# QoS

# QoE and QoS

Customers are willing to pay for the quality of the service received

This should ideally be the **Q**uality **o**f **E**xperience

*the acceptability of a service as perceived subjectively by the end-user*

However, there are problems with directly defining / measuring QoE

and for the same network, QoE depends on application

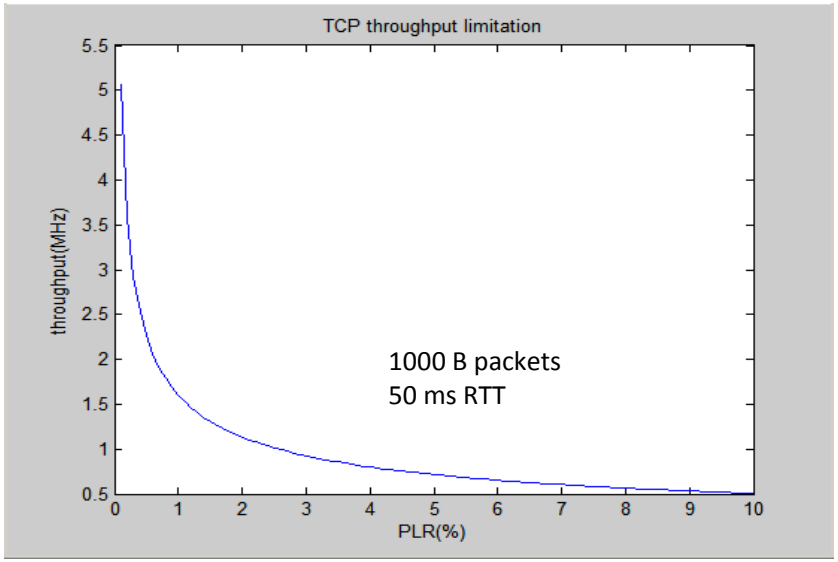
For example

- non-real-time TCP traffic suffers data-rate reduction under packet loss, but is relatively immune to increase in delay
- real-time interactive traffic employing **P**acket **L**oss **C**oncealment is relatively immune to low rates of packet loss, but suffers from delay
- timing distribution is immune to delay
  but sensitive to low frequency packet delay variation

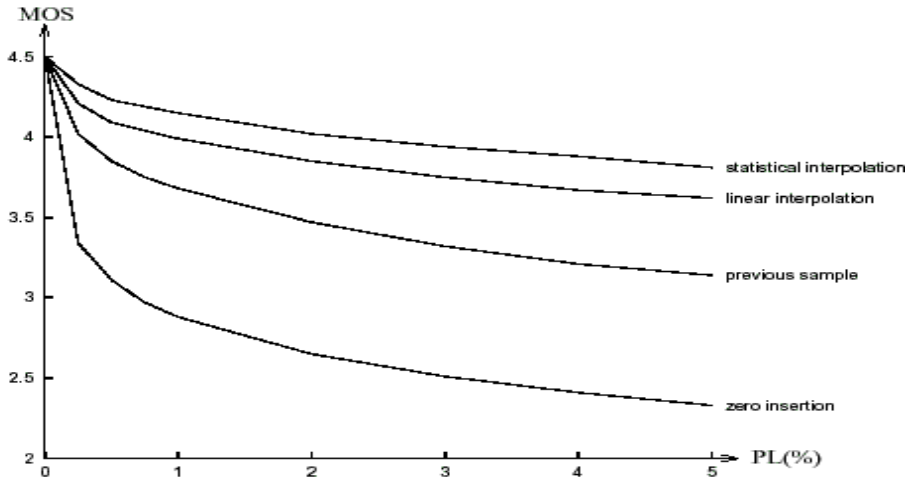
So instead, we use **Q**uality **o**f **S**ervice as a proxy for QoE

QoS means a set of *objective network parameters* that are *easily measured*

# Example effects of QoS on QoE




TCP rate reduction with PLR




TDMoIP voice quality reduction with PLR

# Service Level Agreements

In order to justify recurring payments
the provider agrees to a minimum level of service in an SLA

An SLA is a *legal commitment* between a service provider (SP)
and a customer, for example:

