Mohammad UniversityKheiderBiskra
Faculty of Exact Sciences and Natural and Life Sciences
Computer Science department

Master's course1
Option: RTIC

QUALITY OF SERVICE & MULTIMEDIA COMMUNICANTING SYSTEMS

By:Mrs. Boukhlouf

2023/2024

Course Map

- **Multimedia communicating systems**
  - Principles and architectures
  - Streaming systems and ToIP
  - RTP/RTCP, SIP, RTSP protocols
- **Quality of service**
  - Principles and mechanisms
  - Classification, scheduling, queue management, congestion control, admission control, routing with QoS
  - Protocols IntServ,DiffServ

1. Multimedia communicating systems

Definitions

- Multi: Indicates multiple
- Multimedia (computing): presentation of an application (generally interactive) which integrates elements such as text, graphics, video, sound.
- Media: means/support for the dissemination, distribution or transmission of signals carrying written, sound, visual messages (press, cinema, radio, TV, etc.)
- Multimedia system: Computer and associated software used to run a multimedia application.
- Distributed multimedia system: an SMD that operates on a set of equipment interconnected by a communication network.

Concepts related to*Multimedia*

Multimedia

Text
Image
Video
Audio
Voice
Data

MultimediaSystem

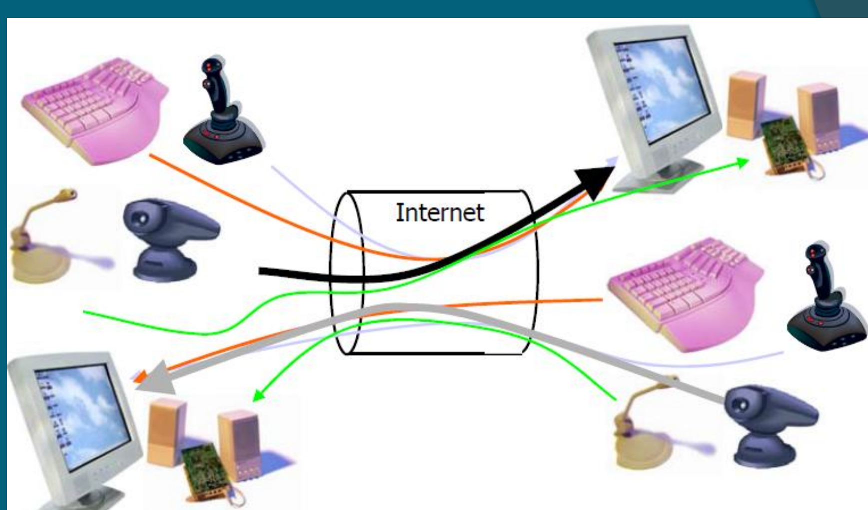
Media management
Security management
Different tools

Distributed Multimedia System

Network

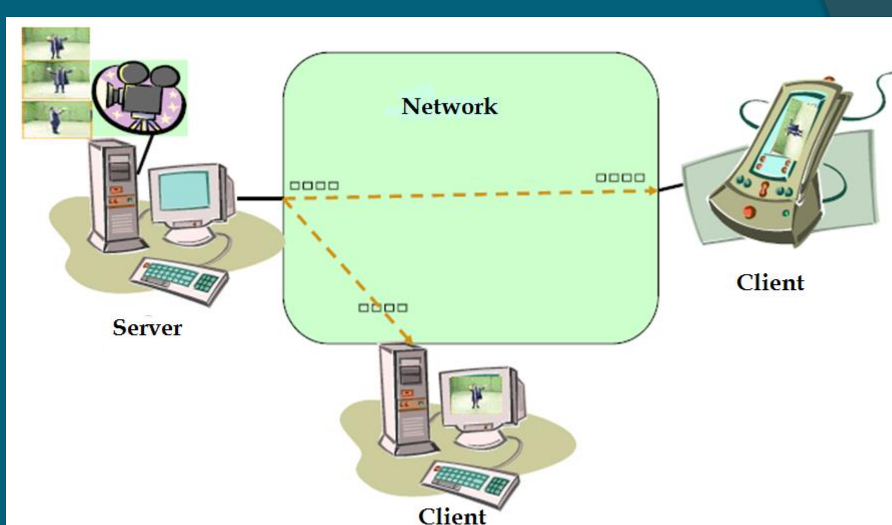
Video streaming
Audio streaming
Client server
Web
Others

Wired equipment for *Multimedia*



Wireless and mobile equipment for the*Multimedia*

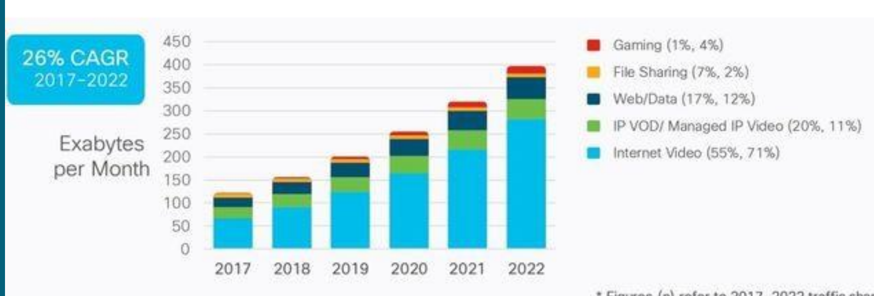
General principle of distributed multimedia applications



Classification of multimedia applications

- **According to the interactivity**
  - Non-interactive: radio and TV, video on demand, e-learning...
  - Interactive: video surveillance, remote control, video conference call,telemedecine,teleshopping,games…
- **According to the criticality**
  - (Very) critical: guidance and supervision, telesurgical operation…
  - (Average) reviews: video conference, stock market, teleshopping
  - Non-critical: TV, radio, games…
- **According to the timings (real time)**
  - *-Sstreaming of previously stored audio/video data*
  - *–Real-time 1-to-m streaming of audio-video data*
  - Interactive audio/video applications

Evolution of Internet traffic



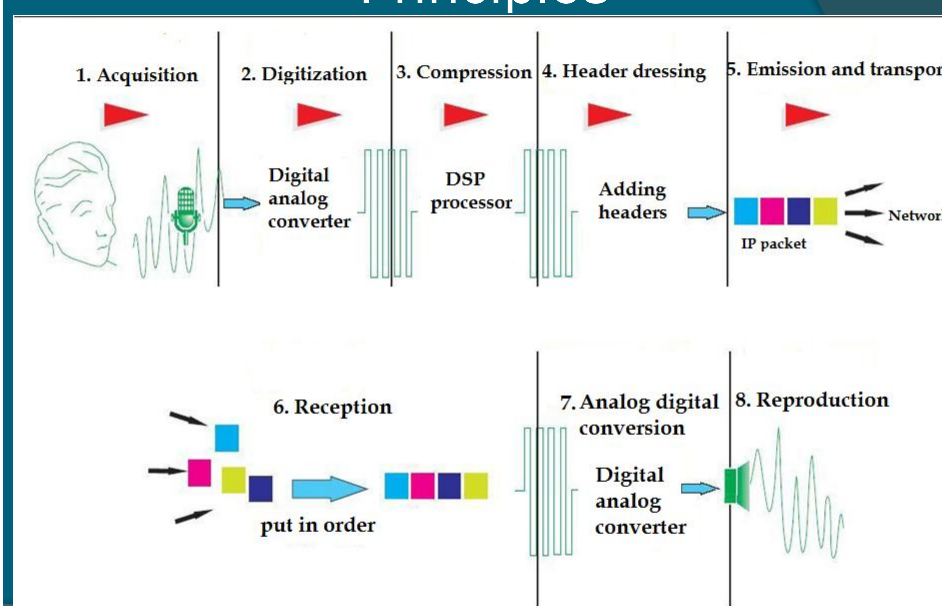
1Exabytes= 1018bytes

Source: Cisco VNI Global IPTraffic

Digitization and compression

- Audio/video support not necessarily digital
  – Digitize content
- Scanned data may be large in size
  – Compress
  – Codecs
- Compression/Decompression
- Choice of codec often imposed by network bandwidth
  – Lossless compression
  – Lossy compression

Principles



audio digitisation

- Pulse Code Modulation – PCM

- Compression techniques
- Voice
  - GSM (13 kb/s), G.729 (8 kb/s), G.723 (6.4 and 5.3 kb/s)
  - proprietary techniques
- CD quality music
  - MP3
  - 96, 128 and 160 kbps
  - splitting into independent files
  - Streaming
- Others: AAFC,Vorbis, …

Video digitization

- Video
- Sequence of images viewed at a certain bit rate
- Picture
- Suite of pixels
- pixels
- Luminance and color
- Encoded in a number of bits

Video compression

- Redundancies
  - Spatial redundancy
  - Temporal redundancy
- MPEG compression standards
  - MPEG 1 CD-ROM video quality – 1.5 Mb/s
  - MPEG 2 high quality DVD video – Digital TV – 3-6 Mb/s
  - MPEG 4 tt type of multimedia applications
  - Inspired by the JPEG standard
- Other standards
  - H.261, 262, 263, 264
  - Owners

QS by Mrs. Boukhlouf 15

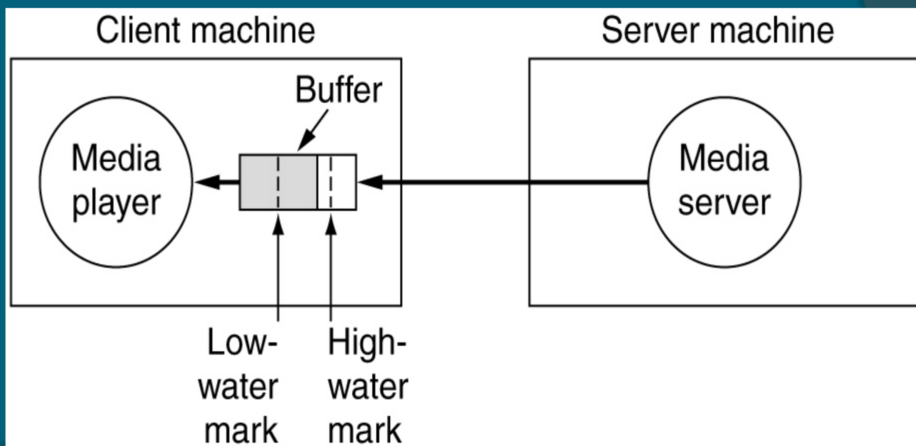
Audio/video streaming

- Definitions
- Play an audio/video stream as it is broadcast
  - No need to have downloaded the whole file
  - The download continues in the background
  - Temporary data storage
  - Alternative to download
- Stored
  - The requested file is previously stored on a server
  - eg. video on demand
- Real time / live
  - Similarity to broadcast radio/television
  - Real-time content processing and delivery

QS by Mrs. Boukhlouf 16

Audio/video streaming

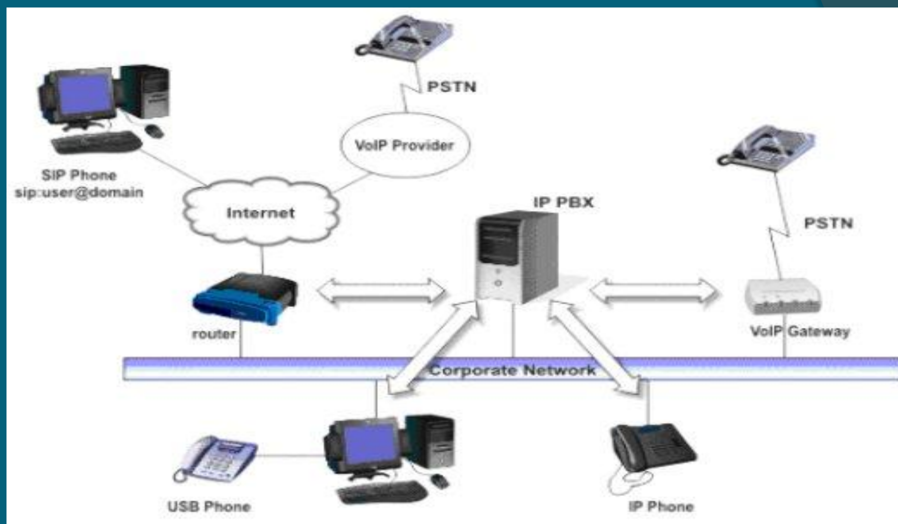
- The “media” client softwareplayer» puts the data in a buffer, and then plays it



Telephony over IP :ToIP

- Differences betweenVoIPand theToIP
- • Voice over IP
- – Transmit an audio signal in the IP world
- • Telephony over IP
- – Application of theVoIP
- – Telephone functions and services around theVoIP
- • which allow telephony
- – IP telephony architecture

Differences betweenVoIPand theToIP



Benefits of ToIP

- Users
- Cost
- Long distances
– Flexibility
- IP phone mobility
- Physical and material mobility
- Operators
– No strong regulation
– Management of a single network
- Voice – data
- Cost
- 60% of thebpallocated to a voice circuit (PSTN) not used

