

Internet Draft

October 2001

Expiration: May 2002

File: [draft-salsano-cops-odra-00.txt](#)

Stefano Salsano

Univ. of Rome "Tor Vergata"

COPS Usage for Outsourcing Diffserv Resource Allocation

Status of this Memo

This document is an Internet-Draft and is in full conformance with all provisions of [Section 10 of RFC2026](#).

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF), its areas, and its working groups. Note that other groups may also distribute working documents as Internet-Drafts.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

The list of current Internet-Drafts can be accessed at <http://www.ietf.org/ietf/1id-abstracts.txt>

The list of Internet-Draft Shadow Directories can be accessed at <http://www.ietf.org/shadow.html>.

Distribution of this memo is unlimited.

Copyright Notice

Copyright (C) The Internet Society (2001). All Rights Reserved.

Abstract

This document introduces a new client type (COPS-ODRA) for the COPS protocol to support dynamic DiffServ admission control. The Policy Decision Point (PDP) acts as a "Bandwidth Broker" for the Policy Enforcement Point (PEP) which is requesting resources. The use of the defined mechanism is suited for (but it is not limited to) the Integrated Services operation over Diffserv networks.

This document represents the updated version of [draft-salsano-iss11-cops-odra-00.txt](#) and is issued only for reference. Another new client type called COPS-DRA (Diffserv Resource Allocation) includes all the functionality of COPS-ODRA adding the support of combined Outsourcing and Provisioning model.

Table of Contents

Abstract.....	1
Glossary.....	2
1 . Introduction	3
2 . Interaction between the COPS-ODRA PEP and PDP/BB	4
3 . Message Content	6
4 . COPS-ODRA Protocol Objects	9
5 . COPS-ODRA Client-Specific data format	12
6 . Security Considerations	14
7 . IANA Considerations	14
8 . Illustrative examples for COPS-ODRA client	14
9 . References	15
10. Author Information and Acknowledgements	16

Glossary

BB	Bandwidth Broker
ER	Edge Router
COPS	Common Open Policy Service. See [COPS]
PDP	Policy Decision Point. See [RAP]
PEP	Policy Enforcement Point. See [RAP]

1. Introduction

The COPS (Common Open Policy Service) protocol [COPS] is a simple query and response protocol that allows policy servers (PDPs) to communicate policy decisions to network devices (PEP). In order to be flexible, the COPS protocol has been designed to support multiple types of policy clients. Each client-type can be described in a different usage draft. The purpose of this draft is to describe the use of COPS for outsourcing the allocation of resources in a Diffserv network. The new client-type is called "COPS-ODRA" (COPS- Outsourcing Diffserv Resource Allocation).

This document represents the updated version of [draft-salsano-iss11-cops-odra-00.txt](#) and is issued only for reference. Another new client type called COPS-DRA [COPS-DRA] (Diffserv Resource Allocation) includes all the functionality of COPS-ODRA adding the support of combined Outsourcing and Provisioning model.

The use of a server to control the admission of traffic within a Diffserv domain has been considered since the very beginning of the discussion about the Diffserv architecture [[2BIT](#)]. The admission control server is typically referred to as Bandwidth Broker (BB). The communality between the BB and the PDP are described in [BBREQ]. In this draft the use of COPS for the communication between the Edge Device and the Bandwidth Broker is proposed. Therefore the Bandwidth Broker will be denoted PDP/BB. An intra-domain scenario is assumed, where the PDP/BB is in charge of controlling resource for a network in a single administrative domain.

Two main models are supported by the COPS protocol: Outsourcing and Provisioning. The COPS-ODRA client relies on the Outsourcing model. The PEP explicitly asks the PDP/BB for a given amount of resources, from an ingress point to an egress point. Note that per-flow state is not stored in the PDP/BB. Instead, resource allocation requests are properly aggregated and only aggregate state information is kept in the PDP/BB, allowing for higher scalability. In this context, resource allocation is strictly related to admission control: the server controls the allocation of resources by admitting or rejecting the requests coming from its clients.

