Material for lecture #6

Meaning of prioritization - reasons and consequences[1]

1.1 Fundamentals

What is the fundamental reason to prioritize something? Prioritize[2] can be defined as "arrange according to priority", which means that we need to define priority. Http://www.dictionary.com gives the following main meanings for priority:

1. Precedence, especially established by order of importance or urgency.
2. a. An established right to precedence.

   b. An authoritative rating that establishes such precedence.

From our viewpoint the conclusion is that prioritization essentially is an act performed by an authority to classify some items by order importance or urgency. This is the nature of the prioritization act - but what is the general objective? Evidently the objective should be related to the task of the authority. For instance, if the main task of an authority is to provide a service, then the prioritization has to somehow support the service provision in a way that is advantageous for the authority.

Authority: Internet Service Provider (ISP)

Service: Internet connectivity

=> Prioritization has to support service provision

This reasoning could appear naïve and be lacking any analytical insight; still I think that we need to keep this simple idea clearly in our mind when further evaluate the meaning and objective of prioritization in packet networks.

There are lot of issues to be considered, such as

- In which way the ISP wants to improve the service provision?
- What is or should be the actual object of prioritization (packet, flow, application, customer, or group of users)?
- The relationship between prioritization and other tools for improving service provision

There seems to be two main situations from service provision viewpoint: the main task of the ISP is to make money directly based on the Internet service, or the ISP[3] has a fundamental task (e.g., to serve the public interest of a society) that cannot be easily converted into economical calculations. Nevertheless, my opinion is that in most cases the possible advantages (and disadvantages) of any service provision

[1] *Prioritization* is used instead of shorter *priorization* because Altavista gives about 100 times more search results for the first one!

[2] According to the dictionary mentioned in the body text *prioritize* is widely regarded as corporate or bureaucratic jargon.

[3] The term ISP is here used in a very broad sense, it covers all organisations that manage Internet type of networks and services

model can be presented by means of utility calculations as discussed in the earlier lecture notes. If that is true, we can simplify the objective of prioritization into the following form

- The objective of prioritization made by an ISP is to increase the total utility defined either as a monetary value, or in another feasible way.

The simple message of this statement is that, in the first place, the ISP has to define what is the total utility of the whole system and then to carefully think what kind of effects different prioritization models have into this total utility - and if something does not serve this purpose, there is no reason to use that prioritization. The objective of this essay is to consider exactly these matters.

1.2 Prioritization architecture

As to the technical level, we can basically start from the assumption that the total utility model is given, and the task is to design a prioritization system that gives an optimal result as regards the utility model.

Then the key question is the basic objective of prioritization, or more generally, the architecture of prioritization. From practical, operational viewpoint it seems that the natural objectives are applications, users, or group of users, where the definition of group depends on the contexts.

One problem is that the common sense rules for prioritizing applications, users and different groups of users can be conflicting. Let us assume that a user belonging to a user group (U_1) is using an application (A_1) and another user belonging to another group (U_2) and uses another application (A_2). Now if the importance relations of these groups and applications are:

$$I(U_1) > I(U_2),\ I(A_1) < I(A_2)$$

there is no clear rule for prioritizing the corresponding flows or packets. This problem is even more complicated because the requirements of each application can be totally differing. Thus, even if $I(U_1) > I(U_2)$ and $I(A_1) > I(A_2)$, it is not obvious that the packets of user 1 are more important than packets of user 2, if application A_1 consumes much more resources than application A_2.

Anyway, we can try to start with the idea that there are a number of user groups ($1...N_U$), and a number of application groups ($1...N_A$), and all flows are classified into one of the available locations in the user-application matrix. It seems that the only task of the operator is to define the importance ordering of the all $N_U * N_A$ locations. But is this enough? No, because the fundamental question of resources is missing - and that means we need at least a third dimension. If the required flows are classified into N_R groups according to their capacity requirements, we have altogether $N_U*N_A*N_R$ locations to be prioritized. From implementation and management viewpoint, the total number of locations could be a problematic issue. For instance, if $N_U = 3$, $N_A = 10$, and $N_R = 20$, the total number of locations is 600, which quite a large number to be managed manually[4].

A further complication is that priority can be established by order of importance or urgency (as the definition mentioned in the beginning indicates), and these two aspects do not necessary adhere to the same ordering. So each location should consist of two priorities, first one related to the importance of the packet and the other one related to the urgency of packet.

[4] Note that 20 means that we can cover the region from 1 kbps to 1 Gbps if it is divided in steps of 2.

Yet, it is easy to identify needs that further complicates the model. For instance, the time of date can have an effect on the order of priorities, because some customers may prefer to get good service on daytime at the expense of worse service after 6 pm, and vice versa. Also, it is quite possible that certain users are allowed to use abundant resources inside a network domain, but only restrictedly those resources that connect the network domain to the outside world. Finally, the operator may reasonably think that network should give higher priority for existing flows than for new attempts, because end-users hate to have interrupted service.

