Quality of service: basic building blocks

In this chapter we introduce the basic buiding blocks of a network architecture for the provisioning of QoS. We will deal with traffic regulators (policers, shapers), admission control and resource provisioning, traffic conditioning, scheduling, packet classification and active queue management

Paolo Giacomazzi

Traffic Conditioning Agreement and Service level Agreement

- In order to guarantee QoS, the provider and the customer must preliminarly stipulate a *Traffic Conditioning Agreement* (TCA) and a *Service Level Agreement* (SLA)
- The SLA specifies the target QoS that the provider is committed to deliver to the customer for a specified set of traffic flows
- However, it is neither possible nor reasonable that the provider is obliged to fulfill the SLA independently on the amount of traffic that the customers offers to the network
- Some upper limit to the amount of customer’s traffic for which the provider must meet the SLA must be specified
- This “upper limit” is specified in the TCA
- The TCA specifies a *traffic profile*
- For a given reference flow, the part of traffic complying with the TCA is called *IN traffic* or *conformant traffic*
- The part of traffic exceeding the TCA is called *OUT traffic* or *non-conformant traffic*

Traffic Conditioning Agreement and Service level Agreement

- The SLA must meet the SLA only for IN traffic, while a number of actions can be made on OUT traffic
- In fact, the provider must protect other established SLAs from excess traffic that a customer may offer to the network
- Examples of actions on OUT traffic
  - *Policing*: OUT traffic is dropped
  - *Shaping*: OUT traffic is delayed until it is possible to send it complying with the TCA
  - *Marking*: OUT traffic is marked and offered to the network; in case of congestion, PUT traffic is dropped first

Traffic Conditioning Agreement and Service level Agreement

- The rationale of this strategy is that in order to guarantee a SLA for a traffic flow or a flow aggregate, the provider must reserve a suitable amount of resources
- This amount of resources depends on both TCAs and SLAs of connections sharing the capacity of a link
- Resources are allocated for IN traffic
- Once resources are allocated, accepting OUT traffic is a risk, because OUT traffic may consume resources allocated for other flows, whose SLA may be in turn violated
- Accepting OUT traffic in the network is a delicate issue; it can be done, but marking such traffic in order to drop it if it degrades other SLAs

Traffic Conditioning Agreement and Service level Agreement

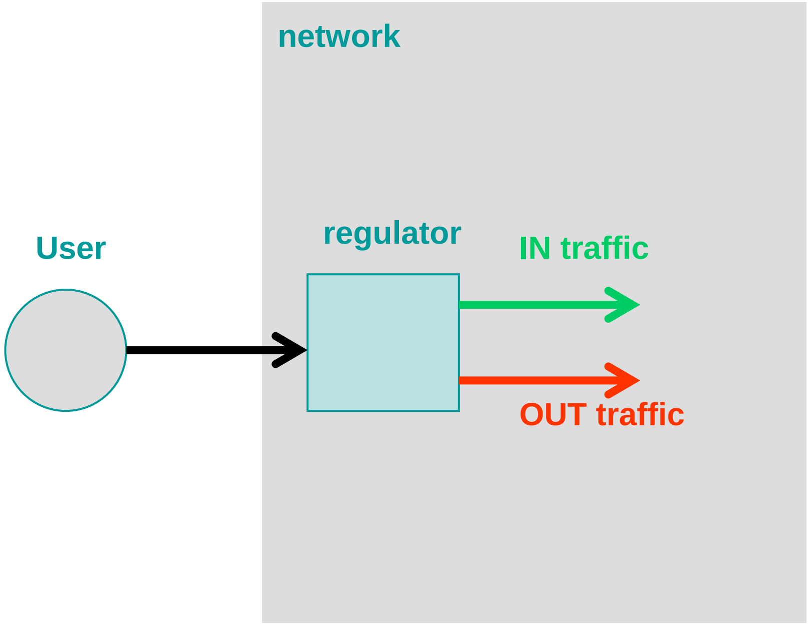
- However, the provider in general would like to transport as much traffic as possible (for example, billing could be based on the volume of transported traffic, among other billing metrics)
- Thus, if the network is not congested, the provider may want to transport OUT traffic in order to use momentarily free resources
- OUT traffic may be
    - Marked (i.e., transported with a higher dropping priority in case of congestion)
    - Transported with lower service priority

