GoS networks

Taking Policy to the
Smart Connected Device

Guarantee of Service: the technology behind GoS

Version 2.0. June 2013

Contents

Overview

Overview

In the drive to introduce new applications to meet commercial opportunities, the simplicity of IP has been an overwhelming advantage. However, new services increasingly demand better performance than 'best effort' IP can deliver. GoS™ technology offers a way of meeting the demands of complexity, without losing the advantage of simplicity.

Summary

This document provides an overview of the primary QoS issues in converged networks, describes the fundamentals of GoS, its two-dimensional quality model, and the basics of GoS processing.

Context

Today's networks are not only transmitting more data, they are also transmitting a wider variety of types of data. As Internet access continues to grow, so does the incentive to carry multiple services, such as video, data and voice, across a single medium. This convergence of different kinds of traffic can create greater efficiency in delivery of existing services, and provide more opportunities to create new ones.

However, the current performance of the Internet is limited by the very aspect that has made it so popular: the sheer simplicity of IP. The standard IP delivery process does not provide any predictable level of quality of service (QoS); that is to say, there is absolutely no guarantee - not even statistically - that given packets will make it across the network in any consistent time, if at all. Some applications, such as VoIP, impose challenging requirements on packet delay and loss, and often fail when they are not met.

This limitation is well known, and there have been many attempts to find suitable solutions. A whole range of signaling and packet-delivery mechanisms have evolved for various purposes in various types of networks. Meanwhile, the Internet Engineering Task Force (IETF) has set out a variety of standards (RFCs) describing types of QoS. However, none of the mechanisms created to date has provided a full solution to the problem of running mixed services over a common network.

The QoS Challenge

The QoS Challenge

A successful QoS mechanism, for today's networks, needs to simultaneously provide reliable quality to different services, each with different needs. VoIP is delay-sensitive (requiring < 150ms end-to-end latency), while steaming video and some business applications are loss-sensitive. To ensure the success of these mixtures of traffic-types, they need be protected from each other, (particularly so called 'bandwidth-hogs' such as large-packet data streams). On top of that, a QoS mechanism should also be cost-effective, scalable and easy to configure.

Features of QoS mechanisms

Packet networks share the same physical links between many different services, even though a data cable can normally transmit only one packet at a time, at a fixed, limited rate. Each service running across a network has its own requirements for sending rate, allowable packet loss, and the maximum acceptable time taken for data to travel from source to destination. "Quality of service" mechanisms, in general, try to satisfy the needs of these services by rearranging the order of packets transmitted on a given link.

In fact, the options available to a QoS mechanism at any particular moment are limited. An arriving packet can be:

- transmitted immediately (making the link unavailable to other services)
- queued for later transmission (increasing the delay experienced by this packet, and potentially the delay and/or jitter experienced by the stream as a whole)
- thrown away (increasing the loss experienced by the flow containing the packet).

Normally, lower-priority packets will be lost or delayed to allow higher-priority packets to overtake them. QoS algorithms are responsible for determining what constitutes priority, based on the services' quality requirements, and for making the "drop or queue" decision on a packet-by-packet basis.

The need to manage contention between services becomes more important as the load on a network link increases, since the contention for the limited resource is greater. Network utilization equals efficiency. To fill the links, and hence reduce the need to overprovision, QoS is an important consideration.

When the network load is such that not all of the requirements can be satisfied, a network administrator will want to apply policies determining which services are more important. In addition, the operator may wish to enforce contracted minimum or maximum levels of quality — a Service Level Agreement (SLA). Traditionally these have been written in terms of bandwidth rather than quality, but this is often insufficient to ensure that services will work. An extreme example: putting hard disks in a truck and driving them to their destination would give an immense bandwidth — but there's no way to use this to provide a voice-over-IP service. Loss and delay parameters can be just as important as throughput when provisioning for real services.