- Telco and subscriber
- ISP and Internet user
- VPN operator and enterprise
- cloud application provider and cloud user

SLAs typically include (financial) penalties for *breaches*

If objectives or penalties are too low, SLA is useless

If objectives or penalties are too high, cost will be prohibitive

Badly defined SLAs may damage operations by setting incorrect goals

# SLAs and QoS parameters

SLAs detail measurable network parameters
that *influence* (correlate with) QoE, such as :

Connectivity parameters
- availability (e.g., the famous five nines)
- time to repair (e.g., the famous 50 ms)

Noise (error) level parameters
- SNR
- BER
- Packet Loss Ratio
- defect densities

Information rate parameters
- bandwidth, throughput, goodput

Information latency parameters
- 1-way delay
- round trip delay

**performance parameters**

# Packet Loss

We often speak of **P**acket **L**oss **R**atio, but packet loss needn't be IID

- for wireless, physical layer BER may lead to significant packet loss
- for modern fiver-optic transmission, physical layer packet loss is rare

Packet loss is often caused by

- overflow of (improperly configured) buffers
- deliberate packet discard for SLA policing
- deliberate packet loss to avoid congestion collapse - (W)RED

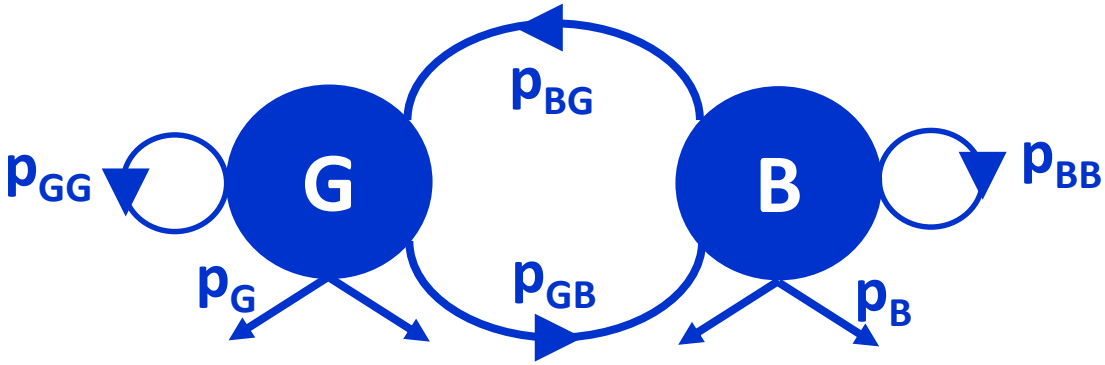
and such packet loss is bursty

QoE may depend on the packet loss' distribution, not only the PLR

- bursty loss with TCP may cause large data-rate cut-back
- bursty loss with VoIP may cause PLC algorithms to fail

# Gilbert-Elliot Model

One common way to describe bursty packet loss is the **Gilbert-Elliot model**




GE is a Markov model with 2 states

- **G** (good, low-loss state)
  - with small probability $p_G$ for packet loss, and $1-p_G$ for no loss
- **B** (bad, lossy state)
  - with large probability $p_B$ for packet loss, and $1-p_B$ for no loss

and transition probabilities $p_{GG}$ $p_{GB}$ $p_{BG}$ $p_{BB}$

Given a sequence of packet loss events, the probabilities can be estimated

The GE probabilities provide more information than PLR

# Throughput (data-rate)

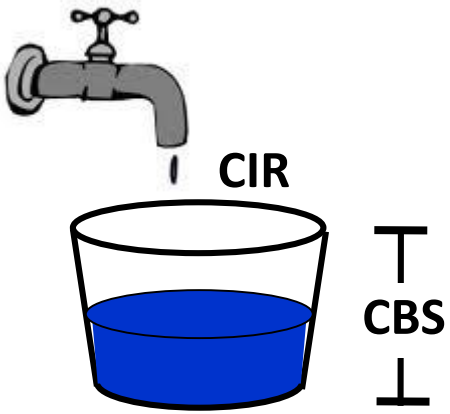
What does X bps mean for packet switched networks ?