Traffic conditioning agreement

- The TCA can include a variety of parameters in order to characteriza IN and OUT traffic
- Such parameters usually include
  - Peak rate of traffic
  - Average rate of traffic
  - Maximum length of bursts (i.e., the maximum number of consecutive packets transmitted at the peak rate of traffic)
  - Maximum length of packets
  - Minimum length of packets
- The TCA specifies the statistical profile of IN traffic, that is, of the traffic for which the SLA must be fulfilled

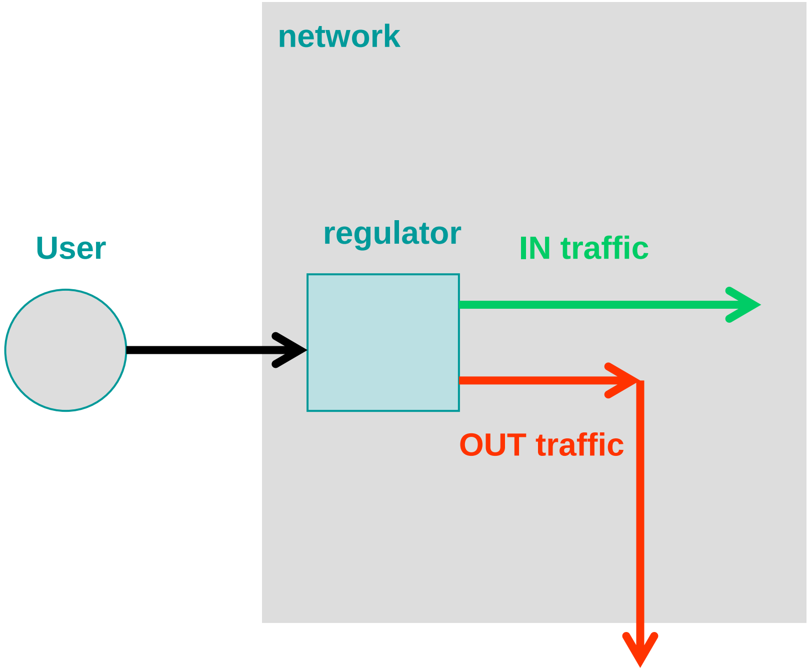
Traffic conditioning agreement and regulators

- Once the TCA and the associated SLA are established, the user-generated traffic is examined, at the network ingress, by a *regulator*
- The regulator splits the user's traffic into (at least) two logically separated flows
  - IN traffic (referred to also as green)
  - OUT traffic (referred to also as red)
- There exists also regulators splitting traffic into three logically separated flows, i.e., green, yellow, and red
- In the figure, a *two-color* regulator is depicted
- If also yellow is distinguished, the regulator is called *three-color*



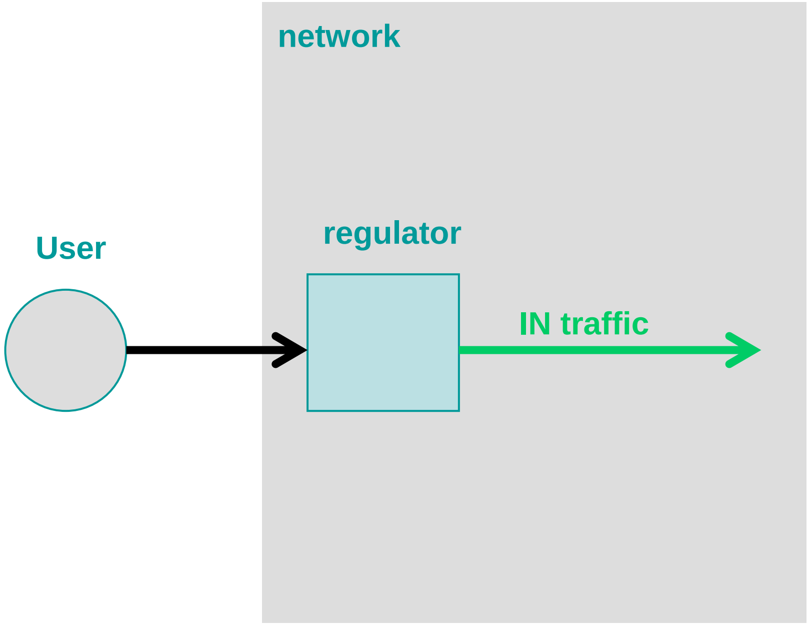
Policers

- Policing regulators (policers) drop OUT traffic
- Only green traffic proceeds into the network
- In this way, other already established traffic flows are always protected from excess traffic of other users
- However, if the network has spare resources, these resources are not used