VOIP protocols

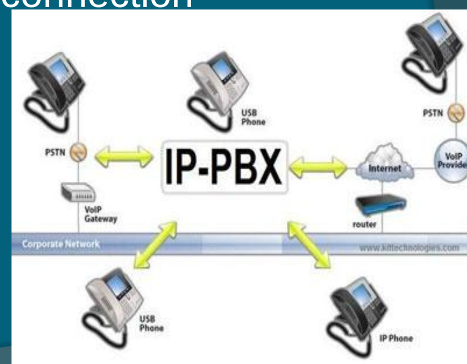
- The main protocols used for establishing connections invoiceover IP are:

Equipment forToIP

• Telephones
–Softphones
• Software to be installed on a system computer science
–Hardphones
• Conventional telephones with a socket ethernet
• Configuration files

Equipment forToIP

- PBXs
– IP-PBX (PABX –Private Automatic Branch exchange)
– Management and interconnection post offices
– Provision of services telephone
– Hardware / software



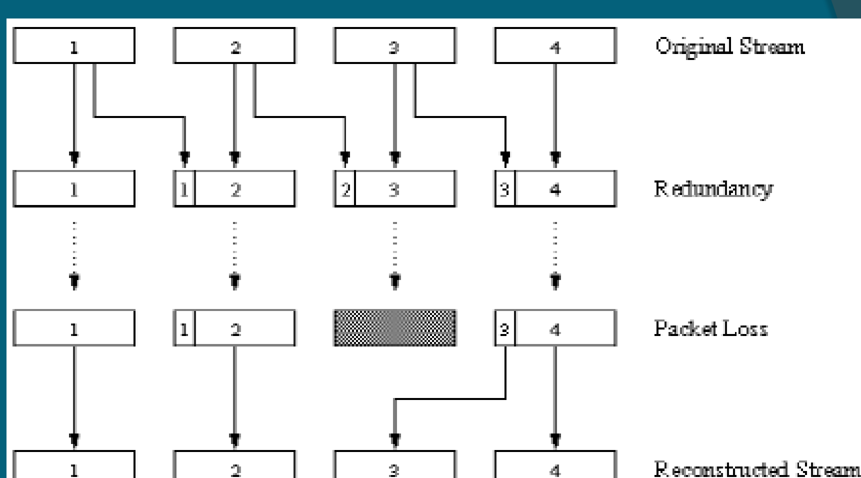
Principle of IP telephony

- Telephone communication: talk, silence, talk...
– Normally: it takes 64 kb/s during the speech phase
- Packets are generated only during speech phases
– Message = a piece of speech (of 160 bytes of data) + header
- Each message is encapsulated in a UDP segment.
- The application sends UDP segments via the UDP-socket every 20 ms during the speech phases. The sending rate is 8 kb/s.
- Up to 10% (or even 20%) packet loss is tolerable.
- Packets with a delay greater than 400 ms are discarded upon receipt.
- Jitter is managed by using packet timestamps, sequence numbers, and by delaying certain packets before they are listened to by the receiver.

Packet Loss Recovery

- As retransmissions are inappropriate in a real-time context, an overlay strategy must be put in place. In the case of IP telephony, two techniques are used to reduce the impact of losses: FEC (*Forward error correction) and Interlacing.*
- **Recovery by FEC**: Add redundancy information by mixing the values of several pieces in a packet.

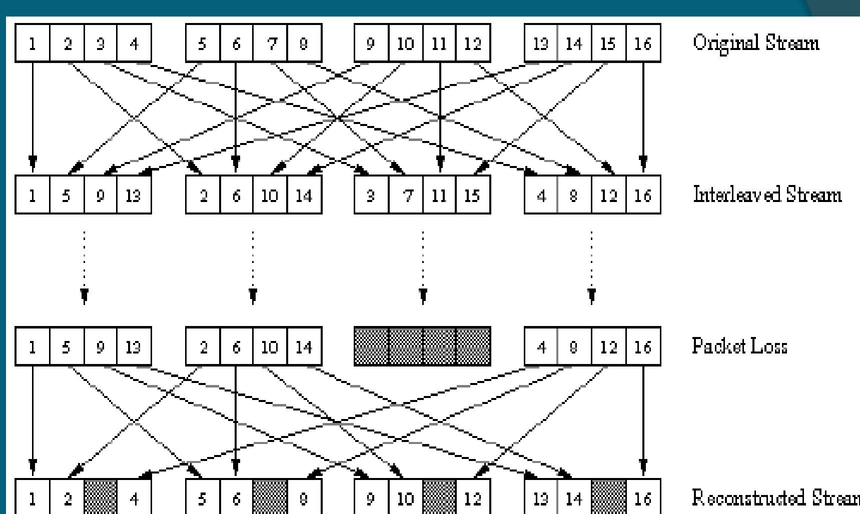
Recovery by FEC



Overlapping by interlacing

- No redundancy, but it may cause delays in the playout.
- Divide 20 msec periods of speech into smaller 5 ms periods and interleave the smaller chunks
- If lost, use incomplete pieces (rather than losing large whole pieces).

Overlapping by interlacing



Characteristics of multimedia applications

- Handling large amounts of 'continuous' data
- Minimum flow rates are required
- Delivery of information respecting timings
- Interactive applications require low round trip times
- Coexistence (and resource sharing) with non-media applications
- Resources required:

– Processors (high performance)
– Powerful servers
– Dedicated main memory (for buffering by the customer)
– Large capacity disk memory
– Network bandwidth with minimal latency

Multimedia application requirements

- Requirements: delay, jitter, throughput

- The required values change with the evolution of the technological offer:

We do not ask the same things for a 56 kb/s Internet connection as for a 10 Mb/s connection.

- The (human) user knows both how to be demanding and how to adapt to what is offered to him.
- Current demand trend: ever shorter lead times, ever higher throughputs, ever lower loss rates.

Multimedia application requirements

■ **Telephony and audio conferencing**

– Low throughput (~ 64 Kb/s), but delays should be short (< 250 ms)

■ Video on demand

– High throughput (~10 Mb/s), non-critical latency

■ **Video conferencing**

– High throughput for each participant (~1.5 Mb/s), low delay (< 100 ms), synchronized states.

■ **Distributed music rehearsal**

– High throughput (~1.4 Mb/s), very low latency (< 100 ms), high media synchronization (drift between sound and image < 50 ms)

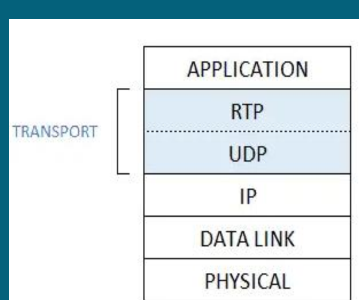
■ **Games**

– A maximum delay of 70 ms is more appreciated by gamers than a delay of 200 ms.

– The jitter should be 20 ms maximum, because the player adapts his strategy to a fixed delay (by shooting the targets for example). High jitter leads to boring gameplay.

Protocols for data transport multimedia Real-Time Protocol (RTP) (1/3)

- RTP: a solution for AMMs with Internet in best effort
- Basically works on top of UDP



Real-Time Protocol (RTP) (2/3)

RTP Header

RTP packet header

– Type of stream (7 bits)

– Sequence number (16 bits): used to detect losses.

– Timestamp (32 bits): Provides the sampling instant of the first byte of the packet.

It is used to absorb jitter.

– Sync Source Identifier (32 bits): identifies the source of the stream.

Each stream in RTP has a source-assigned identifier randomly (but distinct from those that already exist) at the start of the stream.

Real-Time Protocol (RTP)(3/3)

Some types of audio streams supported by RTP

Stream type	Audio Format	Sampling	Rate
0	PCM	8 KHz	64 Kb/s
1	1016	8 KHz	4.8 Kb/s
3	GSM	8 KHz	13 Kb/s
7	LPC	8 KHz	2.4 Kb/s
9	G.722	8 KHz	48-64 K/ps
14	MPEG Audio	90 KHz	----
15	G.728	8 KHz	16 Kb/s

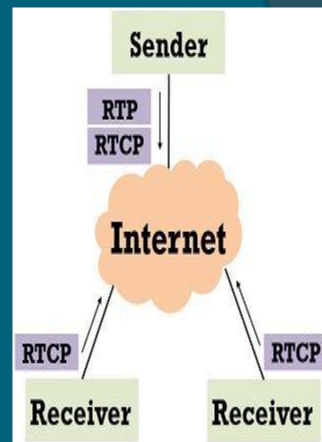
Some types of video streams supported by RTP

Stream type	Video Format
26	Motion JPEG
31	H.261
32	MPEG1 Video
33	MPEG2 Video

Real-Time Control Protocol (RTCP)(1/2)

- RTCP is used to route packets containing reports about a media stream between a source and a receiver.

- Reports contain statistics on: number of packets transmitted, number of packets lost, transfer jitter, etc.
- Report packets are sent by receivers, possibly at the request of sources.
- Report packets are used by the source to modify/adapt its timing to network conditions.



Real-Time Control Protocol (RTCP) (2/2)

- If each receiver sends its report packets to all the other sources/receivers of the stream: significant network overload.

- RTCP adjusts the time intervals between reports based on the number of receivers participating in a stream
- Typically, the bandwidth used for RTCP is limited to 5% of the session bandwidth. This fraction is shared between the report requests issued by the sources (25%) and the reports issued by the receivers (75%)

– *Ts: period of transmission of RTCP packet by the source:*

$$T_s = \frac{Nombre\ de\ sources}{5\%*25\%*Bande\_passante\_session} *Taille\_paquet\_moyen\_RTCP$$

– *Tr: RTCP packet transmission period by a receiver:*

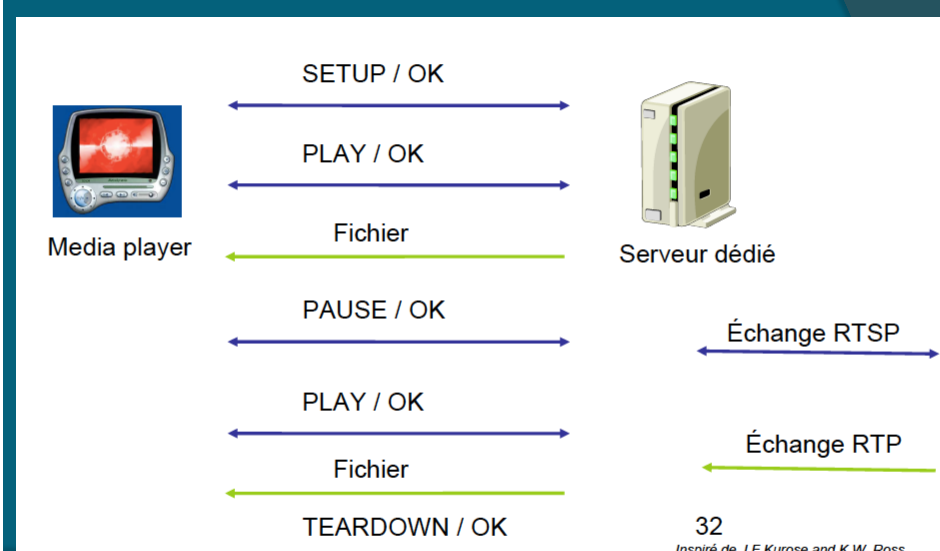
$$T_r = \frac{Number\ de\ récepteurs}{5\%*75\%*Bande\_passante\_session} *Taille\_paquet\_moyen\_RTCP$$

Signaling Protocols: RTSP, SIP.

RTSP: Real-time streaming protocol

- Client/server type application level protocol
- What it does not do
  - Choice of compression techniques
  - Choice of encapsulation
  - Choice of transport protocol
  - Choice of technique for buffering
- What he does
  - Help the media player to control the transmission of an audio/video stream

Flow send control



RTSP session

Session ID chosen by the server
- Used in every message
- History of the state of the client at the server
- Stateful Protocol
- RTSP over UDP or TCP

SIP
Session Initiation Protocol

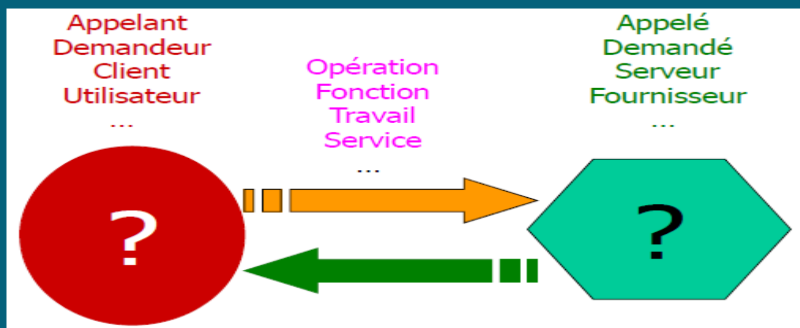
- Lightweight signaling protocol
- ModelClient server
-Mechanisms for establishing/terminating a call on an IP network
  - Prevent the called party from the call
  - Agree on encoding
  - End a call
-Mechanisms for determining which called party's IP address to use
- IP address not necessarily fixed
  - Mobility
  - Multi-terminals
- Call management
  - Changing encoding during a call
  - Invite other participants
  - Call transfer, etc.