An example application scenario for the COPS-ODRA is the Intserv operation over Diffserv networks, as described in [INTDIF]. In this scenario, the Edge Router includes the PEP and interacts with the PDB/BB using COPS-ODRA according to the reservation requests coming from RSVP messages. An architectural definition and scalability analysis of this scenario can be found in [ICC99]

The examples used in this document are biased towards this type of RSVP/Diffserv interworking. However, the COPS-ODRA client type can be used for supporting centralized Resource Allocation control arising from different types of scenario.

Salsano

Expires May 2002

3

COPS Usage for Outsourcing Diffserv Resource Allocation Oct-01

The use of COPS for resource admission control in a Diffserv network is assumed in some studies and prototypes (see for example [BBop], [UCLA-BB]). Anyway, no formal description of the protocol has been proposed so far.

[1.1](#) Discussion on the Outsourcing model

The Outsourcing model is used when there are "Trigger events" in the PEP that require a policy decision (e.g. a dynamic request to admit a new flow). The PEP delegates this decision to an external policy server (PDP). It sends a query message to the PDP, typically waiting for the response decision before admitting the new flow. See Figure 1 below.

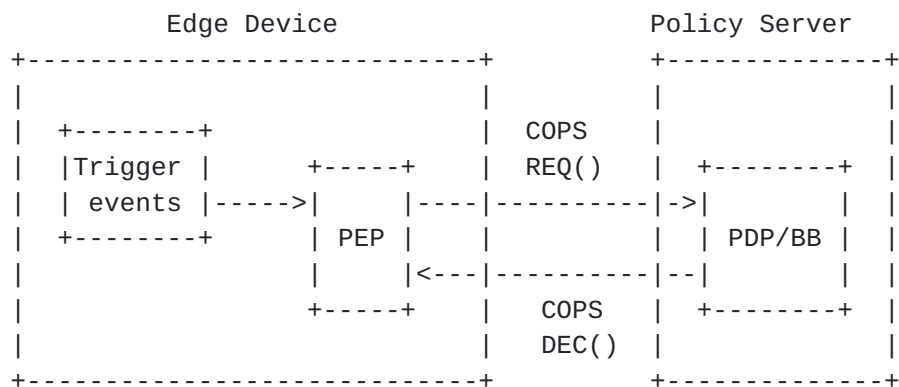


Figure 1: COPS-ODRA Outsourcing Model

On the other hand, the Provisioning model [COPS-PR] foresees that the PDP proactively configures the PEP so that it knows how to run its QoS mechanisms. The mechanisms to exchange the configuration information and to store this information is based on the definition of a "Policy Information Base", see [PIB].

In the Provisioning model, either there are no "Trigger events" at the PEP (i.e. only packet classification, marking, scheduling, etc. is performed at the PEP) or these events must be handled using local

information (i.e. mapped in the available resources provisioned by the PDP).

The Provisioning model is very well suited when there are no such dynamic requests coming to the PEP. In other scenarios, like for example the RSVP/Diffserv interworking the dynamic requests are a fundamental feature in the PEP/Edge Router. In this case a possible solution is to fully rely on the Outsourcing model, so that a very simple PEP can be defined. The drawback is the need of relatively frequent communication with the PDP, but there are scenarios where this solution can be cost-effective.

Note that the Outsourcing model lends itself to more sophisticated solutions if scalability concerns arise. In fact the Outsourcing model can be dynamically paced by the PEP in real-time. A straightforward option is to pre-reserve some amount of bandwidth to limit the pace of Resource Admission Requests. The definition of these mechanisms is out of the scope of this document, which focuses on the protocol specification for the COPS-ODRA client.

Salsano	Expires May 2002	4
COPS Usage for Outsourcing Diffserv Resource Allocation		Oct-01

In general, a combination of Outsourcing and Provisioning model could be used to provide a flexible and general solution for QoS in IP networks, but this is outside the scope of this document.

2. Interaction between the COPS-ODRA PEP and PDP/BB

This section describes the interaction between a PDP and Diffserv Resource Admission Control COPS client.

