Integrated services

- Reading:
  - S. Keshav, “An Engineering Approach to Computer Networking”, chapters 6, 9 and 14

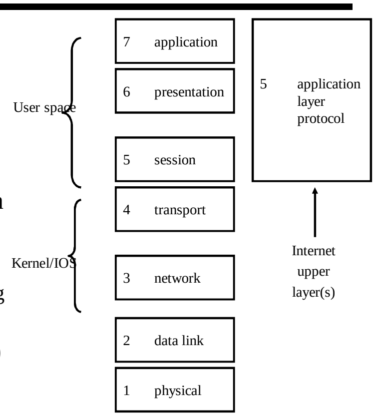
Module objectives

Learn and understand about:

- Support for real-time applications:
  - network-layer and transport-layer
- **Quality of service (QoS)**:
  - the needs of real-time applications
  - the provision of QoS support in the network
- Many-to-many communication - **multicast**
- **Integrated Services Network (ISN)**

Support for real-time applications

- Support in the network:
  - routers, routing
- Support at the end-systems:
  - transport protocols
- Support at the application level:
  - user-network signalling
  - application-level signalling and control
- (Link & physical layers?)



Real-time flows and the current Internet protocols

The “problem” with IP [1]

- Data transfer:
  - datagrams: individual packets
  - no recognition of **flows**
  - connectionless: no signalling
- Forwarding:
  - based on per-datagram forwarding table look-ups
  - no examination of “type” of traffic – no **priority** traffic
- Routing:
  - **dynamic routing** changes
  - no “fixed-paths” → no fixed QoS
- Traffic patterns

The “problem” with IP [2]

- **Scheduling** in the routers:
  - first come, first serve (FCFS)
  - no examination of “type” of traffic
- No priority traffic:
  - how to mark packets to indicate priority
  - IPv4 ToS not widely used across Internet
- Traffic aggregation:
  - destination address
- (QoS: pricing?)

Questions

- Can we do better than **best-effort**?
- What support do real-time flows need in the network?
- What support can we provide in the network?
- Alternatives to FCFS?
- Many-to-many communication?
- Application-level interfaces?
- **Scalability?**

Requirements for an ISN [1]

Today’s Internet

- IPv4: QoS not specified
- TCP: elastic applications
- Many network technologies:
  - different capabilities
  - no common layer 2
- No support for QoS:
  - ToS in IPv4 – limited use
- QoS requirements:
  - not well understood

Integrated Services Packet Network (ISPN)

- QoS service-level:
  - service type descriptions
- Service interface:
  - signalling
- Admission control:
  - access to resources
- Scheduling:
  - prioritisation and differentiation of traffic

Requirements for an ISN [2]

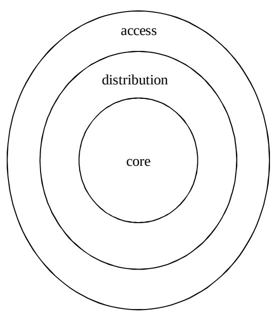
- QoS service-level:
  - packet handling
  - traffic description
  - policing
  - application flow description
- Service interface:
  - common data structures and parameters
  - signalling protocol
- Admission control:
  - check request can be honoured
- Scheduling:
  - packet classification
  - prioritisation of traffic
  - queue management

Traffic and QoS parameters

Network structure [1]

Network hierarchy

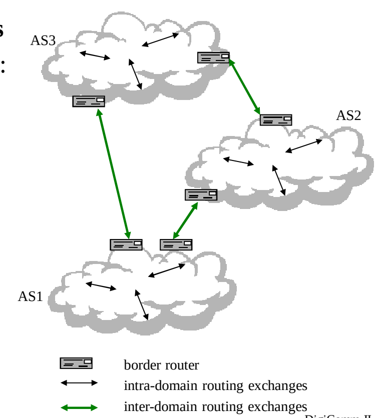
- Access network:
  - low multiplexing
  - low volume of traffic
- Distribution network:
  - interconnectivity at local level
  - medium volume of traffic
  - low multiplexing
- Core network – backbone:
  - high volume of traffic
  - high multiplexing



