INTERNET-DRAFT   Sanjeev Rampal,
Expires January 15th 1998   IBM Corpn.

                                                  Roch Guerin
                                                  IBM Corpn.
                                                  July 15th, 1997.

             Flow Grouping For Reducing Reservation Requirements for
                            Guaranteed Delay Service
                    <draft-rampal-flow-delay-service-01.txt>

Status of this Memo

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

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

   To learn the current status of any Internet-Draft, please check the
   ``1id-abstracts.txt'' listing contained in the Internet- Drafts
   Shadow Directories on ftp.is.co.za (Africa), nic.nordu.net (Europe),
   munnari.oz.au (Pacific Rim), ds.internic.net (US East Coast), or
   ftp.isi.edu (US West Coast).

Abstract

   The purpose of this document is to illustrate when and how flow
   aggregation can be of benefit in the context of the Guaranteed
   Service. Specifically, it identifies simple cases and provides
   generic rules for when grouping of flows allows a reduction in the
   amount of resources (bandwidth) needed to ensure the deterministic
   Guaranteed Service delay bounds of all flows. The benefits of
   grouping should be especially of interest to, say, Internet Service
   Providers, wishing to interconnect sites with Guaranteed Service
   flows. In particular, this document shows that in the case of flows
   with identical traffic characteristics and requirements, e.g.,
   Internet voice, grouping of flows is ALWAYS beneficial.

   This technique is not intended for IETF standardization and is
   being submitted for informational purposes only.

1.0 Introduction

This draft presents a technique for lowering the amount of bandwidth
required by guaranteed service flows [1].

The guaranteed service class provides a strict upper bound on
the end-to-end delay that will be experienced by any packet of a
flow belonging to this class (i.e. 100% of the packets will be
delivered with a network delay of no more than the specified
bound). This service requires each network element in the path
of a flow to reserve a certain bandwidth for this flow.
The provision of such deterministic guarantees does, however, come
at a cost as the amount of bandwidth (service rate) required to
ensure a given delay bound can be high, and in some instances even
larger than the flow's peak rate. As a result, techniques that
offer the opportunity for lowering the required resources while
providing the same guarantees are clearly desirable.

This draft describes one set of such techniques, where reduction in
the amount of bandwidth needed to guarantee the delay bounds of a
given set of flows is achieved by grouping those flows. In other
words, the draft shows when the amount of
bandwidth needed to be reserved for a group is less than the sum
of the reservations that would be required for each flow
individually.

The simplest and potentially most useful case where such a grouping
is beneficial is that of a set of identical flows, i.e., same
traffic characteristics and delay requirements, for which the
bandwidth required by the grouped flows is ALWAYS less than the
sum of the bandwidth needed by individual flows. Such a scenario
could be reasonably common in the case of an Internet Service
Provider offering Internet telephony service between customer sites.
In such a case, the application of grouping as described in this draft
would result in lower bandwidth requirements to support a given
number of voice calls.

In the rest of this draft, we first review the criteria used when
determining candidate flows for grouping, next we provide a simple
example that illustrates both cases when grouping is beneficial
and when it is not, and finally we outline a simple of rules for
determining whether grouping should be attempted or not.

2.0 Outline of Flow Grouping

The first requirement to identify guaranteed service flows that are
potential candidate for grouping is that they all use the same path in
the network. In other words, the same set of flows should

be present on all the links where grouping is performed.
This constraint could be imposed on only a subset
of the end-to-end path, e.g., grouping is only performed on the links
between the ingress and egress routers when crossing a routing domain.
However, this adds complexity to the determination of whether it is
beneficial to group the flows or not.

As a result, in this document we only address the case where flows
are group on their entire path. The reason for choosing only flows
which use the same path is that the delay bound calculated by the
Guaranteed Service is an end-to-end value. If we try to group flows
whose paths share some links initially but subsequently diverge
for example, we will need to characterize the portion of the
end-to-end delay that can be assigned to the common portion of the
path. As mentioned above, this adds complexity and we do not address
this problem in this document. The main contribution of the draft
is to identify specific grouping criteria and guidelines, so that
grouping always results in lowering the required bandwidth allocation.

In particular, we show that by grouping some (but not all) of such
flows appropriately and assigning a bandwidth reservation to the
entire group instead of to the individual flows, we can reserve a
lower amount of bandwidth for the group than the total that would be
required if this grouping were not to be performed. Further we can
still satisfy the delay bounds that are provided for each flow
when such a grouping is not performed. For a given set of flows,
a simple test procedure is outlined using which the network
determines whether the reservation with grouping will be less than
without grouping. Accordingly flows will be grouped only if there
is some bandwidth savings as compared to assigning a separate
reservation for each flow as recommended by the specification [1].

