additional authorisation information received from the PCF.

Editor's note: the exact GGSN action if the "gating" parameters provided by the PCF are not identical with the parameters from the TFT in the PDP context request is for further study.

5.2 PCF

5.2.1 SBLP decisions

5.2.1.1 SBLP authorisation decision

The information needed for the PCF to perform media authorization is passed by the P-CSCF upon receiving a SIP message that contains SDP. The SDP contains sufficient information about the session, such as the end-points' IP address and port numbers and bandwidth requirements.

All media components in the SDP are authorised. The media components contain one or more IP flows each represented by a flow identifier. Cf. the definition of flow identifier in clause 3.1. A flow identifier is expressed as a 2-tuple as follows:

<Media component no, IP flow no.>

where both are numbered starting from 1.

0 3

Media component no.	IP flow no.

As an example, if the second "m=" - line in the SDP information contains one RTP media specification, the following flow identifiers would be assigned:

IP flow	Flow id.
RTP	<2,1>
Associated RTCP	<2,2>

The P-CSCF shall send policy setup information to the PCF upon every SIP message that includes an SDP payload. This ensures that the PCF passes proper information to perform media authorization for all possible IMS session setup scenarios. The policy setup information provided by the P-CSCF to the PCF for each media component shall contain the following:

- Destination IP address;
- Destination port number;
- Transport Protocol id;
- Media direction information;
- Direction of the source (originating or terminating side);
- Indication of the group that the media component belongs to;

Editor's note: The format of this group indication in SIP/SDP is subject to CN1's decision.

- Media type information;
- Bandwidth parameter.

Additionally, upon the P-CSCF receives the ICID in SIP signalling, it shall send the ICID to the PCF.

The PCF stores the authorised policy information, and generates an Authorisation Token to identify this decision. The Authorisation Token is passed back to the P-CSCF for inclusion in the SIP signalling back to the UE.

The Authorisation Token is in the form of a Session Authorisation Data Policy Element as described in [11]. The PCF shall include an AUTH_ENT_ID attribute containing the Fully Qualified Domain Name of the PCF and the SESSION_ID attribute.

Upon receiving the bearer authorization request from the GGSN, the PCF shall authorize the request according to the stored service based local policy information for the session identified by the binding information in the request.

- Decision on the binding information:

  The authorisation shall contain the decision on verifying the binding information. The PCF shall identify whether the binding information indeed corresponds to an initiated SIP session.

  The authorization shall also contain decision on the list of flow_IDs contained in the bearer authorisation request sent by the GGSN representing the list of media components intended to be carried in the same PDP Context. This decision shall verify that these media components are indeed allowed to be carried in the same PDP Context. The PCF shall make this decision by comparing the list of flow_IDs contained in the bearer authorization request received from the GGSN to the media component grouping indication information received from the P-CSCF.

  In case the UE violates the IMS level indication, and attempts to set up multiple IMS media components in a single PDP context despite of an indication that mandated separate PDP contexts, the PCF shall enforce the rejection of this PDP context request by sending the an INSTALL and REMOVE decision to the GGSN.

  If the binding information and the list of flow_IDs are successfully authorised (verified) as per the means described above, the PCF shall also communicate the authorisation details for each media component to the GGSN.

  The authorisation details contain the "Authorised QoS" and the packet classifier(s) of the associated IP flows. In case of an aggregation of multiple media components within one PDP context, the combination of the "Authorised QoS" information of the individual media components is provided as the "Authorised QoS".

  Based on the media direction information and the direction of the source provided by the P-CSCF, the PCF shall define the direction (upstream or downstream) of the "Authorised QoS" and the packet classifier(s).

- Packet classifier(s):

  The PCF shall use the destination IP address(s), destination port number(s) and transport protocol id(s) to formulate a packet classifier(s).

  - The source IP address and source port number, which are part of the standard 5-tuple for packet classifying, are not provided by the P-CSCF. Therefore, the source IP address and source port number are wildcarded by the PCF in the packet classifier.

  Editor's note: The wildcarding of the source IP address maybe updated depending on the SA2's decision.

  - The PCF shall send the destination address and the destination port number for each IP flow associated with the media component.

- "Authorized QoS":

  The "Authorised QoS" information (consisting of maximum DiffServ Class and Data Rate) for a media component is extracted from the media type information and bandwidth parameter of the SDP. The PCF shall map the media type information into a DiffServ Class which is the highest class that can be used for the media. As an example, the audio media type shall be mapped into Expedited Forwarding PHB.

The PCF shall extract the Data Rate value from the "b=AS" SDP parameter. The "b=AS" parameter in the SDP shall contain all the overhead coming from the IP-layer and the layers above, e.g. UDP, RTP. The Data Rate shall also include the overhead coming from the possible usage of RTCP.

NOTE: The overhead coming from the IP-layer and the layers above is also included in the UMTS QoS bitrate parameters and the IP QoS parameters (e.g. RSVP flowSpec).

When the GGSN uses IP QoS parameters for resource reservation, the Data rate value shall be considered as the maximum value of the 'Token Bucket Rate' IP QoS parameter. When the GGSN uses UMTS QoS parameters, the Data rate value shall be considered as the maximum value of the 'Guaranteed bitrate' parameter for real-time bearers.

Editor's note: Mapping the Data rate value for the real time into 'Guaranteed bitrate' or 'Maximum bitrate' parameter is for FFS.

For non-real-time bearers the Data rate value shall be considered as the maximum value of the 'Maximum bitrate' parameter.

In case of an aggregation of multiple media components within one PDP context, the PCF shall provide the "Authorised QoS" for the bearer as the combination of the "Authorised QoS" information of the individual media components. The DiffServ Class in the "Authorised QoS" for the bearer shall contain the highest PHB amongst the ones applied for the individual media components and indicates the highest UMTS traffic class that can be applied to the PDP context.

Editor's note: It shall be possible the group identifiers to restrict the individual media components carried by the same PDP context to have the same PHBs.

The Data Rate of the "Authorised QoS" for the bearer shall be the sum of the Data Rate values of the individual media components and it is used as the maximum Data Rate value for the PDP context.

The PCF may include the gate enabling command as part of the authorisation decision. Alternatively, the PCF may provide a separate decision for opening the gate.

The PCF shall send the IMS charging identifier provided by the P-CSCF as part of the authorisation decision to the GGSN.

Upon receiving the modified SDP information from the P-CSCF, the PCF shall update the media authorization information for the session. The PCF may push this updated authorisation information to the GGSN. Under certain condition e.g. revoke of authorization, the PCF shall push the updated policy decision to the GGSN.

5.2.1.2 SBLP revoke decision

The PCF shall send a revoke authorisation decision to the GGSN upon SIP session release. The revoke authorisation decision shall be sent as a separate decision to the GGSN corresponding to the previous SBLP authorisation decision.

Additionally, when a media component which is bound to a PDP context is removed from a SIP session and the UE has not performed the corresponding modification of the PDP context within an operator specific time the PCF shall revoke the authorisation for the set of media components on that PDP context.

The timer shall be terminated if the PCF receives a new authorisation request with the same handle where that media component has been removed, or by termination of the PDP context.

6 Go protocol

6.1 Protocol support

6.1.1 TCP connection for COPS protocol

The GGSN receives the PCF identifier received as part of the Authorization Token, during the PDP context activation procedure. The GGSN resolves the PCF IP address from the PCF identifier, which is in the form of a fully qualified domain name.

If there is no existing TCP connection to the PCF, the GGSN shall establish a TCP connection for COPS interactions to the PCF. The GGSN shall use an existing TCP connection to the PCF, whenever present.

The TCP connection between the GGSN and the PCF may be pre-established by configuring the PCF addresses on the GGSN.

All communication between the GGSN and the PCFs shall use a standardised Client-Type with a corresponding standardised PIB, as defined in annex B.

The validity of the PCF may be ensured either by using a private DNS for resolving the PCF IP address or by configuring a list of allowed PCF IP addresses on the GGSN.

6.1.2 COPS protocol

The Go interface allows service-based local policy and QoS inter-working information to be "pushed" to or requested by the GGSN from a PCF.

The COPS protocol supports a client/server interface between the GGSN and the PCF. The Go interface shall conform to the IETF COPS framework as a requirement and guideline for Stage 3 work.

The COPS protocol allows both push and pull operations. For the purpose of the initial authorisation of QoS resources the pull operation shall be used. Subsequently the interactions between the PCF and the GGSN may use either pull or push operations.

Policy decisions may be stored by the COPS client in a local policy decision point allowing the GGSN to make admission control decisions without requiring additional interaction with the PCF.