SIP method

- Specified in the first bytes of SIP requests
- Indicates the purpose of the message
- GUEST
  - To initiate a session
- ACK
  - To confirm session establishment
  - Used with INVITE
- CANCEL
  - To cancel a pending INVITE request
- BYE
  - To end a session
- REGISTER
  - To register

2. Basic QoS Concepts and Mechanisms

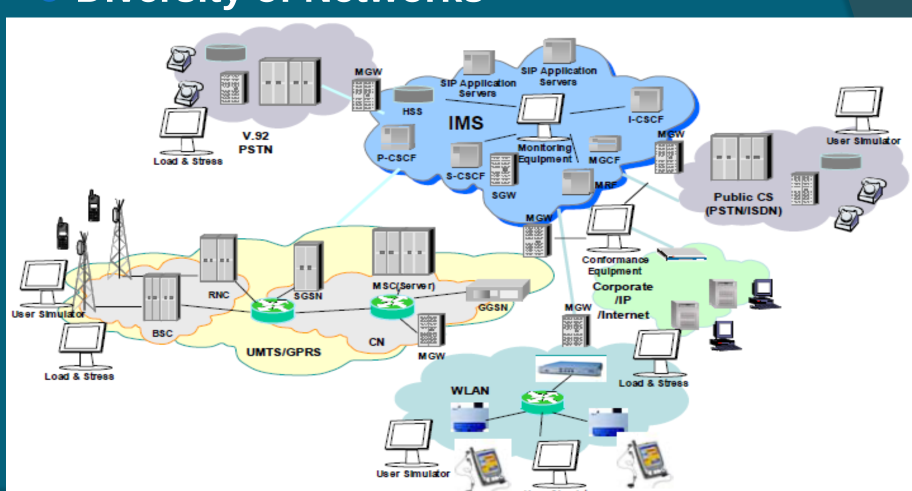
Extent of QoS



Quality
– Required, desired, implicit/explicit, agreed in advance, Trust…
– Measurable (Qualitative/Quantitative) or not
– Measurable online / offline

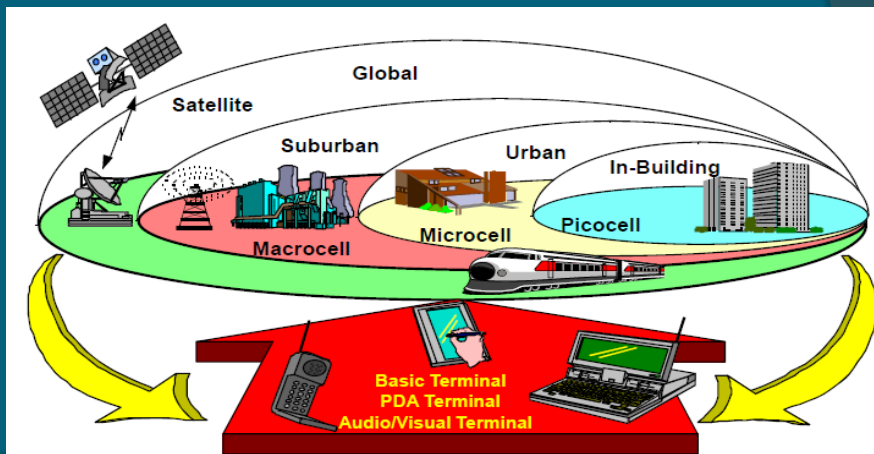
Extent of QoS

- **Diversity of Networks**



Extent of QoS

- Diversity of Networks



Extent of QoS

Diversity of Networks

- Public, private, dedicated networks…
- Wired - Wireless (radio waves, IR, Satellite)
- PAN, LAN, MAN, WAN
- Industrial, embedded, office automation…
- Site: practical work room, station, train, plane, car, etc.
- 1 domain, n domains
- Administration: centralized, distributed, autonomy
- Environment: mountain, tunnel, heat, humidity, radiation...
- Networks: invisible, intelligent, self-organizing, accessible everywhere…
- …

Extent of QoS

Diversity of the public concerned

 Types of stakeholders (users)

– Person, robot, sensor, software object…

– Computer scientist, automation engineer, telecom operator, access provider, military, industrialist… general public

– Very demanding ('the network is supposed to do everything') …., we take what the

network ('comprehensive')

– Accepts to negotiate, wants everything preconfigured…

– Accepts a cost: very high, …, modest, free

– …

 Stakeholder views

– Development of networks and services

– Content and its distribution

– Content and its use

– End-to-end transportation

– Transport on a domain, a router or an antenna

Extent of QoS

Diversity of Applications

 Sectors/areas of activity

– Control/supervision of nuclear power plants

– Military command organization

- Health

- Transportation

– Video surveillance, personnel identification

- Remote control

- E-commerce

– Leisure, Music, Games

– …

 Type of exchanges

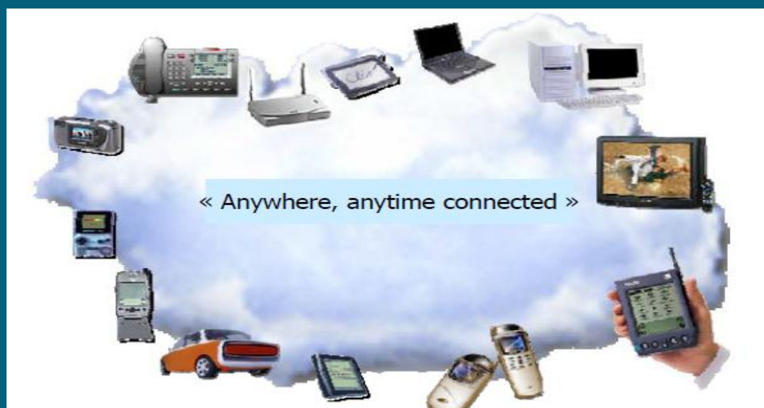
– Critics or not

– Applications: Transactional, Reactive, Interactive…

– 1 to 1, 1 to m, m to 1, n to m

Extent of QoS

- Diversity of Equipment



Extent of QoS

Diversity of Equipment

- General public, private, specialized
- Fixed, mobile (slow or fast mobility)
- Expensive/cheap
- Disappears after use ('sensor') or not
- Listening: always listening, dormant, transmitter, receiver...
- With battery constraints (rechargeable or not)
- Locatable: on demand, always, intelligently
- Smart equipment or not
- …

Extent of QoS

Difficulties talking about QoS

- **Multifaceted (time, security, cost, etc.)**
- **Different views (general public,…, IT specialist)**
- **Different levels (application, network, physical…)**
- **Different mechanisms and means**
- **Course limited to the Network (data transport)**

Concepts and definitions

Definition of ISO (International organization for Standardization)**and ITU-T [ISO/IEC 13236 - X.641; December 1997]**"It is a set of quality requirements on the collective behavior of one or more objects"

■**Definition of IETF (Internet Engineering task Strength)**

"Quality of Service refers to the manner in which the packet delivery service is provided, which is described by parameters such as bandwidth, packet delay, and packet loss rates"

■**Definition of QoS Forum**

"Collective measurement of the level of service provided to the customer. There QoS **may be** characterized by various basic performance criteria including availability, error rate, response time, connection establishment time, data throughput, connection or data loss due to network congestion and the speed of fault detection and correction"

Aspects related to QoS

- Expresses requirements on the behavior of a service provider
- Is expressed by different types of parameters (delay, service availability, etc.)
- Involves different levels of service (deterministic or otherwise)
- Requires the implementation of various mechanisms (reservation, control, etc.)
- Concerns both the network and the applications
- Concerns both different types of equipment and different layers

Classes (levels) of service

- Garantie absolue (déterministe)
- Probabiliste/stochastique/statistique
  - Prédictive, à charge contrôlée
  - Meilleure que le meilleur effort
  - 'Molle'
  - coercitive
- Meilleur effort
- Indifférent

Quel niveau choisir ?

C'est la nature de l'application qui permet de décider

Coût

Service categories

- **Application categories**

– Teleconference

- E-commerce

- Embedded systems

–…

■ Safe

■ Robust

- **Categories ofQoS**

■ Flexible

■ Easy to use

■ Low cost

■…

Parameters of QoS(1/4)

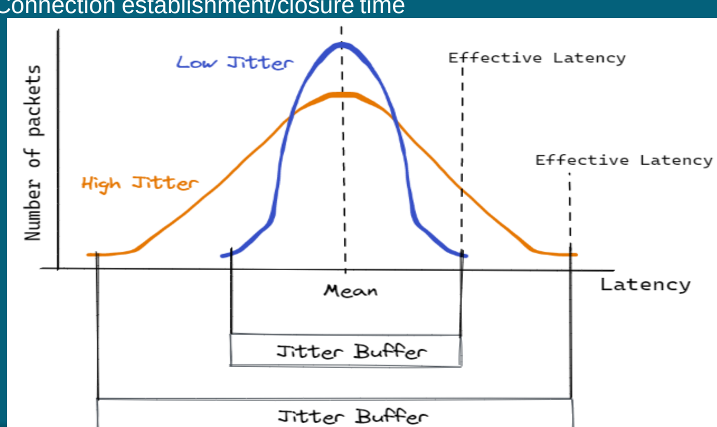
■**Temporal aspects**

– Transfer time, latency, delay

– Jitter

– Response time (round trip)

– Connection establishment/closure time



Parameters of QoS(2/4)

▪**Sensitivity to delay**

– Human beings do not like to wait in general (the applications they use must therefore take this into account). Telephone under current IP, videoconference: jerky sound and images…

– The controlled systems are sensitive to the delay (late reaction, a robot at the end of its travel, etc.).

▪**Jitter sensitivity**

– Human beings are sensitive to sound and image jitter

– Some control systems are sensitive to jitter.

Parameters of QoS(3/4)

▪**Volume**

– Bits/s, Packets/s

– Bandwidth percentage

–…

▪**Reliability/availability/robustness**

- Rate of availability

– MTBF(AverageTimebetween Failure), MTTR(AverageTime ToRepair)

–…

▪**Error settings**

– Error rate, loss rate

– Packet mess rate

– Connection establishment/closure error

Parameters of QoS(4/4)

■**Cost**
– Costs (€, others)
– Penalty, bonus, ...
■**Security**
– Access control capability
– Encryption capability
■ Others (ease of use,maintainability, simplicity, visibility, efficiency, extensibility, scalability, interoperability…)
– Types of attack taken into account and resistance capacity
– Additional cost of security mechanisms
– ...

QS by Mrs. Boukhlouf 59

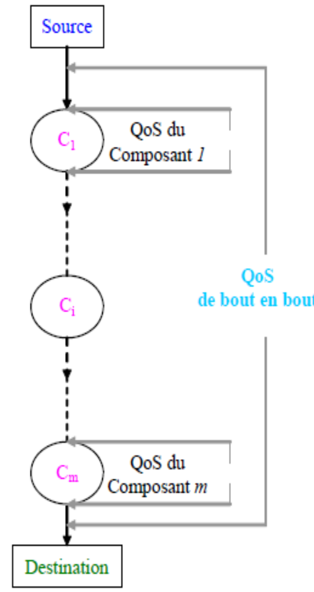
Forms of expression of QoS

■**Determinist**
– A value (delay < 10 ms)
– A range of values (delay in [80 .. 100])
■**Probabilistic**
– With a probability P (delay < 100 ms at 90%)
- More flexible
– Difficulty choosing the right probabilities
■**Statistical**
– Expression on the mean, variance, standard deviation
– Expression on the frequency
– Law of distribution
– Expression in fuzzy logic (e.g. ″high throughput″, ″acceptable delay″, ″reasonable jitter″)

QS by Mrs. Boukhlouf 60

Types of metrics QoS(1/2)

- **Additive**
  - $QoS(C_1 ; C_2) = QoS(C_1) + QoS(C_2)$
  - ex. Délai
- **Multiplicative**
  - $QoS(C_1 ; C_2) = QoS(C_1) * QoS(C_2)$
  - ex. Disponibilité
- **Concave**
  - $QoS(C_1 ; C_2) = \min\{QoS(C_1), QoS(C_2)\}$
  - ex. Débit
- **Autres**
  - Cas des spécifications non homogènes



Types of metrics QoS(2/2)

- **Ascending (increasing)**
  – if Val(QoS1) > Val(QoS2) then QoS1 is better than QoS2
  - ex. debit
- **Descending (decreasing)**
  – if Val(QoS1) < Val(QoS2) then QoS1 is better than QoS2
  - ex. time limit

Concept of Contract
SLA (Service Level Agreement)

- Connection-oriented management vs SLA-oriented management
- SLA = User-Provider Agreement
- Static or Dynamic SLA

