Internet Engineering Task Force
INTERNET-DRAFT
Expires March 2004

L. Westberg
M. Jacobsson
S. Oosthoek
D. Partain
V. Rexhepi
R. Szabo
P. Wallentin
Ericsson

G. Karagiannis
University of Twente

Sept. 2003

Resource Management in Diffserv (RMD) Framework
draft-westberg-rmd-framework-04.txt

Status of this memo

Abstract

This draft presents the work on the framework for the Resource Management in Diffserv (RMD) designed for edge-to-edge resource reservation in a Differentiated Services (Diffserv) domain. The RMD extends the Diffserv architecture with new resource reservation concepts and features. Moreover, this framework enhances the Load Control protocol described in [WeTu00].

The RMD framework defines two architectural concepts:
- the Per Hop Reservation (PHR)
- the Per Domain Reservation (PDR)

The PHR protocol is used within a Diffserv domain on a per-hop basis to augment the Diffserv Per Hop Behavior (PHB) with resource reservation. It is implemented in all nodes in a Diffserv domain. On the other hand, the PDR protocol manages the resource reservation per Diffserv domain, relying on the PHR resource reservation status in all nodes. The PDR is only implemented at the boundary of the domain (at the edge nodes).

The RMD framework presented in this draft describes the new reservation concepts and features. Furthermore it describes the:
- relationship between the PHR and PHB
- interaction between the PDR and PHR
- interoperability between the PDR and external resource reservation schemes

This framework is an open framework in the sense that it provides the basis for interoperability with other resource reservation schemes and can be applied in different types of networks as long as they are Diffserv domains. It aims at extreme simplicity and low cost of implementation along with good scaling properties.

Table of Content

1. Introduction

Today's Internet applications range from simple ones such as e-mail, web browsing and file transfers to highly demanding real-time applications like audio and video streaming, IP telephony and multimedia conferencing. This diversity has influenced the user's and provider's expectations of the Internet infrastructure for satisfying the diverse service needs of the applications. In a highly competitive environment such as the Internet Service Providers' (ISPs) world, satisfying customer needs, whether they are other ISPs or end users, is key to survival. Therefore, the ISPs' zeal to provide value-added services to their customers is natural.

One significant class of such value-added services requires real-time message transport. It can be expected that these real-time services will be popular as they replicate or are natural extensions of existing communication services like telephony.

Moreover, it is expected that next generation ISP backbone networks will have to support a huge real-time traffic (mixed with best effort traffic) volume that is generated by a huge number of users.

Therefore, exact and reliable resource management (such as admission control) is essential for achieving high utilization in networks with real-time transport requirements. Solving this problem is difficult primarily due to scalability issues.

The Differentiated Services (Diffserv) architecture ([RFC2475], [RFC2638], [BeBi99]) was introduced as a result of efforts to avoid the scalability and complexity problems of Intserv [RFC1633]. Scalability is achieved by offering services on an aggregate basis rather than per-flow and by forcing as much of the per-flow state as possible to the edges of the network. The service differentiation is achieved using the Differentiated Services (DS) field in the IP header and the Per-Hop Behavior (PHB) as main building blocks. Packets are handled at each node according to the PHB indicated by the DS field in the message header.

The Diffserv domain will provide to its customer, which is a host or another domain, the required service by complying fully with the Service Level Agreement (SLA) agreed upon. The SLA can either be negotiated statically or dynamically. The transit service to be provided with accompanying parameters like transmit capacity, burst size and peak rate is specified in the technical part of the SLA, the Service Level Specification (SLS).

However, the Diffserv architecture currently does not standardize any solution for dynamic resource reservation. This memo, the RMD framework, defines a dynamic resource reservation scheme that can be used for the dynamic SLS provisioning in an edge-to-edge Diffserv domain. As such, once solutions for resource reservation are introduced, Diffserv needs to be extended with new features. Moreover, this framework enhances the Load Control protocol described in [WeTu00]. The basic functionality in the interior nodes as proposed by that memo is similar to the proposal in this memo.

The RMD framework distinguishes between two types of protocols, the Per Domain Reservation (PDR) and Per Hop Reservation (PHR) protocols:

- A Per Domain Reservation protocol is used to perform resource reservation in the complete Diffserv domain. A PDR protocol is used by the edge nodes (ingress and egress), but not by the interior nodes.

- A Per Hop Reservation protocol is used to perform a per-hop reservation, extending the Diffserv PHB. A PHR protocol is used in all nodes in the Diffserv domain (both edge and interior nodes) on a hop by hop basis.

Furthermore, the RMD framework defines:

- the relationship between the PHR and PHB
- the interaction between the PDR and PHR
- interoperability between the PDR and external resource reservation schemes

The design of the PHR and PDR protocols extends the Diffserv framework with new features necessary for the deployment of the RMD in Diffserv domains. The new features required in this reservation scheme are presented in this framework draft. As this reservation scheme is meant as a solution for a single domain, it is very important that it is able to interoperate with other resource reservation schemes used in other domains, and, as such, be part of end-to-end resource reservation mechanisms. This framework is an open framework in the sense that it provides the basis for interoperability with other resource reservation schemes and is to be applied in different types of networks as long as they are Diffserv domains. Furthermore, it is possible for the RMD framework to co-exist with statically allocated PHBs and SLSs.

The framework scheme presented in this document aims at extreme

simplicity and low cost of implementation along with good scaling properties.

1.1. Definitions/Terminology

DS behavior aggregate (identical to [RFC2475]):

A collection of packets with the same DS codepoint crossing a link in a particular direction.

DS-compliant (identical to [RFC2475]):

Enabled to support differentiated services functions and behaviors as defined in [RFC2474], this document, and other differentiated services documents; usually used in reference to a node or device.

Per Hop Behavior (PHB) (identical to [RFC2475]):

The externally observable forwarding behavior applied at a DS-compliant node to a DS behavior aggregate.

Per Hop Reservation (PHR):

The per-hop resource reservation in a Diffserv domain, extending the Diffserv PHB, e.g., the bandwidth allocated to an AF PHB (see RFC2597]), with resource reservation. It is implemented at both the interior nodes and the edge nodes.

Per Hop Reservation (PHR) protocol:

A type of protocol that is used to perform a per hop reservation. A PHR protocol is used in all nodes in the Diffserv domain (both edge and interior nodes) on a hop by hop basis.

Per Domain Behavior (PDB)(similar to [NiKa01]):

Describes the behavior experienced by a particular set of packets as they cross a DS domain. A PDB is characterized by specific metrics that quantify the treatment that a set of packets with a particular DSCP (or set of DSCPs) will receive as it crosses a DS domain.

Per Domain Reservation (PDR):

The resource reservation in the complete Diffserv domain.

Per Domain Reservation (PDR) protocol:

A type of protocol used to perform a per domain reservation. A PDR protocol is used by edge nodes (ingress and egress), but not by the interior nodes.

Edge nodes:

Nodes that are located at the boundary of a Diffserv domain.

Interior node:

All the nodes that are part of a Diffserv domain and are not edge nodes.

Ingress node:

An edge node that handles the traffic as it enters the Diffserv domain.

Egress node:

An edge node that handles the traffic as it leaves the Diffserv domain.

End Host:

QoS-aware end terminal, either fixed or mobile, i.e. running QoS-aware applications

RMD domain:

A Diffserv domain that uses the RMD framework.

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [RFC2119].

2. Overview of the RMD Framework Protocols

The RMD framework is based on Diffserv principles for QoS provisioning and extends these principles with new ones necessary to provide resource provisioning and control in Diffserv domains.

The RMD operates in a Diffserv domain and therefore support for different levels of Quality of Service (QoS) MUST be provided using Diffserv, as defined in [RFC2475].

The RMD framework will use the Diffserv classes Expedited Forwarding (EF) [RFC2598] and Assured Forwarding (AF) [RFC2597] as QoS classes. This implies that any network supporting the RMD framework MUST be able to classify, mark, police and schedule the traffic accordingly.

It is assumed that different externally defined QoS classes can be translated into these Diffserv classes (Per Hop Behaviors).

In order to maximize the scalability in the Diffserv domain the complexity imposed by the resource reservation scheme has to be moved as much as possible away from the interior nodes. Therefore, the RMD framework separates the problem of a complex reservation within a domain from a simple reservation within a node. This is accomplished by specifying two types of resource reservation protocols.

The first resource reservation protocol type is denoted as Per Hop Reservation (PHR) that enables reservation of resources per PHB in each node within a Diffserv domain. This protocol type is optimized to reduce the requirements placed on the functionality of the interior nodes. For example, the nodes that implement this protocol type do not have per flow responsibilities. This protocol can be either reservation-based or measurement-based. In the reservation-based PHR, each node keeps only one reservation state per PHB. In the measurement-based PHR no reservation states are installed and the resource availability is checked by measuring real average traffic (user) data load.

The second protocol type is denoted as Per Domain Reservation (PDR) and is responsible for the resource reservation within the complete Diffserv domain. The PDR is used by edge nodes (ingress and egress) but not by the interior nodes. This protocol introduces strict and complex requirements on the functionality implemented on the edge nodes. An example of such functionality is the mapping of the traffic parameters signalled by an external QoS request to parameters that are useful to the RMD scheme. In the RMD framework, different PDR

and PHR protocols can be used within a Diffserv domain simultaneously.

The PHR protocol is a new protocol while the PDR protocol can be either a new protocol or (one or more) existing protocols. Examples of such existing protocols can be the Resource Reservation Protocol (RSVP) [RFC2205], RSVP aggregation [RFC3175], Simple Network Management Protocol (SNMP) [RFC1905], Common Open Policy Service (COPS) [RFC2748].

There may be different levels of granularity between external QoS requests and PDR reservations, e.g., one to one, many-to-one. Similarly there may be different levels of granularity between PDR protocol actions and PHR protocol actions, e.g., one to one, one-to-many and many-to-one.

2.1. RMD framework scenarios

Two different scenarios are identified wherein this framework is applied. The first scenario illustrated in Figure 1 includes ingress nodes, egress nodes and interior nodes.

The second scenario, illustrated in Figure 2, includes in addition to the nodes depicted in Figure 1, also an "oracle" (or "agent") that is involved in the per domain reservation, but which does not provide any resources by itself. Note that combinations of the two scenarios may be possible.

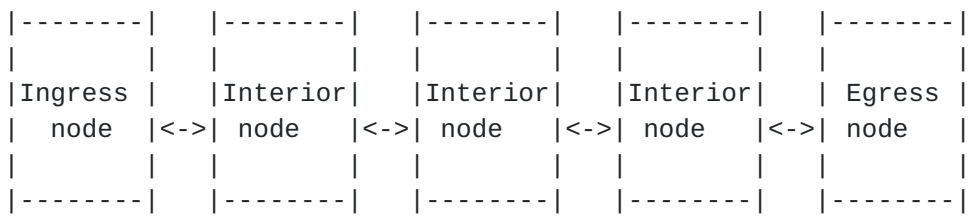


Figure 1: First scenario for the RMD framework

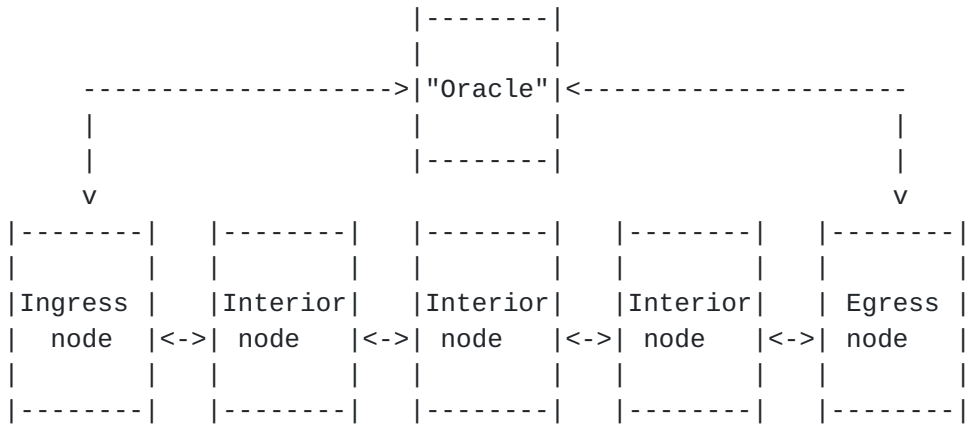


Figure 2: Second scenario for the RMD framework

Figures 3 and Figure 4 depict the peers in the communication of the PDR and PHR protocols in the two different scenarios. In Section 5 below, some examples illustrating the usage of actual PDR and PHR protocols in different scenarios are given.

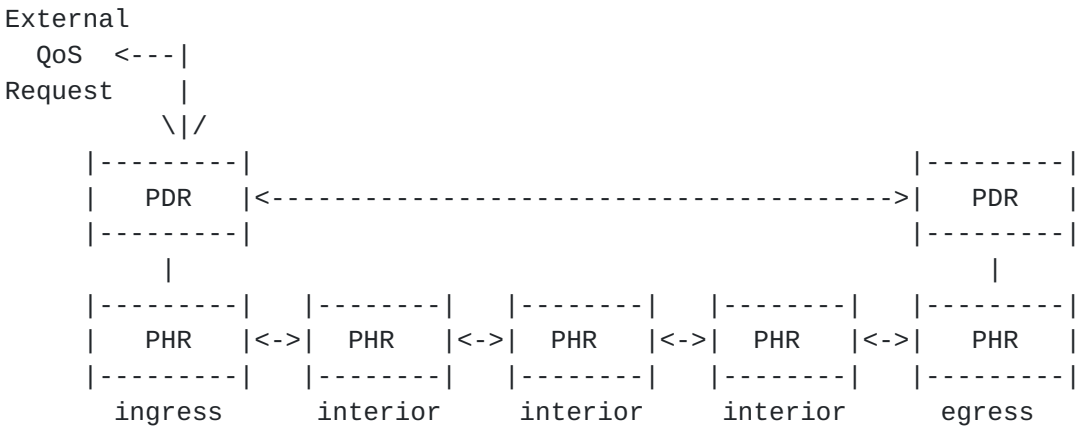


Figure 3: PDR and PHR protocol peers in the first scenario

In the first scenario, the PDR protocol is used between the ingress and egress nodes. The ingress node receives an external QoS request and initiates the per domain reservation. The PHR protocol is used between all nodes on an hop-by-hop basis along the path from the ingress to the egress. The PDR protocol may use the PHR protocol or any underlying protocol for the transport of PDR messages.