Network structure [2]

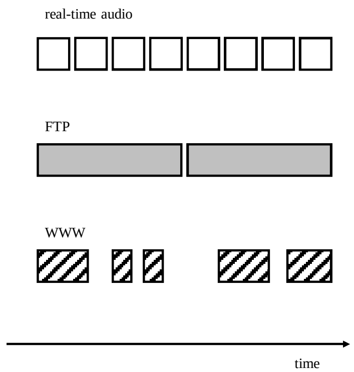
Administrative boundaries

- Autonomous system (AS):
  - **intra-domain routing**
  - internal policy
  - routing metric?
  - protocols: RIPv2, OSPFv2
- Interconnection of ASs:
  - **inter-domain routing**
  - interconnectivity information
  - protocols: BGP



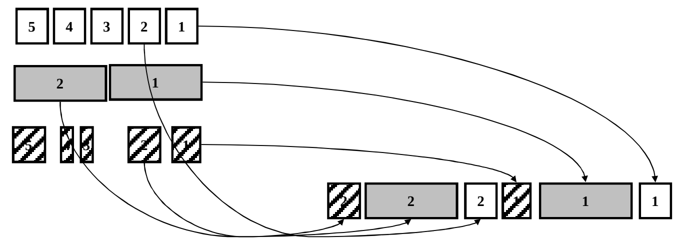
Mixed traffic in the network [1]

- Different applications:
  - traffic (generation) profiles
  - traffic timing constraints
- Routers use FCFS queues:
  - no knowledge of application
  - no knowledge of traffic patterns
- Different traffic types share same network path
- Consider three different applications …



Mixed traffic in the network [2]

- Router:
  - 3 input lines: serviced round-robin at router
  - 1 output line (1 output buffer)



Mixed traffic in the network [3]

- Different traffic patterns:
  - different applications
  - many uses of an application
  - different requirements
- Traffic aggregation:
  - core: higher aggregation
  - many different sources
  - hard to model
- Routing/forwarding:
  - destination-based
  - single metric for all traffic
  - queuing effects
- Large packet size:
  - good for general data
  - “router friendly”
  - “slows” real-time traffic
- Small packet size:
  - good for real-time data
  - less end-to-end delay
  - better tolerance to loss
  - (less jitter?)
  - less efficient (overhead)
  - “not router-friendly”

Delay [1]

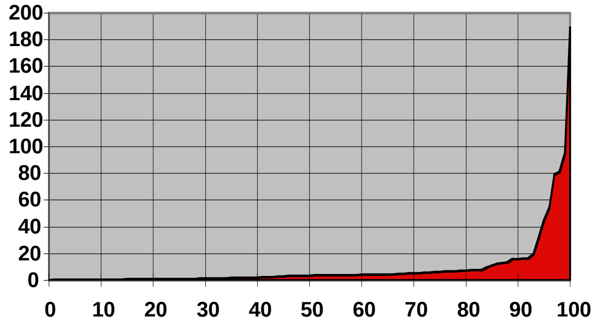
End-to-end delay

- Propagation:
  - speed-of-light
- Transmission:
  - data rate
- Network elements:
  - buffering (queuing)
  - processing
- End-system processing:
  - application specific

Delay bounds?