■ SLA = {

QoS requise,
Spécification de trafic,
Règles de traitement de paquets,

Service Level Specification

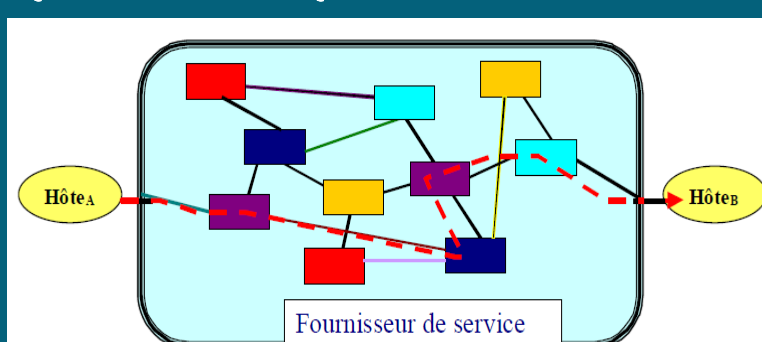
Coûts, pénalisation, bonus,
Aspects juridiques
…
}

QoS end to end

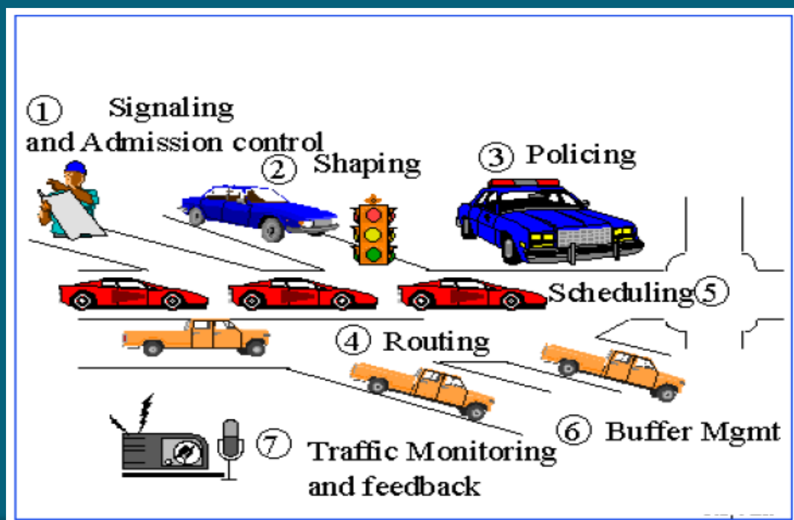
- **QoS end to end –QoS local**

– End user →QoS end to end

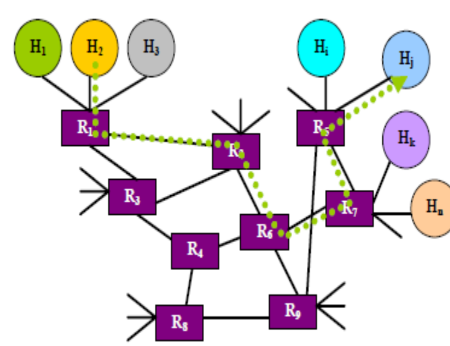
– Service provider → Decomposition of the QoS end to end: QoS local

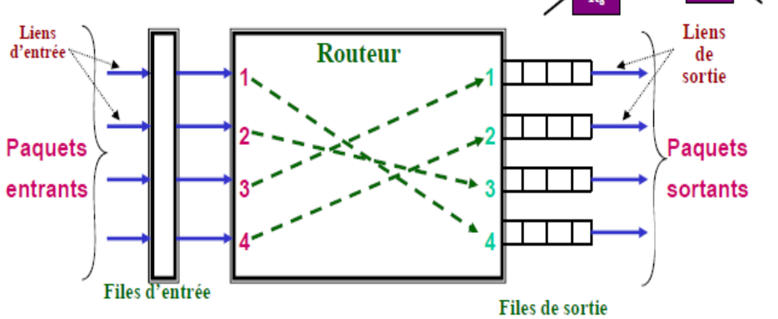


Overview of management functionsQoS



Fonctionnement simplifié d'un routeur/commutateur





Elements of end-to-end delay

- Waiting times in entry queues
- Package build times
- Switching times
- Waiting times in exit queues
- Transmission times
- Propagation delays

Negligible and/or constant delays

Traffic Patterns and Admission Control

Traffic characterization

- Periodic traffic: easy
- Aperiodic traffic

– Distribution of arrival times according to which law (fish, …)?
– Maximum avalanche size?
– Minimum avalanche duration?
– Avalanche size distribution?
– Distribution of lost messages?
– Correlation between packets (to allow loss)?
– Often difficult to model:
choice of parameters for "mathematical convenience"
– Still a lot to do to model random/sporadic traffic

Frequently used traffic models (1/2)

- **Periodic model**

■ Period, Maximum Packet Length

- **Model-1 with burst (Ferrari)**

■*Lpmax: maximum bundle length*
xmin(min time interval between two successive messages)
Xavier(average time interval between two successive messages)
I (time interval over whichXavieris calculated).

- **Model-2 with burst (Cruz)**

■ Average rate ρ and burst size σ:
Total number of packets generated never exceeds σ +ρ *Tin any interval T.*

- **Model 3 with burst (leaky Bucket)**

■ **Average bucket flow rate (ρ) and maximum bucket size (σ).**
Avoid bucket overflow.

- **Model-4 with Burst (Token Bucket)**

■ Average token generation rate (ρ) and maximum number of outstanding tokens (σ). The source can only transmit if it has tokens.

Frequently used traffic models (2/2)

- **IETF Traffic Model (RFC 2215)**

■ Specification using a ***TSpec:***
– Size σ and flow rate ρ of pierced bucket
Frequently used traffic models (2/2)
– Maximum flow p
– Maximum packet size M
■ Upper limit, A(T), of traffic per time interval T:
A(T) ≤ min(M +pT, σ +ρT)

- **Other models: probabilistic, stochastic,…**
- **CA cost and performance depend on traffic characteristics**

Admission control

Objective
– Can the new flow affect theQoSstreams already accepted?
– Can the node offer theQoSrequired by the new flow?
– Does the new flow have the right to use the resources of the node?
– Do all the nodes to be crossed accept the new flow?

 Information Used
– Characteristics of new traffic and theQoSrequested
– Network status and history
 The CA can be done on the basis of connection or SLA
– End dates of accepted traffic
– Possible disturbances of theQoStraffic already accepted
– Resource Usage Policy

Properties
(to be taken into account during the design of a CA)

 Incremental decisions (not always considering all flows)
 Accuracy (complicated due to random phenomena)
 Complexity
– Problem of the diversity of flow models
– Online use without significant additional cost
 Flexibility
– Problem of the diversity of flow models
 Scaling up

Statistical admission control

Why do we need it?

– Most streams are rather random

– Avoid oversizing by rejecting flows that could be accepted if we pay a little more attention to the allocation of resources

 Risks of using statistical CA

– Appearance of congestion situations

– Degradation ofQoS

– Consequence: Statistical CA not suitable for critical applications

 Difficulties of use: mastery of probabilities/statistics

QS by Mrs. Boukhlouf 73

Types of statistical CA

–CA based on average and maximum throughput

–CA based on cumulative effective bandwidth

–AC based on loss curve engineering

–CAs based on maximum variance

–AC based on the theory of large deviations

–Other types

QS by Mrs. Boukhlouf 74

Admission control based on metrics

- **If the flow characteristics are little variable**
  - Using peak and average demand to accept flow
  - Final decision and booking
- **If the flow characteristics are little or not known (traffic imprecision)**
  - Use an initial traffic estimate and reserve resources
  - Carry out traffic measurements and adjust reservations based onD-estimating traffic
  - Accept more streams
  - Cost of measures and actual effectiveness of adjustments
- Problems
  - What should be measured? When ? Or ?
  - How to progressively define traffic patterns?
  - How to evaluate the contribution in relation to the turnover without measurement?

Packet routing

Set Routing ToQoS(*QoS-based routing)*

- “A routing mechanism in which the paths that flows must take are determined by taking into account both knowledge about the availability of network resources and the requirements of QoS flow." [RFC 2386]
- “A dynamic routing protocol that extends its path selection criteria to include parameters for QoS such as available bandwidth, link usage, computing resources of knots, delay and jitter.” [QoSForum]

Functions

- Collect network status information (vital and complex function)
- Find the best path for a new flow based on the QoS required
- Change of path with progressive degradation of theQoS
- Optimize resource utilization and other criteria

Path selection

- On demand (for each packet, for each flow, etc.)
- Periodically and storage in a table
- Pre-calculation based on Bellman-Ford algorithms and Dijkstra(effective for large networks in particular)

Collection and dissemination of information

- On demand or periodic (very simple but can be expensive)
- Every time something changes ('ideal', very accurate, but very expensive)
- Threshold-based policies

– Trigger update if:|NewValue–OldValue| > threshold.

+ Avoid Update useless

- Slow reaction for status changes at many points

- Class-based policies

– Link capacities are subdivided into classes

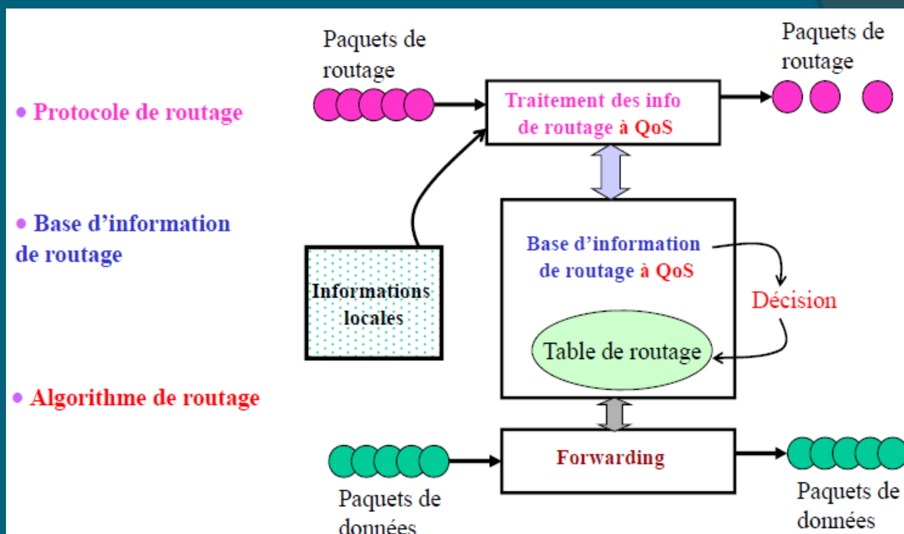
Update triggered when class limits are reached

+Class sizes allow adjustment/tuning of Accuracy/Load ratio

- Class sizes affect path selection

- Major difficulty: optimizing the “Status/Cost Info Accuracy” report

Routing Components



Classes of routing algorithms (1/4)

- **Depending on the number of participants**
  - Unicast
  - Multicast
  - anycast
- **Depending on how the path is calculated**
  - Source Routing
  - Distributed routing (hop-by-hop)
  - Hierarchical routing

Classes of routing algorithms (2/4)

Source Routing

– Each router has a local view of the network (periodic update or not)

– Selection of the path by the source and notification of this path to othersknots

+ Simple

+ more efficient for the management ofQoStemporal

- Approximate knowledge

- Inefficient for large networks

QS by Mrs. Boukhlouf 81

Classes of routing algorithms (3/4)

Routing distributed(hop by hop)

– Selection of the next node only

– Status information exchanged with neighbors

+ More flexible

- Difficulty sharing and exchanging information

Hierarchical routing

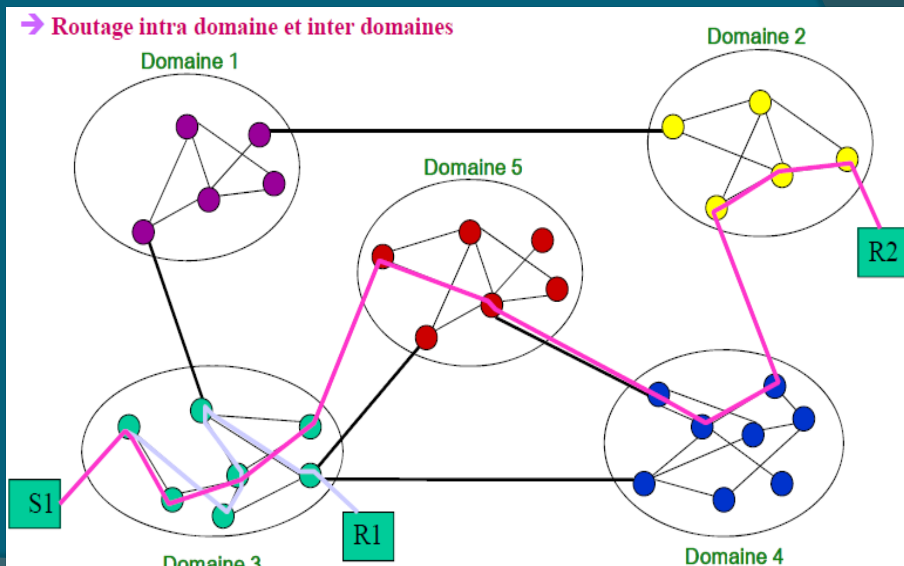
– Prioritization ofknots(aggregation)