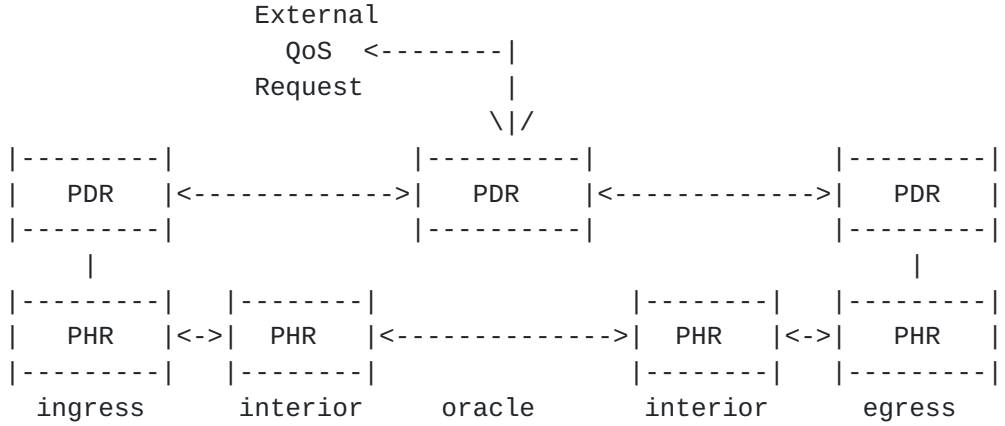


Figure 4: PDR and PHR protocol peers in the second scenario

In the second scenario, the "oracle" receives the external QoS request and uses a PDR protocol towards the ingress and egress nodes to perform the per domain reservation. Note that the "oracle" does

not use the PHR protocol.

In the RMD framework all of the PHR signalling messages are to be generated and discarded at the edge nodes (ingress and egress nodes) and not at the end hosts. Moreover, all of the PDR messages are to be generated and discarded either at the edge nodes or at the oracle.

2.2. PDR protocol functions

A PDR protocol implements all or a subset of the following functions:

* Mapping of external QoS request to a Diffserv Code Point (DSCP).

* Admission control and/or resource reservation within a domain.

* Maintenance of flow identifier and reservation state per flow (or aggregated flows), e.g. by using soft state refresh.

* Modification of an already installed reservation state.

* Notification of the ingress node IP address to the egress node.

* Notification that lost signalling messages (PHR and PDR) occurred in the communication path from the ingress to the egress nodes.

* Notification of resource availability in all the nodes located in the communication path from the ingress to the egress nodes.

* Severe congestion handling. Due to a route change or a link failure, a severe congestion situation may occur. The egress node is notified by PHR when such a severe congestion situation occurs. Using PDR, the egress node notifies the ingress node about this severe congestion situation. The ingress node resolves this situation by using a predefined policy, e.g., refusing new incoming flows and terminating a portion of the affected flows.

2.3. PHR protocol functions

A PHR protocol implements all or a subset of the following functions:

* Admission control and/or resource reservation within a node.

* Management of one reservation state per PHB by using a combination of the reservation soft state and explicit release principles.

* Measurement of the user traffic load.

* Stores a pre-configured threshold value on maximal allowable traffic load (or resource units) per PHB.

* Adaptation to load sharing. Load sharing allows interior nodes to take advantage of multiple routes to the same destination by sending via some or all of these available routes. The PHR protocol has to adapt to load sharing once it is used.

* Severe congestion notification. This situation occurs as a result of route changes or a link failure. The PHR has to notify the edges about the occurrence of this situation.

* Transport of transparent PDR messages. The PHR protocol may encapsulate and transport PDR messages from an ingress node to an egress node.

3. The PDR protocols

3.1. Introduction

A PDR protocol component interacts with external resource requests (via, for example, RSVP [RFC2205]) and with the PHR protocol component for handling resources within the edge-to-edge domain.

A PDR protocol manages the reservation of the resources per Diffserv domain and is implemented at the edges of this domain. This protocol handles the dynamic reservation requests, that is their admission or rejection, and possibly based on the results of the edge-to-edge

domain per hop reservation (PHR). These dynamic reservation requests, shown as "ext. QoS request" in Figures 1 to 4, are generated externally to the Diffserv domain and various protocols might potentially be used to make these requests (RSVP, RSVP aggregation, etc.).

A PDR protocol component should always be able to interpret the resource request and map it into an appropriate DSCP to be used in the edge-to-edge domain. Depending on these external protocols or resource reservation schemes, different PDR protocols can be defined in order to comply with the above requirement. The PDR protocol thus is a link between the external resource reservation scheme and the edge-to-edge PHR.

A PDR protocol should be able to identify and specify any external request for establishment and maintenance of resources using a (possibly aggregated) flow definition, i.e., flow specification identifier (ID).

The flow specification ID is only used by the edge nodes to provide the per-domain reservation (PDR) functionality. Depending on the PDR type used, different flow IDs can be specified. For example, a flow specification ID can be a combination of source IP address, destination IP address and the DSCP field. The flow specification ID is used to identify a (possibly aggregated) state that will only be maintained in the edge nodes.

3.2. Per Domain Reservation (PDR) protocol features

Depending either on the external resource reservation scheme with which the Diffserv domain has to interwork or on the characteristics of the network, the RMD framework MAY specify that several PDRs could use one PHR.

For example, a core network that is applying RSVP aggregation for resource management will use a different PDR than the PDR that has to be used in a wireless access network that is interconnected to the same core network which is using RSVP/Intserv for resource management.

However, both Diffserv domains may use the same reservation-based PHR. For each of these PDRs, there MAY be certain specific functions defined. However, the RMD framework defines a common set of features that need to be realized by any PDR that uses a specific PHR, such as

the RODA PHR [RODA]. These features are described in the sections below. Besides this common set of features, there is also an optional feature described in Section 3.2.6.

3.2.1. Ingress node addressing

There are many situations, such as acknowledgement of a request, when the egress node has to notify the ingress node about the resource reservation status of the communication path between ingress and egress nodes using the PDR protocol, i.e., the request is admitted or is rejected.

This means that the egress node MUST be able to send a PDR signalling message to the ingress node. Depending on the PDR used and consequently also on the flow id specification (see Section 3.1), the IP address of the ingress node can be derived in two ways:

* The egress node can determine the IP address of the ingress node from the available information contained in the header of a received PHR signalling message. This could, for example, be the source IP address of the PHR signalling message received.

* The ingress node has to encapsulate its IP address in the PDR signalling message that is encapsulated in a PHR signalling message. The egress node decapsulating the PHR is able to extract the PDR signalling message and the IP address of the ingress node.

3.2.2. Error control

The PHR signalling messages may be dropped in the communication path from the ingress to the egress nodes.

If a reservation-based PHR is used, these messages might have been received by some of the intermediate interior nodes located in this communication path before being dropped. Some other interior nodes located on the same communication path might not receive these PHR signalling messages. This will mean that the interior nodes that received this PHR signalling message will reserve resources that will not be used.

Should this occur, the PDR protocol MUST be able to handle the

recovery of the dropped reservation-based and measurement-based PHR signalling messages. One possible solution to this is described in Section 5.3.1.

3.2.3. Management of Reservation States

The per-domain reservation functionality MUST support the initiation and maintenance of PDR states. This can be accomplished by using either a new defined PDR protocol or (one or more) already existing protocols. Examples of such existing protocols are the Resource Reservation Protocol (RSVP) [RFC2205], RSVP aggregation [RFC3175], Simple Network Management Protocol (SNMP) [RFC1905], Common Open Policy Service (COPS) [RFC2748]. These states will be identified using the flow specification ID (see Section 3.1) and the related requested resource unit per Diffserv class PHB.

The egress node MUST be able to identify the flow using the flow specification ID after receiving a PHR signalling message. Depending on the PDR protocol type being used, the flow specification ID can be derived in two ways:

* Derived from PHR message: the flow specification ID can be derived from the available information contained in the header of the PHR signalling message received. This could, for example, be the combination of the source and destination IP addresses and the DSCP in the PHR signalling message.

* Derived from PDR message: the flow specification ID is included in the PDR signalling message that is encapsulated by the ingress node into the PHR signalling message. The egress node decapsulating the PHR is able to extract the PDR signalling message and the flow specification ID information.

  Moreover, the PDR signalling message that is sent by the egress node towards the ingress node MUST also contain the flow specification ID information.

The PDR resource reservation states can be either hard or soft states. If these states are hard they will have to be initiated, updated or released explicitly. If these states are soft states then they have to be updated regularly. The PDR soft state can be released by using the refresh timeout or by explicit release of the reserved resources.

3.2.4. Resource Unavailability

When there are insufficient resources available in the communication path between the ingress node and egress node, the ingress node that generated the PHR signalling messages will have to be notified by means of a PDR reporting message. Any interior node that does not admit a reservation request will mark the PHR signalling message that will be sent towards the egress node. The egress node will in return generate and send to the ingress node a marked PDR signalling message to indicate that the communication path is not able to admit the reservation request. Upon receiving this message the PDR functionality in the ingress node will inform the external resource reservation scheme that its associated request for RMD resources is rejected.

When the reservation based PHR group is used, the marked PHR signalling message will also include the number of previous interior nodes that successfully reserved the resources for this PHR reservation signalling message (see [RODA]). This information will be sent to the ingress node as part of the PDR report message. The ingress node will initiate a partial explicit release procedure. This procedure will release the resources that were unnecessarily reserved by the interior nodes located on the same communication path as the interior node that rejected and marked the PHR reservation message (see Section 5.2.1).

Note that when the adaptation to load sharing procedure is applied, (see Section 4.2.3), the partial explicit release procedure should not be used. In this case the resources that were unnecessarily reserved by the interior nodes located on the same communication path as the interior node that rejected and marked the PHR reservation message will be released by using the reservation soft state principle.

3.2.5. Severe congestion handling

Severe congestion can be considered as an undesirable state which may occur as a result of a route change or a link failure. Typically, routing algorithms are able to adapt and change their routing decisions to reflect changes in the topology and traffic volume. In such situations the re-routed traffic will have to follow a new path. Nodes located on this new path may become overloaded, since they suddenly might need to support more traffic than their capacity. Moreover, when a link fails, the traffic passing through it might be

dropped, degrading its performance.

Severe congestion occurrence in the communication path has to be notified to the ingress node that generated the PHR signalling messages. Any interior node that detects the severe congestion will mark (severe congestion bit) the PHR signalling message that will be sent towards the egress node. The egress node will in return generate and send to the ingress node a marked PDR signalling message to indicate the severe congestion occurrence in the communication path. Upon receiving this message the ingress node resolves this situation by using a predefined policy, e.g., refusing new incoming flows and by using the PHR protocol, a portion of the affected by severe congestion flows either is terminated or is preempted (e.g., shifted to an alternative PHB).

When the reservation-based PHR is used, the (severe congestion) marked PHR reservation message will also include the number of previous interior nodes that successfully processed this PHR reservation message (see [RODA]). This information will be sent to the ingress node as part of the PDR report message. The ingress node will initiate a partial explicit release procedure. This procedure will release the resources that were unnecessarily reserved by the interior nodes located on the same communication path as the severe congested interior node.

Note that when the adaptation to load sharing procedure is applied (see Section 4.2.3), the partial explicit release procedure should not be used. In this case the resources that were unnecessarily reserved by the interior nodes located on the same communication path as the severe congested interior node will be released by using the reservation soft state principle.

3.2.6. Modification of a reservation state

The number of resources that were reserved for a certain flow can be modified by using this feature (see also Section 5.5). When the ingress node receives an external QoS request that is requesting a modification on the number of reserved resources then the following process can be realized. When the modification request requires an increase on the number of reserved resources, then the ingress node will have to subtract the old and already reserved number of resources from the number of resources included in the new modification request. The result of this subtraction should be introduced within a PHR request message as the requested resources

value. When the modification request requires a decrease on the number of reserved resources, then the ingress node will have to subtract the number of resources included in the new modification request from the old and already reserved number of resources. The result of this subtraction should be introduced in a PHR release message. Furthermore, if the PDR protocol maintains PDR reservation states then the number of resources that were reserved for a certain flow should also be replaced with the number of resources included in the modification request.

3.2.7. Bi-directional reservations

Bi-directional reservations are an optional feature and do not belong to the common set of features described in Sections 3.2.1 to 3.2.6. This feature is only relevant when using the RMD framework in specific kinds of networks.

One method for bi-directional reservations is based on combining two uni-directional reservations. This is because messages travelling from the reserving entity are likely to follow a different path than messages travelling towards it. The bi-directional reservation imposes a few requirements on the edge nodes, as described below:

* The edge nodes must be able to distinguish between a uni-directional and a bi-directional resource reservation PDR signalling message. This SHOULD be accomplished by using a flag in the header of the PDR signalling messages. Furthermore, these bi-directional packets MUST include the requested resource parameters for initiating a uni-directional reservation in the opposite direction (from the egress to the ingress). Note that the requested resource parameters used for bi-directional reservations are asymmetric, i.e., the value of the requested resources used in the direction from the ingress node towards the egress node could be different than the requested resources used in the direction from the egress node towards the ingress node.

* When an egress node receives a bi-directional reservation message, the egress node will have to construct a uni-directional PDR signalling message and a PHR signalling message that will be sent in the opposite direction. The source IP address of this PHR signalling message that is sent towards the ingress node will be the same as the

destination IP address of the PHR signalling message it received, while the destination IP address of this PHR signalling message will be the same as the source IP address of the PHR signalling message it received.