- Internet paths:
  - “unknown” paths
  - dynamic routing
- Other traffic:
  - traffic patterns
  - localised traffic
  - “time-of-day” effects
- Deterministic delay:
  - impractical but not impossible

Delay [2] #picture#



DigiComm II

Jitter (delay jitter) [1]

End-to-end jitter

- Variation in delay:
  - per-packet delay changes
- Effects at receiver:
  - variable packet arrival rate
  - variable data rate for flow
- Non-real-time:
  - no problem
- Real-time:
  - need jitter compensation

Causes of jitter

- Media access (LAN)
- FIFO queuing:
  - no notion of a flow
  - (non-FIFO queuing)
- Traffic aggregation:
  - different applications
- Load on routers:
  - busy routers
  - localised load/congestion
- Routing:
  - dynamic path changes

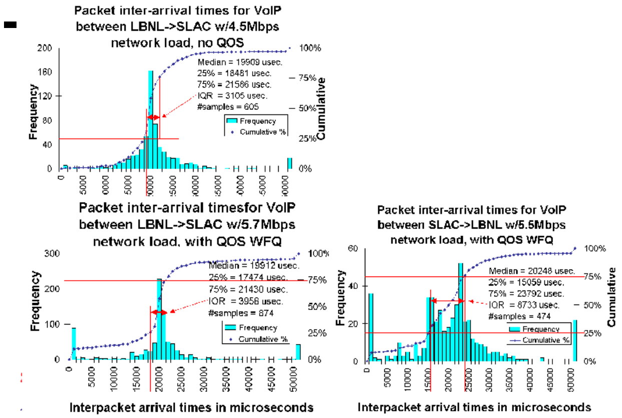
DigiComm II

Jitter (delay jitter) [2] #picture#

- Easiest measure is the variance in inter-packet delay
- Can use lots of other metrics (e.g. other moments of the inter-arrival time distribution
- Not critical to protocols like TCP unless jitter is 1st order (I.e. not 2nd order) effect
- Critical to voice
- (e.g. playout buffer to make sure D2A device or display is not starved…)
- VOIP, Video over IP
- Critical for interactive

DigiComm II

VoIP jitter LBNL<=>SLAC with load



DigiComm II

Loss [1]

End-to-end loss

- Non-real-time:
  - re-transmission, e.g.: TCP
- Real-time:
  - forward error correction and redundant encoding
  - media specific "fill-in" at receiver
- Adaptive applications:
  - adjust flow construction

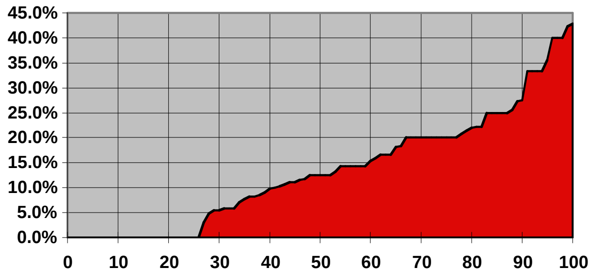
Causes of loss

- Packet-drop at routers:
  - congestion
- Traffic violations:
  - mis-behaving sources
  - source synchronisation
- Excessive load due to:
  - failure in another part of the network
  - abnormal traffic patterns, e.g. "new download"
- Packet re-ordering may be seen as loss

Loss [2] #picture#

- Varies with route
- Depends on link quality
- Depends on router quality
- Varies with time
- Depends on other load
- And interference

Error rates



Data rate [1]

End-to-end data rate

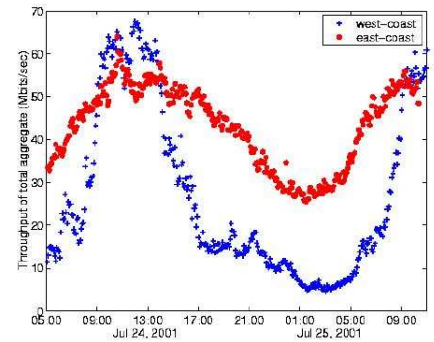
- Short-term changes:
  - during the life-time of a flow, seconds
- Long-term changes:
  - during the course of a day, hours
- Protocol behaviour:
  - e.g. TCP congestion control (and flow control)

Data-rate changes

- Network path:
  - different connectivity
- Routing:
  - dynamic routing
- Congestion:
  - network load – loss
  - correlation with loss and/or delay?
- Traffic aggregation:
  - other users
  - (time of day)

