We Don't Need No Signaling Plane

Michael Welzl and Kashif Munir

Universität Innsbruck, Technikerstr. 21A, 6020 Innsbruck, Austria
{michael.welzl,kashif.munir}@uibk.ac.at
http://dps.uibk.ac.at/nsg

Abstract. Signaling between end systems and edge routers for the sake of per-flow QoS is troublesome: it can have scalability limitations, reveal flaws in equipment and/or the architectural setup of a network, and providing it usually requires manpower. Therefore, in practice, per-flow QoS will often not be made available, at least not for free. We sketch how per-flow QoS for Grids could be obtained in a fully distributed and therefore operationally easier manner while fully utilizing network resources.

Overview

Realizing per-flow QoS guarantees as needed for Advance Reservation in Grids (which should include the network among other resources) is not easy. Even when such QoS services would be available, providing them is an effort for an ISP, meaning that it will also not be done for free. On the other hand, differentiating between a protected traffic aggregate and "all other traffic" is much easier, either by switching a pre-configured type of traffic (with classification via the DSCP, for instance) onto a leased line with MPLS or by treating it as "Expedited Forwarding" (EF) traffic with DiffServ.

On the end system side, there is already a clear logical separation in Grids between short function calls (typically SOAP based messages addressing Grid Services) and bulk data transfers, which can be immensely large and are typically executed by calling an external service such as GridFTP. It is these long file transfers for which Advance Reservations are typically needed in computational Grids. Note that this is a major departure from the traditional QoS requirements of multimedia traffic, where a fixed rate must be sustained for a certain duration, and delay bounds may be required. For Advance Reservation in Grids, the guarantee that is needed is that a file will reach the other end within a certain time. In other words, the traffic is elastic, which makes quite a difference: if a flow quits, other flows will automatically increase their rates because of the underlying congestion control mechanism, thereby leading to an earlier termination time, which can in turn make it possible to admit a flow which could not have been admitted otherwise because the necessary bandwidth has now become available.

In order to guarantee fine-grain QoS, traffic within our protected aggregate must be controlled — but, rather than involving routers, this can be done at

the end systems by communicating with a Resource Broker (a common service in Grids where one can, for instance, request a machine with a certain CPU power; our intention is to extend this element with the ability to grant Advance (Network) Reservation). Specifically, the following communication would have to happen:

- A flow wants to enter the system, requesting a file transfer of a given size to be terminated by a specific deadline. It asks the Resource Broker whether it may join or not.
- The Resource Broker would have to say "yes" or "no" (or "no, but would this other deadline be suitable?").
- When a flow decides to leave the system, it must inform the Resource Broker (alternatively, measurements could be used to automatically decide that a flow has terminated).

This is a standard admission control scenario, which always involves calculations in the element which either grants or rejects a request (the Resource Broker), and usually also involves communication with routers. In the standard Bandwidth Broker scenario, where such signaling is used to ensure per-flow QoS, routers must constantly update the bandwidth broker about their current state, and at least the ingress router close to the newly joining flow must be informed about it in order to detect it and apply the right shaping or policing functions to ensure conforming behavior.

In our scenario, there is no need to check for conforming behavior in routers because there is no incentive for a Grid user to have a single flow break the rules — a reasonable Grid user will correctly assume that the greatest overall utility of the Grid is attained by following them. As for state updates from routers, since our Resource Broker controls all the traffic, knowing when a flow enters and leaves the aggregate, there is no need for such traffic updates. Other information about the network is however needed, and would have to be communicated to the Resource Broker from a constantly active distributed measurement system in the Grid:

Bottleneck link capacities must be known for all bottlenecks of all end-to-end paths. This information can be obtained with the "packet pair" measurement method. Packet pair was was frequently criticized for being imprecise, but the precision can be greatly enhanced by measuring for a long time. Given the longevity of nodes in a Grid, it is feasible to install a (mostly) passive measuring system which mainly listens to flows on the receiver side and only generates traffic if a path has not been used for a very long time.

Shared bottlenecks can be detected by carrying out traceroutes between all nodes, and "filling the holes" that appear because routers do not answer to traceroute with active measurements. These would make use of the fact that congestion controlled traffic (such as TCP) reacts to more aggressively controlled (or uncontrolled) traffic dramatically, making it possible to generate "signatures" in one flow, the imprint of which could be detected in another if they share a bottleneck.