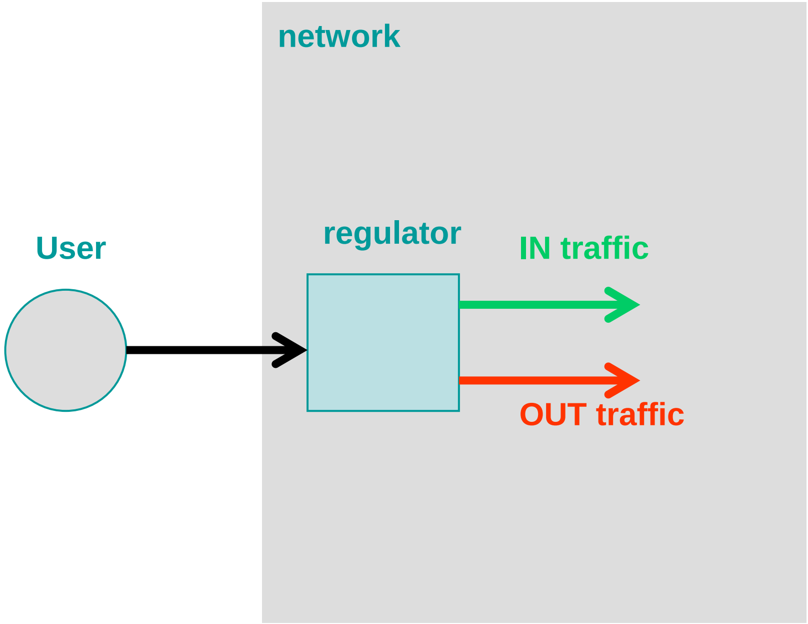
Shapers

- Shaping regulators (shapers) delay OUT traffic in a buffer, inside the regulator, in such a way it is transmitted into the network only when it is possible to do it without exceeding the TCA
- Traffic entering the network is always green
- Also in this case, other already established traffic flows are always protected from excess traffic of other users
- Also in this case, if the network has spare resources, these resources are not used as OUT traffic is delayed, however, resources are in general used more efficiently than with a policer
- The traffic backlog and delay inside the regulator may become large



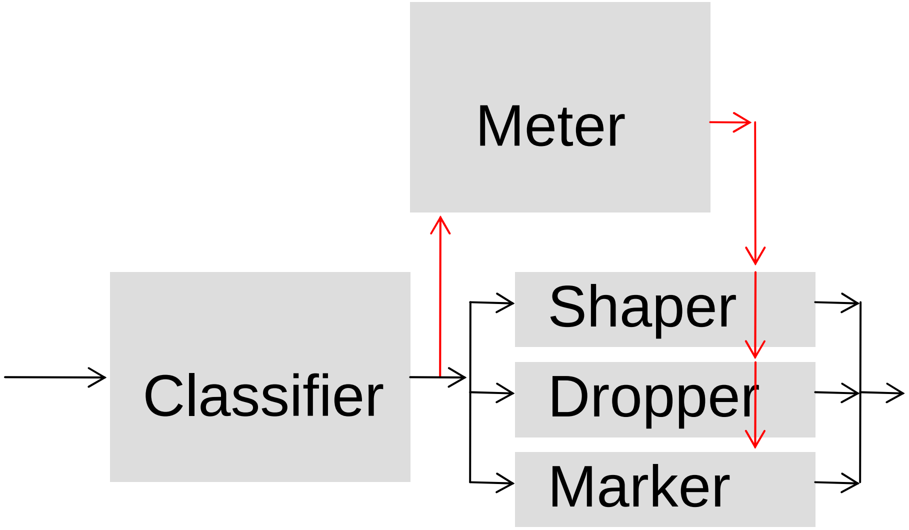
Markers

- Markers let OUT traffic proceed, but this traffic is marked
- Marking can be done in such a way to increase the dropping priotity of this traffic component
- Alternatively, the SLA of this traffic may be downgraded, for exampled, it could be forwarded as Best-Effort
- If the network has spare resources, they can be used more efficently
- However, the issue of congestion management becomes important