2.1. Start of operations

In order to start operations, the PEP must open the dialogue connection with its PDP/BB. First, a TCP connection is established between the client and server and the PEP sends a Client-Open message with the Client-Type = COPS-ODRA. If the PDP supports this client type, it responds with a Client-Accept (CAT) message. If the client type is not supported, a Client-Close (CC) message is returned by the PDP to the PEP. After receiving the CAT message, the PEP can send requests to the server.

2.2. Common operations

Following the terminology in [BBREQ], the PEP will send "Resource Allocation Requests" (RAR) to the PDP/BB once the connection is established.

A Resource Allocation Request will contain:

- the topological information (e.g. ingress point and egress point in the Diffserv domain) which allows the PDP/BB to aggregate different Resource Requests
- the type of the requested resource (e.g. the Diffserv Per Hop Behavior or Behavior Aggregates)
- the amount of the requested resource (e.g. the specification of the bandwidth)

The resource allocation request is used by the PDP to perform the Admission Control procedure. According to the requirements of the requested service, the PDP/BB will properly map the requested edge-to-edge resources into network resources and will assure that such resources are available throughout the Diffserv cloud. The discussion of the Admission Control algorithms in the PDP/BB and of the mechanisms used by the PDP/BB to get topological/routing information from the Diffserv domain are outside the scope of this document. Related work can be found in [IWQOS99], where the use of OSPF and SNMP for the communication between the BB and the Diffserv routers is proposed.

In response to the Resource Allocation request, the PDP sends a reply where it accepts or rejects the RAR. On receiving the answer, the PEP activates its local QoS mechanisms as needed.

Salsano Expires May 2002 5
COPS Usage for Outsourcing Diffserv Resource Allocation Oct-01

2.3 State information in PEP and PDP/BB

The COPS protocol is stateful in the sense that Request/Decision state is shared between PEP and PDP. Depending on the COPS client type, one or multiple states can be installed in the context of a single PEP/PDP relationship. The "handle" object uniquely identifies a single installed state at the PEP and at the PDP side. In case of RSVP client type [COPS-RSVP], a different state is installed for each RSVP flow (actually one PATH state and one RESV state). This implies that a lot of state information is duplicated in the PEP and in the server.

In COPS-ODRA the state information is aggregated: the state represents the set of resources allocated by the PDP/BB to a PEP. Therefore a unique value for the handle object is used in the context of a single PEP/PDP relationship. The handle is inserted by the PEP in the first request and then it is used in every message by the PEP and by the PDP/BB.

The state information in the PDP/BB is represented by the set of the triples (resource type, ingress point, egress point) and the corresponding amount of allocated resources. The PDP/BB keeps this state information separately for each different COPS-ODRA client (i.e. for each connected PEP). The requests for the same resource type, ingress

point, egress point coming from the same PEP are properly composed by the PDP/BB so that only the aggregate information is stored. This aggregate state information stored in PDP/BB is logically shared by the PEP in the sense that it is the result of the sequence of Request messages sent and of the Decision messages received. There is no need in the PEP to evaluate and store the aggregate state information: in the simplest case, the client side (PEP) stores a set of "per flow" reservation information. In more advanced scenarios, the PEP client can evaluate and store aggregate information. Temporary state information per each request must be stored in the PEP and in the PDP/BB in order to correlate requests with decisions. To this purpose the notion of request ID is needed and a corresponding client specific protocol object is defined (see [section 3](#)). This temporary information is deleted in the PDP/BB when the Decision message is sent and in the PEP when this message is received.

[2.4](#) Synchronization

Synchronization procedures are foreseen in COPS specification to cover failure situations. The basic idea is that the PDP/BB can "reset" the state and ask the PEP to rebuild it by sending proper resource allocation Request messages. If the PEP has only stored "per-flow" state, it will send one Request message for each active reservation. If the PEP has stored aggregate states, it can send "aggregate" Resource Allocation requests.

Selective re-submissions (i.e. for resource type, ingress or egress point) can be supported.