The COPS client (PEP) can request a policy decision from the PCF triggered by a QoS signalling request. One PEP request may be followed by one or more asynchronous PCF decisions. Each of the decisions will allow the PCF to notify the PEP in the GGSN whenever necessary to change earlier decisions, generate errors etc.

Protocol stack: IP, TCP and COPS.

6.2 Basic COPS events/messages

The Go interface supports information passed between the GGSN and PCF. In order to allow effective communication between PCF and GGSN, all events associated with control functions are required:

- Coordination of events between the application layer and resource management in the IP bearer layer,

The specific events to the UMTS or IP bearer service are required in order to trigger the request from GGSN to PCF.

6.2.1 Type of messages

The COPS protocol supports several messages between GGSN and PCF. The message content is dependent on the type of COPS operation (e.g. Client-Open/Client-Accept/Client-Close, Request, Decision and Delete Request State).

The Client Open, Client Accept, Client Close, Keep Alive, Synchronize State Request and Synchronize State Complete messages are used for setting up and maintaining the connection between the PCF and the GGSN.

The following messages supported by the COPS layer for Go interface are used for the policy control operations:

- **Request (REQ)** message from the GGSN to the PCF is used by the GGSN to request SBLP and QoS inter-working information.
- **Decision (DEC)** message from the PCF to the GGSN is a response to the Request message or an asynchronous notification from PCF to the GGSN whenever necessary in order to change earlier decisions, generate errors, etc.
- **Report State (RPT)** message from the GGSN to the PCF is used to communicate the success, failure or changes to the client state of the GGSN in carrying out the PCF's decision indicated in the Decision message.
- **Delete Request State (DRQ)** message from the GGSN to the PCF indicates that the state identified by the client handle is no longer available/relevant and the corresponding state may be removed from the PCF.

6.3 Go events/messages

The UMTS-specific information is carried in specific COPS-PR objects, as defined in the 3GPP Go PIB that is given in annex B.

6.3.1 Event descriptions

The Go Interface uses COPS-PR [8] schematics and the UMTS Go PIB. For COPS-PR to support the Outsourcing Model it is required to add a new UMTS Go PIB with objects to:

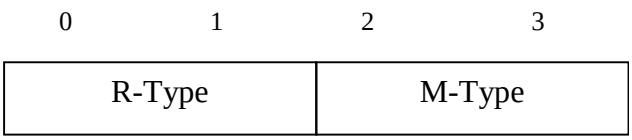
- Describe the Triggering Event Handling.
- Describe the Outsourcing Event.
- Describe the Decision for the Outsourced Event.
- Describe the Termination of the Outsourced Event.
- Describe the resource used for the Outsourced Event.

6.3.1.1 Common Header, Client Type

Client-type is UMTS Go (Client type number to be assigned through IANA).

6.3.1.2 Context Object

C-Num = 2, C-Type = 1



R-Type (Request Type Flag)

0x08 for configuration request

M-Type (Message Type)

0x01 initial capability negotiation

0x02 create event state

0x03 update event state

0x04 terminate event state

6.3.1.3 Client Specific Information (ClientSI) for outsourcing Operation

The binding information consisting of the Authorization Token and flow identifier(s) received by the GGSN are encapsulated inside the Client Specific Information object of the COPS request message sent from the GGSN to the PCF. The PCF identifier is extracted from the token and used inside the GGSN to resolve the address of the actual PCF. However, from the Go messages perspective the token can shall be considered as an opaque entity.

6.3.1.4 Reporting of Device Capabilities and Device Limitations

The functionality of reporting of device capabilities and device limitations is as described in RFC 3084 [8]. In addition, the following shall apply.

The configuration request message serves as a request from the GGSN to the PCF and include provisioning client information to provide the PCF with client-specific configuration or capability information about the GGSN. The capability information to be exchange shall include the PIB objects supported by the GGSN. This information from the client assists the server in deciding what types of policy the GGSN can install and enforce.

The following GGSN capabilities may be provided in the configuration request message:

- Bearer authorisation capabilities:

  The GGSN notifies the PCF that it supports bearer authorisation capabilities. The GGSN will provide the token(s) and media identifier(s) in the REQ for verifying the binding information and the grouping of the media components by the PCF.

- "Authorised QoS" capabilities:

  The GGSN notifies the PCF that it's capable to enforce the combined "Authorised QoS" for the bearer.

- Packet classifier capabilities:

  The GGSN notifies the PCF that it's capable to enforce the packet classifier for each media component direction. Similar to the classification capabilities of DiffServ PIB.

- Open /close the gate capabilities:

  The GGSN informs the PCF that it's capable to enforce a separate decision on opening the gate for the authorised media component and it' s capable to enforce a separated decision from the PCF regarding disabling of the gate.

- Revoke media authorisation capabilities:

  The GGSN notifies the PCF that it's capable to enforce the revoke authorisation for GPRS and IP resources decision from the PCF.

- Charging coordination:

  The GGSN informs the PCF that it's capable to send GCID(s) and GGSN address to the PCF.

  The GGSN informs the PCF that it's capable to receive ICID(s) from the PCF.

- Indication of QoS modifications to 0 kbit/s and from 0 kbit/s:

  The GGSN informs the PCF that it is able to notify when the maximum bit rate for the PDP context is modified to 0 kbit/s or that the maximum bit rate for the PDP context is changed from 0 kbit/s.

- Indication of the maximum number of media authorisation sessions:

  The GGSN may notifies the PCF how many parallel media authorisation sessions can support.

The PCF responds to the configuration request with an initial DEC message.

The R-type = 0x08 for configuration request is used here and M-type = 0x01 initial capability negotiation is used here.

The device capabilities information exchanged by the initial messages shall be stored in the PCF.

6.3.1.5 Initial Go Policy Provisioning

The functionality of initial Go policy provisioning is as described in RFC 3084 [8]. In addition, the following shall apply:

- The DEC message is sent from the PCF to the GGSN in response to the REQ message received from the GGSN. The Client Handle shall be the same as that received in the corresponding REQ message.
- The DEC message is sent as an immediate response to a configuration request with the solicited message flag set in the COPS message header. The PCF informs the GGSN of the capabilities that it supports. The capabilities exchanged shall include the PIB objects supported by the PCF. The PCF shall also inform the GGSN what types of events shall trigger policy control requests over the Go interface.
- The R-type = 0x08 for configuration request is used here and M-type = 0x01 initial capability negotiation is used here.

6.3.2 Message description

The Go interface uses the COPS-PR protocol. The following messages shall be supported:

The following events are available on the Go interface:

- Authorisation_Request (GGSN→PCF):

  This event allows the GGSN to request authorisation details from the PCF. It contains the following information:

  - Client Handle;
  - Binding Information.

  The R-type = 0x08 for configuration request is used here and M-type = 0x02 create event state is used here.

- Authorisation_Decision (PCF→GGSN):

  This event provides the GGSN with the authorisation status, and relevant authorisation decision data if applicable. The event contains the following information:

  - Client Handle;
  - ICID(s);
  - Unidirectional set (this parameter shall appear once for each direction (uplink and downlink)):
    - Direction indicator;
    - "Authorised QoS";
    - Packet classifiers /gate status (this parameter shall appear once for each required filter)
      A gate status (opened/closed) is included with each packet classifier element.

  Editor's note: The ICID issue should still be discussed in SA5.

  The R-type = 0x08 for configuration request is used here and M-type = 0x02 create event state is used here.

  - Filter Specification - The information about the authorised IP end points addresses and ports is detailed below. The Filter Specification contains packet classifiers made of packet filters that have the following data structure. The packet classifier parameters are:
    - Source IP address;
    - Destination IP address;
    - Source ports;
    - Destination ports;
    - Protocol ID.

The Source and Destination ports are described with a range consisting of a minimum and maximum value. If only one port is authorised, the minimum value and maximum value of the range are identical.

- Authorisation_Failure (PCF→GGSN):

  This event provides the GGSN with an indication of an authorisation failure, and may carry additional reason details. The event contains the following information:

  - Client Handle;
  - Authorisation failure (including any provided reason information).

Editor's note: The R-type and M-type shall be specified.

- Approval decision / Removal decision (PCF→GGSN):

  The approval decision indicates to the GGSN that the gate(s) for a media component(s) shall be opened. The removal decision indicates to the GGSN that the gate(s) for a media component(s) shall be closed. The events contain the following information:

  - Client Handle;
  - Unidirectional set (this parameter shall appear once for each direction for which gates are being updated (uplink and/or downlink)):
    - Direction indicator;
    - Packet classifiers /gate status (this parameter shall appear once for each gate to be modified for this direction)
      A gate status (opened/closed) is included with each packet classifier element.