The ingress node that performs a bi-directional reservation, assuming that the above requirements are satisfied, will notify the egress node by means of the PDR signalling messages. On receiving this PDR signalling message, the egress node will initiate a uni-directional reverse PDR signalling message, which will take care of the reservation in the opposite direction.

4. The PHR protocols

4.1. Introduction

The Per Hop Reservation (PHR) protocols extend the PHB in Diffserv by adding resource reservation, thus enabling reservation of resources per Diffserv class PHB per hop in each node within a Diffserv domain.

The RMD Framework currently specifies two different PHR groups:

- The Reservation-Based PHR group

  In this PHR group, each node in the communication path from an ingress node to an egress node keeps only one reservation state per PHB.

  The reservation is done in terms of resource units, which may be based on a single parameter, such as bandwidth, or on more sophisticated parameters. These resources are requested dynamically per PHB (i.e., per DSCP) and reserved on demand on all nodes in the communication path from an ingress node to an egress node.

  Furthermore, this PHR group has to maintain a threshold for each PHB that specifies the maximum number of reservable resource units. This threshold could, for example, be statically configured.

  A reservation-based PHR protocol is described in detail in [RODA]. The RMD framework uses a combination of reservation soft state and explicit release principles.

- The Measurement-based Admission Control (MBAC) PHR group

  This PHR group is used to check the availability of resources before flows are admitted and without installing any reservation state. That is, measurements are done on the real average traffic (user) data load. The main advantage of this PHR group is that the PHR functionality that is executed at the edge and interior nodes will not have to maintain any reservation states. However, the measurement based PHR uses two states that do not have to be maintained by the PHR protocol. One state per PHB that stores the measured user traffic load associated to the PHB and another state per PHB that stores the maximum allowable traffic load per PHB.

Although this is all that is currently defined, new types of PHRs within a PHR group may be defined in the future, as might new PHR groups.

To the extent possible, traffic patterns SHOULD be configured in the nodes rather than signalled. The goal is to simplify the traffic parameter mapping at the interior nodes and keep complexity at the edges. This also simplifies the processing of on-demand requests. For example, some of the token bucket parameters such as token bucket peak rate and bucket size can be configured.

The negotiated parameter within the edge-to-edge Diffserv domain in the RMD framework is the number of the requested resource units. For example, the RODA PHR [RODA] specifies that this parameter is a simple "bandwidth" parameter and can have a maximum value of 2^16 = 65536 resource units. However, this unit may not necessarily be a simple bandwidth value. It might be defined in terms of any resource unit (e.g., effective bandwidth) to support statistical multiplexing at the message level.

A mapping MUST be performed between the type of the resource units requested by an external reservation protocol and the resource units understood by the RMD scheme.

4.2. Per Hop Reservation (PHR) protocol features

The required features for the two PHR groups (the reservation-based and measurement-based (MBAC)) are different for the two groups. These features are described in the sections below.

4.2.1. One reservation state per Diffserv class PHB

The reservation-based PHR installs and maintains one reservation state per PHB, i.e., per DSCP, in all the nodes located in the communication path from the ingress node up to the egress node. This state represents the number of currently reserved resource units that are signalled by the PHR protocol for the admitted incoming flows. Thus, the ingress node generates for each incoming flow a PHR signalling message, which signals only the resource units requested by this particular flow. These resource units if reserved are going to be added to the currently reserved resources per PHB and therefore they will become a part of the per PHB reservation state.

The per PHB reservation states can be created and maintained by combination of the reservation soft state and explicit release principles.

When the reservation soft state principle is used, a finite lifetime is set for the length of the reservation. These reservations are then maintained by sending periodic PHR refresh messages. The length of the refresh period MUST be the same throughout the Diffserv domain and SHOULD be configurable. If this reservation state does not receive a PHR refresh message within a refresh period, reserved resources associated to this PHR message will be automatically released.

The reserved resources for a particular flow can also be explicitly released from a PHB reservation state by means of PHR release message. The usage of explicit release enables the instantaneous release of the resources regardless of the length of the refresh period. This allows a longer refresh period, which will also reduce the number of periodic refresh messages.

Furthermore, each node has to maintain a threshold for each PHB that specifies the maximum number of reservable resource units that could for example, be statically configured.

This feature is specific only to the reservation-based PHR group.

4.2.2. Sender-initiated

In general, a resource reservation scheme can be sender-initiated or receiver-initiated. In a receiver-initiated scheme, such as the Resource reSerVation Protocol [RFC2205], the reservation of the

resources is initiated by the receiver. This means that backward routing information has to be stored in the nodes that are located in the forwarding path between the sender and receiver. This backward routing information will be used by the reservation messages sent by the receiver to the sender. All signalling messages belonging to the same flow will then follow the same backward and forward path.

In order to avoid storing backward routing information in the RMD framework, a sender-initiated scheme is used.

The ingress node will initiate and manage the resource reservation process, meaning that it will generate the PHR signalling messages. Each of these messages may carry either the total amount of the requested resources or a part of the requested resources.

Assuming that typical IP routing protocols are used, i.e., packets are routed based on IP destination address, all the PHR signalling messages that are generated by the edge nodes SHOULD use the IP addresses of the end hosts involved in the resource reservation session as the source and destination IP addresses. However, depending on the PDR used, exceptions should be allowed. For example, the PHR signalling messages may have the IP addresses of the edge nodes as the source and destination IP addresses. This will imply that the traffic (user) data associated with these PHR signalling messages must be encapsulated with the IP addresses of the edge nodes as the source and destination IP addresses.

Both PHR groups MUST be sender-initiated.

4.2.3. Adapts to Load Sharing

Load sharing, also known as load balancing, allows interior nodes to take advantage of multiple routes to the same destination by sending messages via some or all of these available routes. However, load sharing will imply that the traffic (user) data will not follow exactly the same paths as the PHR signalling messages that are used to reserve the transport resources used by the traffic (user) data.

Load sharing can be characterized as equal or unequal cost (see [Doy98]), where cost is specified as a generic term referring to any metric that is associated with the path. Equal cost load sharing (see, for example, [RFC2676]) distributes traffic equally among the multiple paths. Unequal cost load sharing, on the other hand, does not distribute the traffic equally.

An example of this type may be the optimized multi-path (OMP) that is able to distribute loading information, proposing a means for adjusting forwarding and providing an algorithm for making the adjustments gradually enough to ensure stability yet providing reasonably fast adjustment when needed. Note that "reasonably fast" means adaptation in a couple of hours, i.e., daily load fluctuations. OMP discovers multiple paths, not necessarily equal cost paths, to any destinations in the network, but based on the load reported from a particular path, it determines which fraction of the traffic to direct to the given path. Incoming packets are subject to a (source, destination address) hash computation, and effective load sharing is accomplished by means of adjusting the hash thresholds. When combining with multi-protocol label switching (MPLS) forwarding, OMP becomes an effective route optimization engine that can serve the requirements claimed on traffic engineering (TE) in [RFC2702].

Load sharing can be accomplished in different ways:

* Per-destination load sharing: distributes the traffic based on the destination address. All messages for one destination on the network travel on the same path.

* Per-message load sharing (or round robin): given equal cost paths, the first message destined for a particular destination on the network is sent via one path, the next message to the same destination is sent via another path, and so on.

* Using a predefined hash function: the combination of the source and destination IP addresses and the source and destination ports is used in a hash function to determine for each message which load sharing path should be used. In this situation, even if the various paths may have equivalent metrics, the traffic associated with one TCP connection is always routed on a single path.

The Resource Management in Diffserv framework, by means of PHR and PDR functionality, has the necessary support to adapt to load sharing when it is used. This feature is mandatory for both PHR groups. An example of this operation is described in Section 5.4.

4.2.4. Severe Congestion Detection and Notification

Severe congestion may occur as a result of route changes or a link failure. Severe congestion SHOULD always be signalled to the edges by the interior nodes regardless of the type of PHR used. The interior nodes report the severe congestion occurrence to the edges by means of PHR signalling messages. The edges MUST solve this severe congestion state as described in Section 3.2.5.

Severe congestion occurrence in the interior nodes has to be first detected and then the edges are notified. Due to the fact that the interior node does not maintain any flow related information, it is not possible to identify the ID of the passing flow and the IP address of the ingress node. Therefore, the interior node is not able to notify the ingress node that a severe congestion situation occurred.

4.2.4.1. Severe congestion Detection

The PHR functionality in the interior nodes detects the severe congestion and the PDR protocol informs the edge nodes about this severe congestion situation.

A number of possible methods of detecting severe congestion are listed below:

* Link failure: the interior node activates the severe congestion state whenever a link failure occurs.

* Volume measurements: by using measurements on the data traffic volume. If the volume of the data traffic increases suddenly, it is deduced that a possible route change and at the same time, a severe congestion situation occurred.

* Counting: using a counter that counts the number of dropped data packets. The severe congestion state is activated when this number is higher than a pre-defined threshold. This method is similar to the previous one but is much simpler. However, it can only be applied when the traffic characteristics are known.

* Increased number of refreshes: if the number of resource units, per PHB, requested by PHR refresh messages is much higher than the number of resources refreshed in

the previous refresh period, then the node deduces that a severe congestion occurred. This is a very efficient, but it can only be used when the PHR refresh period is small.

The first three detection methods can be applied on both types of PHR, i.e., reservation-based and measurement-based. The last method can only be applied on the reservation-based RMD scheme.

4.2.4.2. Severe Congestion Notification

Once detected the severe congestion should be signalled to the edges. As previously mentioned, the egress node will first be notified, after which the egress will notify the ingress node via the PDR protocol.

Below is a list of several notification methods that can be used:

* Greedy marking: all user data packets which pass through a severe congested interior node and are associated with a certain PHB will be remarked into a domain specific Diffserv code point (DSCP)

* Proportional marking: this method is similar to the previous method, with the difference that the number of the remarked packets is proportional to the detected overload

* PHR message marking: only PHR signalling messages that pass through a severely congested interior node will be marked. The marking is done by setting a special flag in the protocol message, i.e., "S" (see [RODA]). This is an efficient procedure, but it can only be used when the PHR refresh period is small.

The last method can only be applied on the reservation-based PHR, while the other two can be applied on both PHR types. A comparison between different severe congestion solutions is given in [CsTa02]. Furthermore, [CsTa02] demonstrates that there are severe congestion solutions that can efficiently solve the severe congestion situation.

The simple operation in case of a severe congestion is described in Section 5.3.2.

5. Examples of RMD Operation

The RMD framework extends the Diffserv architecture by adding dynamic resource reservations. It is applied edge-to-edge in a dynamically provisioned Diffserv domain. The admission or rejection of the incoming SLS request relies on the result of the PHR signalling protocol. The PDR protocol is the one that links the SLA/SLS request and the PHR protocol. Later in this document, the SLA/SLS request will be referred to simply as a QoS request.

The functional operation of the RMD framework is described as interoperation between the PHR and PDR functions, abstracted from the details in the following scenarios:

* normal operation

* fault handling:
  - loss of PHR signalling messages
  - severe congestion handling

There are two typical example scenarios used for describing the normal operation and fault handling of the RMD framework:

Example 1: PDR protocol will initiate and maintain the PDR states in the ingress/egress nodes. In this scenario it is assumed that the external QoS request does not create any resource reservation states in the ingress/egress nodes.

Example 2: PDR protocol will use (partially or fully) the resource reservation states initiated and maintained by an external protocol as PDR states.

The signalling message types are also explained briefly.

5.1. Examples of signalling Message Types

The RMD Framework classifies the signalling messages into PHR and PDR signalling messages for supporting PHR and PDR functionality, respectively.

5.1.1. PHR signalling message types

There are three types of PHR signalling messages.

5.1.1.1. PHR_Resource_Request

The "PHR_Resource_Request" signalling message is common to both PHR groups, but its role is different in the two PHR groups:

1. The reservation-based "PHR_Resource_Request" signalling message is generated by the ingress node in order to initiate or update the aggregated soft state reservation in the communication path to the egress node.

2. The measurement-based "PHR_Resource_Request" PHR signalling message is generated by the ingress node to check the monitoring status of each node located in the communication path between the ingress node and egress node.

5.1.1.2. PHR_Refresh_Update

The "PHR_Refresh_Update" signalling message is specific to reservation-based PHR group.

The "PHR_Refresh_Update" signalling message is generated by the ingress node in order to initiate, update or refresh the soft state reservation per DSCP in the communication path to egress node.

If possible, all the nodes should process the "PHR_Refresh_Update" messages with a higher priority than the "PHR_Resource_Request" messages.

5.1.1.3. PHR_Resource_Release

The "PHR_Resource_Release" signalling message is used only when the RMD framework supports PHR explicit release procedures.

The "PHR_Resource_Release" signalling message is generated by the ingress edge in order to release a part of, or all the reserved resources per DSCP. Furthermore, this message should specify the amount of resources that have to be explicitly released.

Note that in case that the bi-directional reservation is required, the egress router in addition to the normal processing, it will also respond to a bi-directional "PHR_Resource_Release" message with a unidirectional reservation "PHR_Resource_Release" message that is sent towards the ingress node. This uni-directional reservation message should be processed with a higher priority then other "PHR_Resource_Release" messages.

5.1.2. PDR signalling message types

The PDR signalling messages are processed only by the RMD edge nodes and not by the interior nodes. The PDR protocol can either be an entirely new protocol (see Example 1, Section 5.2.1.1) or it may use one of the existing protocols such as RSVP, RSVP aggregation, SNMP, COPS, etc. (see Example 2, Section 5.2.1.2) as part of its functionality. In order to describe the functionality of the PDR there are several messages denoted in this document, which are not formally specified protocol messages, but represent an exemplification of possible protocol messages used for exchanging the PDR information (such as flow id, address of the ingress) between edge nodes. These PDR signalling messages may also be encapsulated into PHR messages in case it is necessary.