Data rate [2] #picture#

- Link is multiplexed so rate is not just link speed
- Varies with overall load depending on the link sharing rules
- Note latency (RTT) contributed to throughput (e.g. if you have to wait for acks…)
- Lots of different possibile basic link speeds depending on media, modulation, and protocol stack overheads
- “Goodput” is often used for the residual throughput after you allow for all overheads (including retransmissions)



Network probing: a quick note

- Can use probes to detect:
  - delay
  - jitter
  - loss
  - data rate
- Use of network probes:
  - ping
  - traceroute
  - pathchar
- Probes load the network, i.e the affect the system being measured
- Measurement is tricky!
- See:
  - www.caida.org
  - www.nlanr.net

Elastic applications

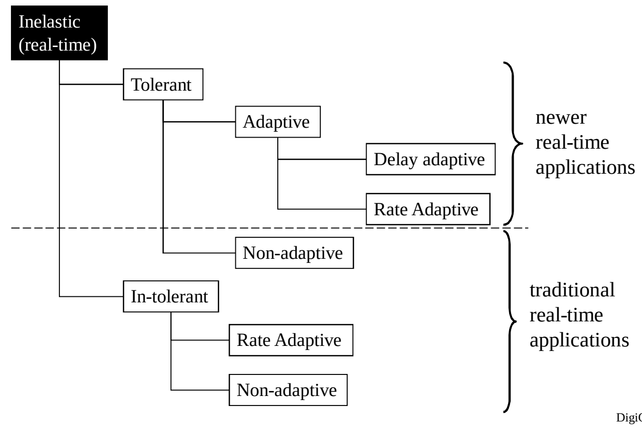
Elastic

- Interactive — e.g. Telnet, X-windows
- Interactive bulk — e.g. FTP, HTTP
- Asynchronous — e.g. E-mail, voice-mail

Examples of elastic applications

- E-mail:
  - asynchronous
  - message is not real-time
  - delivery in several minutes is acceptable
- File transfer:
  - interactive service
  - require “quick” transfer
  - “slow” transfer acceptable
- Network file service:
  - interactive service
  - similar to file transfer
  - fast response required
  - (usually over LAN)
- WWW:
  - interactive
  - file access mechanism(!)
  - fast response required
  - QoS sensitive content on WWW pages

Inelastic applications



Examples of inelastic applications

- Streaming voice:
  - not interactive
  - end-to-end delay not important
  - end-to-end jitter not important
  - data rate and loss very important
- Real-time voice:
  - person-to-person
  - interactive
- Important to control:
  - end-to-end delay
  - end-to-end jitter
  - end-to-end loss
  - end-to-end data rate

QoS parameters for the Internet [1]

Delay

- Not possible to request maximum delay value
- No control over end-to-end network path
- Possible to find actual values for:
  - maximum end-to-end delay, D_{MAX}
  - minimum end-to-end delay, D_{MIN}

Jitter

- Not possible to request end-to-end jitter value
- Approximate maximum jitter:
  - $D_{MAX} – D_{MIN}$
  - evaluate D_{MIN} dynamically
  - D_{MAX}? 99th percentile?
- Jitter value:
  - transport-level info
  - application-level info

QoS parameters for the Internet [2]

Loss

- Not really a QoS parameter for IP networks
- How does router honour request?
- Linked to data rate:
  - hard guarantee?
  - probabilistic?
  - best effort?
- (Traffic management and congestion control)

Packet size

- Restriction: path MTU
- May be used by routers:
  - buffer allocation
  - delay evaluation

QoS parameters for the Internet [3]

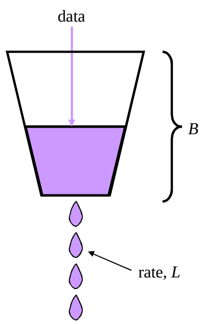
- Data rate:
  - how to specify?
- Data applications are bursty:

$$\frac{\text{peak data rate}}{\text{mean data rate}} \gg 1$$

- Specify mean data rate?
  - peak traffic?
- Specify peak data rate?
  - waste resources?
- Real-time flows:
  - may be constant bit rate
  - can be variable bit rate
- Application-level flow:
  - **application data unit (ADU)**
- Data rate specification:
  - application-friendly
  - technology neutral