NOTE 1: The opening of the gate may occur at the same time / be part of the authorisation decision event.

NOTE 2: The closing of the gate may occur at the same time as the revoke authorisation decision event.

The R-type = 0x08 for configuration request is used here and M-type = 0x03 update event state is used here.

- Report (RPT)s (GGSN→PCF):
  - Authorisation_report; Approval_report; Removal_report:

    The GGSN sends a COPS RPT message back to the PCF reporting that it enforced or not the authorisation decision, or the approval of QoS commit decision or removal of QoS commit decision.

    The events contain the following information:

    - Client Handle;
    - Success / Failure.
  - The report of the initial authorisation decision includes:
    - GCID;
    - GGSN address.
  - Report of state changes:

    The GGSN sends the report of state change message to the PCF reporting that the maximum bit rate for the PDP context is modified to 0 kbit/s or that the maximum bit rate for the PDP context is changed from 0 kbit/s.

    - Client Handle;
    - Maximum bit rate (set to 0kbps / changed from 0 kbps).

- Delete request state (GGSN→PCF):

  The GGSN informs the PCF via the delete request state message, that the PDP context is deactivated and the request state identified by the client handle is no longer available/relevant at the GGSN, so the corresponding state shall also be removed at the PCF.

  The DRQ message includes the reason why the request state was deleted.

  The events contain the following information:

  - Client Handle;

  - Reason code: "Tear", Sub-code: deactivation of the PDP context.

- Remove_Decision (PCF→GGSN):

  The PCF uses the Remove_Decision to inform the GGSN that the SIP session is terminated and the PCF revokes the authorized resources.

  The events contain the following information:

  - Client Handle.

6.4 Go data

The detailed data description is provided in annex B.

Annex A (informative): Information to be incorporated into other specifications

A.1 Capabilities of UE (TS 27.060)

Editor's Note: This clause describes the functional descriptions of capabilities of UE to be incorporated into e.g. TS 27.060.

A.1.1 Binding mechanism

Editor's Note: This clause describes the functionality of "Binding Mechanism" in UE.

The UE shall support the binding mechanism for service-based local policy control. The UE shall include one or more sets of binding information in Activate or Modify PDP Context Request if the PDP Context is for an IMS session and the UE received an authorization token during SIP session negotiation. Each binding information consists of an authorization token and one or more flow identifier(s). The flow identifier identifies a media component for the session and is derived from the media component ordering in SDP, i.e., the nth media component in SDP will have the flow identifier value n. If the UE decides to put multiple media components on the same PDP context e.g. due to the same QoS requirement for those media components, the UE shall include multiple flow identifiers, i.e. one flow identifier for each media component.

Editor's note: The above paragraph must be aligned with the rule for calculating flow ids given in clause 3.1.

Editor's Note: The container for the binding information in Activate or Modify PDP Context Request is defined in TS 24.008. The encoding of the binding information (i.e., Authorization and flow identifier) is defined in TS 29.207.

A.1.2 DiffServ edge function

Editor's Note: This clause describes the functionality of "DiffServ Edge Function" in UE.

A.1.3 RSVP/IntServ function

Editor's Note: This clause describes the functionality of "RSVP/IntServ Function" in UE.

A.1.4 Pre-conditions for SIP QoS assured sessions

Editor's Note: This clause describes the functionality of "Pre-conditions for SIP QoS Assured Sessions" in UE.