Traffic conditioner

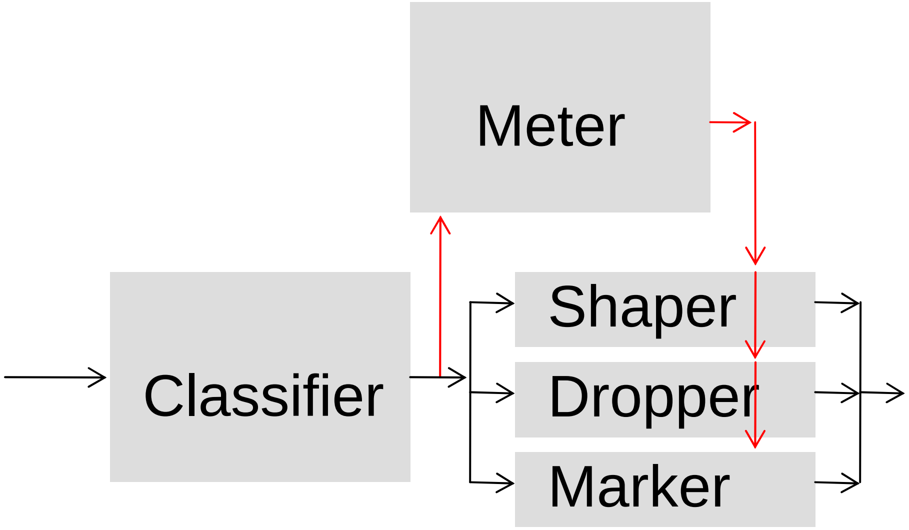
- A traffic conditioner may contain the following elements: meter, marker, shaper, and dropper
- A traffic stream is selected by a classifier, which steers the packets to a traffic conditioner
- A meter is used to measure the traffic stream against a traffic profile and the result of metering can affect a marking, dropping, or shaping action
- The figure shows the block diagram of a classifier and traffic conditioner



- *Black lines represent the flows of packets*
- *Red lines represent control information*

Traffic conditioner

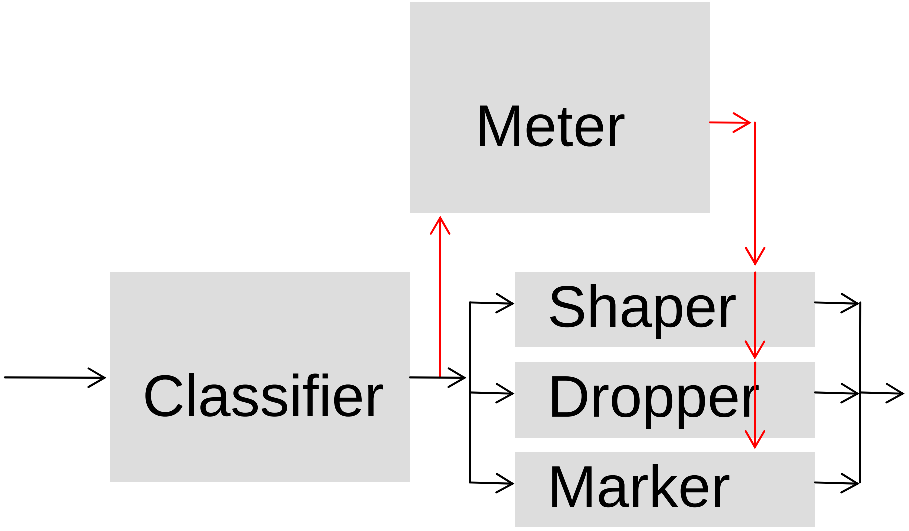
- The traffic meter measures the temporal properties of the stream of packets selected by a classifier against a traffic profile specified in a TCA
- The meter passes state information to other conditioning functions to trigger a particular action



- *Black lines represent the flows of packets*
- *Red lines represent control information*

Traffic conditioner

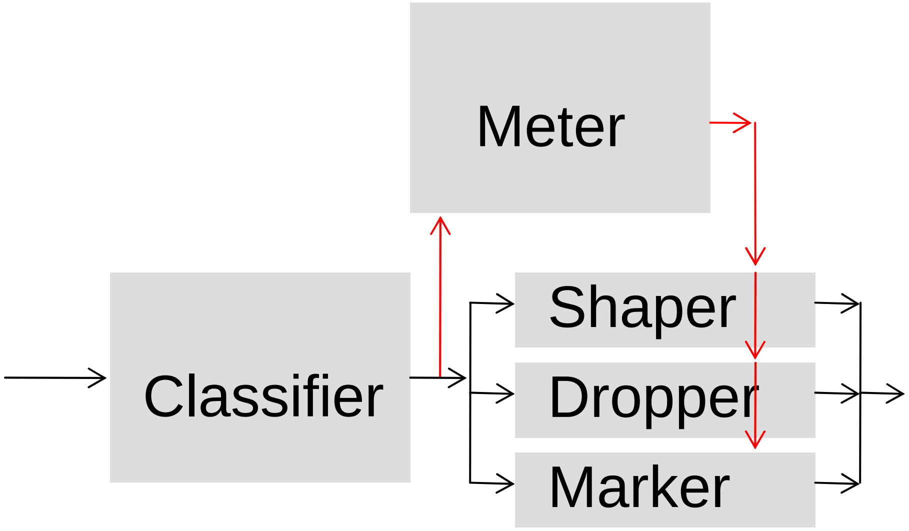
- The marker sets the DS field of a packet to a particular codepoint, adding the marked packet to a particular DS behavior aggregate
- The marker may be configured to mark all packets which are steered to it to a single codepoint
- Alternatively, it may be configured to mark a packet to one of a set of codepoints used to select a PHB in a PHB group, according to the state of a meter
- For example, OUT packets may be re-marked and assigned to an "inferior" PHB