The physical layer is always the same bit-rate
(e.g., 100 Mbps, 1 Gbps, 10 Gbps, 100 Gbps)
but packets are not necessarily sent periodically

If a SLA specifies a X bps rate

- can you transmit 10% of the time at 10X bps ?
- can you transmit 1% of the time at 100X bps ?

Rather than specifying an integration time
it is conventional to use token bucketing algorithms



# Connectivity vs. *the rest*

Basic connectivity (availability) **always** influences QoE

The other parameters **may** influence QoE
depending on service/ application (voice, video, browsing, …)

*Some* services only require basic connectivity

*Some* also require minimum available throughput

*Some* require delay less then some end-end (or RT) delay

*Some* require packet loss ratio (PLR) less than some percentage

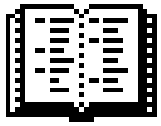
Note: these parameters
are not necessarily independent

For example, TCP throughput drops with PLR

# Some rules of thumb

**M**ission **C**ritical (and life critical) services require

- high availability



If there are any MC services
then system traffic requires high availability too

MC services do not necessarily require strict throughput
but always indirectly require

- a certain minimal average throughput
- bounded delay

If the MC service uses TCP then it requires

- low PLR

Real-time services require

- sufficient throughput

but not necessarily low PLR (audio and video codecs have PLC)

Interactive services require

- low RT delay

# QoS monitoring

RECAP: SLA compliance is the SP's **justification for payment**

To ensure SLA compliance, the SP must :

- monitor the SLA parameters
- take action if parameter is dropping below compliance levels

But how does the SP verify/ensure that the SLA is being met ?

Monitoring is carried out using

**O**perations, **A**dministration, **M**aintenance (**OAM**)

The customer too may use OAM to check that the SP is compliant !

**Technical note:**

OAM is a ***user-plane*** function

but may influence control and management plane operations

for example

- OAM may trigger protection switching, but doesn't switch
- OAM may detect provisioned links, but doesn't provision them

# OAM – FM and PM

The difference between connectivity and performance parameters leads to two types of OAM :

1. **F**ault **M**onitoring **required for maintenance of connectivity** (availability)
   - detection and reporting of *anomalies*, *defects*, and *failures*
   - FM runs continuously/periodically at required rate
   - used to trigger mechanisms in the
     - control plane (e.g. protection switching) and
     - management plane (alarms)
2. **P**erformance **M**onitoring **required for maintenance of all other QoS parameters**
   - measurement of performance criteria (delay, PDV, etc.)
   - PM may run :
     - before enabling a service
     - on-demand
     - per schedule

# QoS assurance : availability

The difference between connectivity and performance parameters leads to 2 types of QoS assurance – availability and performance

Availability is usually specified in “nines”

| nines | up % | permitted down time | typical service |
|---|---|---|---|
| 3 nines | 99.9% | < 7 hour 18 min / month | electric power service |
| 4 nines | 99.99% | < 44 minutes / month | |
| 5 nines | 99.999% | < 4 min 23 sec / month | PSTN |
| 6 nines | 99.9999% | < 26 sec / month | |

In order to ensure high availability, one employs

- FM OAM
- **A**utomatic **P**rotection **S**witching (APS)

# QoS assurance : performance

There are two main approaches to ensuring performance QoS

Integrated Services (guaranteed QoS) – ***hard* QoS**

- define traffic flows (CO approach)
- guarantee QoS attributes for each flow
- reserve resources at each router along the flow
- signaling protocol (e.g., RSVP) needed

$\int \mathbf{dt}$

Differentiate Services (statistical QoS) – ***soft* QoS**

- retain CL paradigm
- no guaranteed QoS attributes
- no resource reservation
- *mark* packets (*differentiated* – e.g., gold, silver, bronze)
  - marking can be by VLAN, P-bits, IP-ToS/DSCP, or general "flow"
- offer special treatment (priority) relative to other packets

$\frac{\boldsymbol{d}}{\boldsymbol{dt}}$

DiffServ is the preferred approach for Ethernet and IP

IntServ is used in MPLS-TE, some SDN scenarios

# QoE

# QoE and MOS

ITU-T defines QoE as

*the acceptability of a service*

*as perceived subjectively by the end-user*

A well-known QoE measure for telephony-grade voice is **M**ean **O**pinion **S**core (MOS) (ITU-T P.800)