+ Reduced state management complexity

- Network partitioning can lead to cliques in the network

QS by Mrs. Boukhlouf 82

Classes of routing algorithms (4/4)



Routing algorithms to QoS

- Dozens of routing algorithms are offered

Ranking criteria

– Constraints taken into account delay, jitter, bandwidth, …)

– Routing strategy (by source, distributed, hierarchical)

– Complexity of the algorithm

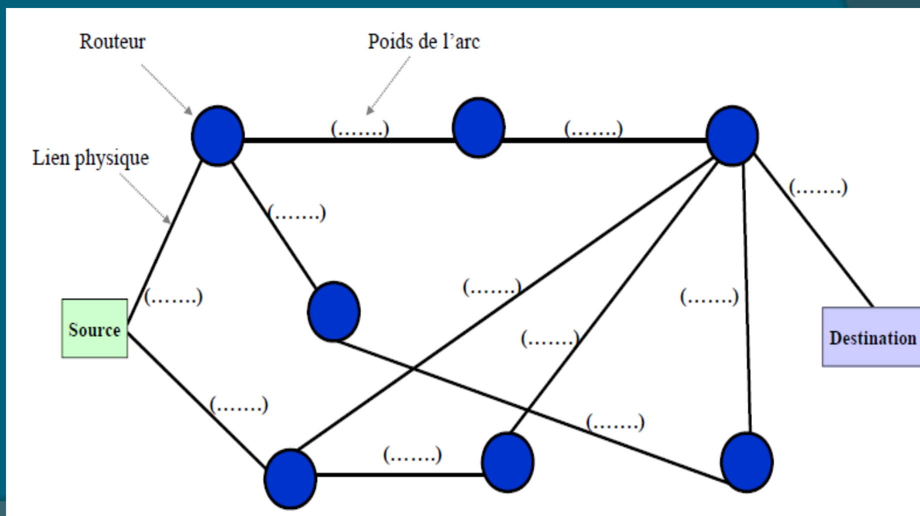
– Complexity of communication to maintain state information

Properties

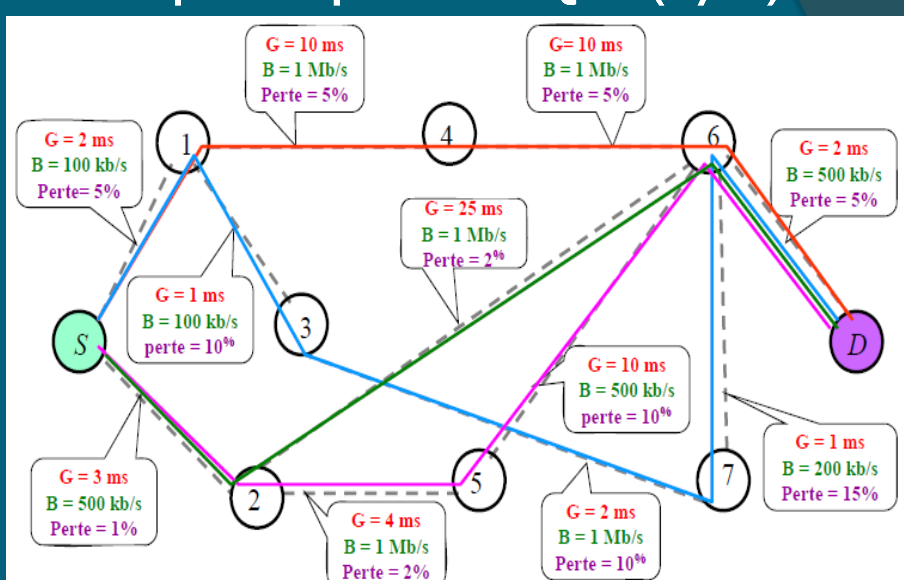
- Complexity (processing and messages) low
- Scaling up
- Routing coexistence atQoSwith best effort routing

Formalization of routing problems atQoS

- Network represented by a graph G = (V, E)



Example of paths toQoS(1/2)



Example of paths toQoS(2/2)

Chemin	Nbre de sauts	Gigue e2e	BP e2e	Taux de perte e2e
S → 1 → 4 → 6 → D	4	24 ms	100 kb/s	18,5%
S → 1 → 3 → 7 → 6 → D	5	**8 ms**	100 kb/s	38%
S → 2 → 5 → 6 → D	4	19 ms	**500 kb/s**	17%
S → 2 → 6 → D	**3**	30 ms	**500 kb/s**	**7,8%**

Elements on the costs of routing toQoS

Routing costs

- Processing/calculation
– Calculation of paths (often NP-complete)
– Calculation related to state exchanges
- Storage
– Network topology information
– Status information (on different metrics)
– Current routing table, pre-calculated routing tables
- Bandwidth (routing related packets)

Routing cost factors

- Path selection frequency
- Metrics
- Complexity factors (number ofknots/links, routing table entries,…)
- Accuracy/additional cost compromise

Examples of Routing AlgorithmsQoS:Reminders on classic routing protocols (1/2)

Distance vector routing (*Distancevector routing)*

- Also called Bellman-Ford Algorithm.
- Each router (periodically) transmits the information it knows only to its immediate neighbors. The information transmitted by each router is a vector which contains, for each destination, the distance between thisnodeand this destination.
- Each router uses the info-vectors it receives to build its routing table.
- Like everynodeonly has part of the topology view, the risk of having loops (during path finding) is high.
- To avoid loops, different techniques are possible, for example Limit the number of hops to 16.

Reminders on classic routing protocols (2/2)

Link-state routing (*Link staterouting)*

- Eachnodemust have an overall knowledge of the network topology.
- Theknotsthe different metrics of each link of the topology are exchanged between them.
- Eachnodemust :
  – Discover neighbors (by sending Hello messages)
  – Measure the transit time to each of these neighbors (by sending Echo messages)
  – Build a special packet saying everything he just learned
  – Broadcast (periodically or when topology changes) this special packet to all other routers in the network.
- Each router uses the algorithm ofDijkstra(or another) to build its routing table containing the best path for each destination.

Example 1: ExtendingDijkstra's Shortest path algorithm (Unicast case)

➔ **Définitions**

G : ensemble des nœuds du réseau

E : ensemble des arcs du réseau

w(x,y) : poids associé au lien du nœud *x* vers le nœud *y*

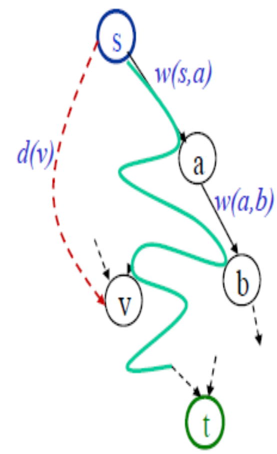
d(v) : poids du chemin depuis la source *s* jusqu'au nœud *v*

s : nœud source

t : nœud destination

Pred(u) : prédécesseur du nœud *u* sur le chemin

Adj(u) : ensemble des nœuds adjacents du nœud *u*.



Examples of Routing AlgorithmsQoS

➔ **Algorithme One-QoS_Dijkstra(V,E, w, s, t)**

```
Step 1 : /* Initialisation */
   For each node v ∊ V do  d[v]← ∞ ;  Pred[v] ← NIL   od
   d[s] ← 0
   Q ← V
Step 2 : /* Détermination du chemin le plus court *//
  while Q ≠ ∅
   { do u ← Extract-min(Q)    /* u tq : d[u] = min{d[y], ∀y ∊ Q} */
              if u = t  then exit
              for each v ∊ Adj[u]
              { if  w(v, u) ⊕ d[u] ≺ d[v]
                       then  Pred[v] ← u
                             d[v] ← w(v, u) ⊕ d[u]
              }
   }
```

Fonction Poids

QoS = délai ⇒ (⊕, ≺) = (+, <)
QoS = Perte ⇒ (⊕, ≺) = (*, <)
QoS = Disponibilité ⇒ (⊕, ≺) = (*, >)

Examples of Routing AlgorithmsQoS

Exemple 2 : EBSP

(Enhanced Bandwidth Shortest Path) J. Wang et K. Nahrstedt 2002

➔ **Objectif**

- Sélection du chemin ayant la bande passante la plus élevée en minimisant le nombre de sauts
- C'est un algorithme saut par saut.
- Il est utilisable dans des contextes où on cherche à fournir la bande passante la plus élevée possible à certains flux (par exemple, flux premium de DiffServ).

➔ **Fonction Poids de EBSP**

$$d(P) = \sum_{i=1}^{n-1} \frac{2^{i-1}}{BP(i,i+1)}$$

2^{i-1} : Facteur qui pénalise les chemins avec plus de noeuds

$BP(i,i+1)$: Facteur qui pénalise les liens avec une faible BP

n : nombre de nœuds du chemin P
BP(i,i+1) : bande passante du lien (i, i+1)

Examples of Routing AlgorithmsQoS

Exemple 2 : EBSP (suite)

➔ **Algorithme EBSP(V,E, BP, s, t)**

```
Step 1 : /* Initialisation */
For each node v ∈ V do  d[v]← ∞ ;  Pred[v] ← NIL   od
d[s] ← 0
Q ← V

Step 2 : /* Détermination du chemin le plus court *//
 while Q ≠ ∅
  { do u ← Extract-min(Q)    /* u tq : d[u] = min{d[y], ∀y ∈ Q} */
                 if u = t  then exit
                 for each v ∈ Adj[u]
                 { if  2d[u] + 1/BP[v,u] < d[v]
                          then  Pred[v] ← u
                                d[v] ← 2d[u] + 1/BP[v,u]
                 }
  }
```

Routing protocols

- **Objective**: Define the formats of the messages conveying the informationrouting andthe rules for exchanging these messages.

Current protocols in the Internet: Best effort

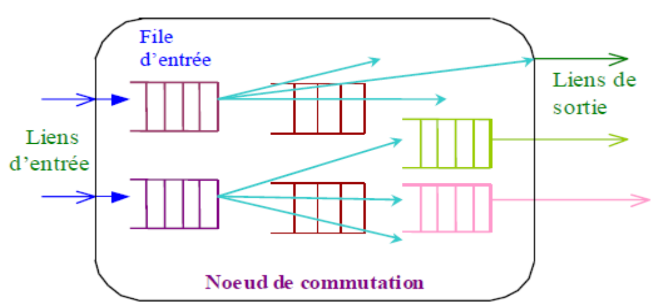
- RIP (RoutingInformation Protocol) – intra-domain; distancevector
- OSFP (Open Shortest Path First) – intra-domain; link state
- IGP (Interior Gateway Protocol) – intra-domain; link state
- IS-IS (Intermediate System - IS – intra-domain; link state
- EGP (ExteriorGateway Protocol) – inter-domain
- BGP-Border Gateway Protocol v.4 – inter-domain; distancevector

Protocols forQoS

- QOSFP: extension of OSPF to take into account theQoS

Packet scheduling

Queue management approaches (1/2)



➔ Attente dans les files d'entrée

➔ Attente dans les files de sortie

➔ Autres techniques

Queue management approaches (2/2)

- **Waiting in exit queues:** Each packet is placed in its output queue as soon as it arrives.

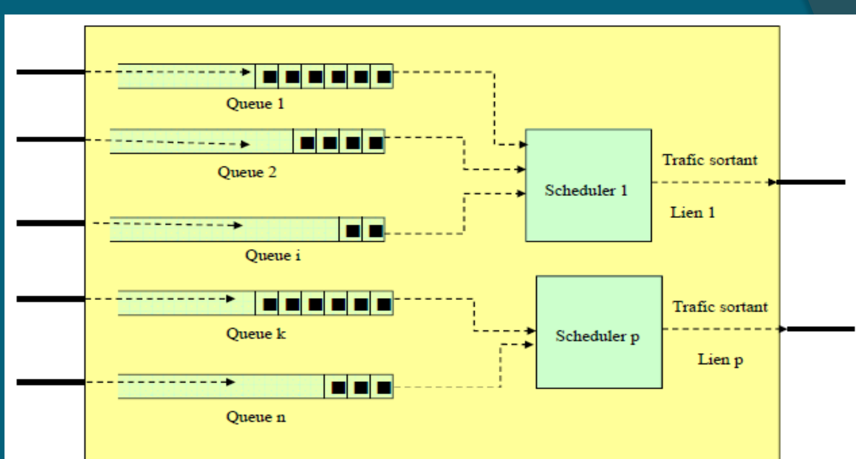
Advantage: it is the most efficient in terms of throughput

- **Waiting in entry queues:** The packets arriving on an input port are placed in a queue associated with this port and served in FIFO.

Inconvenience: « Head-of-lineblocking(when the first packet in the queue is blocked, because its output link is busy, all the other packets are blocked, even if their output link is free).