2.1 Flow grouping technique

The achievable improvement in utilization using flow grouping is
first illustrated through some examples. Next a procedure for
determining when grouping should be performed is outlined.

Consider two flows(F1 and F2) using the guaranteed service class in
a network as defined in [1] and [2]. These two flows are between the
same source and receiver nodes and are routed over the exact
same set of links (which we denote as L1, L2 ... Ln where n is the
number of physical links/hops in the path). Let P denote the maximum
packet size allowed for any connection in the network and let Cj
denote the bandwidth of link Lj (j=1,2 ... n).

Let the traffic characterization of these two flows be given as
(b1, r1, P1) and (b2, r2, P2) respectively where the b term
is the bucket size and the r term is the token generation rate of
the leaky bucket characterizing the respective flows, while the P
terms are the maximum packet sizes. Let D1 and D2 be the maximum
tolerable delay for the respective flows as specified by the
application.

Consider a network in which separate rate reservations of R1 and
R2 have been assigned to these two flows (we assume that
the reservation for a flow is greater than its token generation rate
which is a necessary condition for guaranteed service). The total
amount of bandwidth which needs be reserved on each link in the
path is hence R1 + R2.

In the guaranteed service formulation, the delay bound is inversely
related to the amount of the reservation. Hence to minimize the
amount of reservation for this case, the delay guarantee provided by
the network must equal that required by the flow (since any lower
reservation will increase the delay bound provided by the network
beyond the maximum tolerable limit specified by the application).

Hence we have,

$$D_1 = (b_1/R_1) + (n-1) * (P_1/ R_1) + (\text{sum from } j=1 \text{ to } n) P/C_j \qquad (1)$$

$$D_2 = (b_2/R_2) + (n-1) * (P_2/ R_2) + (\text{sum from } j=1 \text{ to } n) P/C_j \qquad (2)$$

The quantities on the right hand sides of equations (1) and (2)
are the formulation of the end-to-end queueing delay bound as
specified in [4]. The formulation in [1] is slightly different but
is derived from the same. We restrict ourselves to the strict WFQ
type formulation from [4] in this draft. The actual delay bound would
equal the queueing delay bound plus the propagation delay. Given the
route of a flow, the propagation delay component is fixed and the
same for all flows using the same route. Hence without any loss
of generality, we restrict ourselves to queueing delay bounds in
this draft.

Now consider a flow created by grouping these two flows into one
flow group which is characterized by the traffic model
(b_g, r_g, P_g) (the bucket size, token generation rate and maximum
packet size repectively of the resultant flow after grouping). Let
this flow group be assigned a reservation of R_g. Physically, packets
from both flows are simply multiplexed first come first served into
one flow. Packets which arrive at the same instant are reordered
arbitrarily. If the scheduler uses separate queues for different
flows for example, then we simply send packets from all flows in a
flow group to the same queue.

Consider a set of traffic characterization parameters for the flow
group which are obtained as follows.

$$b_g = b_1 + b_2 \qquad\qquad (3)$$

$$r_g = r_1 + r_2 \qquad\qquad (4)$$

$$P_g = \max\{P_1, P_2\} \qquad\qquad (5)$$

In the appendix, we show that these values are sufficient to
characterize the traffic of the flow group (which is the aggregate
traffic consisting of packets from both flows). In other words, if
the individual flows are deemed conforming by a leaky bucket derived

from the individual traffic models, then the traffic of the flow
group will also be deemed conforming by a leaky bucket based on
traffic values for the flow group calculated as above.

The minimum value of the delay bound that can be guaranteed by
the network to this combined flow can be calculated given this
reservation. We note that if the worst case delay bound guaranteed
to the flow group is no more than the minimum of all delay bounds
guaranteed to each individual flow, then the delay seen by any
packet in the flow group is guaranteed to be less than the delay
bound of the group and hence less than the delay bound guaranteed
to its own flow.

The table below shows the minimum delay bound for the individual
and aggregate flows when $R_g = R_1 + R_2$ for two example values of
the traffic parameters (n=2 and $P/C_j = 1$, j=1,2 are assumed).