Annex B (normative): 3GPP Go PIB

```
GO3GPP-PIB  PIB-DEFINITIONS ::= BEGIN

IMPORTS  -- Imports need cleanup
    Unsigned32, Integer32, MODULE-IDENTITY,
    MODULE-COMPLIANCE, OBJECT-TYPE, OBJECT-GROUP
            FROM COPS-PR-SPPI
    InstanceId, Prid
            FROM COPS-PR-SPPI-TC

    InetAddress, InetAddressType
            FROM INET-ADDRESS-MIB
    TruthValue, PhysAddress
            FROM SNMPv2-TC
    SnmpAdminString
            FROM SNMP-FRAMEWORK-MIB;

go3gppPib  MODULE-IDENTITY
    SUBJECT-CATEGORIES   { go3gpp } -- Go 3GPP COPS Client Type
                                    -- to be assigned by IANA
    LAST-UPDATED "200205152200Z"
    ORGANIZATION "3GPP TSG CN WG3"
    CONTACT-INFO "Kwok Ho Chan
                  Nortel Networks
                  600 Technology Park Drive
                  Billerica, MA 01821 USA
                  Phone: +1 978 288 8175
                  Email: khchan@nortelnetworks.com

                  Louis-Nicolas Hamer
                  Nortel Networks
                  100 Constellation Crescent
                  Ottawa, Ontario
                  Canada, K2G 6J8
                  Phone: +1 613 768 3409
                  Email: nhamer@nortelnetworks.com"
    DESCRIPTION
            "A PIB module containing the set of provisioning
            classes that are required for support of policies for
            3GPP's GO interface, Release 5."
    REVISION "Use Version Number of 3GPP TS 29.207"

    ::= { pib xxx } -- xxx to be assigned by IANA


--
-- The root OID for PRCs in the 3GPP GO PIB
--

go3gppCapabilityClasses   OBJECT IDENTIFIER ::= { go3gppPib 1 }
go3gppEventHandlerClasses OBJECT IDENTIFIER ::= { go3gppPib 2 }
go3gppEventClasses        OBJECT IDENTIFIER ::= { go3gppPib 3 }
go3gppReportClasses       OBJECT IDENTIFIER ::= { go3gppPib 4 }
go3gppConformance         OBJECT IDENTIFIER ::= { go3gppPib 5 }



-- ------------------------------------------------------------
--
-- Capability and Limitation Policy Rule Classes
--

--
-- 3GPP GO Capability Table
--



go3gppAuthReqCapTable OBJECT-TYPE
    SYNTAX         SEQUENCE OF Go3gppAuthReqCapEntry
    PIB-ACCESS     notify
```

```
        STATUS         current
        DESCRIPTION
            "The 3GPP Go Authorization Request Capability PRC."
        ::= { go3gppCapabilityClasses 1 }


    go3gppAuthReqCapEntry OBJECT-TYPE
        SYNTAX         Go3gppAuthReqCapEntry
        STATUS         current
        DESCRIPTION
            "An instance of the go3gppAuthReqCap class identifies a
            specific PRC and associated attributes as supported
            by the device."

        PIB-INDEX { go3gppAuthReqCapPrid }
        UNIQUENESS { }
        ::= { go3gppAuthReqCapTable 1 }


    Go3gppAuthReqCapEntry ::= SEQUENCE {
            go3gppAuthReqCapPrid          InstanceId,
            go3gppAuthReqCapBindingInfos  Unsigned32,
            go3gppAuthReqCapFlowIds       Unsigned32
    }


    go3gppAuthReqCapPrid OBJECT-TYPE
        SYNTAX         InstanceId
        STATUS         current
        DESCRIPTION
            "An arbitrary integer index that uniquely identifies an
            instance of the go3gppAuthReqCap class."
        ::= { go3gppAuthReqCapEntry 1 }


    go3gppAuthReqCapBindingInfos OBJECT-TYPE
        SYNTAX         Unsigned32
        STATUS         current
        DESCRIPTION
            "Indication of the maximum number of Binding Information
            the PEP can send with each Authorizating Request.
            The value of zero indicates limit is not specified."
        DEFVAL  { 0 }
        ::= { go3gppAuthReqCapEntry 2 }


    go3gppAuthReqCapFlowIds OBJECT-TYPE
        SYNTAX         Unsigned32
        STATUS         current
        DESCRIPTION
            "Indication of the maximum number of Flow IDs the PEP can
            send with each Authorization Request.
            The value of zero indicates limit is not specified."
        DEFVAL  { 0 }
        ::= { go3gppAuthReqCapEntry 3 }


    --
    -- Component Limitations Table
    --
    -- This table supports the ability to export information
    -- detailing provisioning class/attribute implementation limitations
    -- to the policy control function.
    --

-- The component Limitations Table section needs to be updated.
    -- ---------------------------------------------------------------
    --
    -- 3GPP GO Event Handler Provisioning Classes
    --
    -- PRCs sent from PCF to PEP for indicating how to handle each
    -- kind of event that require actions by the GO interface.
    --
    -- For 3GPP Release 5, PRCs for Event Handling of Authorization
    -- Request containing Binding Information, Flow IDs, and QoS is
    -- specified.
    --
```

```
--
-- 3GPP GO Authorization Request Event Handler Provisioning Table
--

go3gppAuthReqHandlerTable OBJECT-TYPE
    SYNTAX         SEQUENCE OF Go3gppAuthReqHandlerEntry
    PIB-ACCESS     install
    STATUS         current
    DESCRIPTION
        "PRC from PCF to PEP carried by COPS DEC messages
        indicating GO actions to take at the GGSN when an Authorization
        Request Event is detected by the GGSN.  An example of an
        Authorization Request Event is the receive of a PDP Context message."
    ::= { go3gppEventHandlerClasses 1 }


go3gppAuthReqHandlerEntry OBJECT-TYPE
    SYNTAX         Go3gppAuthReqHandlerEntry
    STATUS         current
    DESCRIPTION
        "An instance of the go3gppAuthReqHandler class sent by the PCF to
        the PEP what the PEP should send upon detection of an Authorization
        Request Event."
    PIB-INDEX { go3gppAuthReqHandlerPrid }
    UNIQUENESS { go3gppAuthReqHandlerEnable,
                 go3gppAuthReqHandlerBindingInfo
               }
    ::= { go3gppAuthReqHandlerTable 1 }


Go3gppAuthReqHandlerEntry ::= SEQUENCE {
        go3gppAuthReqHandlerPrid         InstanceId,
        go3gppAuthReqHandlerEnable       INTEGER,
        go3gppAuthReqHandlerBindingInfo  Unsigned32
}


go3gppAuthReqHandlerPrid OBJECT-TYPE
    SYNTAX         InstanceId
    STATUS         current
    DESCRIPTION
        "An arbitrary integer index that uniquely identifies an
        instance of this class."
    ::= { go3gppAuthReqHandlerEntry 1 }


go3gppAuthReqHandlerEnable OBJECT-TYPE
    SYNTAX         INTEGER {
                       enable(1),
                       disable(2)
                   }
    STATUS         current
    DESCRIPTION
        "Controls the usage of 3GPP Authorization Request Events
        to trigger COPS requests to PCF on the go interface."
    DEFVAL { enable }
    ::= { go3gppAuthReqHandlerEntry 2 }


go3gppAuthReqHandlerBindingInfo OBJECT-TYPE
    SYNTAX         Unsigned32
    STATUS         current
    DESCRIPTION
        "Indication of the maximum number of Binding Information
        be associated with a each Authorizating Request.
        The value of zero indicates policy control does not impose
        any limit."
    DEFVAL  { 0 }
    ::= { go3gppAuthReqHandlerEntry 3 }



-- ------------------------------------------------------------
--
-- 3GPP GO Event Classes
--
-- PRCs from PEP to PCF carried by COPS REQ messages
-- indicating the detection of specific events in the GGSN.
```

```
-- Information required for PCF to make decision on behave
-- of GGSN is also defined here to be carried by REQ messages.
--

--
-- 3GPP GO Authorization Request Event Table
--
go3gppAuthReqEventTable OBJECT-TYPE
    SYNTAX         SEQUENCE OF Go3gppAuthReqEventEntry
    PIB-ACCESS     notify
    STATUS         current
    DESCRIPTION
        "PRC for indication of Authorization Request Event
        and its relevant information.
        Sent by PEP to PCF upon receive of an Authorization
        Request.  Using COPS REQ message."
    ::= { go3gppEventClasses 1 }


go3gppAuthReqEventEntry OBJECT-TYPE
    SYNTAX         Go3gppAuthReqEventEntry
    STATUS         current
    DESCRIPTION
        "An entry in the Authorization Request Event Table
        describe a single Event sent by the PEP to the PCF."
    PIB-INDEX { go3gppAuthReqEventPrid }
    UNIQUENESS { }
    ::= { go3gppAuthReqEventTable 1 }


Go3gppAuthReqEventEntry ::= SEQUENCE {
        go3gppAuthReqEventPrid         InstanceId,
        go3gppAuthReqEventBindingInfos Prid
}


go3gppAuthReqEventPrid OBJECT-TYPE
    SYNTAX         InstanceId
    STATUS         current
    DESCRIPTION
        "An arbitrary integer index that uniquely identifies an
        instance of the go3gppAuthReqEvent class."
    ::= { go3gppAuthReqEventEntry 1 }


go3gppAuthReqEventBindingInfos OBJECT-TYPE
    SYNTAX         Prid
    STATUS         current
    DESCRIPTION
        "References the first of a list of go3gppBindingInfo
        class instances that are associated with this
        Authorization Request Event.
        A value of zeroDotZero indicates there are no
        go3gppBindingInfo class instance associated with
        this Authorization Event."
    ::= { go3gppAuthReqEventEntry 2 }


--
-- 3GPP GO Binding Information Table
--
go3gppBindingInfoTable OBJECT-TYPE
    SYNTAX         SEQUENCE OF Go3gppBindingInfoEntry
    PIB-ACCESS     notify
    STATUS         current
    DESCRIPTION
        "PRC representing Binding Information.
        Sent by PEP to PCF as part of an Authorization
        Request.  In a COPS REQ message."
    ::= { go3gppEventClasses 2 }


go3gppBindingInfoEntry OBJECT-TYPE
    SYNTAX         Go3gppBindingInfoEntry
    STATUS         current
    DESCRIPTION
        "An entry in the Binding Information Table
        describing a single Binding Info.
```

```
        Each entry is referenced by go3gppAuthReqEventBindingInfos
        or go3gppBindingInfoNext."
    PIB-INDEX { go3gppBindingInfoPrid }
    UNIQUENESS { }
    ::= { go3gppBindingInfoTable 1 }


Go3gppBindingInfoEntry ::= SEQUENCE {
        go3gppBindingInfoPrid          InstanceId,
        go3gppBindingInfoToken         OCTET STRING,
        go3gppBindingInfoFlowIds       Prid,
        go3gppBindingInfoNext          Prid
}


go3gppBindingInfoPrid OBJECT-TYPE
    SYNTAX         InstanceId
    STATUS         current
    DESCRIPTION
        "An arbitrary integer index that uniquely identifies an
