# 5.2t Transcript

Integrated Services Architecture (ISA) and differentiated services (DiffServ)

These mechanisms serve a simple purpose, to allocate network resources efficiently to maximize effective capacity.

Let’s look at Integrated Service Architecture, ISA, first.

Integrated Services Architecture was defined in IETF RFC 1633, which proposed the resource reservation protocol, RSVP, as a working protocol for signalling in the ISA architecture. This protocol assumes that resources are reserved for every flow requiring Quality of Service at every router hop in the path between receiver and transmitter using end-to-end signalling.

The ISA model for IP QoS architecture defines three classes of service based on applications delay requirements from highest performance to lowest, these are Guaranteed-service class - with bandwidth, bounded delay, and no-loss guarantees. Controlled-load service class - approximating best-effort service in a lightly loaded network, which provides for a form of statistical delay service agreement that will not be violated more often than in an unloaded network.

And Best-effort service class – basically what the Internet currently offers, which is further partitioned into three categories, interactive burst, and example would be Web, interactive bulk something like FTP and asynchronous, e-mail would be an example of this.

The main point is that the guaranteed service and controlled load classes are based on quantitative service requirements, and both require signalling and admission control in network nodes. These services can be provided either per-flow or per-flow-aggregate, depending on flow concentration at different points in the network. Although the ISA architecture need not be tied to any particular signalling protocol, Resource Reservation Protocol is often regarded as the signalling protocol in ISA. Best-effort service, on the other hand, does not require signalling.

RSVP

Using a method similar to the switched virtual circuit (SVC) of asynchronous transfer mode (ATM) networks, ISA uses RSVP between senders and receivers for per-flow signalling.

RSVP messages navigate across the network to request and reserve resources. Routers along the path, including core routers, must maintain a temporary soft state for RSVP flows, this state periodically expires resource reservation, so no specific ‘tear down’ is required. The soft state is periodically refreshed by RSVP messages.

RSVP is a set-up protocol providing a receiver-based, guaranteed, end-to-end Quality of Service pipe.

In order to create a reserved pipe a PATH messages flow from the sender downstream to discover the data path, this is then responded to be a RESV message, originating from a receiver and traveling in the reverse direction of the PATH messages. This attempts to set local Integrated Services standard reservation for the flow. Each node along the path may either admit or reject the reservation subject to capacity or policy admission controls.

The major advantage of ISA is that it provides service classes, which closely match the different application types and their requirements. For example, the guaranteed service class is particularly well suited to the support of critical, intolerant applications.

Controlled load services class can generally support critical, tolerant applications and some adaptive applications and other adaptive and elastic applications are accommodated in the best-effort service class.

Apart from the further sub-classification of best effort service class ISA leaves best-effort unchanged, so it does not involve any change to existing applications. This is an important property since ISA is then capable of providing this class of service as efficiently as the current Internet. ISA also leaves the forwarding mechanism in the network unchanged.

This allows for an incremental deployment of the architecture, while allowing end systems that have not been upgraded to support ISA to be able to receive data from any ISA class.

Differentiated Services

The differentiated services architecture or DiffServ as it is known, originates in Request For Comments (RFC) 2475. It is designed to provide a simple, easy-to-implement, low-overhead tool to support a range of network services that are differentiated on the basis of performance.

There are several key characteristics of DiffServ that contribute to its efficiency and ease of deployment

These key characteristics are IP packets are labelled for differing QoS treatment using the existing IPv4 or IPv6 DSField. This means there is no change is required to IP. A big plus given IP dominance.

There is a service level specification (SLS) established between the service provider and the customer prior to the use of DiffServ.

This avoids the need to incorporate DiffServ mechanisms in applications. This gives the benefit that existing applications need not be modified to use DiffServ. The service level specification is a set of parameters and their values that together define the service offered to a traffic stream by a DiffServ domain.

A traffic conditioning specification (TCS) is a part of the SLS that specifies traffic classifier rules and any corresponding traffic profiles and metering, marking, discarding/shaping rules which are to apply to the traffic stream.

Differentiated services provides a built-in aggregation mechanism, this provides for good scaling to larger networks and traffic loads.

Basically, all traffic with the same DiffServ octet is treated the same by the network service. So, for example, multiple voice connections are not aggregate together as opposed to handled individually.

DiffServ is implemented in individual routers by queuing and forwarding packets based on the DiffServ octet. Routers deal with each packet individually and do not have to save state information on packet flows.

The DiffServ type of service is provided within a DiffServ domain, a differentiated service domain is defined as a contiguous portion of the Internet over which a consistent set of DiffServ policies are administered, usually under the control of one administrative entity.

The services provided across a DiffServ domain are defined in a Service Level Agreement, which will specify the forwarding service that the customer should receive for various classes of packets. A customer can be a user organization or another DiffServ domain.

Under the SLA, the customer submits packets with the DiffServ octet marked to indicate the packet class. The service provider must ensure the agreed QoS for each packet class.

The service provider must ensure that the level of service, the level of quality, is maintained. To do this the provider must configure the appropriate forwarding policies at each router, based on DiffServ octet value. This must be done on an on-going basis to ensure that the SLA is met.

When a customer utilises the network within the DiffServ domain; that is to say destinations within the DiffServ domain, then the agreed level of service is expected.

If packets are destined for outside of the DiffServ domain, then the DiffServ domain will attempt to route packets through other domains requesting the most appropriate service.