Salsano	Expires May 2002	6
COPS Usage for Outsourcing Diffserv Resource Allocation		Oct-01

[3.](#) Message Content

The COPS protocol provides for different COPS clients to define their own "named", i.e. client-specific, information for various messages. This section describes the messages exchanged between a COPS server (PDP) and COPS ODRA clients (PEP) that carry client-specific data objects.

[3.1.](#) Request (REQ) PEP -> PDP

The REQ message is sent by COPS-ODRA clients to issue a 'Resource Allocation Request' to the PDP.

The Resource Allocation Request REQ is used to request new resources, to

modify a previous reservation, to release a reservation.

The Client Handle associated with the REQ message originated by the ODRA client must be unique for that client but otherwise has no protocol significance at this time.

The context object specifies the types of request (Add, Release, Modify, see the description of Context object).

The Client Specific info object is used to specify the requested resources and the request identifier.

Each REQ message contains a single request. The PDP responds to the resource allocation request with a DEC message containing the answer to the query.

The REQ message has the following format:

```
<Request> ::= <Common Header>
               <Client Handle>
               <Context = resource allocation>
               [<ClientSI: RAR data>]
               [<Integrity>]
```

Note that the COPS objects IN-Int, OUT-Int and LDPDecisions are not included in a COPS-ODRA REQ message.

Note that resource allocation request messages can be generated and sent to the PDP in response to the receipt of a Synchronize State Request (SSQ) message.

3.2. Decision (DEC) PDP -> PEP

The DEC message is sent from the PDP to a COPS-ODRA client in response to the REQ message received from the PEP. Unsolicited DEC messages cannot be sent for this client type (therefore the solicited decision flag is always set).

The Client Handle must be the same Handle that was received in the REQ message (it is identical for all requests).

Salsano	Expires May 2002	7
COPS Usage for Outsourcing Diffserv Resource Allocation		Oct-01

The Decision object will contain in the Client Specific info the Request ID, which is needed by the PEP to correlate the reply with the query.

Each DEC message contains a single decision.

The DEC message for the COPS-ODRA client type has the following format:

```
<Decision Message> ::= <Common Header>
```

```
<Client Handle>
<Decision> | <Error>
[<Integrity>]
```

The decision object in turn has the following format:

```
<Decision> ::= <Context>
               <Decision: Flags>
               <Decision: Client SI data>
```

The context object will be the same as contained in the REQ message. The Decision: Flags object will contain the answer in the Command-code field according to the COPS specifications. In particular the Command-code will be "Install" to mean a positive answer and "Remove" to mean a negative answer. The following text clarifies how Install and Remove Decisions map into the different request types.

REQ (Context = Add)

DEC (Dec Flag = Install) -> The requested resources are
successfully allocated

DEC (Dec Flag = Remove) -> The requested resources are not
allocated

REQ (Context = Release)

DEC (Dec flag = Install) -> The resources are released

DEC (Dec Flag = Remove) -> The resources are still allocated.
(This is syntactically correct, but
it make no sense)

REQ (Context = Modify)

DEC (Dec flag = Install) -> The modification is accepted. The
newly requested resources are
allocated, while the previous ones
have been released

DEC (Dec Flag = Remove) -> The modification is not accepted. The
previous allocation is still active.

Salsano

Expires May 2002

8

COPS Usage for Outsourcing Diffserv Resource Allocation Oct-01

(This also makes sense only if the
modification increases the allocated
resource)

The Error object is used to communicate COPS protocol error to the PEP, according to the definition in [COPS]. No client specific error sub-codes are used by COPS-ODRA.

No report is sent by the PEP to confirm the reception of a Decision message. Only in case of specific errors, the PEP will send back a Report State message to the PDP/BB.

3.3. Report State (RPT) PEP -> PDP

For COPS-ODRA client type, the Report State message is sent by the PEP to the PDP in case of problems with a received Decision message. More specifically it is used to communicate that the Decision contains a Request identifier which cannot be correlated to a previous request. This event is the manifestation of abnormal behavior. On reception of a Report State message the PDP could start a Synchronization procedure.