So, what is a reasonable approach to solve this dilemma? "A "capability", merely considered as such, will always seem better to have than not to have" - this is one of quotations mentioned in the notes of 2nd lecture. It is strange how alluring it could be to design an N-dimensional table that defines the whole prioritizing system so that the operator can tune the system very accurately, application by application, user by user, minute to minute, even though the total number of locations in the table could be so huge that the task for filling the table in a reasonable and reliable way is practically impossible.

So let us recall the objective of the whole system, namely, the total utility of the system. The management cost of the system is one part of the story, and any increase of it must be justified by a larger increase in the total utility. It is of extreme importance to remember that the prioritization does not usually increase the total throughput of the system at all; it only divides the limited capacity more efficiently among some entities using a given efficiency criteria. Because of this fact there always is an absolute limit that cannot be exceeded independent of the complexity and intelligence of the system. Usually a limited number of well-designed steps produce the highest reward, whereas the gain obtained by any additional feature could be negligible compared to the associated cost.

As a conclusion, we need a reasonable structure or architecture for the prioritization, not a huge table - whatsoever nice features the table appears to provide.

What could be this kind of straightforward, but efficient prioritization? Most probably it should be based somehow on the amount of price paid by the customer, if the target is to make profitable business. In other words, those connections should be favored that produce the highest earnings per bandwidth. The prioritization shall be dynamic in the sense that if the bit rate is smaller (or otherwise less resources are used), the connection is considered to be more important than another connection with higher bit rate but otherwise similar characteristics. The weak point of this reasoning is that it does not take into account situations, in which the bandwidth is anyway useless, because certain minimum threshold is not exceeded. However, we have to keep the system simple and realistic and any system that requires detailed information about individual applications and their state is quite unrealistic from realistic implementation viewpoint.

1.3 Example

There are a lot of possible motives to use prioritization, but perhaps the most fundamental one, together with the need to use limited resources as efficiently as possible, is the need to minimize the risk when new services are introduced. This phenomenon is illustrated in the following example.

Let us assume that the original state of affairs is that a service provider has certain profitable service with one application, e.g., phone calls. In the starting phase there are on average 1000 users on a link, and the traffic variation is 200 users during the busy hour ($\varepsilon(N_{old}) = 0.2$). Further, we can normalize the capacity calculations by defining that the capacity required by one user is 1, and the utility of a successful call per time unit is 1. The utility of an unsuccessful call is assumed to be -20 (similarly

with the example in lecture notes #5 except that the effect of variability inside one flow is ignored). Under these assumptions, with link capacity of 2000 and with a best effort service, the total utility of the old service is 992.

Now the service provider wants to introduce a new service, for instance, a video streaming service with the following parameters:

Utility of a successful call = 2 per time unit

Utility of a call with insufficient capacity = -10 per time unit

Bandwidth requirement = 10

Uncertainty related to the number of users = 50% of the expected number of users, or $\varepsilon(N_{new})= 0.5$

The framing of a question for the operator could be:

- How much improvement in total utility can be achieved if either the old or the new service is prioritized over the other one compared to a pure best effort service?

The primary problem of the service provider is two-fold. First the operator does not want to risk the business of the old service with a less certain new service. Second, because achievable utility figures are different for the two services, a pure mixture of services is not necessarily the best approach - as discussed earlier, the main objective of prioritization is to exploit the utility differences in order to maximize something similar to the total utility.

We can evaluate this question by making the following simplifying assumptions:

- In best effort case, all connections are assumed to be useless if there is not enough capacity to serve all flows.

- In prioritization model, all the capacity is first filled with the higher priority flows. If there is enough capacity for high priority flows, they generate the utility of successful calls independent of the low priority traffic. The remaining capacity left by the high priority traffic is used to serve the low priority flows, and once again, if that capacity is not enough for all flows, all low priority flows are considered useless.

As to the analysis of the system, the main task is to calculate the probabilities of different traffic load for both of the service categories, everything else is quite straigthforward. In the same way as in the examples of previous lectures, all distributions are assumed to be log-normal. The result of this simplified model are presented in Fig. 1.