In the first case, the minimum delay bound that can be guaranteed
for the combined flow with $R_g = R_1 + R_2$ (denoted
as $D_{(1,2)}$ is less than each of the bounds (D1 and D2) that can be
guaranteed for the flows individually, using separate reservations.
Hence a lower overall reservation (R') can be obtained
when the flows are grouped. As shown, this results in a reduction of
25 percent in the amount of bandwidth which needs to be reserved on
each link in the path of these flows. On the other hand, in the
second case, with $R_g = R_1 + R_2$, the minimum delay bound that
can be guaranteed for the combined flow is greater than that which
can be guaranteed for flow F1 with separate reservations. In order
for the combined flow to obtain a delay bound satisfactory for
both flows, a reservation of greater than R1 + R2 would be required.
Hence there is no gain in grouping for this case and better
utilization would be obtained by keeping the two flows separate and
maintaining individual reservations.

Table 1: Examples of gain (or loss) using flow grouping

(b1, r1, P1)	(R1, D1)	(b2, r2, P2)	(R2, D2)	D_(1,2)	R'	Percent gain
-----------	-------	-----------	-------	------	-	-------
(10, .5, 10)	(1, 22)	(10, .5, 10)	(1, 22)	17	1.5	25
(10, .5, 10)	(1, 22)	(10, .5, 100)	(1, 112)	62	-	Loss

----
n=2, P/Cj = 1, j=1,2

percentage gain is defined as 100 * (R1 + R2 - R')/(R1 + R2)

2.2. A procedure for testing whether grouping should be performed.

As shown above, there may or may not be some gain in grouping

two flows together. The procedure to test whether two flows can
be grouped together is straightforward. The total reservations
required with and without grouping are compared. Grouping can be
performed only if it results in lower total reservation than the
sum of the reservations with no grouping.

This test should be performed everytime a new flow using guaranteed
service enters the network or an existing flow leaves the network
or when the traffic characteristics of an existing flow change.
At such a time the existing sets of flow grouping can possibly be
changed to come up with another grouping which has a lower
total reservation requirement. However this is also a policy
decision which is implementation dependent.

Searching exhaustively among all possible groupings of a set of
flows for one that results in the absolute minimal total reservation
is likely to be computationally intensive and intractable. A
specific implementation may decide to tradeoff achieving the best
possible overall reservation with limiting the scope of the search
for an optimal combination of flows. Again, this draft does not
detail the different such implementation techniques for performance
control. A simple technique is outlined below for checking whether
two flows should be aggregated into a single flow for
the purpose of reservation reduction.

Given the two flows F1 and F2, and their traffic parameters and
delay requirements as above, the procedure for testing whether
they should be combined into a group is as follows.
Typically, F1 would be a new incoming flow or an existing
flow whose traffic characteristics have changed, while F2 represents
an existing aggregated flow to which we are considering adding the flow
F1 also.

1. Calculate the total reservation required with grouping using
equation (6) (which simply inverts the standard delay bound
equation which has the form like equations (1) and (2)).

$$Rg = \max \{r\_g, (b\_g+(n-1)* P\_g)/(D\_g - (\text{sum } j=1 \text{ to } N) P/Cj)\} \qquad (6)$$

where r_g, b_g and P_g are calculated according to equations (3)
through (5).

$$D\_g = \min(D1, \quad D2) \qquad (7)$$

2. Now calculate the total reservation required when grouping is not
performed. This is given by equation (8) below.

$$R\_tot = R1 + R2 \qquad (8)$$

where

$$R1 = \max\{r\_1, (b\_1 + (n-1) * P\_1)/(D1 - (\text{sum } j=1 \text{ to } n) P/Cj)\} \qquad (9)$$

$$R2 = \max\{r_2, (b_2 + (n-1) * P_2)/(D2 - (\text{sum } j=1 \text{ to } n) P/Cj)\} \quad (10)$$

3. If the reservation with grouping (from equation (6)) is smaller,
grouping can be performed else it is better to provide individual
reservations.

4. If it is decided to group two flows, then the new parameters of
the resultant flow group are calculated exactly as in equations (3)
through (5) (and equation (7)).

2.3. The Case of Identical Flows

As mentioned in the beginning for the special case of identical
flows (i.e. identical traffic characteristics and delay requirements)
we do not need to apply any checks to see whether aggregation will
be useful. In fact aggregation will always be useful in such cases
to reduce the reservation requirements. This can be seen by comparing
equations (6), (9) and (10) for the case of $b_g = b_1 + b_2$,
$P_1 = P_2 = P_g$, D1 = D2. Ignoring the pathological case
where the reservation requirements are equal to the average data
rates we will always have R1 + R2 > R_g.