These PDR signalling exemplification messages are listed below. If possible all the nodes should process the "PDR_Refresh_Report" messages with a higher priority than the "PDR_Reservation_Report" messages.

5.1.2.1. PDR_Reservation_Request

The "PDR_Reservation_Request" signalling message is generated by the ingress node in order to initiate or update the PDR state in the egress node.

5.1.2.2. PDR_Refresh_Request

The "PDR_Refresh_Request" message is sent by the ingress node to the egress node to refresh the PDR states located in the egress node.

Any of the "PDR_Reservation_Request" or "PDR_Refresh_Request" messages may either be or not be encapsulated into a PHR message. When any of these PDR messages is encapsulated into one PHR message,

then this PDR message SHOULD contain the information that is required by the egress node to associate the PHR signalling message that encapsulated this PDR message to for example the PDR flow ID and/or the IP address of the ingress node. sh 4 "PDR_Modification_Request"

The "PDR_Modification_Request" message is sent by the ingress node to the egress node to modify the PDR states located in the egress node.

5.1.2.3. PDR_Release_Request

The "PDR_Release_Request" messages are only used when the PDR state does not use a reservation soft state principle. These messages are sent by the ingress node to the egress node to release the flows explicitly.

5.1.2.4. PDR_Reservation_Report

The "PDR_Reservation_Report" messages are sent by the egress node to the ingress node to report that a "PHR_Resource_Request"/"PDR_Reservation_Request" has been received and that the request has been admitted or rejected. The same report message can be used to report that a "PHR_Resource_Request"/"PDR_Modification_Request" received and that the modification request has been admitted or rejected.

5.1.2.5. PDR_Refresh_Report

The "PDR_Refresh_Report" messages are sent by the egress node to the ingress node to report that a "PHR_Refresh_Update"/ "PDR_Refresh_Request" message has been received and has been processed.

5.1.2.6. PDR_Congestion_Report

The "PDR_Congestion_Report" messages are used for severe congestion notification and are sent by the egress node to the ingress node. These PDR report messages are only used when either the "greedy marking" or "proportional marking" severe congestion notification procedures, described in Section 4.2.4, are used.

5.1.2.7. PDR_Request_info

A "PDR_Request_info" message is encapsulated into a PHR signalling message that is sent by the ingress node towards the egress node. This PDR message is containing the information that is required by the egress node to associate the PHR signalling message that encapsulated this PDR message to for example the PDR flow ID and/or the IP address of the ingress node.

5.2. Example of Normal operation

Normal operation refers to the situation when no problems are occurring in the network, such as route or link failure, severe congestion, loss of PHR signalling messages, etc. Normal operation is different for the two PHR groups (the reservation-based PHR and the measurement-based PHR). Both are explained in the following sections.

5.2.1. Normal Operation using the reservation-based PHR

Depending on the functionality of the external resource reservation protocol that interoperates with the RMD domain two scenario types can be identified:

* Example 1, where the external resource reservation protocol does not create any reservation states in ingress/egress nodes.

* Example 2, where the external resource reservation protocol creates reservation states in ingress/egress nodes.

5.2.1.1. Example 1: No Reservation State in Ingress/Egress

In this scenario the external resource reservation protocol that interoperates with the RMD framework does not create any reservation states in ingress/egress nodes.

When a QoS request arrives at the ingress node, the PDR protocol must classify it into an appropriate Diffserv class PHB. It should calculate the associated resource unit for this QoS request, i.e., bandwidth parameter. The PDR state will be associated with a flow

specification ID. If the QoS request is satisfied locally, then the ingress node will generate the "PHR_Resource_Request" signalling message and the "PDR_Reservation_Request", which will be encapsulated in the "PHR_Resource_Request" signalling message. The PDR signalling message MAY contain information such as the IP address of the ingress node and the per-flow specification ID. The PDR_Request_Info message MUST be decapsulated and processed by the egress node only.

The intermediate interior nodes receiving the "PHR_Resource_Request" must identify the Diffserv class PHB (the DSCP type of the PHR signalling message) and, if possible, reserve the requested resources. The node reserves the requested resources by adding the requested amount to the total amount of reserved resources for that Diffserv class PHB.

The behavior of the egress node on admission or rejection of the "PHR_Resource_Request" is the same as in the interior nodes. After processing the "PHR_Resource_Request" message, the egress node decapsulates the "PDR_Reservation_Request" and creates/identifies the flow specification ID and the state associated with it. In order to report the successful reservation to the ingress node, the egress node will send the "PDR_Reservation_Report" message back to the ingress node. After receiving the "PDR_Reservation_Report" the ingress node will inform the external source of the successful reservation, which will in turn send traffic (user) data.

If the reserved resources need to be refreshed (updated), the ingress node will generate a "PDR_Refresh_Request" message in order to refresh the PDR soft state in the egress node. A "PHR_Refresh_Update" is used to refresh the PHR aggregated soft state in both interior and egress nodes. The "PDR_Refresh_Request" will be encapsulated into the "PHR_Refresh_Update". The PHR refresh periods should be equal in all edge and interior nodes.

Interior nodes that receive the "PHR_Refresh_Update" will refresh/update the aggregated reservation state related to the Diffserv class PHB (DSCP).

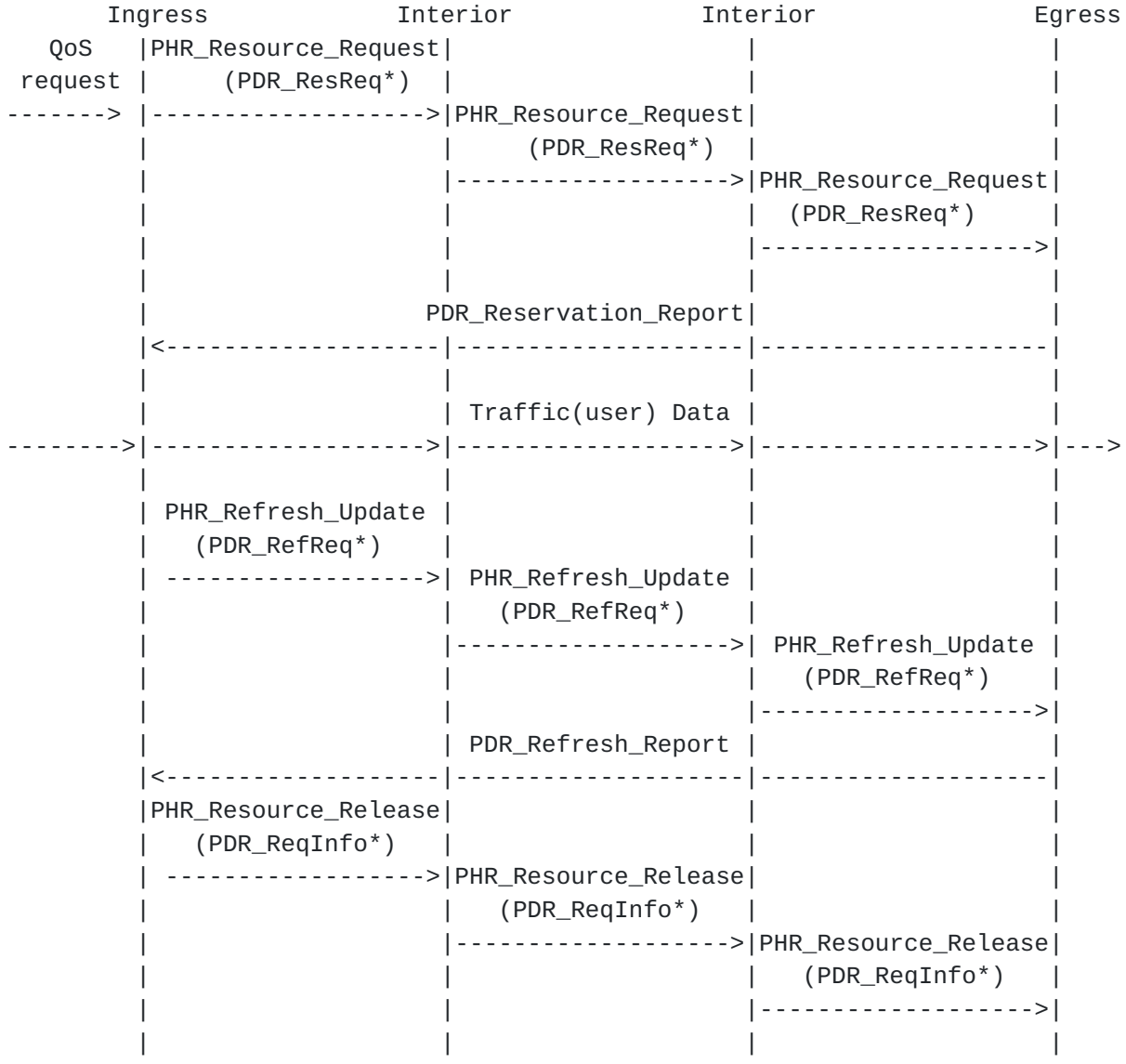
After processing the "PHR_Refresh_Update" message, the egress node MUST identify the flow specification ID carried by either the header of the PHR signalling message or the encapsulated PDR signalling message (see Section 5.1.2). In this way the PDR state associated with this flow specification ID can be refreshed instantaneously. The egress node will send the "PDR_Refresh_Report" signalling message back to ingress node to acknowledge the admission and processing of

the "PHR_Refresh_Update" signalling message.

In addition to the reservation soft state principle, the PHR resources in any node can also be released explicitly by means of explicit release signalling messages. In this case, the ingress node will create a "PHR_Release_Request" message, and it will include the amount of the PHR requested resources specified the PDR reservation state. This message will also encapsulate a "PDR_Request_Info" message. Note that in case the PDR reservation state does not use a reservation soft state principle the "PDR_Request_Info" message will represent a "PDR_Release_Request".

Any node that receives a "PHR_Resource_Release" signalling message must identify the DSCP and release the requested bandwidth associated with it. This can be achieved by subtracting the amount of PHR requested resources, included in the "PHR_Release_Request" message, from the total reserved amount of resources stored in the DSCP state.

The flow diagram showing the normal operation in case of successful reservation for Example 1 is shown in Figure 5.



(PDR_ResReq*) - represents the PDR_Reservation_Request message encapsulated in the PHR_Resource_Request message. This message is processed only by the ingress and egress nodes.

(PDR_RefReq*) - represents the PDR_Refresh_Request message encapsulated in the PHR_Refresh_Update message. This message is processed only by the ingress and

egress nodes.

(PDR_ReqInfo*) - represents the PDR_Request_Info message encapsulated into a PHR message message. This message is processed only by the ingress and egress nodes. Note that in case the PDR reservation state does not use a reservation soft state principle, this message will represent a PDR_Release_Request.

Figure 5: Normal Operation for successful reservation- Example 1

If there are no resources available locally, the ingress node will immediately reject the external QoS request and will not generate any signalling messages related to this request.

If the resources are lacking on the interior or egress of the network, these nodes MUST mark and forward the "PHR_Resource_Request" signalling message they receive in order to indicate the lack of the resources and that no reservation was made to the ingress node.

When a reservation-based PHR group is used, in addition to being marked, a "PHR_Resource_Request" message will also include the number of previous interior nodes that successfully processed this PHR message (see [RODA]). This number can, for example, be identified by the TTL (Time-To-Live) value included in the IP header of the received packet. Note that each time that an IP packet passes a node, its TTL value is decreased by one. Thus if the ingress node is able to initiate the TTL value included in the IP header of any PHR signalling message sent towards the egress node then any interior node will be able to find out how many nodes before it, processed this PHR message.

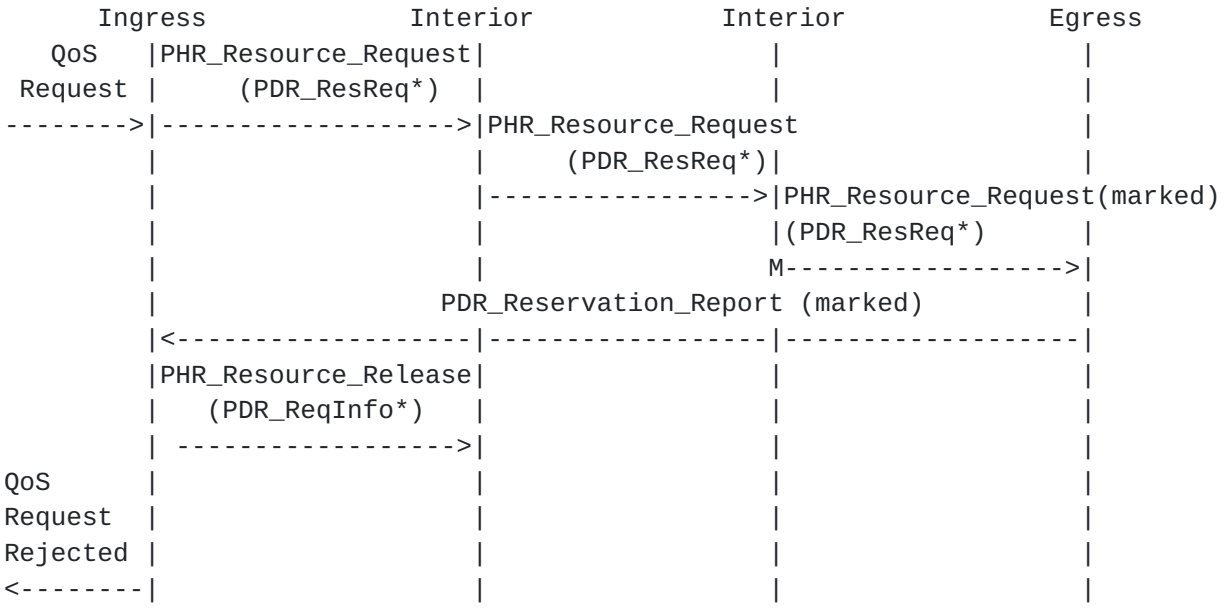
The interior node will copy the TTL value included in the IP header of the received "PHR_Resource_Request" message into the "PDR_Reservation_Request" message encapsulated within the "PHR_Resource_Request" message. For simplicity, we denote this variable as PDR_TTL. Moreover, the "T" field value of the "PHR_Refresh_Update" message is set to "1". This PHR message will be sent towards the egress node. Interior nodes receiving a marked "PHR_Resource_Request" message will not process it. Egress nodes receiving the marked "PHR_Resource_Request" MUST "M" mark the "PDR_Reservation_Report" message that is sent towards the ingress node. Moreover, if the "T" field value is "1" then the PDR_TTL value that was included by the interior node into the "PHR_Resource_Request" message will be copied into the "PDR_Reservation_Report" message.