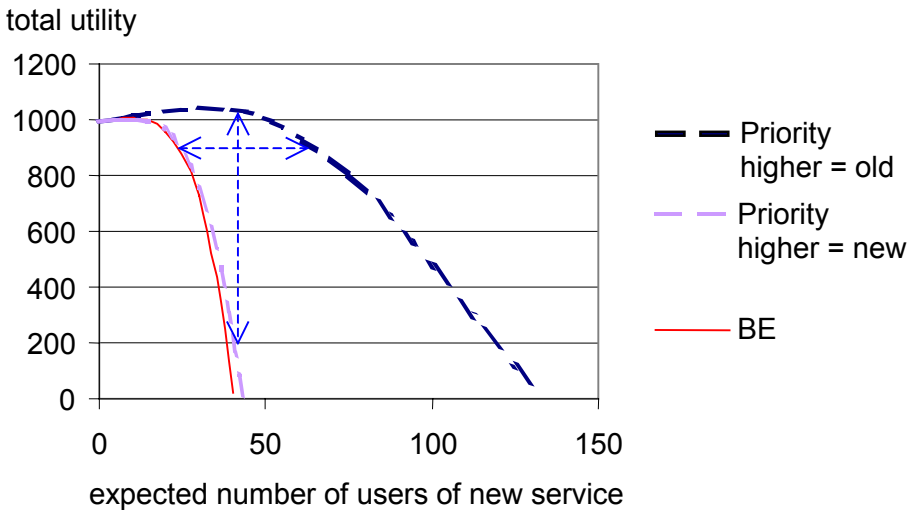

Fig. 1. Total utility with and without prioritization

As mentioned in the beginning of the example, the main tasks for the use of priorities are to decrease the risks of new services and to use limited network resources more efficiently. Fig. 1 illustrates the both aspects: the horizontal difference depicts the risk aspect while the vertical difference depicts the efficiency aspect. In both cases, the result is very clear; the optimal strategy is to give priority for the flows using the old service. Even without any capacity increase, this priority approach can generate some extra utility without deteriorating the old business in any way. The maximum total utility (1044) is reached by an average number of 33 new users. On the reverse, without any prioritization, the utility hardly increases at all, but starts to fall very rapidly when the average number of users of the new service increases.

With $E(N_{new}) = 50$, the total utility is still above the original value (1007 vs. 992) when prioritization is used, whereas with best effort service the total utility is as low as -1224.

Further, this example illustrates very clearly how important it is to design the prioritization in a reasonable way. If the service operator decided, for some reason, to favor the users of new service, the result would be very similar those of the best effort service.

Note particularly that the utility of the new service per time unit is higher than that of the old service. However, the significant issue is not pure utilities, but the utilities per required bandwidth, and in that sense the old service is much more valuable than the new one.

1.4 Prioritization vs. reservations

Let us finally check the list of possible reasons for resource reservations, and look what are the possibilities to satisfy these needs by prioritization.

1. Applications' need of minimum bandwidth

   With priorities the apparent solution is to mark packets with high priority if (and only if) the bit rate is below the minimum bandwidth. There should be also some kind of control or charging method that limits the amount of the high priority traffic.

2. Virtual Private Network or Leased Lines

   Although prioritization can be used to improve the realisation of VPNs, it is not clear whether priorities provide a sufficient tool to manage a large number of VPNs, or if resource reservations are necessary. It seems that the essence of this question is whether the state information of each leased line connection should be kept in every node, or whether it is sufficient to limit the incoming traffic.

3. Pricing model [that indicates that resource reservations are used inside the network]

   This issue is more about psychology and customer relations than technology, because if will be very difficult for any user to check whether a real reservation, whatsoever that means, is performed inside the network, or whether the technical solution is based, for instance, on pure packet level prioritization. Still the service provider may think that the promise to make "real" reservations improves the customer's willingness to pay for the service. Anyway, it is somewhat hard to believe that this reasoning could by itself be a sufficient justification for implementing resource reservations in a packet network.

4. Need to know service availability beforehand

   It is somehow easy to think that this kind of advance booking definitely requires a real reservation. However, that is not at all clear on packet level, because the system can be designed in way that the reservation just means a right to mark packets with high priority during the given period. The system has to keep track on these rights, but that does not imply that the network nodes have to make any definite reservations.

5. Emergency calls

   At least for me, an emergency call is an issue that is most naturally solved by packet level prioritization with an appropriate control to limit the use of this call category. It is hard to imagine any model that could somehow work better.

6. Need to limit traffic sent into the network

   Prioritization *is* a tool to limit the traffic sent into the network. Particularly, a proper prioritization scheme limits the volume of high priority (or important) traffic, which usually is the primary objective. If the network is build in a way that even a very high load on lower priority levels has only a negligible effect on the service quality of higher priority traffic, prioritization is a very efficient tool to solve the problems that operator tries to solve by limiting the incoming traffic. Note that usually even excessive traffic as such is not the actual problem, but rather the damaging consequences that are a combined result of the excessive traffic and defective traffic control.

7. Technological reasons (reservations may improve the use of network resources in some specific cases)

   The relative efficiency of different approaches should be assessed case by case. Still it is likely that in some cases prioritization is not the most efficient way to handle the traffic but bandwidth reservations should be used at least with certain applications.

8. The need to favour existing connections over new attempts

   This seems to be really difficult to implement without keeping per-connection state information and without a kind of connection admission control. Indeed, this property is an integral part of the service in any circuit switched network with no or very limited variety of applications. However, the important question is, rather than the way of implementation in packet networks, how critical this issue is. It may have a significant effect on the user satisfaction and service provider business, but that is not sure in cases where the variety of applications is one of the most prominent characteristics. Even if this property definitely seems advantageous from one application viewpoint, it is anything but clear whether all kind of old connections should be favoured over all kind of new connections.