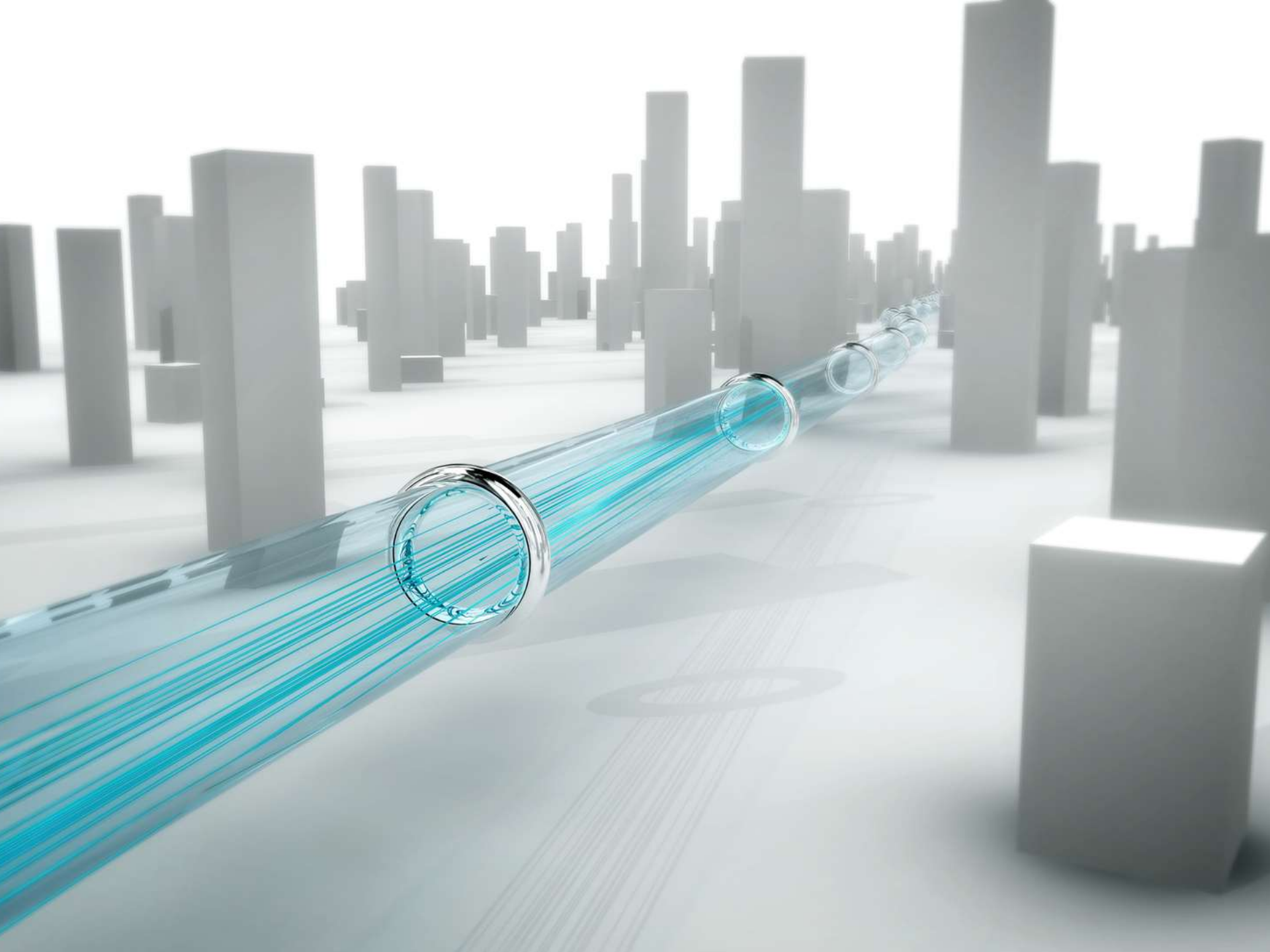


Online Games: is the Internet prepared for them?

GTC
Communication
Technologies Group

Jose Saldana
University of Zaragoza
Spain

Index

- I. The two problems
- II. Online games
- III. Multiplexing FPS
- IV. MMORPG adaptation

Index

- I. The two problems
- II. Online games
- III. Multiplexing FPS
- IV. MMORPG adaptation

The two problems

- The current size of the Internet does not allow us to deploy quick global modifications (e.g. IPv6 is from Dec. 1998)
- The changes must be accepted and progressively introduced, and they have to be backwards compatible

The two problems

- The last big change: January 1st 1983.
  - NCP was substituted by TCP/IP
  - 400 nodes

The two problems

- Which services was the Internet thought for?
  - e-mail, file transfer
- The Internet was not designed for real-time services (delivery deadline).