2.4 Implementation of flow grouping

The emphasis of this draft is to bring out the benefits of the flow
grouping technique without outlining any implementation details.
However we note briefly that one possible method of implementing
this is by using IP-in-IP encapsulation to create an IP tunnel
corresponding to each flow group. A signalling mechanism such as
RSVP could then be used signal a single reservation for this flow
group instead of for each flow within the group. Alternative methods
also exist for creating some form of IP tunnel corresponding to
each flow group so that a single low reservation can be requested
for the single flow corresponding to the flow group rather than
for each individual flow within it. This draft does not seek to
go into any more detail on such possible implementation
techniques.

3.0 Additional Issues

3.1 Performance control policies and optimal reservation reduction

As mentioned earlier, in order to obtain maximal bandwidth
reduction, the network should always attempt to re-assign
flow groupings whenever any flow changes its characteristics
or flows enter or leave the network, so that overall
reservation is kept as low as possible. However, this
requires performing the test for grouping frequently.

Each such test can involve O(m) calculations where m is the
number of flows in a path, since in the worst case, we have
to compare an incoming or changing flow against all existing
flows.

The network can implement several policies which reduce the
amount of calculations and resource re-allocations by
compromising on the achievable reduction in reservation. For
instance, if the only the tolerable delay for a flow increases,
it may continue in its current group (thereby avoiding any
resource re-allocation or calculations), even though a greater
reduction in the amount of reservation could have been achieved
if this flow was now assigned to another flow group.

This draft does not present any algorithm to come up with
the optimal grouping of a given set of flows (one that
results in the lowest overall bandwidth reservation) since
that is likely to be a computationally intractable problem.

3.2 Policing

Even though flows are grouped, policing should still be
performed on individual flows rather than on flow groups. This
ensures that a flow which violates its traffic specification
can be detected even if the combined traffic in a flow group may
still be in comformance with the calculated parameters for
the group.

4.0 References

[i] S. Shenker, C. Partridge, R. Guerin "Specification of Guaranteed
Service," draft-ietf-intserv-guaranteed-svc-08.txt, work in progress,
Integrated Services Working Group, Internet Engineering Task Force
(IETF), July 1997.

[2] R. Braden, D. Clark, and S. Shenker, "Integrated Services
in the Internet Architecture: an Overview," RFC 1633, June 1994.

[3] R. Braden et. al., "Resource ReSerVation Protocol (RSVP) -
Version 1 Functional Specification," draft-ietf-rsvp-spec-16.txt,
work in progress, June 1997.

[4] A.K. Parekh and R.G. Gallager, "A Generalized Processor Sharing
Approach to Flow Control in Integrated Services Networks - The
Multiple Node Case," Proc IEEE Infocom '93 pp.521-530.

[5] The ATM Forum, "User-Network Interface (UNI) Specification
v3.1," September 1994.

5.0 Appendix

We show here that the traffic parameters calculated using equations
(3), (4), (5) and the delay bound of (7) are sufficient to
characterize a flow group formed from two individual flows. Consider
the flows F1 and F2 with traffic parameters (b1, r1, P1) and
(b2, r2, P2) and delay upper bounds D1 and D2 as above.

If flow F1 conforms to its traffic specification,
the amount of data provided by it to the source node over any time
interval (t1 , t2) is at most b1 + r1*(t2 - t1). Similarly the amount
of data provided by flow F2 in the same time interval cannot exceed
b2 + r2*(t2 - t1).
Now consider the flow obtained by grouping these two flows (Fg),
with traffic parameters (b_g, r_g, P_g) and delay bound
D_g, where these quantities are calculated according to
equations (3), (4), (5) and (7) above. Over time interval (t1, t2)
hence, the aggregate flow is required to generate no more
than b_g + r_g*(t2 - t1). However, this is easily true because when
equations (3) and (4) are true, we have

b_g + r_g*(t2 - t1) = (b1 + r1*(t2 - t1)) + (b2 + r2*(t2 - t1))

Also, the greater of P1 and P2 is clearly the
maximum packet size of the combined flow. Finally, if every packet
of the combined flow is guaranteed an end to end delay which is the
minimum of the tolerable delays of all flows making up the flow group,
clearly, every flow will see an acceptable delay even when it is
combined with other flows in a group. Hence equations (3), (4), (5)
and (7) represent valid parameter values for the flow obtained by
grouping flows F1 and F2.

6.0 Acknowledgemnts

The authors would like to thank Steve Blake of IBM for
helpful comments and suggestions.

7.0 Author Information

Sanjeev Rampal
C305/B664,
IBM Networking Division,
Research Triangle Park, NC 27709.

Voice 919-254-4801
email sanjeev@raleigh.ibm.com

Roch Guerin
IBM T.J. Watson Research Center
Yorktown Heights, NY 10598

Voice 914-784-7038
email guerin@watson.ibm.com