After receiving the marked "PDR_Reservation_Report", the ingress node will reject the external QoS request. Simultaneously, the ingress node will start a partial explicit release procedure, for releasing the unnecessarily reserved resources in some interior nodes for the rejected flow.

In this case, the ingress node will generate a "PHR_Release_Request" message, and it will include the amount of the PHR requested resources specified the PDR reservation state. It will also insert the received PDR_TTL value in the TTL - IP header field of the "PHR_Resource_Release" message. Moreover, this message will encapsulate a "PDR_Request_Info" message.

Any node that receives a "PHR_Resource_Release" signalling message must identify the DSCP and release the requested bandwidth associated with it. This can be achieved by subtracting the amount of PHR requested resources, included in the "PHR_Release_Request" message, from the total reserved amount of resources stored in the DSCP state. Moreover, its TTL value is decremented by one. When this value becomes zero, the "PHR_Resource_Release" message reached the interior node that marked the "PHR_Resource_Request" message and it will be destroyed. This means that this message will not release any resources in this node.

Figure 6 depicts the normal operation for an unsuccessful reservation for Example 1.



(PDR_ResReq*) - represents the PDR_Reservation_Request message encapsulated in the PHR_Resource_Request message. This message is processed only by the ingress and egress nodes.

(PDR_ReqInfo*) - represents the PDR_Request_Info message encapsulated into a PHR message message. This message is processed only by the ingress and egress nodes. Note that in case the PDR reservation state does not use a reservation soft state principle, this message will represent a PDR_Release_Request.

Figure 6: Normal Operation for unsuccessful reservation - Example 1

5.2.1.2. Example 2

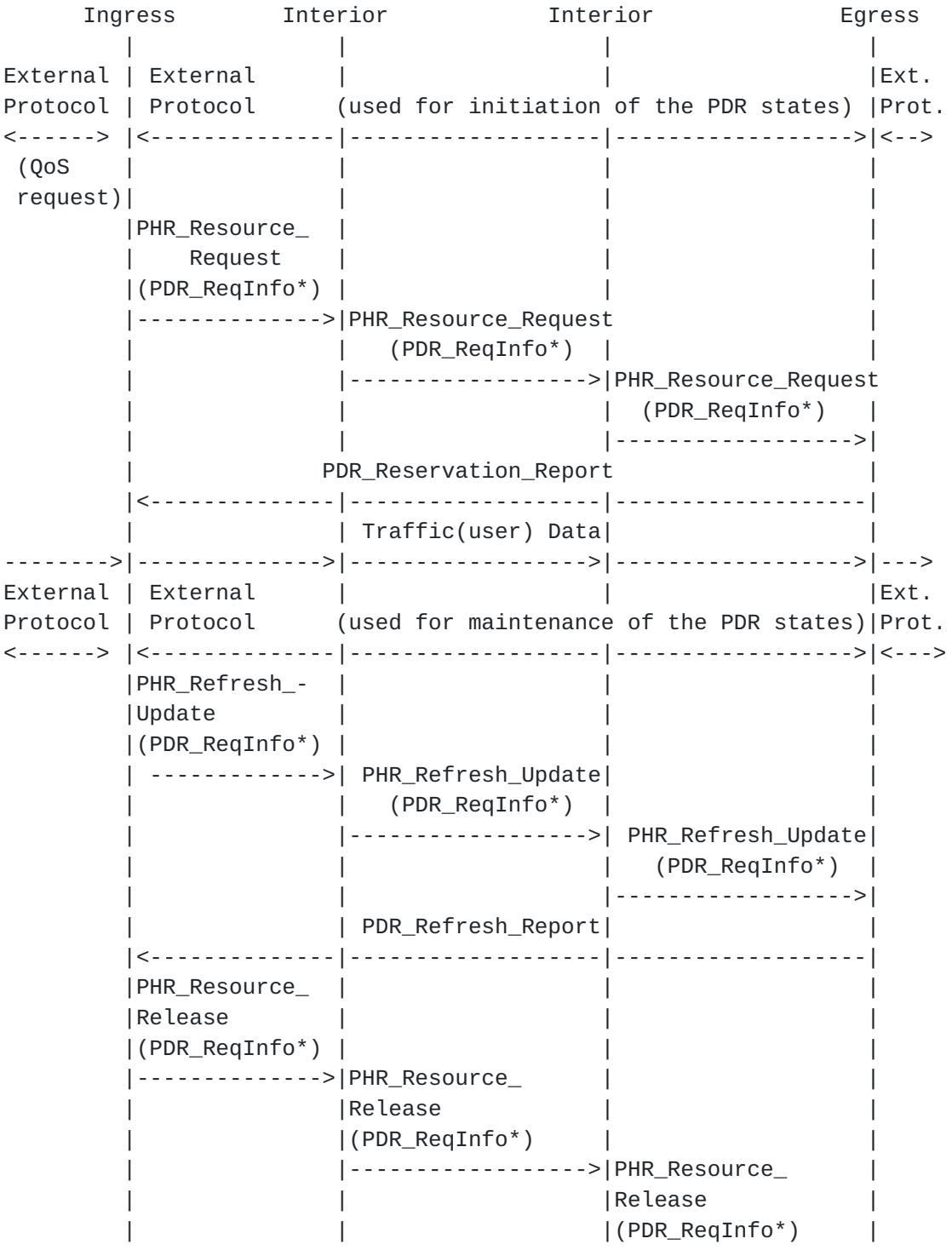
In this scenario the external resource reservation protocol that interoperates with the RMD domain creates reservation states in ingress/egress nodes that are used (partially or completely) by the RMD framework as PDR resource reservation states.

In this scenario as already mentioned an external protocol (such as RSVP, RSVP aggregation) initiates and maintains the states (per flow

or per aggregates) in the ingress and egress nodes. In the RMD framework these states (fully or partially) are to be used by the PDR handling the resource reservation in the Diffserv domain as PDR states, which will consist of for example a flow id and a DSCP.

Furthermore, in this scenario the "PHR_Resource_Request", "PHR_Refresh_Update" and "PHR_Release_Request" messages are encapsulating "PDR_Request_Info" messages that are used to associate the PHR signalling message that encapsulated this PDR message to for example the PDR flow ID and/or the IP address of the ingress node. Apart from this the rest of the functionality in generating and processing the PDR and PHR signalling messages by the edge and interior nodes is the same as in previous case (see Example 1).

Figure 7 shows the flow diagram for normal operation in case of successful reservation for Example 2.



(PDR_ReqInfo*) - represents the PDR_Request_Info
message encapsulated into a PHR message
message. This message is processed only by
the ingress and egress nodes.

Figure 7: Normal Operation for successful reservation - Example 2

When there are no resources available in ingress/egress nodes or interior nodes, the operation is similar to the one in Example 1, with the difference that the PDR resource reservation states are handled by the external protocol.

Figure 8 depicts the normal operation for an unsuccessful reservation for Example 2, where X represents the external protocol states related to the unsuccessful reservation, that need to be released in this particular case based on the soft state principle by the external protocol.

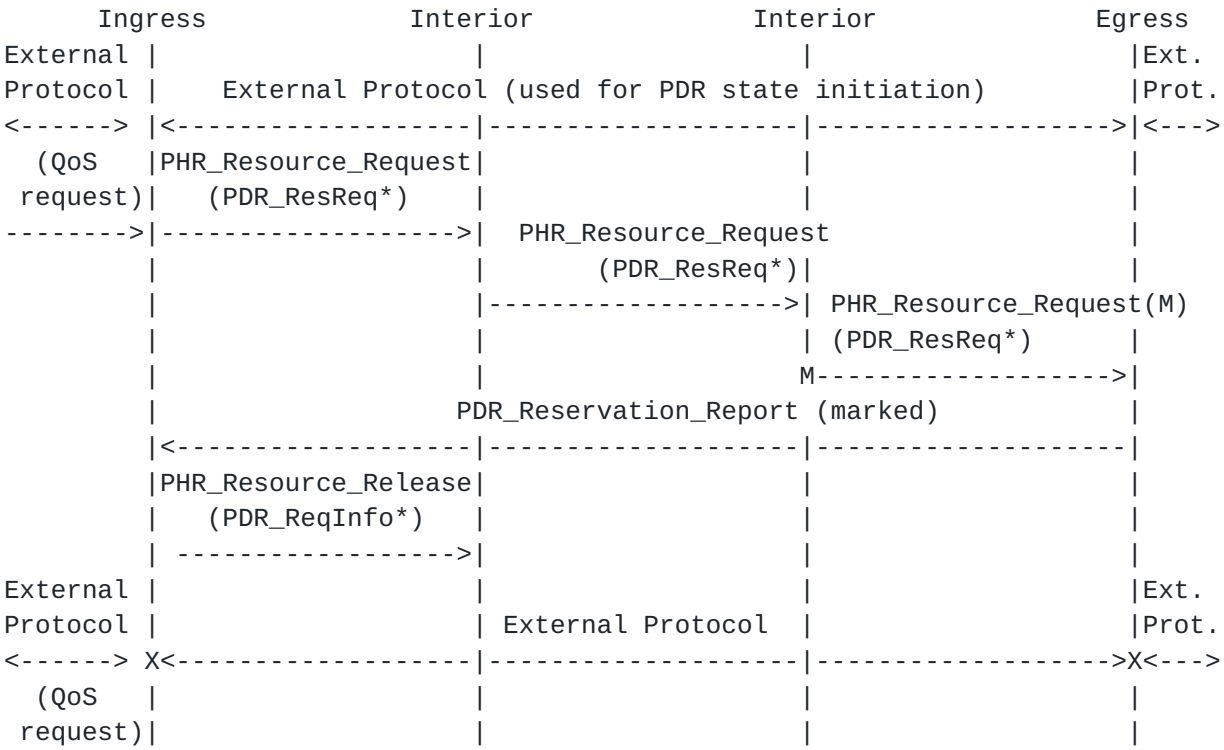


Figure 8: Normal Operation for unsuccessful reservation - Example 2

5.2.2. Normal operation using the measurement-based PHR

This RMD functionality is quite similar to that which uses the reservation-Based PHR. As with the reservation_based PHR, in this case both of the example scenarios are considered and the same differences between the two in the manner of handling the PDR states,

applies here as well. The classification of the QoS request is done as described in Sections 5.2.1.1 and 5.2.1.2 respectively.

The difference with the reservation-based is that the measurement-based PHR relies on a measurement algorithm on admission or rejection of the resource requests. As such, it does not have to maintain any resource reservation state per PHB in the edge or interior nodes. However, the measurement based PHR uses two states that are not maintained by the PHR protocol. One state per PHB that stores the measured user traffic load associated to that PHB and another state per PHB that stores the maximum allowable traffic load per PHB. However, the edges maintain a PDR resource reservation state (see Section 3.2.3).

The initiation and maintenance of the PDR resource reservation states is accomplished in an identical way as described in Section 5.2.1.1 and Section 5.2.1.2 respectively. If the QoS request can be satisfied locally, the ingress node will start the process of generating the "PHR_Resource_Request" message. In addition, depending on how the external resource reservation protocol initiates and maintains the PDR resource reservation states at the edges, the ingress node will also create either the "PHR_Resource_Request" message or the "PDR_Request_Info" message (see Section 5.2.1.1). On receiving the "PHR_Resource_Request" signalling message, the interior node has to check the monitoring status by, for example, measuring the real average traffic (user) data load per PHB. By "monitoring status", we specify how much of the resources allocated to a particular PHB have been consumed.

If the sum of the value of the PHR requested resources and the value specified by the monitoring status is less than or equal to the maximum node capacity associated with the given PHB, then the request is accepted.

Otherwise, the node does not have the requested amount of resources. Therefore, "PHR_Resource_Request" is marked as not admitted.

The behavior of the egress node on admission or rejection of the "PHR_Resource_Request" is the same as in the interior nodes. The reporting process used to inform the ingress node about the monitoring status is similar to the process explained in Section 5.2.1.1.

5.3. Example of Fault Handling Operation

Fault Handling Operation refers to the situations when there are problems in the network, such as loss of the PHR signalling messages, route change, link failure, etc. Two typical situations will be described: the loss of the PHR signalling messages and severe congestion. The fault handling operation described here is in general independent from the type of the example scenarios, thus it can be applied in both cases.

5.3.1. Loss of PHR signalling messages

The PHR signalling messages and subsequently the PDR signalling messages might be dropped, for example due to route or link failure. The loss of the PHR signalling messages is especially problematic for the reservation-based PHR since the dropped signalling messages might have reserved resources in some interior nodes in the communication path that will now not be used. This does not present a problem for the measurement-based PHR since there are no reservation states.

The ingress nodes are responsible for handling the loss of the PHR signalling messages. When sending a "PDR_Reservation_Request", a "PDR_Refresh_Request" or a "PDR_Request_Info" message as encapsulated in a PHR message, the ingress node will start a timer. The ingress node will then wait for a predefined amount of time to receive an acknowledgement, either as a "PDR_Reservation_Report" or "PDR_Refresh_Report" message. If the ingress node does not receive this acknowledgement within the predefined amount of time, it will conclude that an error has occurred. Moreover, it will also know that this error occurred during the resource reservation process for the flow session that is associated with the "PDR_Reservation_Request", "PDR_Refresh_Request" or "PDR_Request_Info" message it sent previously.

