# Diffserv Policies and Their Combinations in OpenView/JP1 PolicyXpert
(Extended abstract)

Yasusi Kanada
IP Network Research Center, Hitachi, Ltd.
Totsuka-ku Yoshida-cho 292, Yokohama 244-0817, Japan
kanada@crl.hitachi.co.jp

Brian J. O'Keefe
Hewlett-Packard Company
3404 East Harmony Road, Ft. Collins, CO 80528-9599, USA
brian_okeefe@hp.com

## 1. Introduction

In policy-based networks, two or more policies often work in cooperation. For example, in a QoS-assured network service such as Diffserv, packets from service subscribers are classified and policed at an edge router, and queued and scheduled in each router that the flow passes through. Thus, policies for classification, policing, and queuing/scheduling must cooperate to assure QoS. If the service is typical Diffserv, the policy for classification specifies the class or the DSCP of the flow, and the policy for queuing/scheduling specifies the testing of the DSCPs to determine the queuing and scheduling method required by packets in that class. These policies can be regarded as components of a network-wide QoS policy. Although DSCPs can be used for implicit cooperation of policies, some other services require explicit cooperation. The concept of policy combination was developed for an explicit specification of a *positive* relationship between policies [Kan 01]. The three types of Diffserv policies and the methods and application of policy combination in a policy server called OpenView PolicyXpert and JP1/PolicyXpert[1] [HP 00] are described here.

## 2. Diffserv Policies

A policy is a sequence of condition-action rules. The conditions are evaluated from left to right, and only the action which corresponds to the first condition to be matched is taken.

There are three policy types for Diffserv in PolicyXpert.

1. *Traffic Classifier (CL) Policy*: A CL Policy classifies packets and assigns virtual flow labels (See Section 3) called CIDs.
2. *Traffic Conditioner (TC) Policy*: A TC Policy polices, marks, and/or drops packets.
3. *Queue Control (QC) Policy*: A QC Policy queues and schedules, or drops packets randomly (e.g., by WRED).

CL and TC Policies are usually deployed to edge router interfaces, and a QC Policy is usually deployed to core router interfaces. TC and QC Policies are natural representation of Diffserv functions as policies; i.e., collections of condition-action rules.

## 3. Method of Policy Combination

To combine policies, both the dataflow and the control flow between policies must be specified. A specific dataflow of packets can be detected by using *flow labels* [Kan 01]. They are labels attached to a packet or flow. Flow labels are of two types.

1) *Real flow labels*: labels written inside the packet (e.g. DSCP).

2) *Virtual flow labels* (*VFLs*): labels external to the packet.

Here, we focus on the usage of VFLs.

Dataflows between policies are specified by defining and using VFL values in the rules of the policies. If Policy 2 is executed after Policy 1, a VFL can be defined by one or more of the rules (in actions) of Policy 1, and the VFL, i.e., the data defined in the rule, can be referred to in one or more of the rules (in conditions) of Policy 2.

VFLs in PolicyXpert are classified into three categories:

1. *Classifier ID* (*CID*): A CID combines rules in CL and TC policies.
2. *Traffic ID* (*TID*): A TID combines rules in a TC Policy.
3. *Queue Set ID* (*QID*): A QID combines rules in a QC Policy.

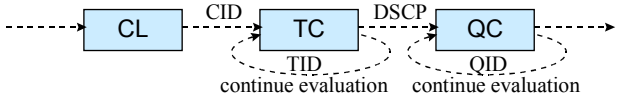
An example of a CID used in TC/QC Policy rules are shown:

TCrule: if (SourceIPAddress is 192.168.1.1) { CID = "G"; };

QCrule: if (CID is "G") { Priority = 6 /*high*/; … };

TIDs and QIDs only combine rules within a single TC/QC Policy because there is only one instance of a TC/QC Policy for an interface. DSCPs are used for combining a TC and QC Policy.

Flow of control can be explicitly specified by using a properly-defined policy language [Kan 00]. However, the order of policy evaluation can be predefined as part of the definition of a specific policy. The order of CL, TC and QC Policies is predefined as shown in the figure below. "Continue evaluation" must be explicitly specified in a rule when the same policy is evaluated next. If it is not specified in a QC Policy rule, the rule is not followed by another QC Policy rule.




## 4. Application of Policy Combination

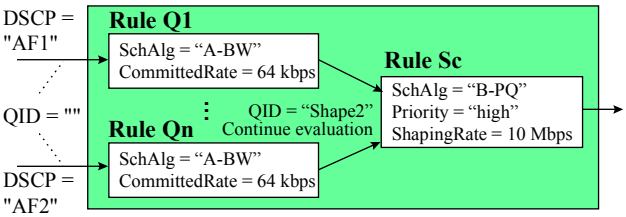
### 4.1 Separation of subscriber and service policies

Both network services and service subscribers, i.e., end customers, can be managed by using policies. Subscriber and service policies can be separated by using a policy combination. In a Diffserv network, gold, silver, and best-effort service classes, can be defined. The same DSCP can be used for both gold and silver classes, but the policing rates for them can be different. Then, two different DSCPs and three different CIDs (which represent sub-classes of DSCP-based classes) are used for gold, silver, and best-effort classes. Each class of services is specified by a TC Policy rule by a network administrator, and each service subscriber is specified by a CL Policy rule by network operators. CIDs are used for mapping subscribers into service classes. The policy rules that define these services are not modified when a subscriber is added or removed. This separation of subscriber and service policies clearly separates the task of the network administrator from the task of the network operator.

### 4.2 Hierarchical shapers and policers

In multi-service networks, hierarchical schedulers and shapers can be used for harmonizing various types of traffic. These functions can be represented by using QIDs and a QC Policy. Each QC Policy rule represents a simple queuing/scheduling method. So a simple method can be represented by a single rule. However, QC Policy rules can be combined by QIDs to represent a complex queuing/scheduling method.

For example, a hierarchical shaper can be outlined as shown in the figure below. Here, in rules Q1, …, Qn, the input traffic is assumed to have the QID value "", the output traffic has the QID value "Shape2", and "continue evaluation" is specified. Rule Sc inputs the aggregation of the shaped traffic from Q1, …, Qn, and outputs traffic at a maximum of 10 Mbps.




A hierarchical policer can be represented in a similar way to the above shaper, but the details are omitted here.

## 5. Conclusion

The policy combination enables the representation of complex Diffserv policies. TC and QC Policies, and TIDs and QIDs can be used in constructing such representations. Policy combination also allows sub-classing of DSCP-based service classes, and the separation of service and subscriber policies. The careful design of Diffserv policies has enabled simple Diffserv policies to be represented in a simple form.

## References

[HP 00] HP OpenView PolicyXpert 2.0 Users Guide, Edition 1, Hewlett-Packard, October 2000.

[Kan 01] Kanada, Y., "Taxonomy and Description of Policy Combination Methods", *Workshop on Policies for Distributed Systems and Networks* (*Policy 2001*), LNCS, No. 1995, pp. 171–184, Springer, January 2001.

[1] OpenView and PolicyXpert are trademarks of Hewlett-Packard Company. JP1 is a trademark of Hitachi, Ltd. PolicyXpert Version 2.0 was developed jointly by Hewlett-Packard and Hitachi.