The RPT message for the COPS-ODRA client type has the following format:

```
<Report State Message> ::= <Common Header>
                             <Client Handle>
                             <Report Type>
                             <ClientSI: Req ID>
                             [<Integrity>]
```

3.4. Synchronize State Request (SSQ) PDP -> PEP

The Synchronize State Request message is sent by the PDP to the PEP to "reset" the state information. It requests the PEP to send the set of resource allocation REQ messages needed to rebuild the state. The SSQ can apply to the whole set of PEP active reservations PEP, or to a specific resource type and ingress-egress couple, depending on the information contained in the Client SI object.

```
< Synchronize State> ::= <Common Header>
                          <Client Handle>
                          [<ClientSI: SSQ scope>]
                          [<Integrity>]
```

Note that the COPS specification in [COPS] does not foresee a ClientSI object in the SSQ message.

3.5. Synchronize State Complete (SSC) PEP -> PDP

The Synchronize State Complete message is sent by the PEP to the PDP to inform that all the REQ messages needed to rebuild the state have been sent. The Client SI object is the same received in the SSQ message and specifies the scope of the synchronization procedure which has been completed.

```

< Synchronize State Complete> ::= <Common Header>
                                   <Client Handle>
                                   [<ClientSI: SSQ scope>]
                                   [<Integrity>]

```

Note that the COPS specification in [COPS] does not foresee a ClientSI object in the SSC message.

4. COPS-ODRA Protocol Objects

A new COPS client type for outsourcing Diffserv resource allocation is defined:

Client Type = 0x4001; Outsourcing Diffserv Resource Allocation Client (ODRA Client)

COPS messages sent between an ODRA client and a COPS server contain a COPS Common Header with this ODRA Client type specified:

0	1	2	3
Version	Flag	Op Code	Client Type = 0x4001
+-----+-----+-----+-----+			
		Message Length	
+-----+-----+-----+-----+			

The COPS-ODRA uses new COPS protocol objects that carry client-specific information. This section defines those new objects.

The format for these new objects is as follows:

S-Num and S-Type are similar to the C-Num and C-Type used in the base COPS objects. The difference is that S-Num and S-Type are used only for ClientSI specific objects.

Length is a two-octet value that describes the number of octets (including the header) that compose the object. If the length in octets does not fall on a 32-bit word boundary, padding must be added to the end of the object so that it is aligned to the next 32-bit boundary before the object can be sent on the wire. On the receiving side, a subsequent object boundary can be found by simply rounding up the previous stated object length to the next 32-bit boundary.

4.1. Request ID Object (ReqID)

The Request ID object is used to correlate the Requests sent by the COPS-ODRA client with the responses coming from the PDP/BB. The ReqID

Salsano Expires May 2002 10
COPS Usage for Outsourcing Diffserv Resource Allocation Oct-01

object is chosen by the PEP and it is opaque to the PDP/BB. A different ReQID object is chosen by the PEP for each Request.

The ReqID object is carried in the Client Specific Information Object for Request message (PEP -> PDP) and in the Decision ClientSI Data Object (PDP -> PEP).

If the PEP receives a Decision with an unrecognized ReqID object, it will send a Report State Message to the PDP to communicate the error.

S-Num = 1, S-Type = 1

0	1	2	3
Length	S-Num = 1	S-Type = 1	
Request ID			

[4.2.](#) Ingress Point Address (IPA)

This object specifies the address of the Ingress Point into the Diffserv domain.

The IPA object is carried in the Client Specific Information Object.

In the RSVP Diffserv interaction scenario the Ingress point coincides with the requesting PEP itself.

S-Num = 2

S-Type = 1, Ipv4 Address.

0	1	2	3
Length	S-Num = 2	S-Type = 1	
IPv4 Address format			

S-Type = 2, Ipv6 Address.