MOS is measured by having a number of listeners listen and **score** speech on a scale from **1** (bad) to **5** (excellent) and averaging over these scores (finding the **mean**)

- *Toll quality* voice has **MOS = 4**
- *Cellphone voice* has **MOS ≈ 3.5**
- *Synthetic* or *military* voice has **MOS = 2** and below

# QoE and QoS

QoE for a given application is a function of QoS parameters

$$QoE = f(\text{service}; QoS_1, QoS_2, \ldots QoS_n)$$

Researchers have found various functional forms
for the dependence of QoE on a particular QoS parameter

| form | expression | examples |
|---|---|---|
| *Linear* | $QoE \sim QoS_k$ | perceived download time vs. PLR |
| *Logarithmic* | $QoE \sim \log(QoS_k)$ | perceived download time vs. datarate |
| *Exponential* | $QoE \sim \exp(QoS_k)$ | VoIP MOS vs. PLR |
| *Power Law* | $QoE \sim QoS_k^p$ | perceived streaming video quality vs. PDV |

see e.g., work of Markus Fiedler (BTH, Sweden)

# Absolute vs. Comparative QoE

QoE measures may be

absolute determined by observing the degraded message or

comparative determined by comparing the degraded message to the original

Comparative measures are often more accurate

but can not be used unintrusively on a live network scenarios

Absolute measures can be used *single-ended (non-intrusively)*

MOS variations

**A**bsolute **C**ategory **R**ating (ACR) : listeners hear only the degraded speech

**D**egradation **C**ategory **R**ating (DCR) : listeners hear first the original and then the degradation score 1 = very annoying degradation … 5 inaudible degradation

**C**omparative **C**ategory **R**ating (CCR) : listeners hear the original and the degraded speech in random order and score