When a "PHR_Resource_Request" message is dropped, then the ingress node will not send any new PDR and PHR signalling messages associated with the same flow session during the first subsequent refresh period. In this way all the possible unused reserved resources will implicitly be released within one refresh period.

When a "PHR_Refresh_Update" message is dropped, the ingress node, depending on which PDR type was used, will send a PDR and "PHR_Refresh_Update" message during either the first or second subsequent refresh period. In the first case, one or more interior

nodes may reserve double the amount of the required resources, while only half of the amount of these reserved resources will be used. In the second case, the ingress node will not send any new PDR and "PHR_Refresh_Update" messages associated with the same flow session during the first subsequent refresh period. In this way all possible unused reserved resources will implicitly be released. However, the application may experience a possible QoS degradation during one refresh period.

In case a "PHR_Release_Request" message gets dropped the ingress node will rely on the reservation soft state principle for the release of the unnecessary reserved PHR resources.

5.3.2. Severe Congestion Handling operation

As explained in Section 4.2.4 above, severe congestion can be detected by any interior node by using different methods. Moreover, the severe congestion situation can be notified by any interior node to egress nodes by using three approaches, i.e., "Greedy Marking", "Proportional Marking" and "PHR message marking". The "PHR message marking" can only be applied on the reservation-based PHP, while the other two methods can be applied on both PHR types.

In this section the "PHR message marking" and "Proportional Marking" severe congestion notification methods are used.

5.3.2.1. PHR message marking

Using this severe congestion notification method, only PHR signalling message that pass through a severely congested interior node will be marked. If the severe congestion occurs in the interior or the egress node, then these nodes will set the "severe congestion" flag [RODA] in the PHR signalling message and will forward it to the egress node. The egress node will inform the ingress node by sending a PDR report message with the "severe congestion" flag set. After receiving this message, the ingress node will discard all new incoming requests for the severely congested path for a predefined time.

A flow diagram showing the severe congestion handling is depicted in Figures 9 and 10, where in (a) the severe congestion notification is performed by the "PHR_Resource_Request" and in (b) this notification is performed by the "PHR_Refresh_Update".

If the severe congestion notification is performed by the "PHR_Resource_Request" message (see Figure 9 (a)), after detecting the severe congestion, the "S" flag is set. The egress node will detect the marked "PHR Resource_Request" message and by "S" marking a "PDR_Reservation_Report" will inform the ingress node that a severe congestion situation occurred. The ingress node will then not admit any new QoS requests for that communication path.

If using the reservation-based PHR group, the "PHR_Resource_Request", besides being "S" marked, will include the number of previous interior nodes that successfully processed this PHR message (see [RODA]). This number is calculated and used by the PHR functionality in a similar way as in the "normal operation for unsuccessful reservations", i.e., using the TTL field. The interior node will copy the TTL value included in the IP header of the received "PHR_Resource_Request" message into the "PDR_Reservation_Request" message encapsulated within the "PHR_Resource_Request" message. For simplicity, we denote this variable as PDR_TTL. Moreover, the "T" field value of the "PHR_Refresh_Update" message is set to "1". This PHR message will be sent towards the egress node.

Interior nodes receiving a marked "PHR_Resource_Request" message will not process it.

Egress nodes receiving the "S" marked "PHR_Resource_Request" MUST mark (with the "S" bit) the "PDR_Reservation_Report" message that is sent towards the ingress node. Moreover, if the "T" filed included in the "PHR_Resource_Request" is "1" the egress node will include the received PDR_TTL value into the PDR_Reservation_Report.

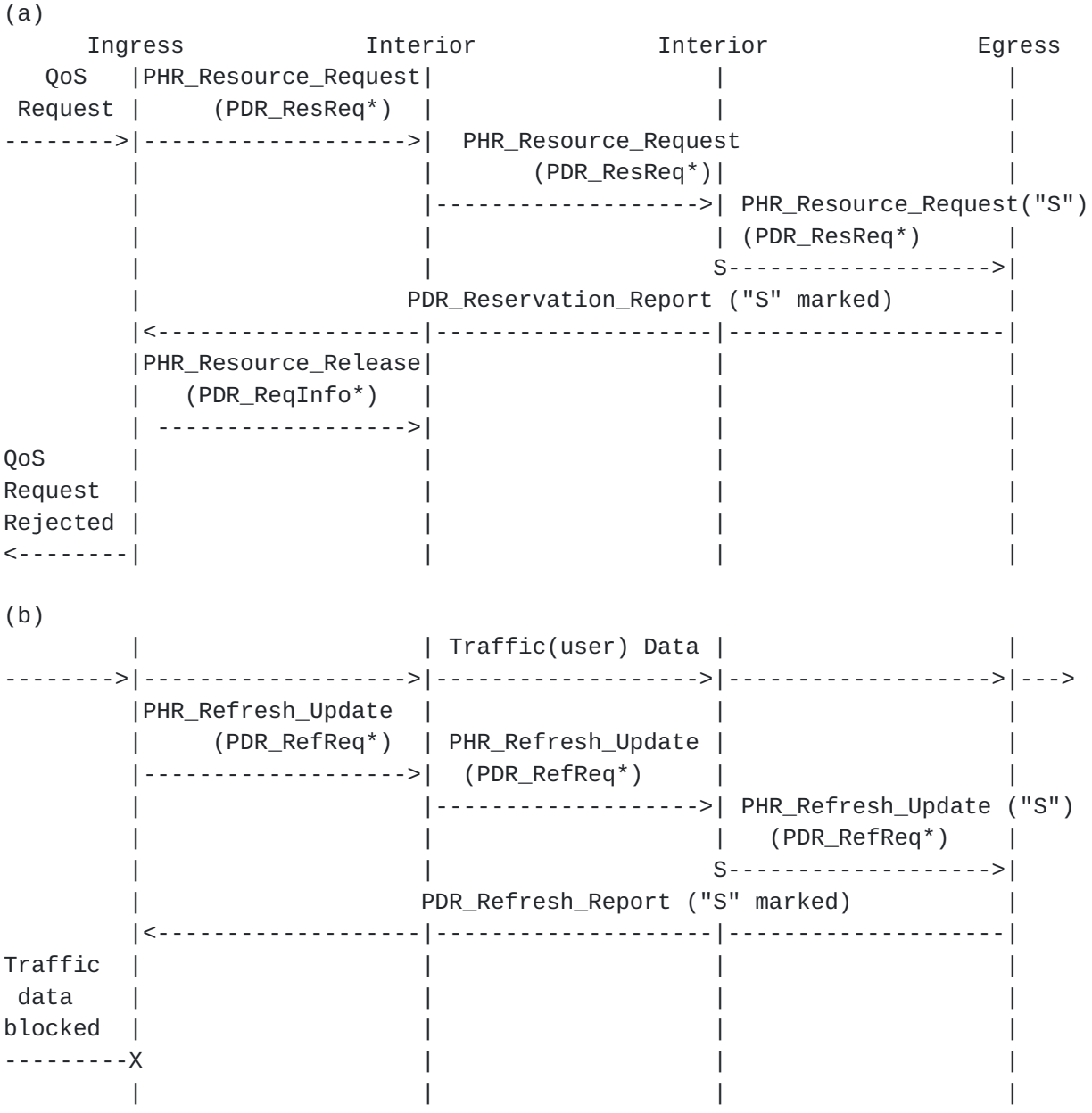
After receiving the "S" marked "PDR_Reservation_Report", the ingress node will not admit new QoS requests for that communication path. Simultaneously, the ingress node will start a partial explicit release procedure. It will generate a "PHR_Release_Request" message, and it will include the amount of the PHR requested resources specified in the PDR reservation state and insert the PDR_TTL received in the "PDR_Reservation_Report" message in the TTL - IP header field of the "PHR_Resource_Release" message. This message will also encapsulate a "PDR_Request_Info" message.

Any node that receives a "PHR_Resource_Release" signalling message must identify the DSCP and release the requested resources associated with it. This can be achieved by subtracting the amount of PHR requested resources, included in the "PHR_Release_Request" message, from the total reserved amount of resources stored in the PHB

reservation state. Moreover, its TTL value is decremented by one. When this value becomes zero, the "PHR_Resource_Release" message reached the interior node that marked the "PHR_Resource_Request" message and it will be destroyed. This means that this message will not release any resources in this node.

In case the severe congestion notification is performed by the "PHR_Refresh_Update" message (see Figure 9 (b)), the procedure is the same as when this is done by the "PHR_Resource_Request" message, but in this scenario the partial release will not be used.

Note that this separation is only for illustrative purposes. Figure 9 illustrates the scenario that is denoted in Section 5.2.1.1 as "Example 1" and Figure 10 illustrates the scenario that is denoted in Section 5.2.1.2 as "Example 2". The "PHR message marking" procedure is efficient, but it can only be used when the PHR refresh period is small.



(PDR_ResReq*) - represents the PDR_Reservation_Request message encapsulated in the PHR_Resource_Request message. This message is processed only by the ingress and egress nodes.

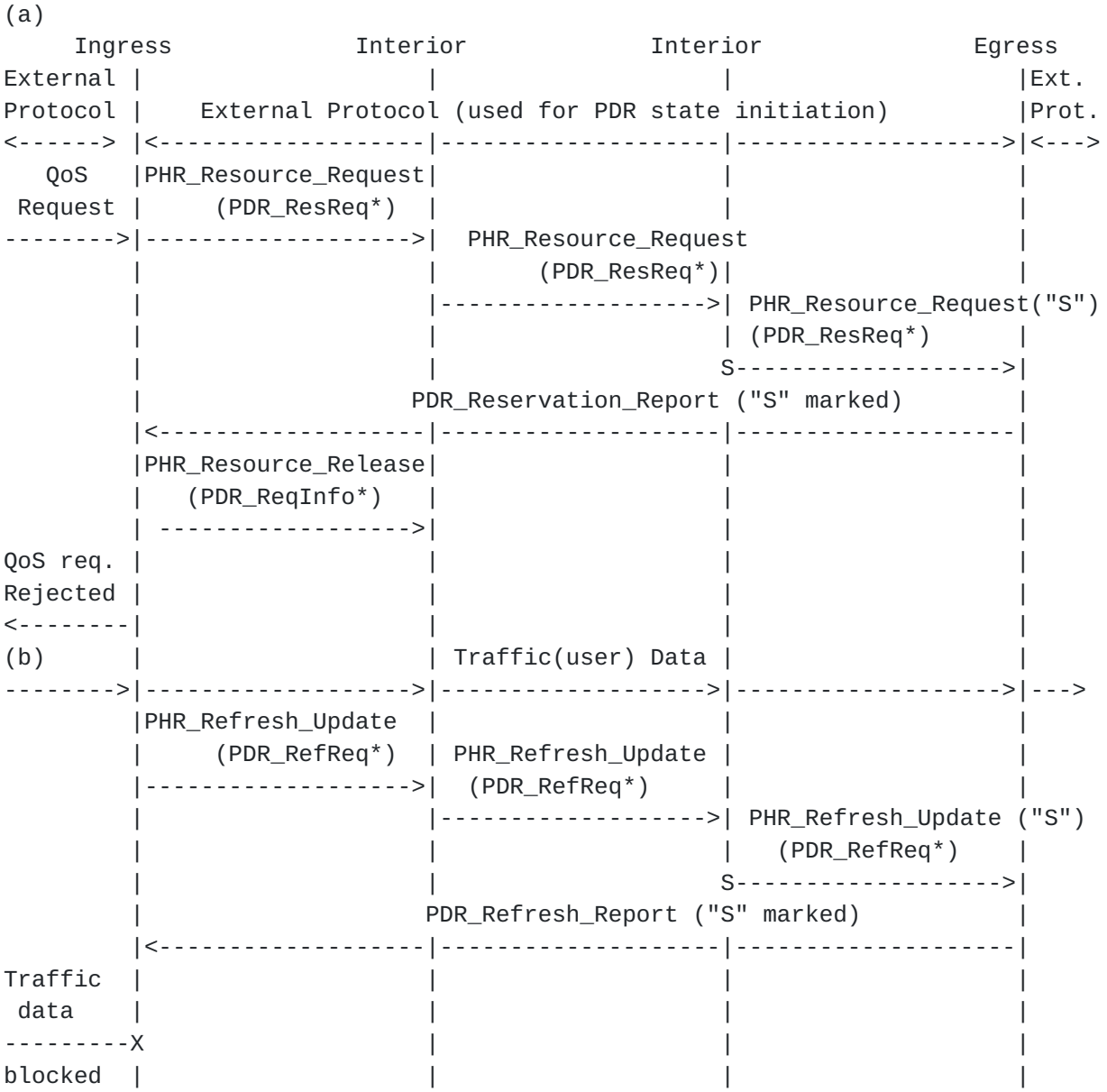
(PDR_RefReq*) - represents the PDR_Refresh_Request message encapsulated in the PHR_Refresh_Update message. This message is processed only by the ingress and egress nodes.

(PDR_ReqInfo*) - represents the PDR_Request_Info message encapsulated into a PHR message message. This message is processed

only by the ingress and egress nodes. Note that in case

the PDR reservation state does not use a reservation soft state principle, this message will represent a PDR_Release_Request.

Figure 9: Severe Congestion handling Operation applied to Example 1



(PDR_ReqInfo*) - represents the PDR_Request_Info message encapsulated into a PHR message message. This message is processed only by the ingress and egress nodes.

Figure 10: Severe Congestion handling Operation applied to Example 2

5.3.2.2. Proportional marking

Using this severe congestion notification method, after detecting the severe congestion situation, the interior node will notify the egress node by using remarking of user data packets that pass through the node. Proportionally to the detected overload the interior node will remark a number of user data packets which are passing through a severe congested interior node and are associated with a certain PHB, into a domain specific DSCP.