0	1	2	3
Length	S-Num = 2	S-Type = 2	

```

|
+
//                               Ipv6 Address format
+
|
+-----+-----+-----+-----+

```

Salsano Expires May 2002 11
COPS Usage for Outsourcing Diffserv Resource Allocation Oct-01

4.3. Egress Point Address Object (EPA)

This object specifies the address of the Egress Point from the Diffserv domain.

The EPA object is carried in the Client Specific Information Object.

S-Num = 3

S-Type = 1, Ipv4 Address.

S-Type = 2, Ipv6 Address.

See Ingress Point Address for object coding.

4.4. Resource Type Object (RType)

This object specifies the type of the requested resources. In a Diffserv scenario, the simplest option is to specify a DS codepoint. The Rtype object is carried in the Client Specific Information Object.

S-Num = 4

S-Type = 1 Diffserv Codepoint (DSCP)

```

          0          1          2          3
+-----+-----+-----+-----+
|          Length          | S-Num = 4 | S-Type = 1 |
+-----+-----+-----+-----+
|          ////////////////          | // | DSCP |
+-----+-----+-----+-----+

```

More complex Diffserv edge-to-edge services can be listed defining different "S-Type".

4.5. Traffic Descriptor Object (TD)

This object specifies the amount of the requested resources.

The TD object is carried in the Client Specific Information Object.

S-Num = 5

S-Type = 1 (Bandwidth)

0	1	2	3
Length	S-Num = 5	S-Type = 1	
Bandwidth (bytes/s)			

This description can be used if there is not a more accurate indication on the policing at the ingress interface.

Salsano Expires May 2002 12
COPS Usage for Outsourcing Diffserv Resource Allocation Oct-01

S-Type = 2 (Token Size and Measurement Interval)

0	1	2	3
Length	S-Num = 5	S-Type = 2	
Token Size (bytes)			
Measurement Interval (msec)			

This description can be used if there is an indication on the policing at the ingress interface. This information could be used by the PDP/BB in the admission control algorithm.

4.6. Reject Reason Object (RejRea)

The RejRea object is used to provide the reason for a negative Decision. It is carried in the Decision Client SI Data object.

S-Num = 6, S-Type = 1

0	1	2	3
Length	S-Num = 6	S-Type = 1	
Reason code			

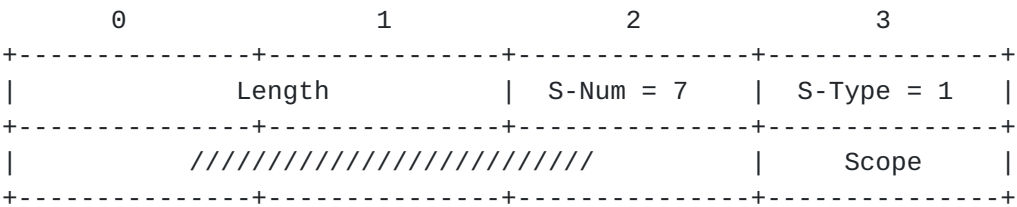
Reason codes are:

Reason code = 1 : Resource unavailable
Reason code = 2 : Unsupported resource type
Reason code = 3 : Unacceptable Ingress Point
Reason code = 4 : Unacceptable Egress Point

4.7. Synchronize State scope (SSQscope)

The SSQscope object is used to specify the scope of a Synchronize State operation. The SSQscope object is carried in the Client Specific Information Object for the SSQ and SSC messages.

S-Num = 7, S-Type = 1



Allowed value for Scope field are:

Salsano	Expires May 2002	13
COPS Usage for Outsourcing Diffserv Resource Allocation		Oct-01

0 : Generic scope

1 : Specific scope

If the Scope is Generic, the Synchronize State operation refers to the whole set of resource types and ingress-egress pairs.
If scope is Specific, the Synchronize State Request refers to the specific resource type and ingress-egress pair contained in the rest of the ClientSI Object for SSQ and SSR messages (see [section 5](#)).

5. COPS-ODRA Client-Specific data format

5.1. Context object

The context object specifies the type of request that triggered the query. It is included in the REQ and in the DEC message. For COPS-ODRA client the Request Type flag is always set to 0x02 (Resource-Allocation request). The Message Type flag is used by the COPS-ODRA client to distinguish between add, release and modify requests. Depending on the M-Type, a different content of the ClientSI object for REQ message is used.