Leaky bucket

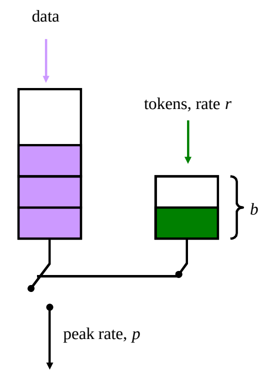
- Two parameters:
  - *B*: bucket size [Bytes]
  - *L*: leak rate [B/s or b/s]
- Data pours into the bucket and is leaked out
- *B*/*L* is maximum latency at transmission
- Traffic always constrained to rate *L*



Token bucket

Token bucket

- Three parameters:
  - *b*: bucket size [B]
  - *r*: bucket rate [B/s or b/s]
  - *p*: peak rate [B/s or b/s]
- Bucket fills with tokens at rate *r*, starts full
- Presence of tokens allow data transmission
- Burst allowed at rate *p*
- data sent < *r*t + *b*



Real-time media flows

Interactive, real-time media flows

- Audio/video flows:
  - streaming audio/video
  - use buffering at receiver
- Interactive real-time:
  - only limited receiver buffering
  - delay <200ms
  - jitter <200ms
  - keep loss low
- Effects of loss:
  - depend on application, media, and user
- Audio:
  - humans tolerant of “bad” audio for speech
  - humans like “good” audio for entertainment
- Video:
  - humans tolerant of “low” quality video for business
  - humans like “high” quality video for entertainment
- Audio – video sync:
  - separate flows?

Audio

QoS requirements

- Delay < 400ms:
  - including jitter
- Low loss preferable:
  - loss tolerant encodings exist
- Data rates:
  - speech ≤ 64Kb/s
  - “good” music ≥ 128Kb/s
- Time domain sampling
- Example – packet voice:
  - 64Kb/s PCM encoding
  - 8-bit samples
  - 8000 samples per second
  - 40ms time slices of audio
  - 320 bytes audio per packet
  - 48 bytes overhead
(20 bytes IP header)
(8 bytes UDP header)
(20 bytes RTP header)
  - 73.6Kb/s

Example audio encoding techniques

G.711

- PCM (non-linear)
- 4KHz bandwidth
- 64Kb/s

G.722

- SB-ADPCM
- 48/56/64Kb/s
- 4-8KHz bandwidth

G.728

- LD-CELP
- 4KHz bandwidth
- 16Kb/s

G.729

- CS-ACELP
- 4KHz bandwidth
- 8Kb/s

G.723.1

- MP-MLQ
- 5.3/6.3Kb/s
- 4KHz bandwidth

GSM

- RPE/LTP
- 4KHz
- 13Kb/s

Video

QoS requirements

- Delay < 400ms:
  - including jitter
  - same as audio
  - inter-flow sync
- Loss must be low
- Data rate – depends on:
  - frame size
  - colour depth
  - frame rate
  - encoding
- Frequency domain:
  - discrete cosine transform (DCT)
- Example - packet video:
  - ###

Example video encoding techniques

MPEG1

- upto 1.5Mb/s

MPEG2

- upto 10Mb/s (HDTV quality)

MPEG4

- 5-64Kb/s (mobile, PSTN)
- 2Mb/s (TV quality)
- MPEG7, MPEG21

H.261 and H.263

- $n \times$ 64Kb/s, $1 \leq n \leq 30$

Summary

- IPv4 and current Internet:
  - not designed for QoS support
- Need to add support for ISN:
  - service definitions
  - signalling
  - update routers
- Need to describe traffic:
  - QoS parameters
- Audio and video have different requirements