When the marked packets arrive at the egress node, the egress node will generate a "PDR_Congestion_Report" message and send it to the ingress node containing the over-allocation volume of the flow in question, e.g., a blocking probability. For each flow ID, the egress node will count the number of marked bytes (# marked bytes) and the number of unmarked bytes (# unmarked bytes).

Based on this information the egress node will have to calculate the blocking estimation of data. The egress node will actually calculate the blocking probability (Pdrop), which will be used by an ingress node to block this particular flow.

The blocking probability is calculated as the ratio between the dropped bytes and the maximum number of bytes that can be supported by the interior node:

Pdrop = (# marked bytes)/(# marked bytes + # unmarked bytes)

This blocking probability will be included in the "PDR_Congestion_Report" message that will be sent to the ingress.

The ingress node, based on this blocking probability, might terminate the flow, i.e., for a higher blocking probability there is a higher chance that the flow is terminated.

If a flow needs to be terminated, then for this flow, the ingress node will generate a "PHR_Release_Request" message.

The operation of this severe congestion solution is given in Figure 11.

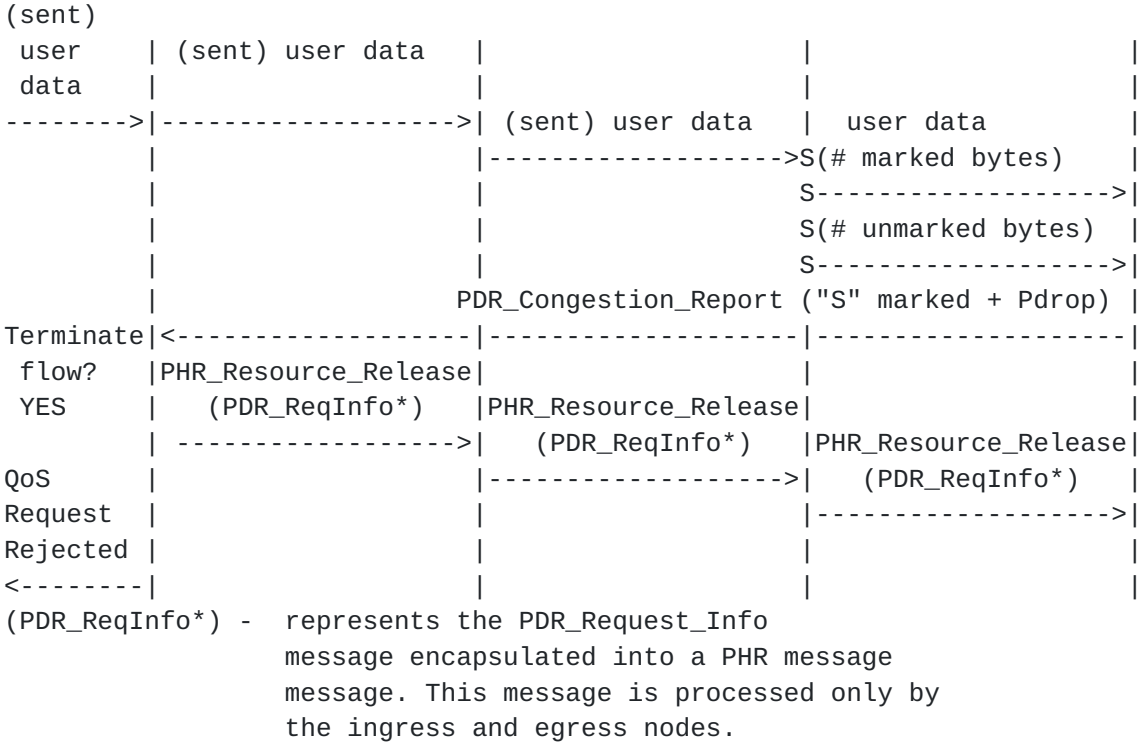


Figure 11: Severe Congestion handling Operation: with proportional marking

5.4. Example of Adaptation to equal cost path load sharing operation

Due to load sharing (see e.g., [RFC2676]), a node, influenced by the applied routing protocol, may cycle between different routes in order to balance the load. This will imply that the traffic (user) data will not follow exactly the same paths as the PHR messages used to reserve or refresh the transport resources used by this traffic (user) data. As such, interior nodes MUST be able to observe when a load sharing situation occurs.

In case a network domain is using a routing protocol which is applying an equal cost path load sharing principle, any interior node will be able to know the number, e.g., "N", of multiple equal cost paths that the routing protocol will use to provide the load sharing principle. Subsequently, for each arrived PHR message which is affected by the load sharing principle, the interior node according to [RODA] will be able to create "N" number of PHR messages of

identical type as the original one. Each of these generated PHR messages will contain in its "Requested Resources" field a value equal to the requested resources value which was included in the "Requested Resources" field of the original PHR message divided by the number of equal cost paths, i.e., "N". Moreover, each of these generated PHR messages SHOULD also contain in its "Shared %" field a new value that is calculated by dividing the shared percentage value, included in the "Shared %" field of the original PHR message, by the number of equal cost paths, i.e., "N".

When the egress node receives a "PHR_Resource_Request" or "PDR_Refresh_Update" message it must send a "PDR_Reservation_Report" or "PDR_Refresh_Report", respectively, back to the ingress node. In other words the egress node will have to copy the "M" and "S" fields from the PHR signalling message into a PDR report message. Furthermore, the PDR report messages have to include the PDR state information, i.e., flow specification ID and the IP address of the egress node. Moreover, the shared percentage value included in the received PHR message, i.e., the "Shared %" field (see [RODA]) is copied into the PDR report message. When the ingress node receives any PDR report message it must check if the shared percentage value included in the PDR report message is equal to 100. Note that any ingress node sets the "Shared %" value of any PHR message sent into the RMD domain, to 100.

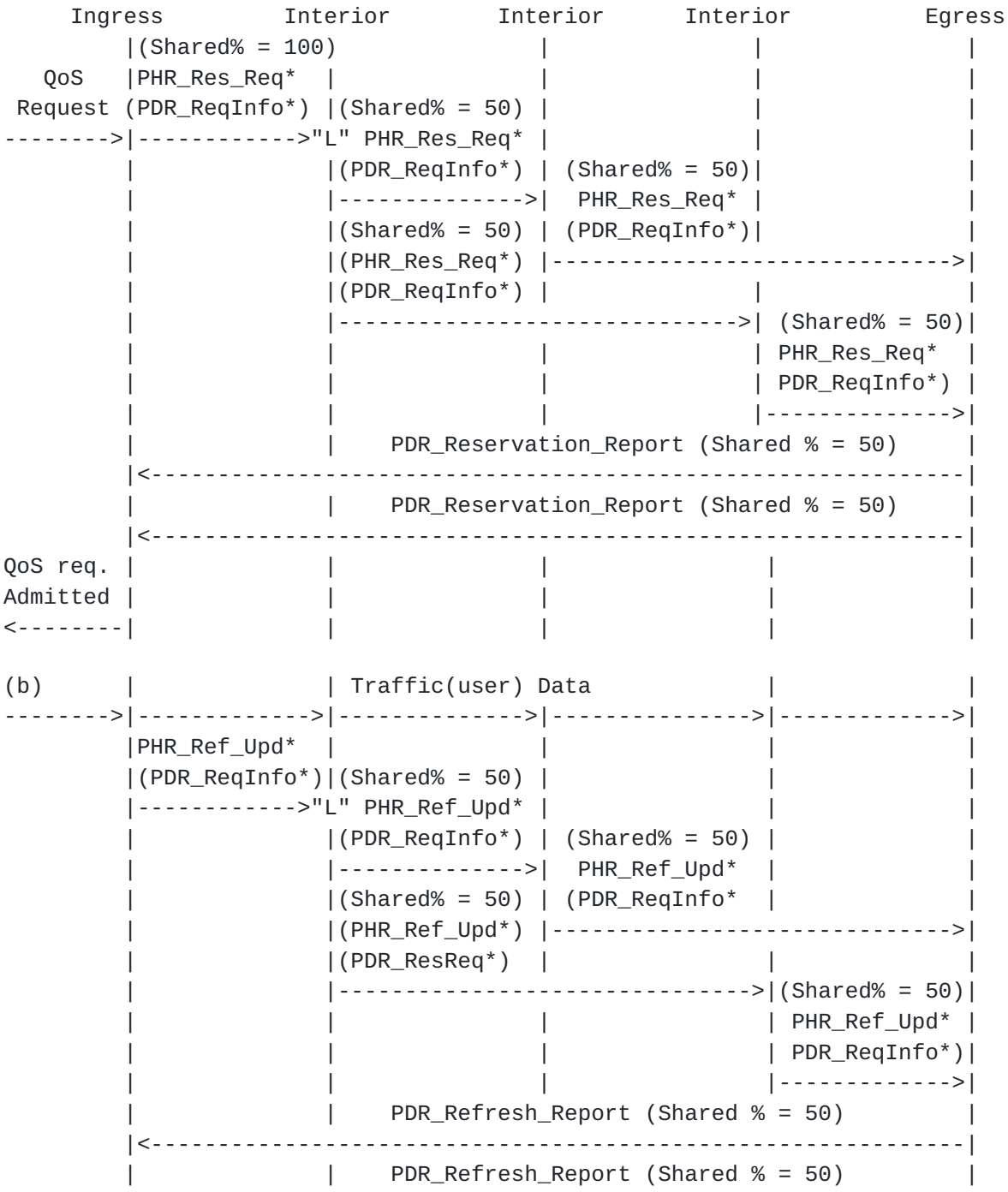
If that is the case then the ingress node will deduce that no load sharing took place state. If this values is not equal to 100, then the ingress node deduces that a load sharing in the communication path occurred. Moreover, the ingress node has to store the IP source address of the message, i.e., IP address of the egress node, and the shared percentage value included in the received PDR report messages. In this example we call the shared percentage value that was carried by the initial PHR request message as initial_shared_percentage.

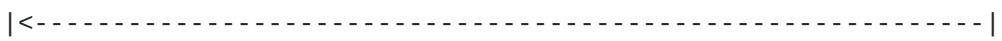
Each time that a new PDR report message associated to the same PDR state arrives, the ingress node must check the IP source address of this message with the IP address of the egress node that sent the previous PDR report message. If these two addresses are equal then the ingress node deduces that the same egress node sent the two PDR report messages. Otherwise, the ingress node will deduce that different egress nodes sent the PDR messages. Depending on the policy used and if the external QoS protocol does not adapt to load sharing then the ingress node may deduce that the reservation was unsuccessful. Otherwise, the ingress node must add the shared

percentage value of the previous received PDR report, i.e.,

initial_shared_percentage, with the shared percentage value included in the current received PDR report message. If the result of this addition is equal to 100, it means that the ingress node received all expected PDR report messages associated with this PDR state. Otherwise, more PDR report messages associated to the same PDR state, will have to arrive. The same procedure explained above is repeated until all the PDR report messages associated to the same PDR state are received.

The operation of this adaptation to load sharing operation for "PHR_Resource_Request" and "PHR_Refresh_Update" messages is given in Figure 12(a) and Figure 12(b), respectively.





"L" - the routing protocol detects two equal cost paths

PHR_Res_Req* - represents the PHR_Resource_Request message

PHR_Ref_Upd* - represents the PHR_Refresh_Update message

(PDR_ReqInfo*) - represents the PDR_Request_Info message encapsulated into a PHR message message. This message is processed only by the ingress and egress nodes.

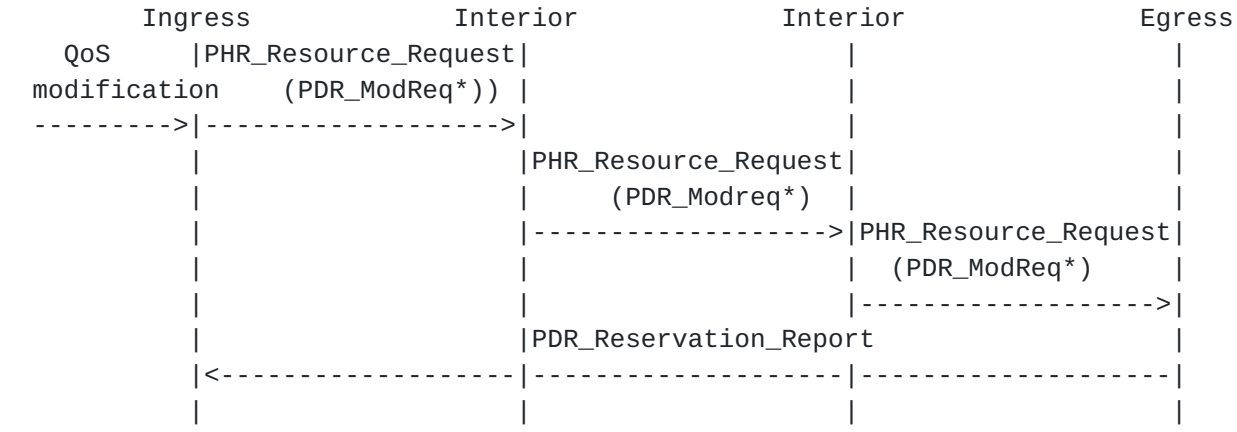
Figure 12: Adaptation to equal cost path load sharing operation

5.5. Example of modification of reservation state

This section presents an example that describes how the modification of a reservation state procedure can be specified.