        instance of the go3gppBindingInfo class."
    ::= { go3gppBindingInfoEntry 1 }


go3gppBindingInfoToken OBJECT-TYPE
    SYNTAX         OCTET STRING
    STATUS         current
    DESCRIPTION
        "The Authorization Token associated with this
        instance of the go3gppBindingInfo class.
        Each Binding Information must have a Token."
    ::= { go3gppBindingInfoEntry 2 }


go3gppBindingInfoFlowIds OBJECT-TYPE
    SYNTAX         Prid
    STATUS         current
    DESCRIPTION
        "References the first of a list of FlowIds associated
        with this instance of go3gppBindingInfo class.
        This is the anchor of a list of go3gppFlowIdEntry
        Instances.
        A value of zeroDotZero indicates an empty list which
        is an error condition."
    DEFVAL  { zeroDotZero }
    ::= { go3gppBindingInfoEntry 3 }


go3gppBindingInfoNext OBJECT-TYPE
    SYNTAX         Prid
    STATUS         current
    DESCRIPTION
        "References the next of a list of go3gppBindingInfo
        instances associated with an Authorization Request.
        A value of zeroDotZero indicates this is the last of
        a list of go3gppBindingInfo instances associated with
        an Authorization Request."
    DEFVAL  { zeroDotZero }
    ::= { go3gppBindingInfoEntry 4 }


--
-- 3GPP Go Authorization Request FlowID Table
--
go3gppFlowIdTable OBJECT-TYPE
    SYNTAX         SEQUENCE OF Go3gppFlowIdEntry
    PIB-ACCESS     notify
    STATUS         current
    DESCRIPTION
        ""
    ::= { go3gppEventClasses 3 }


go3gppFlowIdEntry OBJECT-TYPE
    SYNTAX         Go3gppFlowIdEntry
    STATUS         current
    DESCRIPTION
        "Each entry describes a single FlowID."
```

```
        PIB-INDEX { go3gppFlowIdPrid }
        UNIQUENESS { }
        ::= { go3gppFlowIdTable 1 }


    Go3gppFlowIdEntry ::= SEQUENCE {
            go3gppFlowIdPrid         InstanceId,
            go3gppFlowIdFlowId       Unsigned32,
            go3gppFlowIdNext         Prid
    }


    go3gppFlowIdPrid OBJECT-TYPE
        SYNTAX         InstanceId
        STATUS         current
        DESCRIPTION
            "An arbitrary integer index that uniquely identifies an
            instance of the go3gppFlowId class."
        ::= { go3gppFlowIdEntry 1 }


    go3gppFlowIdFlowId OBJECT-TYPE
        SYNTAX         Unsigned32
        STATUS         current
        DESCRIPTION
            "The FlowId itself."
        ::= { go3gppFlowIdEntry 2 }


    go3gppFlowIdNext OBJECT-TYPE
        SYNTAX         Prid
        STATUS         current
        DESCRIPTION
            "References the next FlowId in the list associated with the
            same Binding Information of an Authorization Request.
            This points to a list of go3gppFlowIdEntry Instances.
            A value of zeroDotZero indicates end of the list."
        DEFVAL  { zeroDotZero }
        ::= { go3gppFlowIdEntry 3 }



    -- ------------------------------------------------------------
    --
    -- 3GPP Go Authorization Request Decisions
    --
    -- PRCs for carrying the Event Decision send from PCF to PEP,
    -- carried by the COPS DEC message.
    -- These PRCs include support for Gates/Filters, QoS, ICIDs.
    --

--
    -- We can define Failure Decisions by use of COPS-PR DEC message
    -- containing first an install decision (with objects indicating
    -- what failed and some indication to the GGSN how to react to this
    -- Error Decision), and second a remove decision (for cleanup of
    -- the installed Error Decision Object).
    --

-- Failures indicated by PCF to GGSN
--   Authorization Failure
--



    --
    -- Authorization Request Decision Table
    --
    go3gppAuthReqDecTable OBJECT-TYPE
        SYNTAX         SEQUENCE OF Go3gppAuthReqDecEntry
        PIB-ACCESS     install
        STATUS         current
        DESCRIPTION
            ""
        ::= { go3gppEventClasses 4 }


    go3gppAuthReqDecEntry OBJECT-TYPE
```

```
        SYNTAX         Go3gppAuthReqDecEntry
        STATUS         current
        DESCRIPTION
            "Each go3gppAuthReqDecEntry is per Authorization Request."
        PIB-INDEX { go3gppAuthReqDecPrid }
        UNIQUENESS { }
        ::= { go3gppAuthReqDecTable 1 }


    Go3gppAuthReqDecEntry ::= SEQUENCE {
            go3gppAuthReqDecPrid       InstanceId,
            go3gppAuthReqDecIcids      Prid,
            go3gppAuthReqDecDirDecs    Prid
    }


    go3gppAuthReqDecPrid OBJECT-TYPE
        SYNTAX         InstanceId
        STATUS         current
        DESCRIPTION
            "An arbitrary integer index that uniquely identifies an
            instance of the go3gppAuthReqDec class."
        ::= { go3gppAuthReqDecEntry 1 }


    go3gppAuthReqDecIcids OBJECT-TYPE
        SYNTAX         Prid
        STATUS         current
        DESCRIPTION
            "References the first of a list of IcIDs associated
            with this instance of go3gppAuthReqDec class.
            There should be one IcID on this list for each Binding
            Information in the corresponding Authorization Request.
            A value of zeroDotZero indicates an empty list and there
            is no IcID change associated with this Authorization Request
            Decision."
        DEFVAL  { zeroDotZero }
        ::= { go3gppAuthReqDecEntry 2 }


    go3gppAuthReqDecDirDecs OBJECT-TYPE
        SYNTAX         Prid
        STATUS         current
        DESCRIPTION
            "References the first of a list of Directional Decisions
            associated with this instance of go3gppAuthReqDec class.
            A value of zeroDotZero indicates an empty list and there
            is no Directional Decisions associated with this
            Authorization Request Decision.
            There should be at least one and at most two Directional
            Decisions per Authorization Request Decision."
        ::= { go3gppAuthReqDecEntry 3 }


--
-- 3GPP Go ICID Table
--
    go3gppIcidTable OBJECT-TYPE
        SYNTAX         SEQUENCE OF Go3gppIcidEntry
        PIB-ACCESS     install
        STATUS         current
        DESCRIPTION
            ""
        ::= { go3gppEventClasses 5 }


    go3gppIcidEntry OBJECT-TYPE
        SYNTAX         Go3gppIcidEntry
        STATUS         current
        DESCRIPTION
            ""
        PIB-INDEX { go3gppIcidPrid }
        UNIQUENESS { go3gppIcidValue }
        ::= { go3gppIcidTable 1 }


    Go3gppIcidEntry ::= SEQUENCE {
            go3gppIcidPrid             InstanceId,
```

```
            go3gppIcidValue            OCTET STRING,
            go3gppIcidNext             Prid
    }


    go3gppIcidPrid OBJECT-TYPE
        SYNTAX         InstanceId
        STATUS         current
        DESCRIPTION
            "An arbitrary integer index that uniquely identifies an
            instance of the go3gppIcid class."
        ::= { go3gppIcidEntry 1 }


    go3gppIcidValue OBJECT-TYPE
        SYNTAX         OCTET STRING
        STATUS         current
        DESCRIPTION
            "The ICID itself, Using this as a place holder for now."
        ::= { go3gppIcidEntry 2 }


    go3gppIcidNext OBJECT-TYPE
        SYNTAX         Prid
        STATUS         current
        DESCRIPTION
            "References the next go3gppIcidEntry of a list of IcIDs
            associated with this instance of go3gppAuthReqDec class.
            There should be one IcID on this list for each Binding
            Information in the corresponding Authorization Request.
            A value of zeroDotZero indicates the end of the list of
            IcIDs associated with an Authorization Request Decision."
        DEFVAL  { zeroDotZero }
        ::= { go3gppIcidEntry 3 }


--
-- 3GPP Go Authorization Request Directional Decision Table
--
    go3gppAuthReqDirDecTable OBJECT-TYPE
        SYNTAX         SEQUENCE OF Go3gppAuthReqDirDecEntry
        PIB-ACCESS     install
        STATUS         current
        DESCRIPTION
            "This table represents the authorization request decision for a  unique direction (e.g.
uplink and downlink)."
        ::= { go3gppEventClasses 6 }


    go3gppAuthReqDirDecEntry OBJECT-TYPE
        SYNTAX         Go3gppAuthReqDirDecEntry
        STATUS         current
        DESCRIPTION
            "There should be one of these per direction per AuthReqDec."
        PIB-INDEX { go3gppAuthReqDirDecPrid }
        UNIQUENESS { }
        ::= { go3gppAuthReqDirDecTable 1 }


    Go3gppAuthReqDirDecEntry ::= SEQUENCE {
            go3gppAuthReqDirDecPrid      InstanceId,
            go3gppAuthReqDirDecDirection Prid,
            go3gppAuthReqDirDecQos       Prid,
            go3gppAuthReqDirDecGates     Prid,
            go3gppAuthReqDirDecNext      Prid
    }


    go3gppAuthReqDirDecPrid OBJECT-TYPE
        SYNTAX         InstanceId
        STATUS         current
        DESCRIPTION
            "An arbitrary integer index that uniquely identifies an
            instance of the go3gppAuthReqDirDec class."
        ::= { go3gppAuthReqDirDecEntry 1 }

-- Need to change Prid to something else.
    go3gppAuthReqDirDecDirection OBJECT-TYPE
```

```
        SYNTAX         Prid
        STATUS         current
        DESCRIPTION
            "References a Direction Type definition."
        ::= { go3gppAuthReqDirDecEntry 2 }


    go3gppAuthReqDirDecQos OBJECT-TYPE
        SYNTAX         Prid
        STATUS         current
        DESCRIPTION
            ""
        ::= { go3gppAuthReqDirDecEntry 3 }


    go3gppAuthReqDirDecGates OBJECT-TYPE
        SYNTAX         Prid
        STATUS         current
        DESCRIPTION
            "References the first instance of a list of go3gppGate class."
        ::= { go3gppAuthReqDirDecEntry 4 }


    go3gppAuthReqDirDecNext OBJECT-TYPE
        SYNTAX         Prid
        STATUS         current
        DESCRIPTION
            "References the next instance of a list of
            go3gppAuthReqDirDec class."
        ::= { go3gppAuthReqDirDecEntry 5 }

       go3gppAuthReqDirection OBJECT-TYPE
        SYNTAX         INTEGER {
                            uplink   (0),
                            downlink (1)
                       }
        STATUS         current
        DESCRIPTION