The QoS Challenge

What QoS can and can't do

Can do

- guarantee levels of service, even in network overload
- provide differential quality treatment to gove applications the quality they require
- protect against "chatty applications":
  - stop elastic traffic like TCP from hogging quality
  - guard against network denial-of-service attacks
  - enforce limits on groups of users or services
- predict performance ahead of time, on a statistical basis
- optimise network utilisation:
  - fill the link while keeping services working
  - increase "goodput"

Can't do

- undo degradation that's already happened
- enforce impossible policies
  - not all combinations of bandwidth/ loss/ delay and jitter are mathematically possible
  - the amount of quality available in the link is finite
- queue large numbers of packets without introducing delays
- predict what's coming next, no way of knowing that a high priority packet is about to arrive
- output packets faster than the link rate

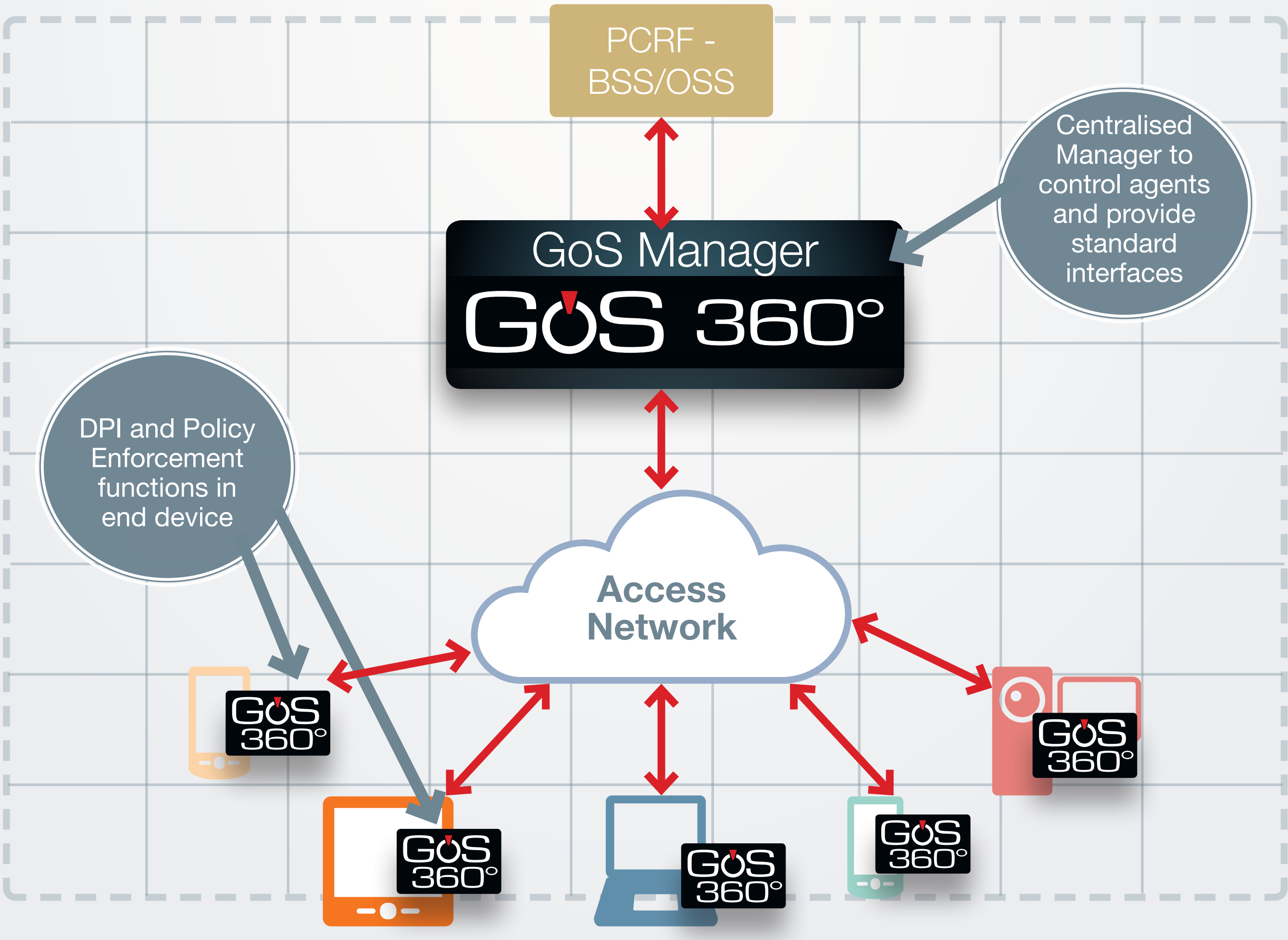
GoS Technology

GoS Technology

Guarantee of Service (GoS) is a technology to manage network congestion - the principle cause of loss and delay in IP and other packet networks - in a controlled and predictable way.