C-num = 2, C-Type = 1

R-Type = 0x02

M-Type = 1 : Add

M-Type = 2 : Release

M-Type = 3 : Modify

5.2. Client Specific Information Object for REQ message.

The Client Specific Information Object carried in the REQ messages for COPS-ODRA client contains the description of the resources and has a different format depending on whether the type of the request is Add/Release or Modify, as specified by the context object.

If the Context object specifies an Add or Release request, the ClientSI object has the following format:

```
<Client SI: RAR data> ::= <Request ID>
                           <Ingress Point Address>
                           <Egress Point Address>
                           <Resource Type>
                           <Traffic Description>
```

If the Context object specifies a Modify request, the previous values for Resource Type and Traffic Description are appended at the end of ClientSI object:

```
<ClientSI: RAR data> ::= <Request ID>
                           <Ingress Point Address>
```

Salsano	Expires May 2002	14
COPS Usage for Outsourcing Diffserv Resource Allocation		Oct-01

```
<Egress Point Address>
<Resource Type> (new value)
<Traffic Description> (new value)
<Resource Type> (old value)
<Traffic Description> (old value)
```

5.3. Decision Client Specific Info Data object.

The Decision ClientSI data object carries the information needed to correlate the decision with the request and some optional information to explain negative Decisions. It has the following format:

```
<Decision: Client SI data> ::= <Request ID>
                               [<Reject Reason>]
```

5.4. Client Specific Information Object for SSQ and SSC messages.

The Client Specific Information Object carried in the SSQ messages for COPS-ODRA client describes the scope of the requested (SSQ message) and

performed (SSC message) synchronize operation. This object has the following format:

```
<ClienSI: SSQ scope> ::= <SSQ scope>
                        [<Resource Type>]
                        [<Ingress Point Address>]
                        [<Egress Point Address>]
```

If the "Specific scope" value is expressed in the SSQ scope object, then the Resource Type, Ingress Point Address and Egress Point Address object must be included.

5.5 Report Type Object

The Report Type is contained in the Report State message. Only failures are communicated.

C-num = 12, C-Type = 1

Report-Type = 2 : Failure

5.6. Client Specific Information Object for Report State message.

The Client SI object for Report State message is used to communicate to the PDP that a Request ID was not recognized.

```
<ClientSI: ReqID> ::= <Request ID>
```

Salsano	Expires May 2002	15
COPS Usage for Outsourcing Diffserv Resource Allocation		Oct-01

6. Security Considerations

This document relies on COPS for its signaling and its security. Please refer to section "Security Considerations" in [COPS].

7. IANA Considerations

Following the consideration in [COPS], the COPS Diffserv Resource Allocation has been temporarily assigned a client-type value in the range for private use. Future versions of the draft could define new client-type value from the "First Come First Served" range.

8. Illustrative examples for COPS-ODRA client

This section details two examples related to a successful and to an unsuccessful reservation respectively.

In the following example the PEP requests the reservation of 1 Mb/s of EF traffic between ingress point 192.168.1.1 and the egress point 192.168.129.1.

```
PEP --> PDP   REQ := <Handle H>
                  <Context: Resource allocation, Add>
                  <ClientSI:
                    Request ID: X
                    IPA: 192.168.1.1
                    EPA: 192.168.129.1
                    Request type (DSCP): EF
                    TD (Bandwidth): 125kBytes/s
                  >
```

The PDP accepts the reservation.

```
PDP --> PEP   DEC := <Handle H>
                  <Context: Resource allocation, Add>
                  <Decision: Command = Install>
                  <Decision: Client SI
                    Request ID: X
                  >
```

In the following example the PEP requests the reservation of 2 Mb/s of EF traffic between ingress point 192.168.1.1 and the egress point [192.168.129.1](#). Note that the same Handle is used, while the Request ID is different.

```
PEP --> PDP   REQ := <Handle H>
                  <Context: Resource allocation, Add>
                  <ClientSI:
                    Request ID: Y
                    IPA: 192.168.1.1
```