-3 2nd is much worse than 1st … 3 2nd is much better than 1st

Even simpler : AB test – simply report which sounds better

# Subjective vs. Objective QoE

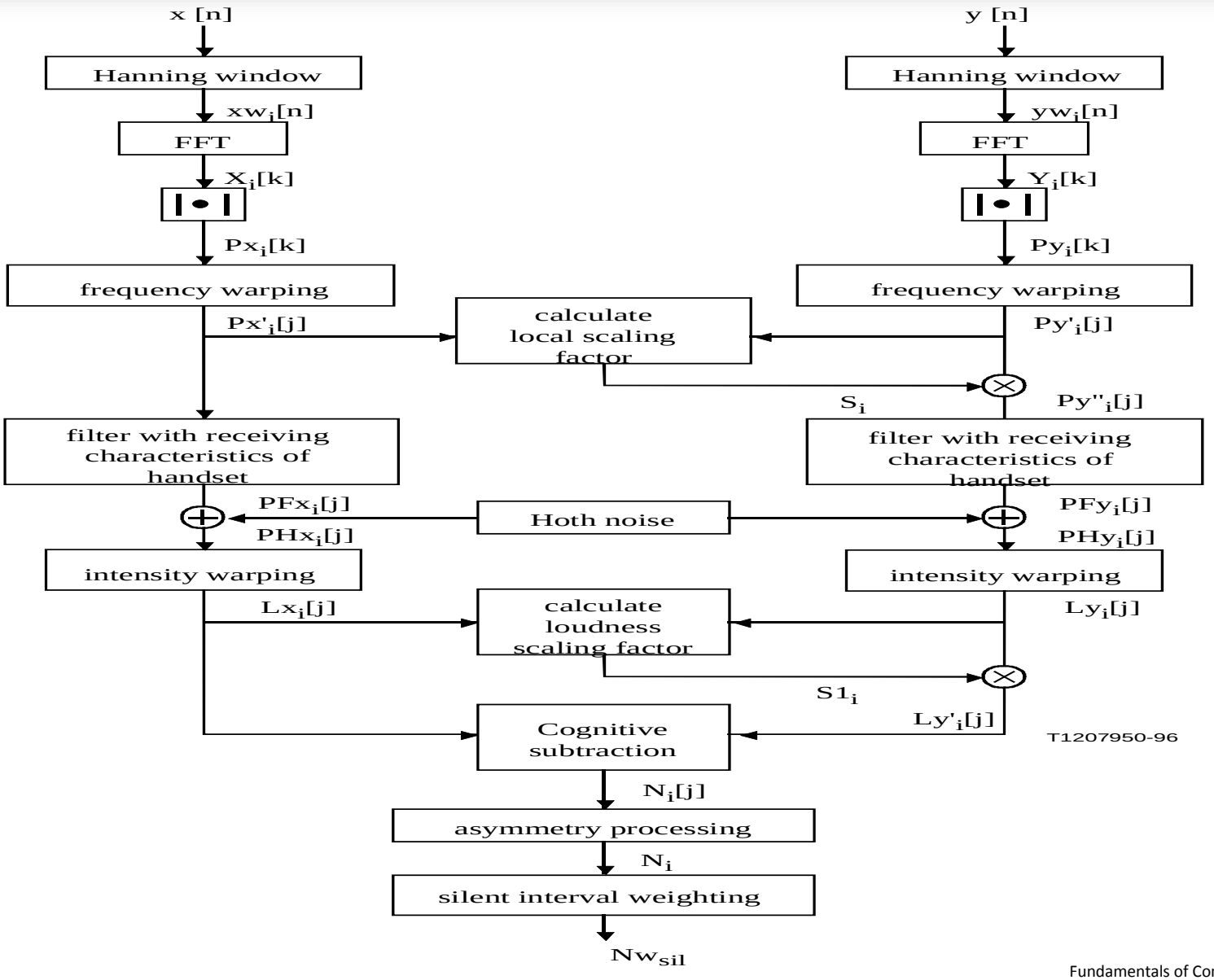
Direct human QoE scoring is expensive and time-consuming
ITU-T has defined *objective measures* that can be *automated*
These entail algorithms that produce scores
    that *correlate well with human QoE*

PSQM (ITU-T P.861) and PESQ (ITU-T P.862)
    are *objective comparative* MOS-like measures for telephone grade speech
They model the human auditory perception system (Bark scale, masking, etc.)

PEAQ (ITU-R BS-1387) similarly scores wideband audio
These were selected in competitions
    to have highest correlation with human MOS

ITU-T P.563 is a *single-ended* (absolute) objective MOS-like score
It determines un-naturalness of telephone-grade speech sounds
    and the amount of non-speech-like noise

# PSQM processing

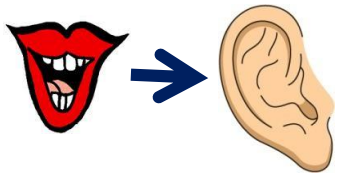


# E-model

The E-model defined in ITU-T G.107 is a *planning* tool

It predicts a "mouth-to-ear" *transmission rating factor* R

- between 0 and 100
- higher values signify better voice quality
- should be uniquely convertible to a MOS level



$R = f(QoS_1, \ldots QoS_n)$ and is additive in individual $QoS_k$ degradations

R starts with the basic signal to noise ratio

R is reduced to account for various impairments, including

- simultaneous impairments (loudness, sidetone, clipping, quantization noise)
- delay impairments (delay, echo delay and loudness)
- equipment impairments (codec distortion, packet loss)

R is increased when there are additional advantages such as mobility (cellphone receives A=10)

$$\mathbf{R = R_0 - I_s - I_d - I_e + A}$$

# R value meanings

| R values | meaning | Equivalent MOS |
|---|---|---|
| 90 - 100 | Very satisfied | 4.3-5.0 |
| 80- 90 | Satisfied | 4.0-4.3 |
| 70-80 | Some users dissatisfied | 3.6-4.0 |
| 60-70 | Many users dissatisfied | 3.1-3.6 |
| 50-60 | Nearly all users dissatisfied | 2.6-3.1 |
| Below 50 | Not recommended | 1-2.6 |

# VQMON

VQmon is a single-ended method
for estimating the E-model factors for VoIP audio, based on
- codec type
- packet loss statistics (Gilbert-Elliot parameters)
- delay