The SLA will cover specific issues to the DiffServ framework, including Traffic Conditioning Agreements (TLA), Service Level Specifications, SLS and Traffic Conditioning Specifications, TCS. It is important to note because DiffServ are unidirectional the two directions of flow across the boundary will need to be considered separately.

From the original IETF DiffServ draft, these were specified as:

The TCS specifies detailed service parameters for each service level.

Such parameters include:

1. Detailed service performance parameters such as expected throughput, drop probability, latency.

2.Constraints on the ingress and egress points at which the service is provided, indicating the `scope' of the service.

3.Traffic profiles which must be adhered to for the requested service to be provided, such as token bucket parameters.

4.Disposition of traffic submitted in excess of the specified profile.

5.Marking services provided.

6.Shaping services provided.

In addition to the details in the TCS, the SLS may specify more general service characteristics such as:

1.Availability/Reliability, which may include behavior in the event of failures resulting in rerouting of traffic

2.Encryption services

3.Routing constraints

4.Authentication mechanisms

5.Mechanisms for monitoring and auditing the service

6.Responsibilities such as location of the equipment and functionality, action if the contract is broken, support capabilities

7.Pricing and billing mechanisms

At the start of talking about differentiated services, I mentioned the DSField. This field is part of the IPv4 and IPv6 header.

The codes are divided into three pools, they are designated by the least significant bits, any code with the first five bits as either zero or one, but ending in zero is reserved for Standards action. Any with the first 4 bits as either one or zero and the final 2 bits are one, one are reserved for experimental or local use.

The final pool, codepoints ending in zero, one are also reserved for experimental or local use but may be allocated for future standards action as needed.

The next field along on both IPv4 and 6 headers is the ECN, Explicit Congestion Notification. ECN uses two bits to encode four different code points:

00 – Non ECN-Capable Transport, Non-ECT

10 – ECN Capable Transport, ECT(0)

01 – ECN Capable Transport, ECT(1)

11 – Congestion Encountered, CE.

It’s worth noting that outside of the direct reading, anything else you read you may come across the term Type of Service, TOS.  RFC 2474 reserved the first six bits of the IPv4 ToS field for the Differentiated Services Code Point (DSCP), and RFC 3168 reserved the last two bits for Explicit Congestion Notification.

We’ve covered a lot about QoS, however there is a growing realisation that QoS by itself is not fully sufficient in providing performance guarantees, because they do not take into account the user perception of network performance and service quality.

It is this that has led to the evolving field of Quality of Experience, QoE.

Informally, QoE refers to the user perception of a particular service, this can be dependent upon the device being used, different screen size, bandwidth capabilities, codecs and so on.

The perception of the quality of experience is subjective and therefore is difficult to quantify, it requires a multidisciplinary approach, bringing together experts in the fields of cognitive processes, multimedia signal processing, and social psychology, and people like us. Focusing on understanding the user perception of quality.

There are models emerging that bring QoE and QoS together, one such model is the QoE/QoS layered approach.  There are four layers in this approach, User, Service, Application and Network.

The User interacts with the service. It is their degree of enjoyment from using the service that is to be measured.

Being linked to human perception, QoE is hard to describe in a quantitative way, and it varies from person to person. The complexities of QoE at the user level stem from the differences between individual user characteristics, these could include gender, age, attitudes, prior experience, expectations, socio-economic status, and many others.

Therefore, it becomes a challenge to derive unified QoE metrics for all users and their contexts.

The service level provides a virtual level where the user’s experience of the overall performance of the service can be measured. It is the interface where the user interacts with the service.

It is also where tolerance thresholds are measured. An example of quality of experience from the user perspective for streaming applications could be start up time, audio/visual quality, channel change delay, and buffering interruptions.

However, the QoE measures for web browsing applications could be page load waiting times.

There have been many surveys over the last few years showing that the average person grows frustrated after waiting 16 seconds for a webpage to load, 25 seconds for traffic signal to change.

The first two layers are primary concerned with the Quality of experience, the next two layers are primarily concerned with the Quality of Service. It should be noted that this model includes QoS as part of the QoE Domain.

Application-level deals with the control of application-specific parameters such as content resolution, bit rate, frame rate, colour depth, codec type, layering strategy, and sampling rate.

The network capacity often dictates the bandwidth that will be allocated to a service for transmission.

The final layer is the Network- level, this level is concerned with the low-level network parameters such as service coverage, bandwidth, delay, throughput, and packet loss.

Much of this a Diffserv domain addresses.

Once we have knowledge that our user is satisfied or dissatisfied is there anything we can actually do?

In the last few years researches have been investigating and presenting various algorithms, to offer a metric for QoE/QoS. Obviously if we have some sort of measure, if our consumer is dissatisfied we can attempt to improve the overall QoE/QoS of our system.

One such system is Network Layer QoE/QoS Mapping Models for Video Services.

The network layer QoS/QoE mapping models rely solely on NQoS metrics gathered from the TCP/IP stack except for the application layer, remember back in week 1, transport, network, link, and physical layers. In a 2010 paper, Ketyko et al. proposed the following parameter-based quality model for estimating video streaming quality in 3G environment.

The equation in the slides shows the various metrics where AL and VL refer respectively to audio and video packet loss rates, AJ and VJ represent respectively audio and video packet jitter (VJ), and RSSI is the received signal strength indicator.

There are other algorithms that utilise the network layer QoS/QoE and there are also algorithms utilising the application QoS metrics.

How and which we decide to use in our network design depends upon many factors, what type of network are we delivering over, what application the user is engaging with. It is for us as the designer to ultimately make decisions on this and implement the best system we can.