- *Black lines represent the flows of packets*
- *Red lines represent control information*

Traffic conditioner

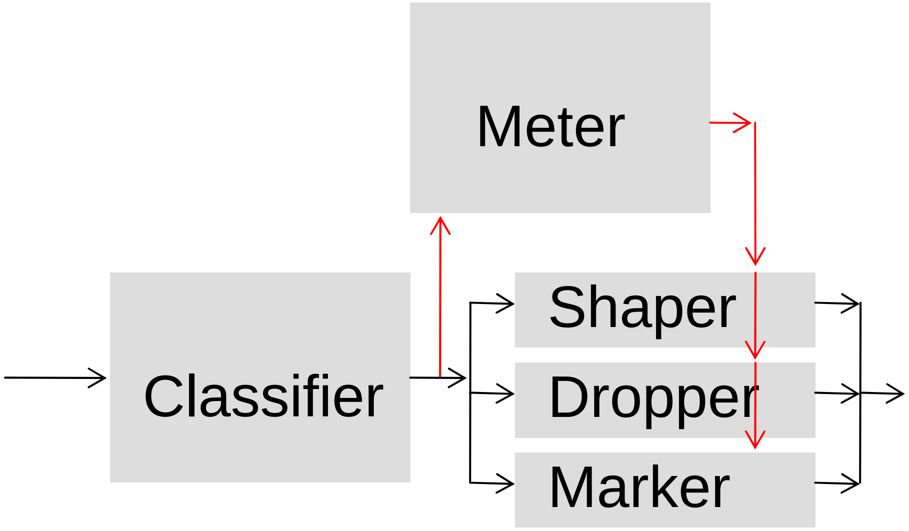
- The shaper delays some or all of the packets in a traffic stream in order to bring the stream into compliance with a traffic profile
- A shaper usually has a finite-size buffer, and packets may be discarded if there is not sufficient buffer space to hold the delayed packets
- Actually, practical shapers can be referred to as "shapers-droppers"



- *Black lines represent the flows of packets*
- *Red lines represent control information*

Traffic conditioner

- Droppers discard some or all of the packets in a traffic stream in order to bring the stream into compliance with a traffic profile
- This process is know as "policing" the stream
- Note that a dropper can be implemented as a special case of a shaper by setting the shaper buffer size to zero (or a few) packets



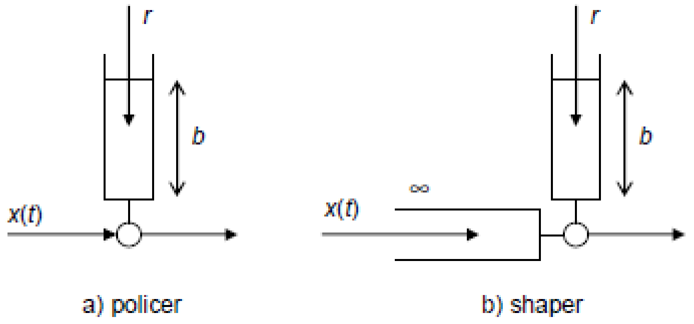
- *Black lines represent the flows of packets*
- *Red lines represent control information*

Admission control

- The difficult problem of admission control is determining the admission region, which depends of the statistical characterization of traffic, on the link's capacity, on the SLAs and on the scheduling policy
- The difficulty comes from the complexity of the traditional analytical methods used to carry out the calculation
- The difficulty increases when multi-hop paths are considered (this is the most realistic and common case)
- We will see that with the most recent methods of the statistical network calculus, this problem can be solved in a **relatively** easy way