GoS incorporates novel queuing, rate-limiting and traffic-shaping mechanisms, more sophisticated than any other network scheduling method. GoS technology:

- is stable and predictable: it can deliver the configured set of QoS parameters at all network loadings.
- allows the user or network management software to know about the quality bounds that will be delivered, before the configuration is applied.
- permits the evaluation of different "what-if" configurations, allowing a network administrator to experiment with different rules for managing expensive network resources to meet business requirements before committing to one.



GoS Technology

Features of GoS

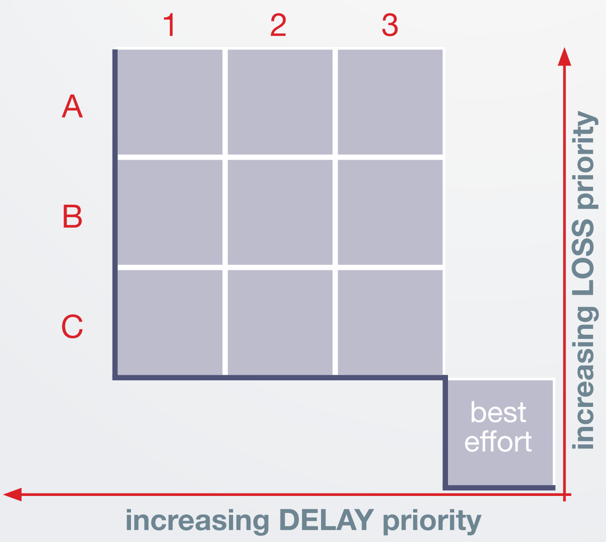
GoS Networks' GoS offering implements a set of algorithms that provide the network administrator with:

- guaranteed bandwidths
- enforced bandwidth limits
- reuse of unclaimed bandwidth
- separate control of loss and delay priority
- fair sharing of quality, not just bandwidth
- network stability under increasing load
- predictions of loss and delay behavior under saturation
- live monitoring of delivered quality

A new model

GoS is based on the fundamental relationship between three related aspects of each packet stream: loss, delay and throughput. The essential insight behind GoS is that these variables are interdependent: if one is fixed, a relationship is created between the other two, and if two are fixed, the third is determined by default, given the current operational behavior of the network. Thus there are two degrees of freedom that the user can determine. So, for example, it is possible to manage a contention point to reduce loss for certain streams, but only at the cost of increased delay for those or other streams.

Using this principle, GoS technology allows different levels of treatment for many different data streams, simultaneously. For instance, a VoIP stream cannot tolerate high delay, but it can accept a moderate proportion of packet loss, without a perceptible drop in quality. For business-critical transactions using certain protocols, on the other hand, delay is less important; what matters is that all packets arrive at their destination eventually.



GoS Networks calls this two-dimensional classification of network quality the 2D-QoS Matrix, which it uses for QoS configuration. GoS essentially provides differentiated services based on a two-dimensional matrix of loss and delay classes, where matrix elements are called classes.

The Basics of GoS processing

Predictable QoS

GoS is the only technology currently available that can give the user a confident prediction of the quality of service that they can expect under a given configuration, even under heavy load conditions. The user can specify differential treatment per stream, according to its individual needs; GoS then predicts the maximum levels of loss and delay that will result.

Because GoS is based on a rigorous mathematical model, these predictions take the form of strong statistical predictions of service. When the congestion point is relatively quiet, the GoS device will be almost transparent; but as congestion grows heavier, or even when the congestion point experiences momentary bursts of traffic, the predicted levels of service will hold. Thus, users can be sure that their critical streams will still receive a certain throughput with tight bounds on loss and delay, under all conditions.