- **Waiting in virtual exit queues (avoids the "Head-of-lineblocking"):** Each input port is associated with as many queues as there are output links used by the packets arriving at this port. Any packet waits in its virtual output queue, before being served.
- **Waiting in a single queue:** All packets arriving at the router are placed in a single queue before being served. It is the simplest, but the least effective for the guarantee ofQoS.
- **Combination of waiting in entry and exit queues**

Restriction: exit queues



- Algorithme d'ordonnancement de paquets = Discipline de service (terminologie des files d'attente)

Properties

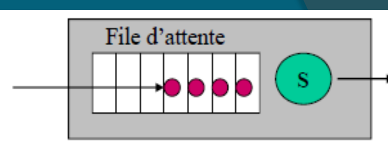
- Ease of implementation and low additional cost
  - Processing for each packet
    - need for very rapid treatment
  - Ideal complexity *O(1) - Complexity O(Number_packets) to avoid*

Guarantee (best effort, statistical, deterministic) verifiable: delay, loss, etc.

- Flow isolation
  - A malfunctioning flow should not disrupt others.
- Equity
  - Equitable distribution of resources between streams
  - Fairness leads to *best effort, but not at the bounds guarantee*
- Scaling (Scalability)
- Admission control
  - Simple to implement
  - Efficient (for better admission and use of resources)

Law of conservation ofKleinrock(1971) (1/2)

Formule de Little (1956) : $E(t) = E(n) / \lambda$



$E(t)$: moyenne de temps de réponse

$E(N)$: moyenne du nombre de clients dans la file

λ : taux d'arrivée des clients

La loi de conservation de Kleinrock stipule que :

Si l'ordonnanceur est conservatif alors, quelque soit la discipline choisie :

$$\sum_{k=1}^{N} r_k d_k q_k = constante$$

$r_k d_k q_k$ peut être considéré comme un délai pondéré pour la connexion k.

N connexions géré par un ordonnanceur.

r_k le débit moyen de la connexion k.

d_k le délai moyen de traitement par paquet de la connexion k.

q_k le délai moyen de séjour en file d'attente par paquet de la connexion k.

Disciplines of service

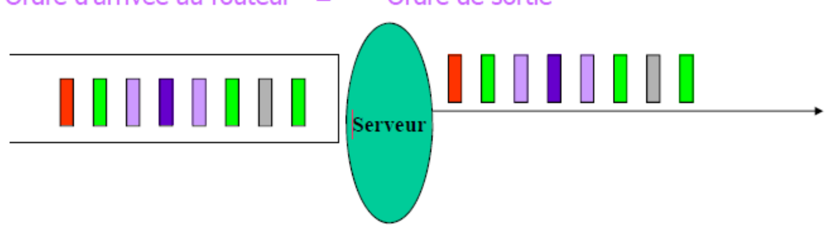
- Conservatives
  - –PR (Fixed Priority)
  - –QF (Fair Queuing)
  - –WFQ (Weighted Fair Queuing)
  - –WF2Q (Worst-caseFairWeightedFairQueuing)
  - –SCFQ (Self-clocked Fair Queuing)
  - –VirtualClocks
  - –Delay EDD (Delay Earliest Due Date)
  - –Others
- Non-conservative
  - –JitterESD
  - –Stop-and-Go
  - –HRR (HierarchicalRound Robin)
  - –RCSP (Rate Controlled Static Priority)
  - –Others

FIFO and FP scheduling

FIFO (First in First Out) – FCFS (First Come First Serve)

- Naturelle (la première qui vient à l'esprit)
- Non équitable
- Ne permet pas la garantie de QoS (en général)

Ordre d'arrivée au routeur = Ordre de sortie



PR (Fixed Priority) (1/2)

FP (Fixed Priority) (1/2)

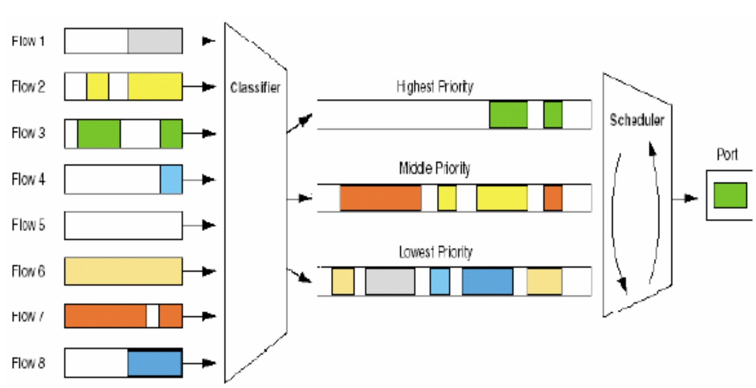
- FP (Fixed Priority) = PQ (Priority Queueing)
- Une priorité fixe est associée à chaque flux (connexion) ou à chaque paquet

Priorité	
Haute	Service urgent et critique
↓	Service temps réel audio et vidéo
Basse	Trafic best-effort

- Il y a un mapping entre les priorités initiales et les priorités de l'ordonnanceur.
- Si le nombre de priorités de l'ordonnanceur est faible, cela peut conduire à un service non-conforme aux priorités initiales.

PR (Fixed Priority) (2/2)

- Les paquets de priorité élevée sont servis d'abord

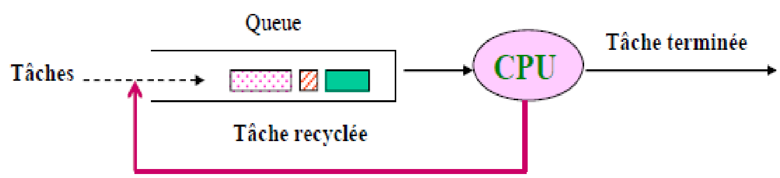


- Risque de famine pour les paquets de priorités faibles

Round Robin Scheduling

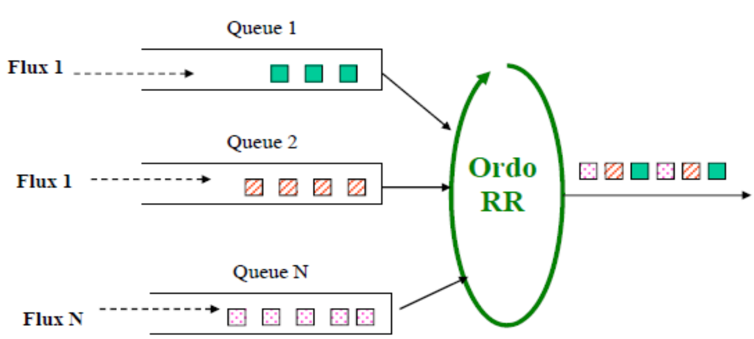
Round Robin (RR) de base pour les tâches

- Une seule queue pour toutes les tâches (processus).
- Servir pendant Δt chaque tâche. Si la tâche n'a pas fini la recycler en queue.
- Ordonnancement largement utilisé dans les systèmes non temps réel.



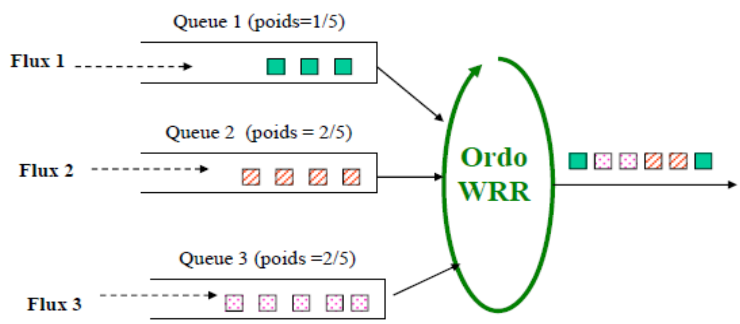
Round Robin (RR)

- Associer une queue à chaque flux. Servir les queues à tour de rôle.
- Avantages : simplicité, possibilité de réalisation câblée, équité.
- Inconvénients : ne permet pas la garantie de QoS. Pas d'équité si les paquets sont de tailles différentes.



Weighted Round Robin (WRR):principlebasic

- Associer une queue à chaque flux. Associer à chaque flux un poids normalisé en fonction de la taille moyenne de paquet du flux.
- Servir les queues (non vides) à tour de rôle et en fonction de leurs poids.
- Avantages : prise en compte de l'importance (poids) de chaque flux. Protection des flux les uns contre les autres.
- Inconvénients : pénalise les flux à faibles poids.



Deficit Round Robin (DRR) (1/2)

- Idée de base : extension de RR pour des paquets de taille variable.
  Economiser des crédits pour transmettre.
- Principe

```
• Associer un compteur C[k], initialisé à 0, à chaque queue k
• Lorsque la connexion k est visitée par DRR
   ▪ Si la queue k est non vide
      { C[k] = C[k] + quantum ;
      Si Taille(tetequeue[k]) <= C[k]
             { Le paquet est transmis;
             C[k] = C[k] - taille du paquet transmis;
             Si la queue k est vide { C[k] = 0; }
      }
   ▪ Passer à la queue suivante
```

Le quantum est choisi pour permettre la transmission de paquet de taille minimale

DeficitRound Robin (2/2)

- Advantages: easy to implement; more equity than RR
- Disadvantages: does not allow the guarantee ofQoS(in general).
- There are other forms of RR strategies

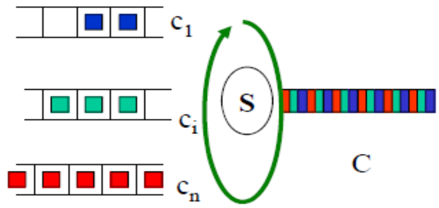
– HRR (HierarchicalRR)

– BWRR (BudgetedWRR)tieofQoS(in general).

PGPS and WFQ scheduling

PS « Processor Sharing »

- Temps partagé simple du processeur (pour l'ordonnancement de tâches)



$$\forall i, j \quad c_i = c_j = \frac{C}{n}$$

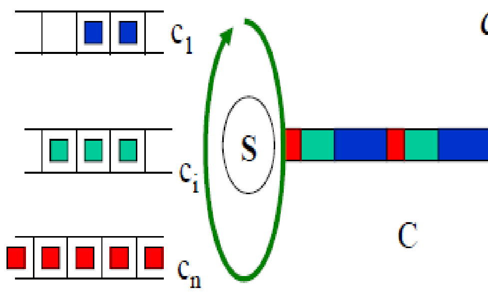
c_k : vitesse d'exécution de la tâche k

C : vitesse d'exécution du processeur

- PS n'est pas implantable pour les paquets (sinon on risque de transmettre des paquets contenant moins d'un bit).

GPS « Generalized Processor Sharing »

– PS + équité en tenant compte de l'allocation préalable des tâches (poids ϕ_i)



$$c_i = \frac{\phi_i}{\sum_j \phi_j} C$$

$$\sum_j \phi_j \leq 1$$

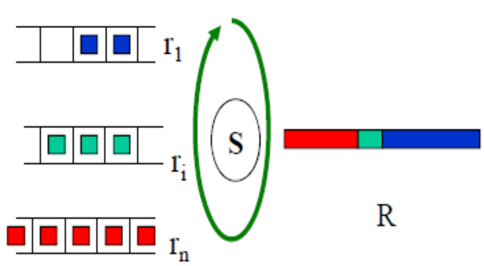
– GPS garantit un temps d'exécution (c_i) selon le poids ϕ_i

Technique « Weighted Fair-Queueing » (Demers, Keshav et Shenker 1989)
PGPS « Packet Generalized Processor Sharing » (Parekh et Gallager 1993)

– GPS signifie que l'interruption de tâche peut se faire à n'importe quel moment (PGS non applicable directement aux réseaux)

– PGPS = version de GPS appliquée aux réseaux

– PS + équité en tenant compte de l'allocation préalable des connexions (poids ϕ_i)



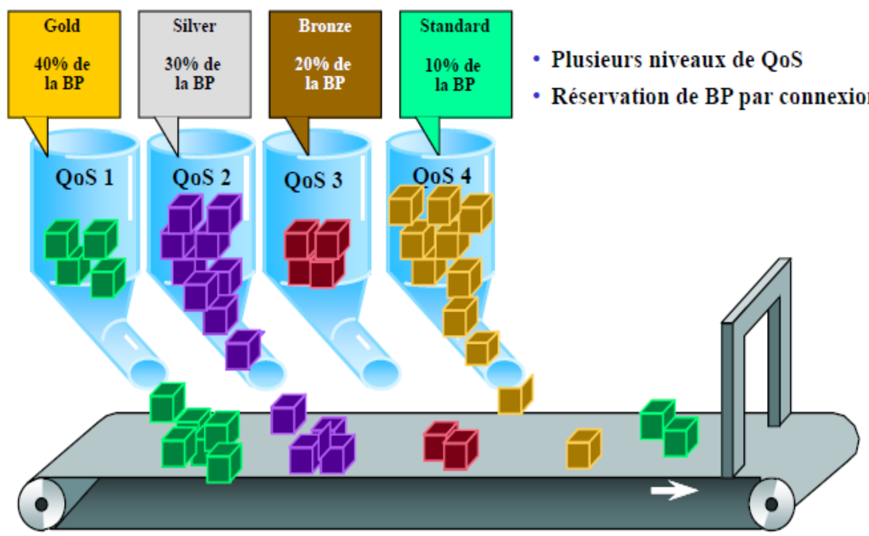
$$r_i = \frac{\phi_i}{\sum_j \phi_j} R$$

$$\sum_j \phi_j \leq 1$$

r_k : débit du flux k R : débit du lien

– GPS garantit le débit

General principle of WFQ



- Plusieurs niveaux de QoS
- Réservation de BP par connexion

DEDD and JEDD scheduling

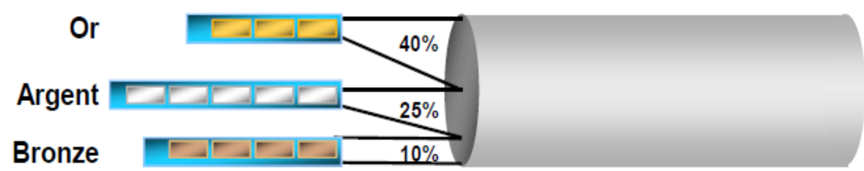
Ordonnancement à priorité DEDD

➔ **Delay Earliest Due-Date [Ferrari et Verma]**

- EDF (Earliest Deadline First) pour les paquets
- Garantie de contraintes de délai (de bout en bout en cas de réseau homogène)
- Modèle orienté connexion
- Flux périodique

CBQ scheduling

Class-basedQueuing (1/3) [Floyd, Jacobson 95]



- Chaque classe se voit réserver une part (selon son poids) de bande passante
- Les poids garantissent une bande passante minimum
- Les flux sont regroupés en classes
- La bande passante laissée libre par un flux d'une classe et d'abord utilisée par les flux des classes 'sœurs'.