Policing and shaping

- The token bucket policing regulator (Figure a) has a counter of credits (tokens) with maximum value *b* [traffic units], an is referred to as *token bucket size*
- The unit measure of *b* can be bits, bytes or packets
- The credit counter is increased every 1/*r* s, where *r* is the token rate
- One traffic unit (bit, byte, or packet) of offered traffic is allowed to pass through the regulator if the counter is positive (then, the counter is decremented
- Otherwise, if the counter is equal to zero, the traffic unit is dropped

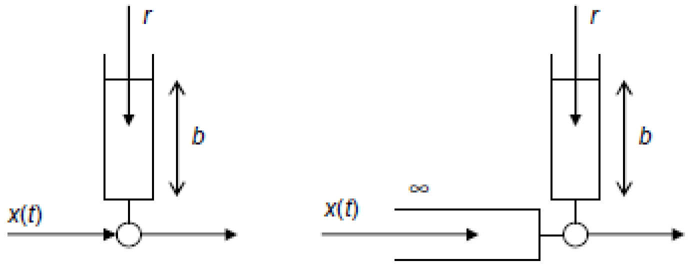


a) policer

b) shaper

Policing and shaping

- Figure b shows a shaping regulator
- The credit counter works as for the policer
- An incoming bit passes through the regulator instantaneously if, at its arrival, the counter is positive and the infinite input buffer is empty
- Otherwise, if the buffer is not empty and/or the counter is null, the incoming traffic unit is buffered
- When the input buffer is not empty, one traffic unit is fetched from the buffer as soon as a token is generated.
- The *r* and *b* parameters of both types of regulators have an intuitive physical meaning
- The r parameter controls the average rate of the through traffic, as the regulator cannot output more than *r* bit/s on the average
- The *b* parameter controls the length of output traffic bursts
- If the token counter is full (i.e., it traffic units at maximum rate
- Then, it must stop to wait further tokens