The two problems

- Traditional services are called *Best-effort*
- This means that the network will try to deliver the information as soon as possible
- But there are no maximum delay guarantees (like a postcard)

The two problems

- But real-time services are being widely used: VoIP, video conference, online gaming

The two problems

- Quality problem: Using a *best effort* network for a real-time service.
- E.g. in IP Telephony, users demand a quality similar to the one they used to have with traditional telephony
- We need to measure quality

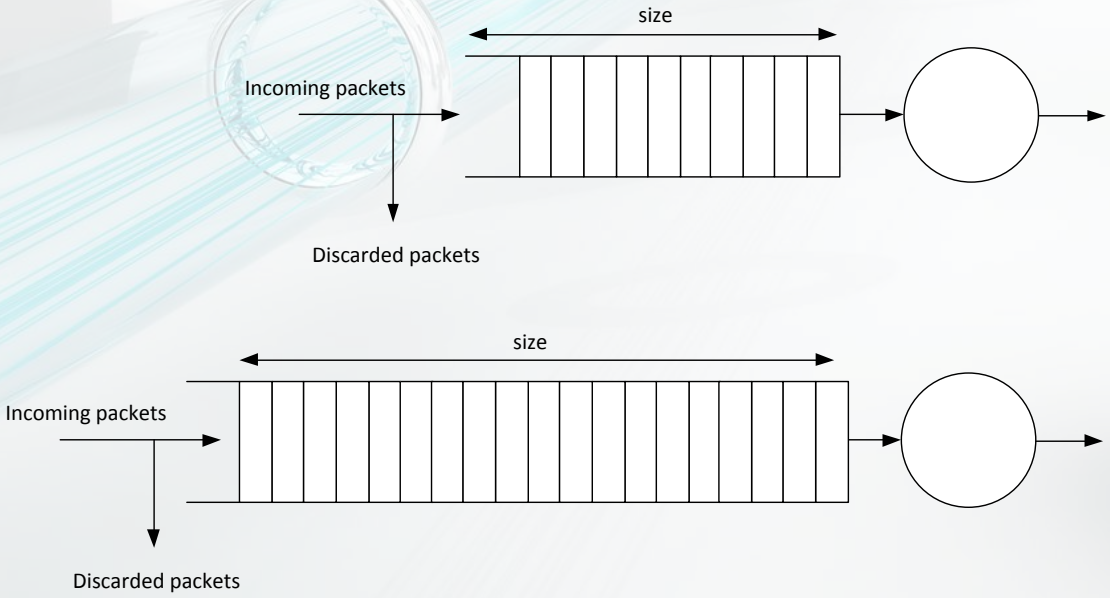
The two problems

- But there are network impairments:
  - Delay: If it is big, interactivity is lost. Causes:
    - Network equipment
    - Applications
    - Speed of light