Basics of GoS processing

Principles

GoS traffic management is based on dividing packets into different Quality Groups (QGs). Incoming packets are directed into one of several QGs, based on a classification step (which may be part of the GoS datapath, or may be handled beforehand). QGs may be defined as receiving

- guaranteed access to a certain share of the link bandwidth, and
- a certain level of priority treatment relative to other QGs.

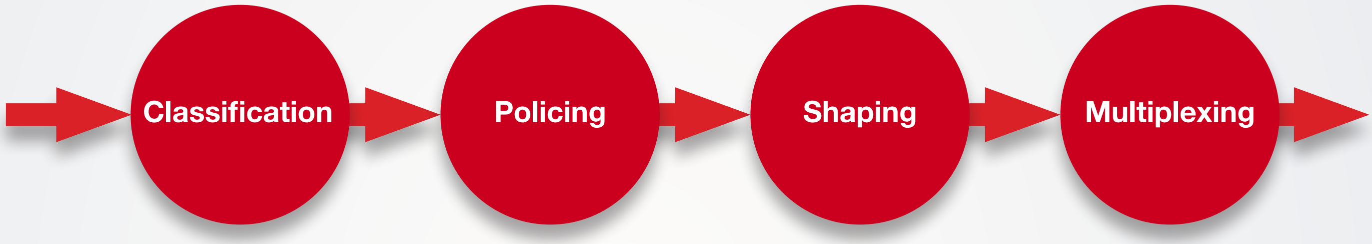
On a normal network link, before GoS control is applied, all traffic competes for bandwidth on a best-effort basis. With GoS control, traffic that is entitled to priority treatment will automatically displace best-effort traffic when necessary, up to its bandwidth limit. This allows spare capacity to be used freely for best-effort traffic, without affecting priority traffic.

When the link is heavily loaded, even with bandwidth control, packets will still be subject to delay (because they have to queue for transmission) or, occasionally, loss (because the queue may be full). In GoS, one of the attributes of a QG is a GoS Class (see diagram above), which defines the relative treatment its packets should receive in this case. A QG may be defined as highly sensitive to delay (in which case, its packets will queue-jump less urgent traffic), or highly sensitive to loss (in which case, extra queue spaces will be made available).

The Basics of GoS processing

Stages

There are four stages of processing carried out by the GoS datapath:



- **Classification**: separate incoming traffic according to its quality requirements
- **Policing**: enforce bandwidth guarantees and limits
- **Shaping**: re-space packets for fairness and predictability
- **Multiplexing**: merge separated flows according to their priority

Stage 1: Classification

In the GoS reference datapath, packets are identified by a Classifier component, which marks packets based on information in the packet headers as belonging to a particular Traffic Type (TT). Traffic Types are then assigned to QGs based on the classification marking.

When GoS is embedded in a network device, packets may already have been marked in some way (for example, with DiffServ Code Points) to determine their quality requirements; in this situation, GoS can assign packets to QGs based on this marking information, instead of using its own classifier.

Stage 2: Policing - bandwidth control

Each QG is fed through its own bandwidth-controlling component, called a Policer. A QG may be specified as receiving either “strict policing” or “CAR policing”.

Strict policing
Strict policing allows the network administrator to set an absolute limit for traffic in this QG (its rate). Traffic arriving at a rate below this level is allowed through, and receives the loss and delay priority treatment assigned to the QG. Traffic arriving above the configured rate is discarded. This bandwidth is guaranteed to be available to the QG whenever it is demanded.

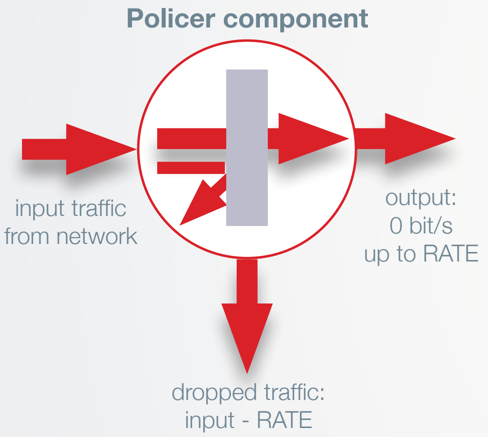
CAR policing
CAR (committed access rate) policing provides a way for traffic in this QG to reuse bandwidth that is assigned to other QGs but which they are not currently using. The “extra” bandwidth is made available on a best-effort basis — that is, it is not regulated, and is treated with the lowest loss and delay priority.