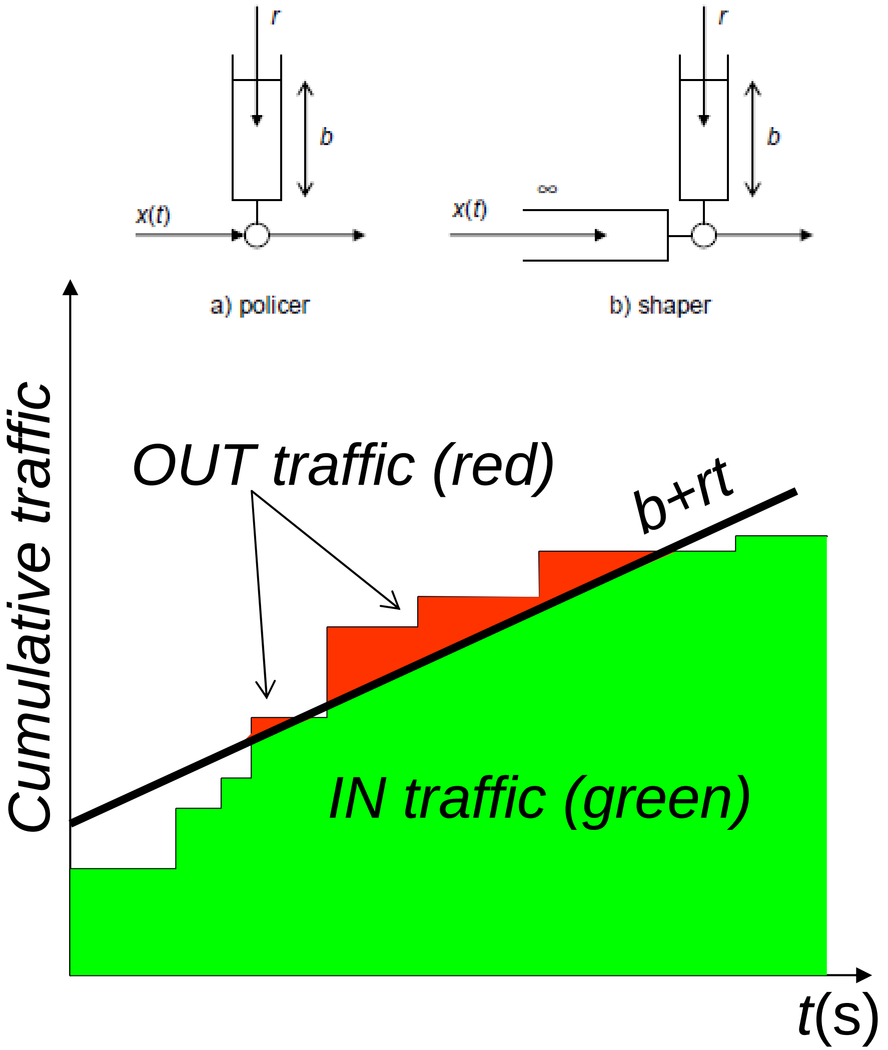


a) policer

b) shaper

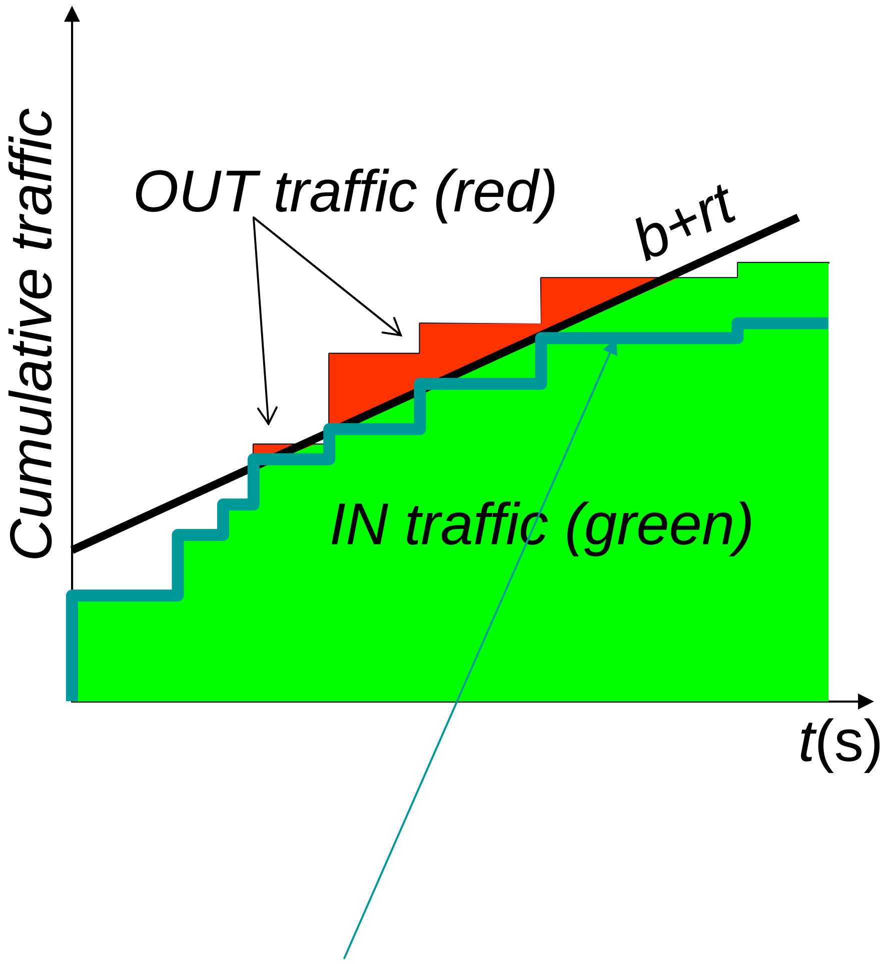
Policing and shaping: constraint functions

- Both regulators implement a *constraint function*
- This constraint function is the line *b*+*rt* and it represents the maximum number of traffic units, corresponding to IN traffic, that the regulator lets pass into the network
- In a time interval of duration *t*, the maximum In traffic that the regulator lets pass is equal to *b*+*rt*
- Excess traffic is OUT traffic
- This OUT traffic is treated differently by the policer and the shaper