            "References the media authorisation direction."
        ::= { go3gppAuthReqDirDecEntry 6 }


--
-- 3GPP Go QoS Table
--
    go3gppQosTable OBJECT-TYPE
        SYNTAX         SEQUENCE OF Go3gppQosEntry
        PIB-ACCESS     install
        STATUS         current
        DESCRIPTION
            "This table represents the Authorised QoS"
        ::= { go3gppEventClasses 7 }


    go3gppQosEntry OBJECT-TYPE
        SYNTAX         Go3gppQosEntry
        STATUS         current
        DESCRIPTION
            "There should be one of these per direction per AuthReqDec."
        PIB-INDEX { go3gppQosPrid }
        UNIQUENESS { }
        ::= { go3gppQosTable 1 }



    Go3gppQosEntry ::= SEQUENCE {
            go3gppQosPrid               InstanceId,

            go3gppQosServiceClass       INTEGER,
            go3gppQosDataRateUnit       INTEGER,
            go3gppQosDataRate           Unsigned32
    }


    go3gppQosPrid OBJECT-TYPE
        SYNTAX         InstanceId
        STATUS         current
        DESCRIPTION
```

```
            "An arbitrary integer index that uniquely identifies an
            instance of the go3gppQos class."
        ::= { go3gppQosEntry 1 }



    go3gppQosServiceClass OBJECT-TYPE
        SYNTAX         INTEGER
        STATUS         current
        DESCRIPTION
            "A Service Class Indication using DSCP Encoding."
        ::= { go3gppQosEntry 2 }


    go3gppQosDataRateUnit OBJECT-TYPE
        SYNTAX         INTEGER {
                         bps     (0),
                         kbps    (1),
                         Mbps    (2)
                       }
        STATUS         current
        DESCRIPTION
            "Indication of the unit of measure for go3gppQosDataRate."
        ::= { go3gppQosEntry 3 }


    go3gppQosDataRate OBJECT-TYPE
        SYNTAX         Unsigned32
        STATUS         current
        DESCRIPTION
            "The Data Rate with unit of measure indicated by
            go3gppQosDataRateUnit."
        ::= { go3gppQosEntry 4 }



--
-- 3GPP Go Gate Decision Table
--
--
-- There could be one of these per direction per GateDec.
--
-- This is for changing Gating Status only when used alone
-- (not as part of Direction Decision).

-- go3gppGateDec is sent in a different COPS DEC message
-- from the DEC message carrying go3gppAuthReqDec.  PCF must
-- have sent a go3gppAuthReqDec before using go3gppGateDec.

    go3gppGateDecTable OBJECT-TYPE
        SYNTAX         SEQUENCE OF Go3gppGateDecEntry
        PIB-ACCESS     install
        STATUS         current
        DESCRIPTION
            "This table represents an updated gating decision."
        ::= { go3gppEventClasses 8 }


    go3gppGateDecEntry OBJECT-TYPE
        SYNTAX         Go3gppGateDecEntry
        STATUS         current
        DESCRIPTION
            "There should be one of these per direction per AuthReqDec."
        PIB-INDEX { go3gppGateDecPrid }
        UNIQUENESS { }
        ::= { go3gppGateDecTable 1 }



    Go3gppGateDecEntry ::= SEQUENCE {
            go3gppGateDecPrid            Prid,
            go3gppGateDecDirection       Prid,
            go3gppGateDecGates           Prid,
            go3gppGateDecNext            Prid
    }


    go3gppGateDecPrid OBJECT-TYPE
```

```
        SYNTAX         InstanceId
        STATUS         current
        DESCRIPTION
            "An arbitrary integer index that uniquely identifies an
            instance of the go3gppGateDec class."
        ::= { go3gppGateDecEntry 1 }


    go3gppGateDecDirection OBJECT-TYPE
        SYNTAX         INTEGER {
                            uplink   (0),
                            downlink (1)
                       }
        STATUS         current
        DESCRIPTION
            "References the gate direction."
        ::= { go3gppGateDecEntry 2 }

     go3gppGateDecGates OBJECT-TYPE
        SYNTAX         Prid
        STATUS         current
        DESCRIPTION
            "References the first instance of a list of go3gppGate class."
        ::= { go3gppGateDecEntry 3 }


    go3gppGateDecNext OBJECT-TYPE
        SYNTAX         Prid
        STATUS         current
        DESCRIPTION
            "References the next instance of a list of go3gppGateDec class."
        ::= { go3gppGateDecEntry 4 }









--
-- 3GPP Go Gate Table
--
    go3gppGateTable OBJECT-TYPE
        SYNTAX         SEQUENCE OF Go3gppGateEntry
        PIB-ACCESS     install
        STATUS         current
        DESCRIPTION
            "PRC representing a Gate."
        ::= { go3gppEventClasses 9 }


    go3gppGateEntry OBJECT-TYPE
        SYNTAX         Go3gppGateEntry
        STATUS         current
        DESCRIPTION
            "Each instance represent one Gate."
        PIB-INDEX { go3gppGatePrid }
        UNIQUENESS { }
        ::= { go3gppGateTable 1 }


    Go3gppGateEntry ::= SEQUENCE {
            go3gppGatePrid              InstanceId,
            go3gppGateFilter            Prid,
            go3gppGateStatus            INTEGER,
            go3gppGateNext              Prid
    }


    go3gppGatePrid OBJECT-TYPE
        SYNTAX         InstanceId
        STATUS         current
        DESCRIPTION
            "An arbitrary integer index that uniquely identifies an
            instance of the go3gppGate class."
```

```
    ::= { go3gppGateEntry 1 }


go3gppGateFilter OBJECT-TYPE
    SYNTAX         Prid
    STATUS         current
    DESCRIPTION
        "References an instance of the go3gppFilter class."
        A value of zeroDotZero indicates no go3gppFilter is
        used with this go3gppGate."
    ::= { go3gppGateEntry 2 }


go3gppGateStatus OBJECT-TYPE

    SYNTAX         INTEGER {
                        close  (0),
                        open (1)
                   }
    STATUS         current
    DESCRIPTION
        "Indicates if this gate will allow traffic to flow."
 DEFVAL  { close }
    ::= { go3gppGateEntry 3 }



go3gppGateNext OBJECT-TYPE
    SYNTAX         Prid
    STATUS         current
    DESCRIPTION
        "Reference the next Gate on a list of go3gppGate instances.
        A value of zeroDotZero indicates this is the last Gate
        on the list."
    ::= { go3gppGateEntry 4 }









--
-- The Classification classes group
--

go3GPPClassifierClasses
           OBJECT IDENTIFIER ::= { ? }
--
-- The Base Filter Table
--

go3GPPBaseFilterTable OBJECT-TYPE
    SYNTAX         SEQUENCE OF go3GPPBaseFilterEntry
    PIB-ACCESS     install
    STATUS         current
    DESCRIPTION

        "The Base Filter class.  A packet has to match all
        fields in an Filter.  Wildcards may be specified for those
        fields that are not relevant."

::= { go3GPPClassifierClasses 1 }


go3GPPBaseFilterEntry OBJECT-TYPE
    SYNTAX         go3GPPBaseFilterEntry
    STATUS         current
    DESCRIPTION
        "An instance of the go3GPPBaseFilter class."

    PIB-INDEX { go3GPPBaseFilterPrid }

    ::= { go3GPPBaseFilterTable 1 }
```

```
go3GPPBaseFilterEntry ::= SEQUENCE {
        go3GPPBaseFilterPrid         InstanceId
}


go3GPPBaseFilterPrid OBJECT-TYPE
    SYNTAX         InstanceId
    STATUS         current
    DESCRIPTION
        "An integer index to uniquely identify this Filter among all
        the Filters."

    ::= { go3GPPBaseFilterEntry 1 }



--
-- The Go 3GPP IP Filter Table
--

go3gppFilterTable OBJECT-TYPE
    SYNTAX         SEQUENCE OF Go3GPPIpFilterEntry


    PIB-ACCESS     install
    STATUS         current
    DESCRIPTION
        "Filter definitions.  A packet has to match all fields in a
        filter.  Wildcards may be specified for those fields that
        are not relevant."

    ::= { go3gppEventClasses 10 }

go3GPPIpFilterEntry OBJECT-TYPE
    SYNTAX         Go3GPPIpFilterEntry
    STATUS         current
    DESCRIPTION
        "An instance of the go3GPPIpFilter class."

    EXTENDS { go3GPPBaseFilterEntry }
    UNIQUENESS {
                 go3GPPIpFilterAddrType,
                 go3GPPIpFilterDstAddr,
                 go3GPPIpFilterDstPrefixLength,
                 go3GPPIpFilterSrcAddr,
                 go3GPPIpFilterSrcPrefixLength,
                 go3GPPIpFilterProtocol,
                 go3GPPIpFilterDstL4PortMin,
                 go3GPPIpFilterDstL4PortMax,
                 go3GPPIpFilterSrcL4PortMin,
                 go3GPPIpFilterSrcL4PortMax }

    ::= { go3gppFilterTable 1 }



Go3GPPIpFilterEntry ::= SEQUENCE {
        go3GPPIpFilterAddrType         InetAddressType,
        go3GPPIpFilterDstAddr          InetAddress,
        go3GPPIpFilterDstPrefixLength  InetAddressPrefixLength,
        go3GPPIpFilterSrcAddr          InetAddress,
        go3GPPIpFilterSrcPrefixLength  InetAddressPrefixLength,
        go3GPPIpFilterProtocol         Integer32,
        go3GPPIpFilterDstL4PortMin     InetPortNumber,
        go3GPPIpFilterDstL4PortMax     InetPortNumber,
        go3GPPIpFilterSrcL4PortMin     InetPortNumber,
        go3GPPIpFilterSrcL4PortMax     InetPortNumber
}



go3GPPIpFilterAddrType OBJECT-TYPE

    SYNTAX         InetAddressType
    STATUS         current
    DESCRIPTION
```

```
        "The address type enumeration value [INETADDR] to specify
         the type of the packet's IP address."

    ::= { go3GPPIpFilterEntry 1 }

go3GPPIpFilterDstAddr OBJECT-TYPE

    SYNTAX         InetAddress
    STATUS         current
    DESCRIPTION
        "The IP address [INETADDR] to match against the packet's
         destination IP address. go3GPPIpFilterDstPrefixLength
         indicates the number of bits that are relevant. "

    ::= { go3GPPIpFilterEntry 2 }


go3GPPIpFilterDstPrefixLength OBJECT-TYPE
    SYNTAX         InetAddressPrefixLength
    STATUS         current
    DESCRIPTION
        "The length of a mask for the matching of the destination
         IP address.  Masks are constructed by setting bits in
         sequence from the most-significant bit downwards for
         go3GPPIpFilterDstPrefixLength bits length. All other bits in
         the mask, up to the  number needed to fill the length of
         the address go3GPPIpFilterDstAddr are cleared to zero. A zero
         bit in the mask then means that the corresponding bit in