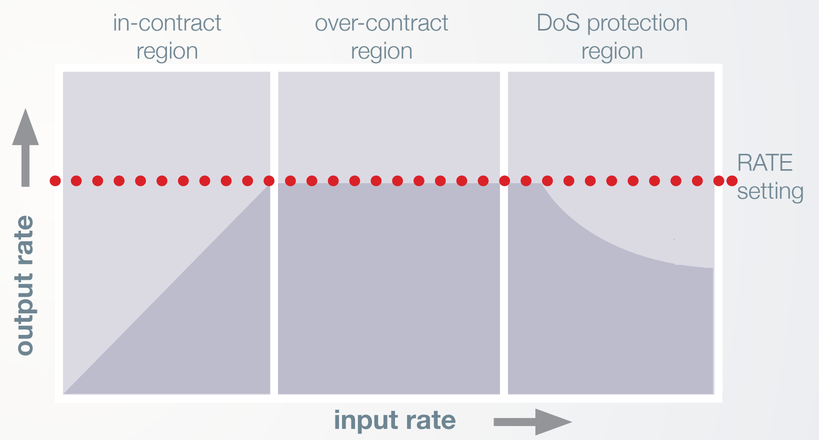
The Basics of GoS processing

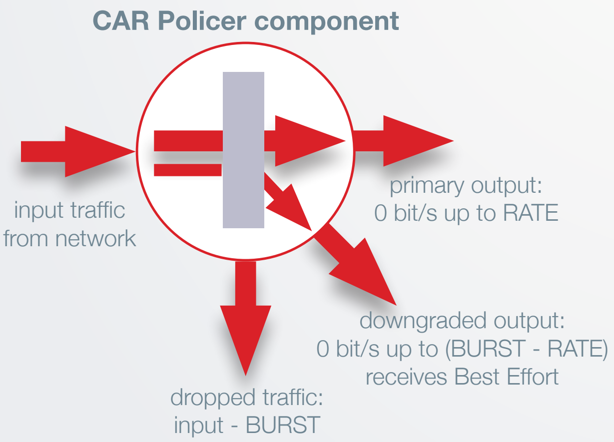
Two parameters are set for a CAR policed group:

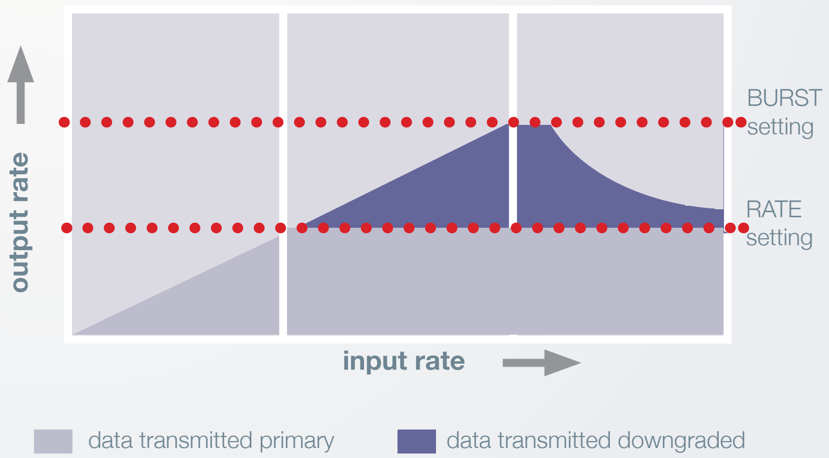
- The rate setting assigns a bandwidth guaranteed exclusively to this QG. Traffic arriving inside this limit will be prioritized according to its assigned GoS Class. (This is explained in the section on Multiplexing, below.)
- The burst setting is the maximum total bandwidth that this QG is allowed to use when borrowing unused capacity from other groups. This extra traffic will be downgraded to the best-effort GoS Class.

Data arriving in a CAR-policed QG at a rate greater than the burst setting will be thrown away.