➔ **C'est la discipline la plus implantée actuellement par les routeurs dits à QoS.**

Summary

FIFO	Pas de distinction entre les paquets – Pas de réservation de ressources
FP	Les queues de haute priorité sont servies d'abord. Garantie de transmission de trafic critique/urgent
RR	Les queues sont servies à tour de rôle et de la même manière
WRR	Les queues sont servies à tour de rôle et en tenant compte de leur poids
DRR	Les queues sont servies à tour de rôle en fonction de la taille de leurs paquets et du crédit accumulé.
WFQ	Garantie d'équité entre les flux – Les queues sont traitées en RR
D-EDD	Paquets servis selon EDF. Garantie de délai
J-EDD	Paquets servis selon EDF + rétention de paquets. Garantie de gigue
CBQ	Classification, réservation et service selon les classes. Garantie de bande passante et éventuellement de délai

Congestion control and buffer management
Resource Allocation (1/4)

- Ressources = CPU, mémoire, bande passante…
- QoS fournie dépend des ressources allouées pour le service.
- Allocation de ressources ⇒ Politique d'allocation
  (droits d'utiliser des ressources, coûts, …)
- Allocation de ressources
  - Sans Négociation : rigide (tout ou rien), sûre
  - Avec Négociation : à la connexion, flexible, complexe
  - Avec Renégociation : s'adapter au réseau à tout moment, transmettre à moindre coût, (très) complexe

Resource Allocation (2/4)

- **Resource reservation issues**
- ■ Resources reserved but not used: Loss
- ■ Minimum/average resources to reserve: difficult to determine

Resource Allocation (3/4)

➔ **Gestion de buffers (tampons ou queues)**

- Gestion de buffers (un flux / une file, une file partagée entre plusieurs flux)
- Mémoire de stockage des paquets limitée ⇒ Contrôler son utilisation
- Deux décisions majeures : Quand rejeter les paquets ?
  Quels paquets rejeter ?

➔ **Contrôle et traitement de trafic**

- Contrôle de congestion
- Contrôle de trafic de l'utilisateur
- Façonnage du trafic de l'utilisateur (« traffic shaping »)
- Marquage de paquets

congestion problem

- Random streams + Limited memory + limited bandwidth ⇒ Possibility of congestion
- Negative effects: high loss rate high latency high load,
- Need for congestion control

Congestion control techniques

- Methods: reactive vs. preventive
– RED (“Random Early Detection”)
– Variants of RED (FRED, WRED…)
– ECN (“Explicit Congestion Notification”)
- Others

FIFO technique for queue management

- FIFO: the simplest technique for managing router queues.
– only one queue per output interface
– serve the package in the lead
– queue the incoming packet if the queue is not full
– reject the last packet if queue full

FIFO technique for queue management

++ FIFO has the merit of not posing any problem to the operation of the Internet based on the principle of best effort. No constraints for the installation of IP routers.

--Impossible to differentiate the traffics (because we have only one queue)

--There is no flow isolation.

--Large buffers imply high latencies.

RED technique (Random Early Detection)

- RED: most popular technique for congestion avoidance
- Proposed by Floyd and Jacobson (1993) to manage TCP streams
- Queue management with thresholds - active (preventive) management technique

Algorithm of the RED technique

- Probabilistic rejection/marking as a function of average tail size:

Qavg = (1-w)Qavg + w*Qreel w<<1 (ex. w=0.002)

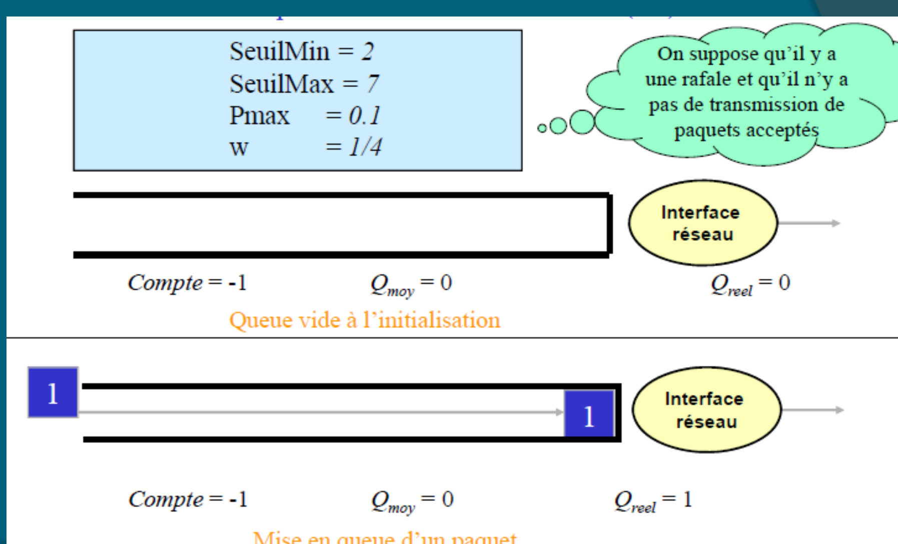
- Whether *Qavg<MinThreshold: no rejection/marking of incoming packet*
- Whether *Qavg≥ThresholdMax: systematic rejection/marking ofEasterarriving*
- Whether *MinThreshold≤Qavg<ThresholdMax: rejection/marking with a probabilitypTo* calculated as follows:

bp=Pmax(Qavg-MinThreshold)/(ThresholdMax–MinThreshold)*

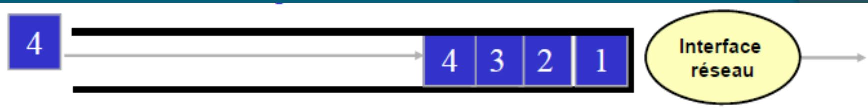
*not=bp/(1 –bp*account)*

Count = number of packets received since the last packet flagged/rejected while condition *MinThreshold≤Qavg<ThresholdMaxstay*satisfied.

Example of how RED works

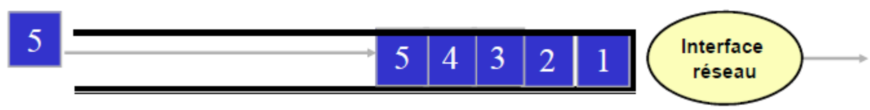


Example of how RED works



Compte = -1 Q_{moy} = 1.27 Q_{reel} = 4

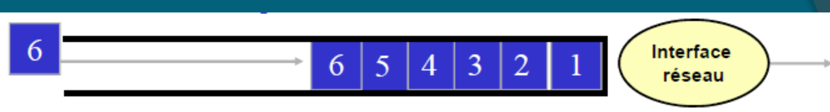
Après le 4[ème] paquet



Compte = -1 Q_{moy} = 1.95 Q_{reel} = 5

Après le 5[ème] paquet

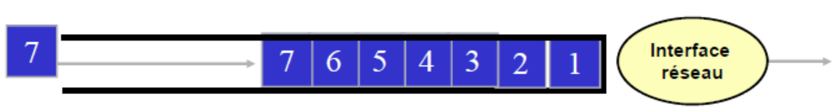
Example of how RED works



Compte = 0 Q_{moy} = 2.71 Q_{reel} = 6

Le *SeuilMin* est dépassé, on calcule $p_a = 0{,}014$. On tire une valeur aléatoire pour savoir si on rejette ou non le paquet. On suppose ici que l'on ne rejette pas le paquet.

Après le 6[ème] paquet

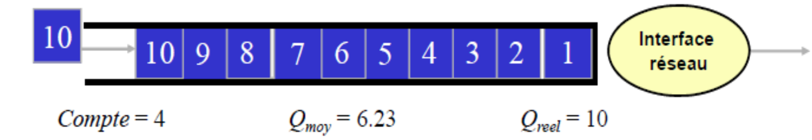


Compte = 1 Q_{moy} = 3.53 Q_{reel} = 6

Le *SeuilMin* est dépassé, on calcule $p_a = 0{,}03$. On tire une valeur aléatoire pour savoir si on rejette ou non le paquet. On suppose ici que l'on ne rejette pas le paquet.

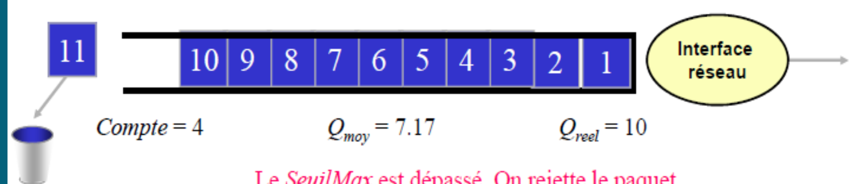
Après le 7[ème] paquet

Example of how RED works



Le *SeuilMin* est dépassé, on calcule $p_a = 0{,}127$. On tire une valeur aléatoire pour savoir si on rejette ou non le paquet. On suppose ici que l'on ne rejette pas le paquet.

Après le 10ème paquet



Le *SeuilMax* est dépassé. On rejette le paquet.

Après le 11ème paquet

Limitations (disadvantages of RED)

- Disturbance by misbehaving flows
- Sensitive to the number of sources
- Difficulties in choosing the parameters (thresholds and Pmax)

Flow RED Technique (FRED)

- FRED: an extension of RED proposed by Lin and Morris (1997)
- Objective: protect TCP flows from aggressive flows
- Basic principle:

• Maintain a single FIFO queue, but keep track of the number of packets from each connection

• Drop packets from a connection when the connection exceeds its queue sharing capacity

– Rejections are proportional to the bandwidth used

– Aggressive flows are identified and penalized

• The principle of calculating the estimated average tail size,Qavg, is optimized compared to RED: evaluation ofQavgat each arrival and departure of a packet,Qavgis not updated in case of packet rejection.

QS by Mrs. Boukhlouf 131

Flow RED Technique (FRED)

- Paramètres de FRED (en plus de ceux de RED) :

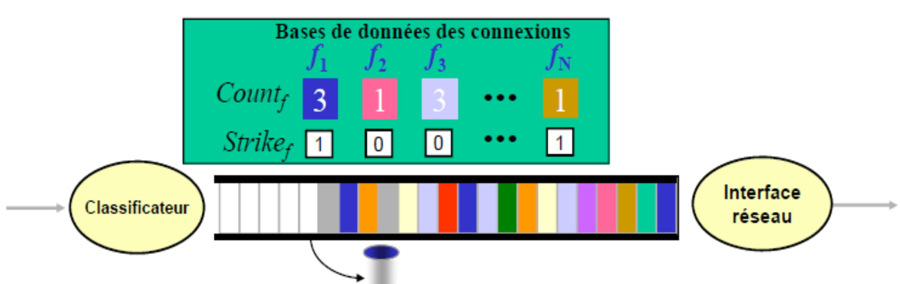
Min_Q : nombre minimum de paquets dans la queue par flux. Aucun rejet ne peut être appliqué à un flux qui n'a pas plus de *Min_Q* paquets dans la queue.

Max_Q : nombre maximum de paquets dans la queue par flux.
$Count_f$: nombre de paquets dans la queue pour le flux *f*.

$Strikes_f$: nombre de fois que le flux *f* a émis un paquet en excès.
Les flux avec une valeur élevée de *Strike* sont pénalisés en premier.

NbFlux : nombre de flux actifs.

MoyFlux : estimation d'une taille moyenne de queue par flux. Les flux qui ont moins de paquets que *MoyFlux* sont favorisés par rapport à ceux qui en ont plus.