         the address always matches."

    ::= { go3GPPIpFilterEntry 3 }



go3GPPIpFilterSrcAddr OBJECT-TYPE
    SYNTAX         InetAddress
    STATUS         current
    DESCRIPTION
        "The IP address to match against the packet's source IP
        address. go3GPPIpFilterSrcPrefixLength indicates the
       number of bits that are relevant. "

    ::= { go3GPPIpFilterEntry 4 }

go3GPPIpFilterSrcPrefixLength OBJECT-TYPE
    SYNTAX         InetAddressPrefixLength
    UNITS          "bits"
    STATUS         current


    DESCRIPTION
        "The length of a mask for the matching of the source IP
         address. Masks are constructed by setting bits in sequence
         from the most-significant bit downwards for
         go3GPPIpFilterSrcPrefixLength bits length. All other bits in
         the mask, up to the  number needed to fill the length of
         the address go3GPPIpFilterSrcAddr are cleared to zero.  A
         zero bit in the mask then means that the corresponding bit
         in the address always matches."

    ::= { go3GPPIpFilterEntry 5 }

go3GPPIpFilterProtocol OBJECT-TYPE
    SYNTAX         Integer32 (-1 | 0..255)
    STATUS         current
    DESCRIPTION
        "The IP protocol to match against the packet's protocol.
        A value of -1 means match all."

    ::= { go3GPPIpFilterEntry 6 }

go3GPPIpFilterDstL4PortMin OBJECT-TYPE
    SYNTAX         InetPortNumber
    STATUS         current
    DESCRIPTION
        "The minimum value that the packet's layer 4 destination
        port number can have and match this filter. This value must
        be equal to or lesser that the value specified for this
```

```
            filter in go3GPPIpFilterDstL4PortMax."

        ::= { go3GPPIpFilterEntry 7 }


    go3GPPIpFilterDstL4PortMax OBJECT-TYPE
        SYNTAX          InetPortNumber


        STATUS          current
        DESCRIPTION
            "The maximum value that the packet's layer 4 destination
            port number can have and match this filter. This value must
            be equal to or greater that the value specified for this
            filter in go3GPPIpFilterDstL4PortMin."

        ::= { go3GPPIpFilterEntry 8 }


    go3GPPIpFilterSrcL4PortMin OBJECT-TYPE
        SYNTAX          InetPortNumber
        STATUS          current
        DESCRIPTION
            "The minimum value that the packet's layer 4 source port
            number can have and match this filter. This value must
            be equal to or lesser that the value specified for this
            filter in go3GPPIpFilterSrcL4PortMax."

        ::= { go3GPPIpFilterEntry 9 }


    go3GPPIpFilterSrcL4PortMax OBJECT-TYPE
        SYNTAX          InetPortNumber
        STATUS          current
        DESCRIPTION
            "The maximum value that the packet's layer 4 source port
            number can have and match this filter.  This value must be
            equal to or greater that the value specified for this filter
            in go3GPPIpFilterSrcL4PortMin."

        ::= { go3GPPIpFilterEntry 10 }







    -- ------------------------------------------------------------
    --
    -- 3GPP Go Reports
    --
    -- PRCs for carrying the Decision enforcement result sent from PEP to PCF,
    -- carried using the COPS REPORT message.
    -- These PRCs include support for the success or failure of the PEP in
    -- carrying out the PCF's decision or -change of the state in the GGSN.
    --

    go3gppReportTable OBJECT-TYPE
        SYNTAX          SEQUENCE OF Go3gppReportEntry
        PIB-ACCESS      notify
        STATUS          current
        DESCRIPTION
            "This table represents the success or failure of the decision enforcement and state
changes in the PEP."
        ::= { go3gppReportClasses 1 }


    go3gppReportEntry OBJECT-TYPE
        SYNTAX          go3gppReportEntry
        STATUS          current
        DESCRIPTION
            ""
        PIB-INDEX { go3gppReportPrid }
        UNIQUENESS { go3gppReportX }
        ::= { go3gppReportTable 1 }
```

```
go3gppReportEntry ::= SEQUENCE {
        go3gppReportPrid       InstanceId,
        go3gppReportStatus     INTEGER,
        go3gppReportDetails    Prid }


go3gppReportPrid OBJECT-TYPE
    SYNTAX         InstanceId
    STATUS         current
    DESCRIPTION
        "An arbitrary integer index that uniquely identifies an
        instance of the go3gpgReport class."
        ::= { go3gppReportEntry 1 }


go3gppReportStatus OBJECT-TYPE
    SYNTAX         INTEGER {
                             success (0),
                             failure (1),
                             usage   (2) }
    STATUS         current
    DESCRIPTION
        "When Status is:
           success: Indicates the successful implementation of the
                    decision.
                    go3gppReportDetails:
                      Reference an instance of go3gppRprtGPRSChrgInfo
                      for initial authorization request decision;
                      References nothing otherwise (contains the value
                      zeroDotZero).

           Failure: Indicates the failure of implementing the decision.

                    go3gppReportDetails may references an Error object,
                    or may have the value zeroDotZero when no error
                    object is needed.
           Usage:   go3gppReportDetails references an instance of
                    go3gppRprtUsage class."

    ::= { go3gppReportEntry 2 }


go3gppReportDetails OBJECT-TYPE
    SYNTAX         Prid
    STATUS         current
    DESCRIPTION
        "May reference an instance of go3gppRprtGPRSChrgInfo,
        go3gppRprtError, or go3gppRprtUsage class, or may have
        the value of zeroDotZero depending on the value of
        go3gppReportStatus."
    ::= { go3gppReportEntry 3 }


go3gppRprtGPRSChrgInfoTable OBJECT-TYPE
    SYNTAX         SEQUENCE OF Go3gppRprtGPRSChrgInfoEntry
    PIB-ACCESS     notify
    STATUS         current
    DESCRIPTION
        "This table represents the GPRS Charging information"
    ::= { go3gppReportClasses 2 }


go3gppRprtGPRSChrgInfoEntry OBJECT-TYPE
    SYNTAX         go3gppRprtGPRSChrgInfoEntry
    STATUS         current
    DESCRIPTION
        "This entry represents the GPRS Charging Identifier and GGSN address."
    PIB-INDEX { go3gppRprtGPRSChrgInfoPrid }
    UNIQUENESS { go3gppRprtGPRSChrgInfoGGSNAddr,
                 go3gppRprtGPRSChrgInfoGCID }
    ::= { go3gppRprtGPRSChrgInfoTable 1 }


go3gppRprtGPRSChrgInfoEntry ::= SEQUENCE {
        go3gppRprtGPRSChrgInfoPrid       InstanceId,

        go3gppRprtGPRSChrgInfoGGSNAddr   INETADDR,
```

```
        go3gppRprtGPRSChrgInfoGCID       OCTET STRING }


go3gppRprtGPRSChrgInfoPrid OBJECT-TYPE
    SYNTAX         InstanceId
    STATUS         current
    DESCRIPTION
        "An arbitrary integer index that uniquely identifies an
        instance of the go3gpgRprtGPRSChrgInfo class."
        ::= { go3gppRprtGPRSChrgInfoEntry 1 }


go3gppRprtGPRSChrgInfoGGSNAddr OBJECT-TYPE
    SYNTAX         INETADDR
    STATUS         current
    DESCRIPTION
        "Contains the IP Address of the GGSN providing the GCID
        upon successful handling of an Authorization Request."
        ::= { go3gppRprtGPRSChrgInfoEntry 2 }


go3gppRprtGPRSChrgInfoGCID OBJECT-TYPE
    SYNTAX         OCTET STRING
    STATUS         current
    DESCRIPTION
        "The GPRS Charging ID related to this Authorization Request."
    ::= { go3gppRprtGPRSChrgInfoEntry 3 }


--
-- Notice go3gppRprtError PRC is currently not defined because all
-- error condition handling is satifactoryly covered by using the
-- standard COPS-PR error handling mechanism and error objects.
-- go3gppRprtError PRC should only be used for 3GPP GO Application
-- error indications when necessary.
--

go3gppRprtUsageTable OBJECT-TYPE
    SYNTAX         SEQUENCE OF Go3gppRprtUsageEntry
    PIB-ACCESS     notify
    STATUS         current
    DESCRIPTION
        ""
    ::= { go3gppReportClasses 3 }


go3gppRprtUsageEntry OBJECT-TYPE
    SYNTAX         go3gppRprtUsageEntry
    STATUS         current
    DESCRIPTION
        "This entry represents the PEP state changes."
    PIB-INDEX { go3gppRprtUsagePrid }
    UNIQUENESS { go3gppRprtUsageIndication }
    ::= { go3gppRprtUsageTable 1 }


go3gppRprtUsageEntry ::= SEQUENCE {
        go3gppRprtUsagePrid        InstanceId,
        go3gppRprtUsageIndication  INTEGER }



go3gppRprtUsagePrid OBJECT-TYPE
    SYNTAX         InstanceId
    STATUS         current
    DESCRIPTION
        "An arbitrary integer index that uniquely identifies an
        instance of the go3gpgRprtUsage class."
        ::= { go3gppRprtUsageEntry 1 }


go3gppRprtUsageIndication OBJECT-TYPE
    SYNTAX         INTEGER {
                     chngdTo0kbs   (0),
                     chngdFrom0kbs (1) }
    STATUS         current
    DESCRIPTION
        "Indication of GPRS Usage change.
```

```
            chngdTo0kbs indicates changing to 0kbs,
            chngdFrom0kbs indicates changing from 0kbs."
            ::= { go3gppRprtUsageEntry 2 }


   -- ------------------------------------------------------------
   --
   -- Conformance Section
   --

   go3gppCompliances         OBJECT IDENTIFIER ::= { go3gppConformance 1 }
   go3gppGroups              OBJECT IDENTIFIER ::= { go3gppConformance 2 }

   MODULE -- this module
    MANDATORY-GROUPS {
    -- Include here a list of PRCs in Framework PIB used by 3GPP GO PIB.
    -- These PRC include ones used to indicate which 3GPP GO PIB PRCs are
    -- Supported by a particular PEP/GGSN. The complete set of PRCs defined in the Go 3GPP PIB are
to be included in this section since they are all deemed part of the minimum functionality for the
PEP.
    }

   MODULE -- this module
    MANDATORY-GROUPS {
    -- Include here the Group and PRC Conformance definitions for 3GPP GO PIB
    -- PRCs.
    }

-- Security considerations
-- The security mechanisms described in COPS [7] and COPS-PR [8] are
-- re-used in 3GPP. No security concerns have been identified beyond
-- those that the COPS base protocol security have already addressed
-- and provide the necessary protection against security threats.

   END
```