"Best effort" groups

A third class of QG receives best-effort treatment only. These QGs may use whatever bandwidth is left over after non-BE traffic has been dealt with. The group can burst up to the whole capacity of the network link, but is treated as lowest priority.

The Basics of GoS processing

Stage 3: Shaping

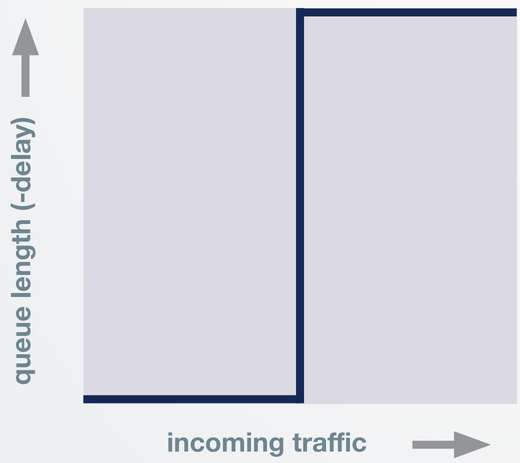
Some vendors use the term "shaping" interchangeably with "policing". In GoS, these are two separate and clearly different operations. The shaping stage adds random perturbations to the delays experienced by a QG. This randomness allows streams to be recombined in a predictable, stable manner. The shaping stage is to:

- avoid starvation
- merge streams in a fair manner
- make network performance gracefully degrade in overload, rather than catastrophically collapse
- allow stochastic methods to be used to predict behavior at the merge point

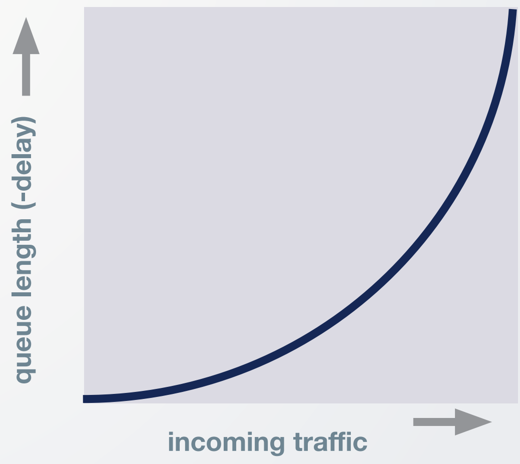
The GoS architecture uses one Shaper component per QG, and one per GoS Class.

Some effects of this shaping can be illustrated by a "road junction" analogy (see figure below).

Conventional first-in-first-out (FIFO) queuing does not cope with overload: an increase in traffic past the critical point may suddenly and dramatically increase the delay and loss in a network, destroying its ability to successfully carry real-time traffic. GoS shaping circumvents this unpredictable mode of behavior (see diagrams below).



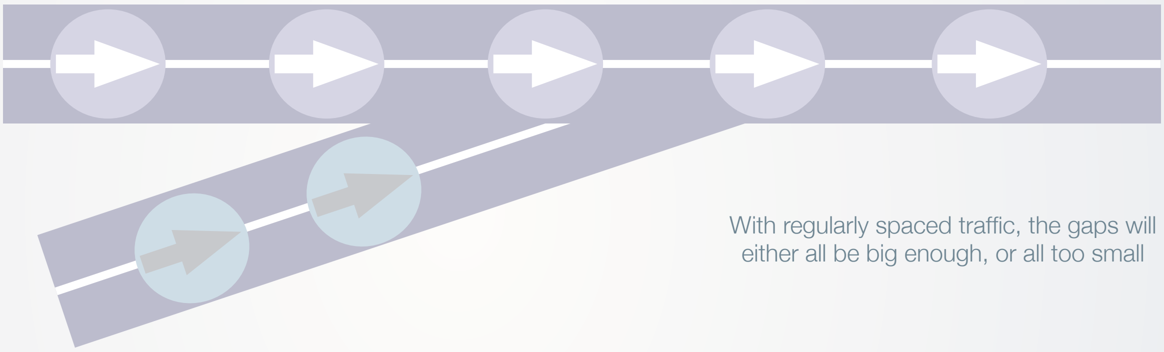
Conventional queuing disciplines can suddenly produce high delay when the network load passes a critical threshold