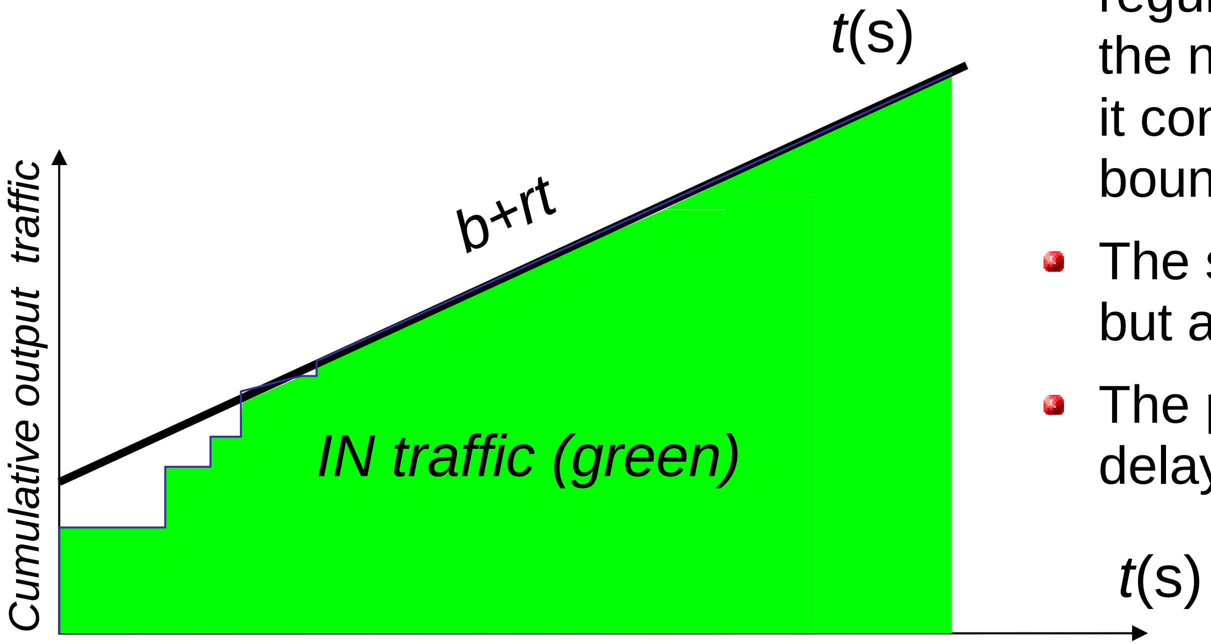
Token bucket policers

- A token bucket policers drops OUT traffic
- In this way, only In traffic enters the network
- The result is shown in the lower figure, representing the cumulative traffic entering the network
- The traffic pattern shown in the lower figure is a Linearly Bounded Arrival Process (LBAP)
- A token bucket policer assures that the traffic offered to the network is linearly bounded



IN traffic actually output by the policer

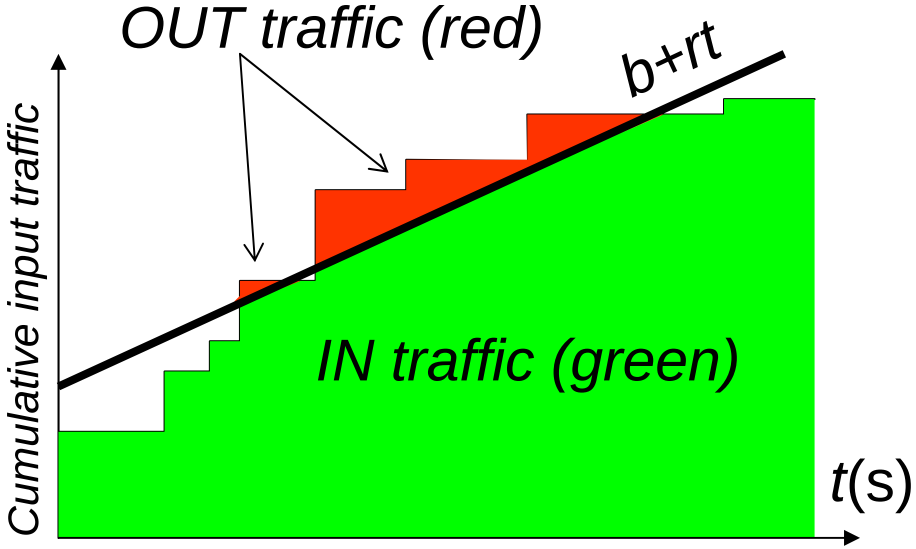
Token bucket shapers



- A token bucket shaper behaves diferently
- OUT traffic is delayed in the regulator's buffer and it is sent into the network when it is possible to do it complying with the LBAP upper bound on cumulative traffic
- The shaper eliminates packet loss but adds delay
- The policer has (practically) no delay but it introduces packet loss

Token bucket marker

- A token bucket marker works as the policer does, but it does not drop OUT traffic
- OUT traffic is marked and it is forwarded into the network
- Inside the network, as soon as congestion arises, marked packets are dropped before In packets


OUT traffic (red)
b+rt
Cumulative input traffic
IN traffic (green)
t(s)