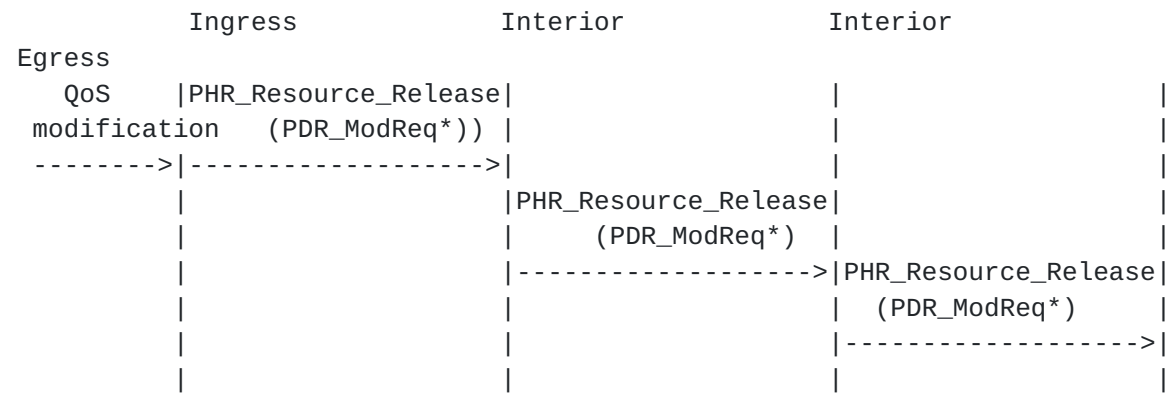
When the RMD functionality of the ingress node receives an external QoS request that is requesting a modification on the number of reserved resources then the following process can be realized. When the modification request requires an increase on the number of reserved resources (see Figure 13), then the RMD functionality of the ingress node will have to subtract the old and already reserved number of resources from the number of resources included in the new modification request. The result of this subtraction should be introduced within a PHR_Resource_Request message as the requested resources value. If a node is not able to reserve the number of requested resources, then the PHR_Resource_Request will be marked. In this situation the PHR and PDR protocol functionality associated with an unsuccessful reservation procedure will be applied for this case. When the modification request requires a decrease on the number of reserved resources (see Figure 14), then the ingress node will have to subtract the number of resources included in the new modification request from the old and already reserved number of resources. The result of this subtraction should be introduced in a PHR release message. Furthermore, if the PDR protocol part maintains PDR reservation states (e.g., Example 1) then the number of resources that were reserved for a certain flow should also be replaced with the number of resources included in the modification request. In this situation the above used PHR messages, i.e., PHR_Resource_Request and PHR_Resource_Release will have to encapsulate the PDR_Modification_Request message. If the PDR protocol does not maintain PDR reservation states (e.g., Example 2) then the above used PHR messages, i.e., PHR_Resource_Request and PHR_Resource_Release

will have to encapsulate the PDR_Request_Info message.



(PDR_ModReq*) - represents the PDR_Modification_Request message encapsulated into a PHR message message. This message is processed only by the ingress and egress nodes.

Figure 13: Modification of reserved resources when new number of resources is higher than number of already reserved resources



(PDR_ModReq*) - represents the PDR_Modification_Request message encapsulated into a PHR message message. This message is processed only by the ingress and egress nodes.

Figure 14: Modification of reserved resources when new number of

resources is
        lower than number of already reserved resources

6. Interoperability with external resource reservation schemes

The RMD framework is initially designed for a single edge-to-edge Diffserv domain. As part of the global Internet, this single edge-to-edge Diffserv domain will have to interoperate with other domains that may or may not be Diffserv-capable and which may use different resource reservation schemes. The RMD framework, which is specified as an open framework, MUST be able to interoperate with these external resource reservation schemes. That is, the PDR functionality will have to take care of interoperability between the external resource reservation schemes and the PHR protocol. The external resource reservation scheme could be applied either on an end-to-end or an edge-to-edge basis.

In order to describe this interoperability, the two most typical scenarios are chosen:

- Interoperability of the RMD framework with an RSVP/Intserv domain

  For a description of this interoperability, the Integrated Services over Differentiated Services framework [RFC2998] is used as a reference.

  The framework for Integrated Services (Intserv) operation over Differentiated Services (Diffserv) views the two architectures as complementary towards deploying end-to-end QoS. It is primarily intended to support the quantitative (guaranteed) end-to-end services that have not been commercially deployed yet by RSVP/Intserv due to the lack of scalability. The specific realization of the RSVP/Intserv - Diffserv interoperation depends on Diffserv resource management and on Diffserv network region RSVP awareness. Resource management in Diffserv can either be static (managed by human agents) or dynamic (via protocols). When the resource management in Diffserv is performed using the RSVP protocol, then according to the scenario described in Section 5.2.1.2, the RSVP protocol will have to be used as an external resource reservation protocol that will initiate and maintain the PDR resource reservation states used at the edges of the RMD domain. Independently of the Diffserv resource management, the service mapping of Intserv-defined services to Diffserv-defined services is essential for Intserv-over-Diffserv operation, unless Diffserv is used only as transmission medium. Service

mapping depends on appropriate selection of PHB, admission control and policy control on the Intserv request based on the available resources and policies in the Diffserv domain. In this framework, it is the edge nodes that will perform the service mapping on receiving of the RESV message.

- Dynamically Assigned Trunk Reservations

  In this case, the SLAs/SLSs between different Diffserv domains are negotiated in a dynamic way. In this scenario, RSVP aggregation [RFC3175] is used to signal QoS requests, that is, negotiate the SLAs/SLSs between Diffserv domains. Furthermore, the DSCP marking is performed in a domain outside the RMD domain, such as the neighboring Diffserv domain located upstream. When the RSVP aggregation protocol is used to dynamically assign the trunk reservations, then according to the scenario described in Section 5.2.1.2, the RSVP aggregation protocol will have to be used as an external resource reservation protocol that will initiate and maintain the PDR resource reservation states used at the edges of the RMD domain.

7. Applicability scope of the RMD framework

The RMD framework is designed to be applicable to core networks and any type of access networks, wired and wireless, as long as they are using the Diffserv architecture edge-to-edge.

As a particular example, the RMD framework applicability to wireless cellular access networks, that is, IP-based Radio Access Networks (RANs), is considered.

The specific characteristics of the RAN (see [PaKa01]) constrain the resource management strategies applied in the IP-based RAN with strict requirements, which are explained in [PaKa01] in detail. These requirements are not satisfactorily met by the current resource management strategies (see [PaKa01]). The RMD framework design on the other hand satisfies these specific resource management requirements, which gives the RMD framework an advantage over the current resource management strategies.

In order to fulfill the specific requirements related to resource management strategies applied in the IP-based RAN given in [PaKa01],

the PDR protocol in the RMD framework MUST be able to support the bi-directional reservations. This means that the PDR protocol MUST support the bi-directional feature described in Section 3.2.7. in addition to the mandatory ones given in Section 3.2.1 to 3.2.6.

8. Tunneling

When PHR/PDR signalling messages are tunneled within the RMD Diffserv domain, the tunneling messages MUST include the PHR/PDR option field.

9. Security Considerations

The general security and tunneling considerations stated in Section 6 of [RFC2475] apply also to this RMD framework.

In addition, unlike Differentiated Services PHBs, and PDBs, the RMD framework allows the edge nodes to reserve bandwidth or other QoS parameters dynamically. This flexibility makes it more vulnerable to erroneous reservations and sabotage. In order to keep functioning properly, the edge nodes MUST be certain that any flow reserving resources in the core network is allowed to do this and only up to that flow's agreed-upon limit. If the edge node detects erroneous or malicious behavior, it MUST police that flow to the agreed-upon limits or reject it entirely.

Because of the use of soft state, the RMD framework can recover relatively easily from incorrect reservations. Thus, it is quite safe to deploy the RMD framework in a well-controlled network with trustworthy edge nodes.

In order to prevent abuse of the QoS capabilities of the core network, the ingress nodes SHOULD filter any PHR or PDR related header information coming from the outside before sending it through the core network. Whether this information needs to be preserved and later re-inserted or if it should be discarded from the packet or if the entire packet should be discarded is an open issue.

10. Conclusions

The Resource Management in Diffserv (RMD) framework presented in this memo is an open framework, which by means of the PHR and PDR functionality provides a scalable and simple solution for resource

reservation in a single edge-to-edge Diffserv domain. Furthermore, the Resource Management in Diffserv framework provides the necessary functionality for interoperability with other external resource management strategies, which makes it a part of the effort to achieve end-to-end QoS deployment.

Also of particular importance is to note that the RMD framework can be applied on any IP network that has to support a huge real-time traffic (mixed with best effort traffic) volume which is generated by a huge number of users. Such networks can for example be next generation ISP backbone networks, and various wired and wireless access networks.

11. References

[CsTa02] Csaszar, A., Takacs, A., Szabo, R., Rexhepi, V., Karagiannis, G., "Severe Congestion Handling with Resource Management in Diffserv On Demand", submitted to Networking 2002, May 19-24 2002, Pisa - ITALY.

[Doy98] Doyle, J, "CCIE Professional Development: Routing TCP/IP", Volume 1, CISCO Press, 1998.

[NiKa01] Nichols, K., Carpenter, B., "Definition of Differentiated Services Per Domain Behaviors and Rules for their Specification", Internet Draft, Work in progress.

[RODA] Westberg, L., Karagiannis, G., Partain, D., Oosthoek, S., Jacobsson, M., Rexhepi, V., "Resource Management in Diffserv On DemAnd (RODA) PHR", Internet Draft, Work in progress.

[PaKa01] Partain, D., Karagiannis, G., Westberg, L., "Resource Reservation Issues in Cellular Access Networks", Internet Draft, Work in progress.

[RFC1633] Braden, R., Clark, D., Shenker, S., "Integrated Services in the Internet Architecture: An Overview", IETF RFC 1633, 1994.

[RFC1905] Case, J., McCloghrie, K., Rose, M. and S. Waldbusser, "Protocol Operations for Version 2 of the Simple

Network Management Protocol (SNMPv2)", RFC 1905, 1996.

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC2119, March 1997.

[RFC2205] Braden, R., Zhang, L., Berson, S., Herzog, A., Jamin, S., "Resource ReSerVation Protocol (RSVP) -- Version 1 Functional Specification", IETF RFC 2205, 1997.

[RFC2474] Nichols, K., Blake, S., Baker, F. and D. Black, "Definition of the Differentiated Services Field (DS Field) in the IPv4 and IPv6 Headers", RFC 2474, December 1998.

[RFC2475] Blake, S., Black, D., Carlson, M., Davies, E., Wang, Zh., Weiss, W., "An Architecture for Differentiated Services", IETF RFC 2475, 1998.

[RFC2543] Handley, M., Schulzrinne, H., Schooler, E., Rosenberg, J., "SIP: Session Initiation Protocol", IETF RFC 2543, 1999.

[RFC2597] Heinanen, J., Baker, F., Weiss, W., Wroclawski, J., "Assured Forwarding PHB group", IETF RFC 2597, 1999.

[RFC2598] Jacobson, V., Nichols, K., Poduri, K., "An Expedited Forwarding PHB", IETF RFC 2598, 1999.

[RFC2638] Nichols, K., Jacobson, V., Zhang, L., " A two-bit Differentiated Services Architecture for the Internet", IETF RFC 2638, 1999.

[RFC2676] Apostolopoulos, G., Willians, D., Kamat, S., Guerin, R., Orda, A., Przygienda, T., "QoS Routing Mechanisms and OSPF Extensions", IETF Experimental RFC 2676, August 1999.

[RFC2702] Awduche, D., Malcolm, J., Agogbua, J., O'Dell, M., McManus, J., "Requirements for Traffic Engineering Over MPLS", IETF Informational RFC 2702, Sept. 1999.

[RFC2748] Durham, D., Boyle, J., Cohen, R., Herzog, S., Raja, R., Sastry, A., "The COPS (Common Open Policy Service)

Protocol" IETF RFC 2748, January 2000.

[RFC2859] Fang, W., Seddigh, N., Nandy, B., "A Time Sliding Window Three Colour Marker (TSWTCM)", IETF Experimental RFC 2859, June 2000.

[RFC2998] Bernet, Y., Yavatkar, R., Ford, P., baker, F., Zhang, L., Speer, M., Braden, R., Davie, B., "Felstaine, E., "Framework for Integrated Services operation over Diffserv Networks", IETF RFC 2998, 2000.

[RFC3175] Baker, F., Iturralde, C. Le Faucher, F., Davie, B., "Aggregation of RSVP for IPv4 and IPv6 Reservations", IETF RFC 3175, 2001.

[WeTu00] Westberg. L., Turanyi Z. R., Partain, D., "Load Control of Real-Time Traffic", Internet Draft, Work in progress.

12. Acknowledgements

Special thanks to Geert Heijenk for reviewing this and providing useful input.

13. Authors' Addresses

Lars Westberg
Ericsson Research
Torshamnsgatan 23
SE-164 80 Stockholm
Sweden
EMail: Lars.Westberg@era.ericsson.se

Martin Jacobsson
Ericsson EuroLab Netherlands B.V.
Institutenweg 25
P.O.Box 645
7500 AP Enschede
The Netherlands
EMail: Martin.Jacobsson@eln.ericsson.se

Georgios Karagiannis
University of Twente

P.O. BOX 217
7500 AE Enschede
The Netherlands
EMail: karagian@cs.utwente.nl

Simon Oosthoek
Ericsson EuroLab Netherlands B.V.
Institutenweg 25
P.O.Box 645
7500 AP Enschede
The Netherlands
EMail: Simon.Oosthoek@eln.ericsson.se

David Partain
Ericsson Radio Systems AB
P.O. Box 1248
SE-581 12 Linkoping
Sweden
EMail: David.Partain@ericsson.com

Vlora Rexhepi
Ericsson EuroLab Netherlands B.V.
Institutenweg 25
P.O.Box 645
7500 AP Enschede
The Netherlands
EMail: Vlora.Rexhepi@eln.ericsson.se

Robert Szabo
Net Lab
Ericsson Hungary Ltd.
Laborc u. 1
H-1037 Budapest
Hungary
EMail: robert.szabo@eth.ericsson.se

Pontus Wallentin
Ericsson Radio Systems AB
P.O. Box 1248
SE-581 12 Linkoping
Sweden
EMail: Pontus.Wallentin@era.ericsson.se