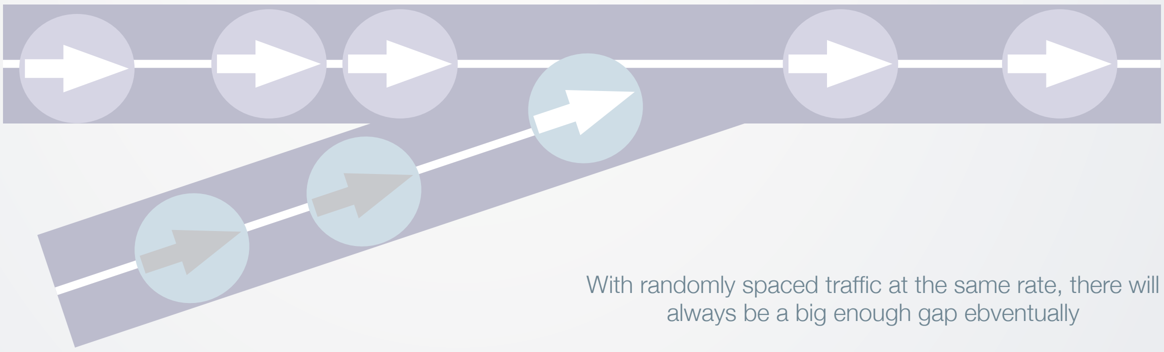
GoS shaping provides a smoother response to increasing network load, avoiding the possibility of abrupt failure

The Basics of GoS processing

Question...
If a vehicle is waiting to turn onto a major highway, it must find a gap in the traffic big enough to let it in. If the highway traffic is regularly spaced, either all of the gaps will be big enough (so the car joins immediately), or none are (so it never has the chance to join). This is an example of catastrophic collapse: as the density of traffic gradually increases, a completely working system suddenly flips into a completely non-working state.



With regularly spaced traffic, the gaps will either all be big enough, or all too small



With randomly spaced traffic at the same rate, there will always be a big enough gap ebventually

Now imagine that, instead of being evenly spaced, the gaps between vehicles on the highway are random. Sooner or later, there will always be a gap large enough to fit into. As the amount of traffic increases, these gaps occur less frequently. Thus the ability of waiting cars to join degrades gradually, rather than failing suddenly.

In GoS, the "shaping" stage randomizes the gaps between successive packets, thus allowing streams to merge even when they are very congested.

The Basics of GoS processing

Stage 4: Multiplexing

This is the point at which packets are brought back together to feed into a single output link. The order in which packets are output is finalized here.

The GoS CUBEMux component uses a two-stage algorithm that allows loss and delay priority to be set separately. This means that the priority settings have definite meaning — unlike many traditional algorithms, where the notion of “priority” doesn’t correspond to a desirable real-world behaviour.

Advantages of GoS

The advantages of GoS

By policing and shaping different streams and the aggregate traffic flow, GoS technology offers these advantages:

- It manages contention for network resources in a controlled and predictable way. Contention between and within streams is managed within the GoS-enabled device, rather than happening in an uncontrolled manner further along the network path.
- It allows loss and delay to be managed simultaneously and independently.
- It allows statistical predictions of loss and delay performance to network traffic types, which remain true irrespective of network load and the demands of other network users.
- It polices all traffic streams fairly; that is, no one stream is allowed to use excessive network resources. This feature has the added benefit of giving some protection against 'denial of service' attacks and similar packet floods.

Unlike conventional IP bandwidth-management technologies, GoS is composable: that is, GoS instances can be cascaded to provide a general, network-wide solution to QoS problems. By deploying GoS technology at the major contention points, it will become possible to provide a guaranteed level of service across an arbitrary number of network hops. Thus, GoS has the potential to offer guaranteed QoS, per stream, from end to end of a standard IP network. It overcomes the quality limitations of IP, while retaining the simplicity and flexibility that has made the Internet the platform of choice for delivering every kind of new service.

GoS Networks
National Software Centre,
Mahon,
Cork
Ireland

Tel: +353 (0) 21 230 7050
Email: contact@gosnetworks.com

www.gosnetworks.com

Version 2.2. June 2013