Salsano	Expires May 2002	16
COPS Usage for Outsourcing Diffserv Resource Allocation		Oct-01

```
                  EPA: 192.168.129.1
                  Request type (DSCP): EF
                  TD (Bandwidth): 250kBytes/s
                  >
```

The PDP rejects the reservation.

```
PDP --> PEP   DEC := <Handle H>
```

```

<Context: Resource allocation, Add>
<Decision: Command = Remove>
<Decision: Client SI
    Request ID: Y
>

```

9. References

- [2BIT] K. Nichols, V. Jacobson, L. Zhang " A Two-bit Differentiated Services Architecture for the Internet " [RFC 2638](#), July 1999
- [BBop] First Internet2 bandwidth broker operability event
<http://www.merit.edu/internet/working.groups/i2-qbone-bb/inter-op/index.htm>
- [BBREQ] R. Neilson, J. Wheeler, F. Reichmeyer, S. Hares (Editors), "A Discussion of Bandwidth Broker Requirements for Internet2 Qbone Deployment" Version 0.7
- [COPS] D. Durham, Ed., J. Boyle, R. Cohen, S. Herzog, R. Rajan, A. Sastry "The COPS (Common Open Policy Service) Protocol", IETF [RFC 2748](#), January 2000
- [COPS-DRA] S. Salsano "COPS usage for Diffserv Resource Allocation (COPS-DRA)", <[draft-salsano-cops-dra-00.txt](#)>, September 2001, Work in Progress
- [COPS-PR] K. Chan, J. Seligson, D. Durham, S. Gai, K. McCloghrie, S. Herzog, F. Reichmeyer, R. Yavatkar, A. Smith "COPS Usage for Policy Provisioning (COPS-PR)", IETF [RFC 3084](#), March 2001.
- [ICC99] A. Detti, M. Listanti, S. Salsano, L. Veltri, "Supporting RSVP in a Differentiated Service Domain: an Architectural Framework and a Scalability Analysis", ICC '99, June 1999, Vancouver, Canada.
- [INTDIF] Y. Bernet, R. Yavatkar, P. Ford, F. Baker, L. Zhang, M. Speer, R. Braden, J. Wrocklaski, E. Felstaine, A Framework for Integrated Services Operation Over Diffserv Networks, IETF [RFC 2998](#), November 2000
- [IWQoS99] O. Schelen, A. Nilsson, J. Norrgard, S. Pink: Performance of QoS Agents for Provisioning Network Resources. In Proceedings of IFIP Seventh International Workshop on Quality of Service (IWQoS'99), London, UK, June 1999.
- [PIB] M. Fine, K. McCloghrie, J. Seligson, S. Hahn, K. Chan, A. Smith, "Framework Policy Information Base", <[draft-ietf-rap-frameworkpib-03.txt](#)>, November 2000, Work in Progress
- [RAP] R. Yavatkar, D. Pendarakis, R. Guerin, "A Framework for Policy Based Admission Control", IETF [RFC 2753](#), January 2000.

Salsano	Expires May 2002	17
COPS Usage for Outsourcing Diffserv Resource Allocation		Oct-01

[UCLA-BB] A. Terzis, J. Ogawa, S. Tsui, L. Wang, L. Zhang "A prototype

Implementation of the Two-Tier Architecture for Differentiated Services" IEEE Workshop on QoS Support for Real-Time Internet Applications, June 2-4, 1999 Vancouver, Canada

10. Author Information and Acknowledgements

Special thanks to Roberto Mameli, Andrea Ferraresi and Eleonora Manconi for their comments and suggestions and their work on the prototype implementation.

Stefano Salsano
DIE - University of Rome "Tor Vergata"
Via di Tor Vergata, 110
00133 Roma - ITALY
CoRiTel - Consorzio di Ricerca sulle Telecomunicazioni
email: salsano@coritel.it