See:
- ETSI TIPHON TS 101 329-5 Annex E
- ITU-T G.113

Takes human perception phenomena into account (e.g., recency effect)

VQmon was later extended to
- non-telephony audio (MOS-A)
- video (MOS-V)
- audio-video (MOS-AV)

# Video quality

ITU-R produced BT.500 for subjective assessment of TV quality

Similar to MOS :

- television sequences are shown to a group of viewers
- subjective opinions are averaged

ITU-T has produced many Recommendations for video and multimedia quality :

- Subjective (P.9xx, J.140)
- Objective (J.143, J.144, J.147, J.148, J.24x, J.34x)

Since 1997 the **V**ideo **Q**uality **E**xperts **G**roup (VQEG)
has been producing standards and tutorials

**PEVQ** (J.247) is a comparative pixel by pixel objective measure
that models the human visual tract
and returns a 5-point MOS score and further KPIs

# QoE for other applications

**G.1011** is a reference guide to existing standards for QoE and provides a taxonomy


**G.1010** discusses many applications, including

- conversational voice, voice messaging, streaming audio
- videophone, one-way video
- web-browsing, bulk data transfer, email, e-commerce,
- interactive games
- SMS, instant messaging

and gives performance targets for delay, PDV, and PLR

**G.1050** gives an IP network model for evaluating the performance of IP streams based on QoS parameters (delay, PDV, PLR).

**J.163** treats real-time services over cable modems

**X.140** defines QoS parameters for public data networks

# Network planning tools

In addition to subjective/objective methods to quantify the QoE of a specific (live or simulated) service instance

**Network planners** need tools to predict service quality in order to efficiently allocate resources

**G.1030** provides network planners with end-to-end (E-model-like) tools for applications over IP networks

It includes an appendix devoted to web browsing that presents empirical perception of users to response times and proposes a MOS measure

**G.1070** proposes an algorithm for network planners to estimate videophone quality

# Apdex

The Apdex Alliance is a consortium of companies
functions as a IEEE-ISTO (Industry Standards and Technology Organization)

Apdex develops open standardized methods to
- report
- benchmark and
- track

application performance.

The **Apdex** (Application Performance Index)
- is a number between 0 and 1
- is meant to capture user satisfaction from an application
- 0 means no user would be satisfied
- 1 means that all users would be satisfied

# Apdex (cont.)

To compute the Apdex N users are divided into 3 categories

- satisfied (S users) e.g., web page completely loads within 2 seconds
- tolerating (T users) e.g., web page completely loads within 8 seconds
- frustrated (F users) e.g., web page takes > 8 seconds to load

The Apdex is given by Apdex = ( S + T/2 ) / N

Apdex hierarchically deconstructs application transactions into
sessions processes tasks turns protocols packets

Sessions consist of the entire connect time
Processes are interactions accomplishing a goal
Tasks are individual interactions

The user is mainly aware of the **task response time**
since must wait for the task to complete before proceeding!

# Behavioral QoE

All of the above subjective and objective QoE measures
are service/application-specific.

But new services and applications are created every day
and different users use different features of a single application

So it is no longer feasible to study each application in depth

A new approach is **behavioral QoE estimation**
the user's satisfaction is estimated based on actions / reactions

Example : there is a high measured correlation between
a user being unsatisfied with a service level
his aborting the application
(or at least waiting until the service level improves)

Behavioral QoE can be used *instead* of traditional QoE measurement
or to automatically find QoE(new app, QoS)

# Research topics

- Fault *isolation* is well understood, but performance degradation less so.
  For example, QoE degradation may not be additive, or even monotonic.
  QoS is not well defined when there are parallel paths or multipoint.
  How can we predict QoS/QoE for complex networks?
- What are the QoE(QoS) functions for other applications ?
- QoS routing is an active research area.
  How can one optimize the end-to-end QoS for given services ?
  What if QoS estimates on subnetworks may be highly variable or in error ?
  What can be done by distributed routing
  and what requires centralized path computation ?
- Best effort services are often available free of charge.
  How should QoS-guarantees / promises be priced?