Annex C (informative): Change history

Change history							
Date	**TSG #**	**TSG Doc.**	**CR**	**Rev**	**Subject/Comment**	**Old**	**New**
2001-07		N3-010284			Version 0.0.0 presented to CN3 #18 – Dresden	x.y.z	0.0.0
2001-07		N3-010335			Tdocs N3-010286 and N3-010325 are agreed at CN3 #18 – Dresden, Germany and incorporated. Raised to Version 0.1.0.	0.0.0	0.1.0
2001-10		N3-010480			Tdoc N3-010460 is agreed at CN3 #19 – Brighton, U.K. and incorporated. Deletion of clause 5.4 is also agreed. Raised to Version 0.2.0.	0.1.0	0.2.0
2001-11		N3-010577			Tdocs N3-010574, N3-010573, N3-010546, N3-010553, and N3-010525 are agreed with some modifications at CN3 #20 – Cancun, Mexico and incorporated. Raised to Version 0.3.0.	0.2.0	0.3.0
2001-11		N3-010611			Tdoc N3-010547 is agreed at CN3 #20 – Cancun, Mexico and incorporated. Raised to Version 0.4.0.	0.3.0	0.4.0
2001-11		N3-010614			The figure 4.2-1 is modified based on comments. Raised to Version 0.5.0.	0.4.0	0.5.0
2002-02		N3-020120			Tdocs N3-020028 and N3-020109 are agreed at CN3 #21 – Sophia Antipolis, France and incorporated. Raised to Version 0.6.0.	0.5.0	0.6.0
2002-02		N3-020157			Tdocs N3-020152, N3-020132, N3-020129, N3-020145, N3-020133, N3-020130, N3-020156, N3-020126, N3-020137, N2-020128, N3-020136 and N3-020138 are agreed with some modifications at Go drafting session in CN3 #21 Bis – Sophia Antipolis, France and incorporated. Raised to Version 0.7.0.	0.6.0	0.7.0
2002-02		N3-020158			Tdocs N3-020151 (restructuring), N3-020160, and N3-020159 are agreed with some modifications at Go drafting session in CN3 #21 Bis – Sophia Antipolis, France and incorporated. Raised to Version 0.8.0.	0.7.0	0.8.0
2002-02		N3-020166			Tdocs N3-020163 (additions to gate function) and N3-020161 (UMTS Go PIB) are agreed with some modifications at last day of CN3 #21 Bis – Sophia Antipolis, France and incorporated. Raised to Version 0.9.0.	0.8.0	0.9.0
2002-02		N3-020168			Addition of security consideration regarding the UMTS Go PIB. Raised to Version 0.9.0.	0.9.0	0.10.0
2002-02		NP-020078			Some editorial cleaning - presented to NP#15 for information	0.10.0	1.0.0
2002-04		N3-020364			Tdocs N3-020244, N3-020248, N3-020305, N3-020306, N3-020319, N3-020320, N3-020321, N3-020325, N3-020335, N3-020337, N3-020338, N3-020339, N3-020341, N3-020342, N3-020343, and N3-020347 are agreed at CN3 #22 – Fort Lauderdale, Florida, USA and incorporated. Raised to Version 1.1.0.	1.0.0	1.1.0
2002-05		N3-020514			Tdocs N3-020367, N3-020391, N3-020393, N3-020443, N3-020444, N3-020447, N3-020449, N3-020464, N3-020482, N3-020483, N3-020487, N3-020488, N3-020489, N3-020497, N3-020498, N3-020502, and N3-020511 are agreed at CN3 #23 – Budapest, Hungary and incorporated. Raised to Version 1.2.0.	1.1.0	1.2.0
2002-05		N3-020517			Comments agreed at Go drafting session in CN3 #23 – Budapest, Hungary are incorporated. Raised to Version 1.3.0.	1.2.0	1.3.0
2002-05		N3-020522			Tdocs N3-020389, N3-020465, N3-020516, and N3-020520 are agreed at CN3 #23 – Budapest, Hungary and incorporated. Raised to Version 1.4.0.	1.3.0	1.4.0
2002-05		N3-020524			Comments agreed at CN3 #23 – Budapest, Hungary are incorporated. Raised to Version 1.5.0.	1.4.0	1.5.0
2002-06	NP#16	NP-020167			Presented to NP#16 as v2.0.0 for approval	1.5.0	2.0.0