Service levels

- The term "**level of service**" (in English *Servicelevel*) defines the level of requirement for the ability of a network to provide a point-to-point or end-to-end service with atrafficgiven. Generally, three levels are defined.QoS:
- **Best Effort** (***best effort***), providing no differentiation between several network streams and allowing no guarantee. This level of service is sometimes called *lack of QoS*.
- **Differentiated service** (***differentiated service*** **Or** ***soft QoS***), making it possible to define priority levels for the various network flows without, however, providing a strict guarantee.
- **Guaranteed Service** (***guaranteed service*** **Or** ***hard QoS***), consisting of reserving network resources for certain types of streams. The primary mechanism used to achieve such a level of service is RSVP (*Resources reservation Protocol*, translate *Resource reservation protocol*).

Approaches to providing QoS

IETF (Internet Engineering task Force) proposed multiple means to take into account the needs of QoS

- **Two approaches to providing QoS**
- **Integrated Services (IntServ)**[RFC 1633 -June1994]

connection oriented

- **Differentiated Services (DiffServ)**[RFC 2475 - December 1998]

Focused on stream aggregation and packet marking

- **Protocols forQoS**
- **RSVP**(ResourcereservationProtocol) [RFC 2205 – September 1997]
- **MPLS**(Multi Protocol Label switching) [RFC 3031 – January 2001]
- **Others**

IntServ(Integrated Servies) (1/3)

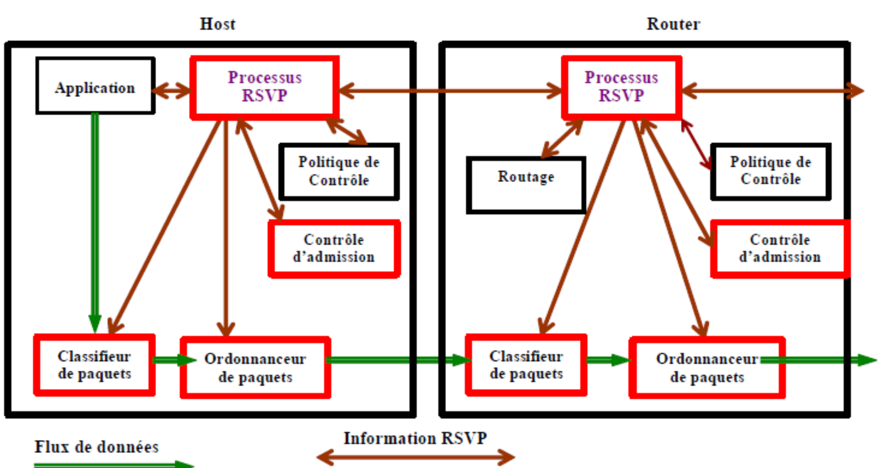
- Flux de données = paquets ayant la même source et destination et même port
- IntServ adopte le modèle orienté connexion
- IntServ se base sur RSVP pour la réservation de ressources
- Les routeurs qui n'implantent pas IntServ ignorent les nouvelles classes de QoS
- Trois classes de service selon les besoins des applications
  - Service garanti (garantie absolue)
  - Service à charge contrôlée (garantie statistique)
  - Services Best-effort de trois types (pour le Web, FTP, Mail, etc.)

QS by Mrs. Boukhlouf 135

IntServ(IntegratedServing) (2/3)

- Implantation de IntServ : 4 composants

Protocole de signalisation, Contrôle d'admission, Classifieur, Ordonnanceur de paquets



QS by Mrs. Boukhlouf 136

IntServ

→Paramètres de caractérisation de QoS et trafic [RFC 2215 – 1997]

- NON-IS-HOP : indique si, sur le chemin du flux, il y a des noeuds qui n'implantent pas IntServ.
- NUMBER_OF_IS_HOPS : compte le nombre de noeuds qui implantent IntServ sur le chemin du flux.
- AVAILABLE_PATH_BANDWIDTH : fournit l'information sur la bande passante disponible sur le chemin du flux.
- MINIMUM_PATH_LATENCY : fournit la latence minimale (i.e. avec délai de séjour en file d'attente -*queuing delay*- égal 0) pour les noeuds traversés.
- PATH_MTU : fournit la *Minimum Transmission Unit* sur le chemin du flux.
- TOCKEN_BUCKET_TSPEC : décrit les caractéristiques du trafic par le débit du seau percé (*r*), la capacité maximum du seau (*b*), le débit de crête (*p*), la taille minimum de paquet (*m*) et la taille maximum de paquet (*M*).

Limits/disadvantages

- The number of individual streams can be very large. Consequently, the number of control messages can be high and requires a lot of resources at each router.
- Policies should be in place to determine when, where, and for whom resources can be reserved.
- Security rules should be in place to ensure that unauthorized users cannot make resource reservations.
- Few manufacturers have implemented IntServ in large scale.
- IntServ: suitable only for small networks (scaling problem)

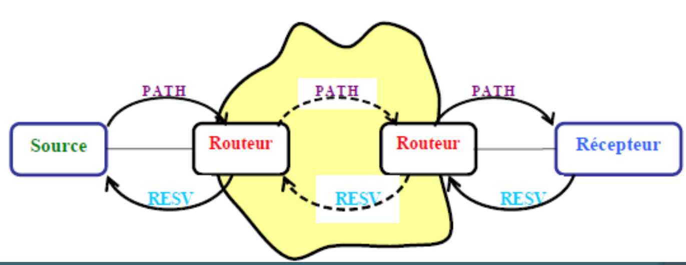
Reservation Protocol (RSVP) RFC 2205

→ C'est quoi RSVP?

- RSVP est un protocole de signalisation pour demander la réservation de ressources dans un réseau IP.
- Principales caractéristiques de RSVP
  - C'est l'application qui initie le processus de réservation (granularité fine de réservation) au moment du démarrage d'un flux.
  - Modèle de réservation orienté Récepteur
  - Les réservations sont faites pour chaque flux individuel
  - Les flux peuvent être *unicast* ou *multicast*
  - Il supporte les réservations hétérogènes et permet la renégociation de réservation
  - Il permet un bon partage des ressources réservées pour de multiple flux.
  - La gestion des réservations s'effectue en mode état *soft*.

RSVP

- Un chemin *unicast* ou *multicast* est déterminé par un algorithme de routage (sans être sûr que ce chemin réponde à la QoS exigée).
- La source du flux transmet un message PATH pour indiquer les caractéristiques de son flux.
Chaque routeur traversé garde la trace du flux et des ressources demandées.
- Chaque récepteur a ses propres capacités et spécifie dans un message RESV ses exigences.
Chaque routeur sur le chemin du message RESV confirme les réservations ou les annule.
Quand le message RESV atteint la source, les réservations sont acceptées le long du chemin et le flux de données peut commencer.



Disadvantages

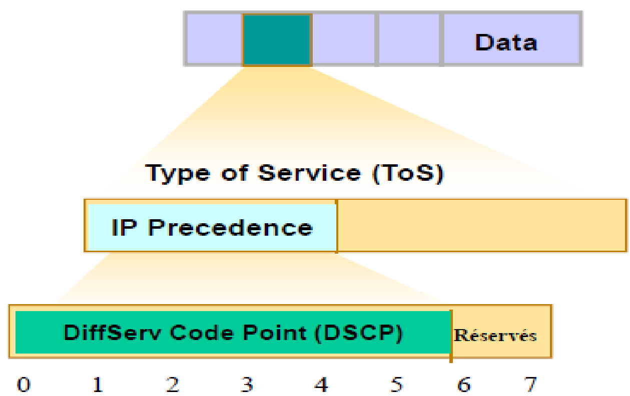
- Maintaining soft states means that all routers must constantly monitor and update states for each individual flow. The consequence may be network congestion.

Basic principles and characteristics of DiffServ

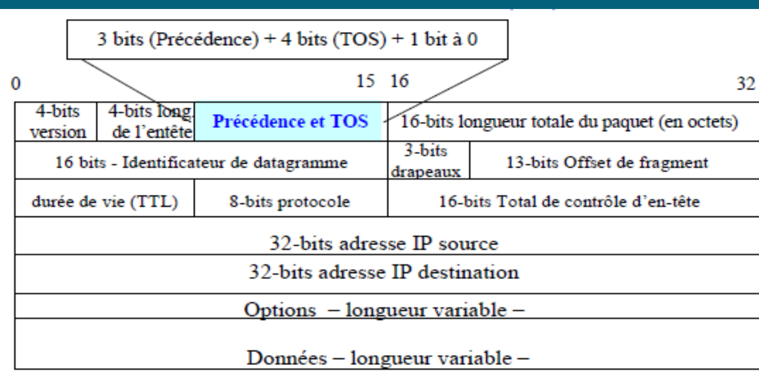
- Pallier les inconvénients de passage à l'échelle de IntServ
- Introduction du concept d'agrégation de flux pour simplifier les traitements de classification et marquage de paquets
- Caractéristiques importantes
  - Pas de gestion d'état par flux
  - Pas (ou peu) de classification de paquets à l'intérieur du réseau
  - Gestion sur la base de SLA et non de connexion
  - Réalisable par des mécanismes simples (câblés)

Coded DiffServ

- Champ DiffServ (or DSCP) = 6 bits dans l'entête IPv4 ou IPv6
  Il indique le niveau de QoS pour traiter le paquet



CodedDiffServ



Entête de paquet IPv4
[RFC 791 ; Sep. 1981]

TOS = 1xxx : Minimiser le délai
01xx : Maximiser le rendement
001x : Maximiser la fiabilité
0001 : Minimiser le coût
0000 : Service normal

On ne peut pas exiger deux types de QoS en même temps

Coded DiffServ

Version (4 bits)	Classe de trafic (8 bits)	Etiquette de flux pour la QdS (20 bits)		
Longueur de données (16 bits)			Entête suivant (8 bits)	Nombre de sauts (8 bits)
durée de vie (TTL)	8-bits protocole	16-bits Total de contrôle d'en-tête		
Adresse source (16 octets)				
Adresse destination (16 octets)				

Entête de paquet IPv6
[RFC 2460 – Déc. 1998]

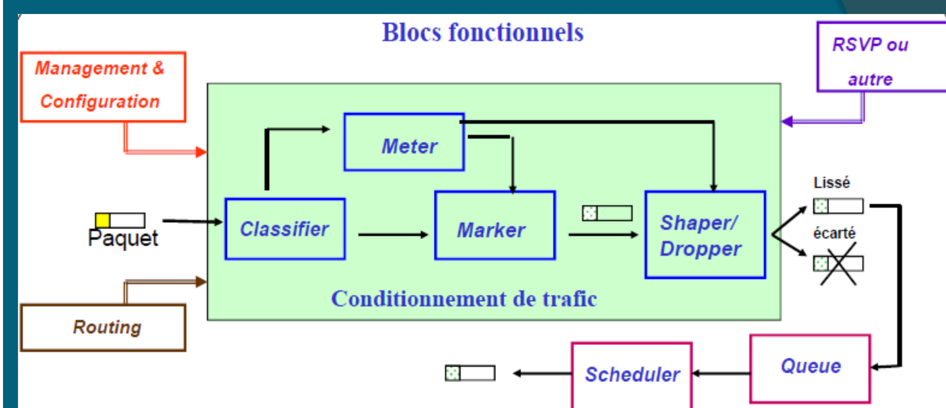
4 bits de priorité/classe

Les priorités de 0 à 7 : pour le trafic dont la source dispose d'un mécanisme de contrôle de congestion : 0 (trafic non caractérisé), 1 (NNTP), 2 (SMTP),

4 (FTP et NFS), 6 (Telnet et terminal X), 7 (SNMP)

Les priorités de 8 à 15 : pour le trafic qui ne prend pas en compte les situations de congestion (trafic temps réel).

Architecture of DiffServ



- *Classifier :* sélectionne les paquets en fonction de leur entête.
- *Meter :* effectue des mesures pour savoir si le paquet est conforme au contrat de trafic
- *Marker :* réécrit ou change le DSCP
- *Shaper :* retarde certains paquets pour les rendre conformes à certain rythme.
- *Dropper :* écarte (élimine) certains paquets non conformes ou ayant un taux de rejet exigé plus élevé que celui des autres paquets (seulement en cas de congestion)

Limits of DiffServ

– Provide QoS to aggregate flows does not allow (except for the premium class) always to offer the QoS end-to-end for each flow taken individually.

–DiffServ assumes statically configured SLAs. However, user needs and network topology can change over time.

–DiffServ is sender oriented. However, sometimes it is necessary to take into account the capacities and needs of the receiver.

– The number of DSCPs is not high enough to really differentiate the streams. We oversize often (eg. Put together two streams that require two latencies of 10 and 50 ms).

IntServ vs.DiffServ

	Intserv	Diffserv
Coordination pour la différenciation de service	End-to-end	Local (PHB)
Etendue de service	Chemins unicast ou Multicast	Partout dans le réseau
Sclabilité	Limité par le nombre de flux	Limité par le nombre de classes de service
Comptabilité	Basée sur les caractéristiques de flux et exigences de QoS	Basée sur l'utilisation de classe
Gestion de réseau	Similaire aux réseaux à commutation de circuits	Similaire aux réseaux IP
Déploiement interdomaine	Accords multilatéraux	Accords bilatéraux