Lecture 23: April 27

Lecturer: Prashant Shenoy *Scribe: Bhuvan Urgaonkar*

The first part of this lecture discusses the need for integrated services networks and two approaches to design such networks. The second half discusses the IntServ and the DiffServ architectures in the context of the present Internet.

23.1 Need for Integrated Services Networks

The Internet offers a single class of "best effort" service; that is, there is no admission control and the network offers no assurance about when, or even if, packets will be delivered. Traditional data uses such as remote terminal, file transfer and electronic mail are rather elastic in nature, in that they can tolerate packet delays and packet losses rather gracefully, and so they are rather well served by the current Internet's best effort service. Moreover, because of this elasticity, they can decrease their transmission rate in the presence of congestion. With the emerging widespread use of multimedia applications, computers are now processing voice and video in addition to their more traditional tasks. Therefore, the networks that connect these computers, such as the Internet, must be prepared to cope with the traffic emanating from such applications. Real time applications like video and audio have very different characteristics and requirements than data applications, and thus their emergence is likely to significantly alter the nature of the Internet's traffic load. In particular, as traditionally implemented real time applications are typically less elastic — less tolerant of delay variations — than data applications.

23.2 Two Approaches to designing IS Networks

Here are two kinds of approaches with their pros and cons:

- **Change the Implementation** This approach advocates addressing the lack of elasticity at the application level. An example is to use a larger buffer at a client receiving video content to help mask variable delays over the Internet. Such an approach has the advantage that no changes are needed in the current Internet and so it is easy to employ. However, since the network delivers the same class of service to all applications, there is no assurance of quality of service. Application development becomes more complicated. Also, this approach can benefit only certain kinds of applications. For example, using a larger buffer at the client can help applications receiving stored video, but not applications involved in tele-conferencing.

- **Change the Architecture** This approach requires fundamental changes in the existing Internet. The basic philosophy is the following — One can always increase the efficacy of an architecture by supplying more bandwidth; faster speeds mean lower delays and therefore higher utility values. However, one can also increase utility by keeping the bandwidth fixed and delivering a wider variety of services than just a single class of best-effort service. Such an extension would allow different applications to get different qualities of service; the general intuition is that the closer the services are aligned with application needs, the higher the efficacy. For instance, if different applications have widely different sensitivities to delay, then offering two priority classes will likely increase the efficacy of the network (when compared to a single class of service).

23.3 The IntServ architecture

IntServ (Integrated Services) architecture is a framework developed to provide individualized quality of service guarantees to individual application sessions. Two key features lie at the heart of IntServ architecture:

- *Reserved resources* A router is required to know what amounts of its resources (buffers, link bandwidth) are already reserved for ongoing sessions.
- *Call setup* A session requiring QoS guarantees must first be able to reserve sufficient resources at each network router on its source-to-destination path to ensure that its end-to-end QoS requirement is met. This call setup process requires the participation of each router on the path. Each router must determine the local resources required by the session, consider the amount of its resources that are already committed to the ongoing sessions, and determine whether it has sufficient resources to satisfy the per-hop QoS requirement of the session at this router without violating local QoS guarantees made to an already admitted session.

Here are some of the difficulties associated with the IntServ model and per-flow reservation of resources:

- Scalability Per-flow resource reservation implies the need for a router to process resource reservations and to maintain per-flow state for each flow passing through the router. Since the number of flows through a typical backbone router is several hundred thousands, this represents a considerable overhead in large networks.
- Application design more difficult This is because the applications need to characterize their flows.

23.4 The DiffServ architecture

These considerations have led to the DiffServ (Differentiated Services) architecture. The basic idea behind this architecture is to place only simple functionality within the network core, with more complex control operations being implemented at the "edge" of the network. The differentiated services architecture is flexible in the sense that it does not define specific services or service classes. Instead, it provides the functional components, that is, the pieces of a network architecture, with which such services can be built.

References

- SCOTT SHENKER, Fundamental Design Issues for the Future Internet