8,000km / 300,000km/sec = 0,026 sec

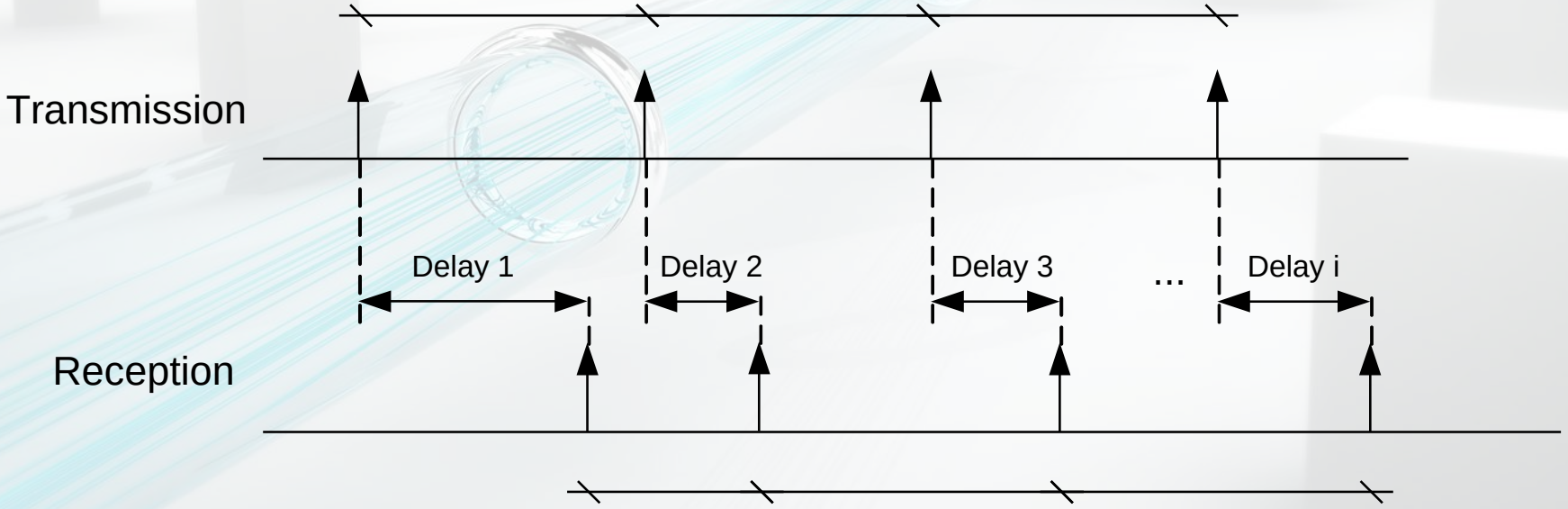
The two problems

- **Packet loss: There are network bottlenecks, so buffers are used. They have a limited capacity, so sometimes they drop packets:**



The two problems

- Network jitter: different delay for different packets. Caused by bursty traffic (e.g. web browsing)

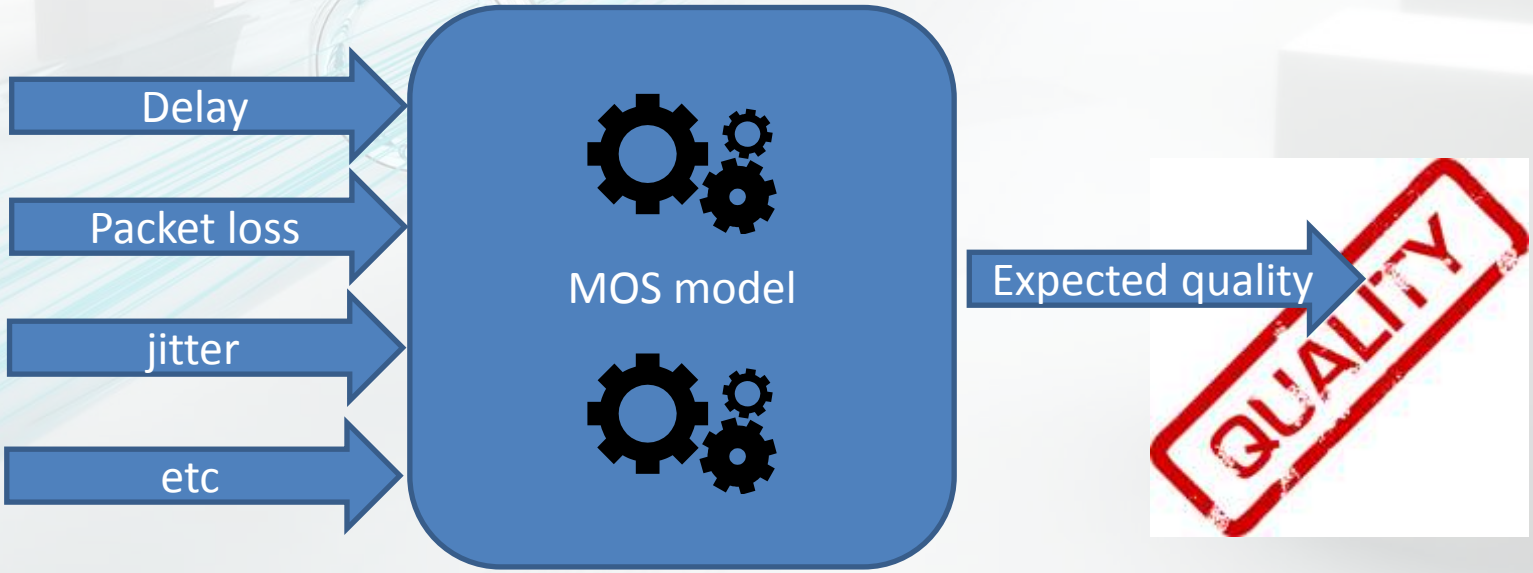


The two problems

- E-Model: ITU's solution that estimates perceived quality of voice, as a function of delay, packet loss, jitter, and other things: codec, etc.
- Battery of surveys in order to obtain a MOS *(Mean Opinion Score)* model

The two problems

- A mathematical model is obtained, and it is able to generate an estimation of the Quality we would obtain with that network conditions

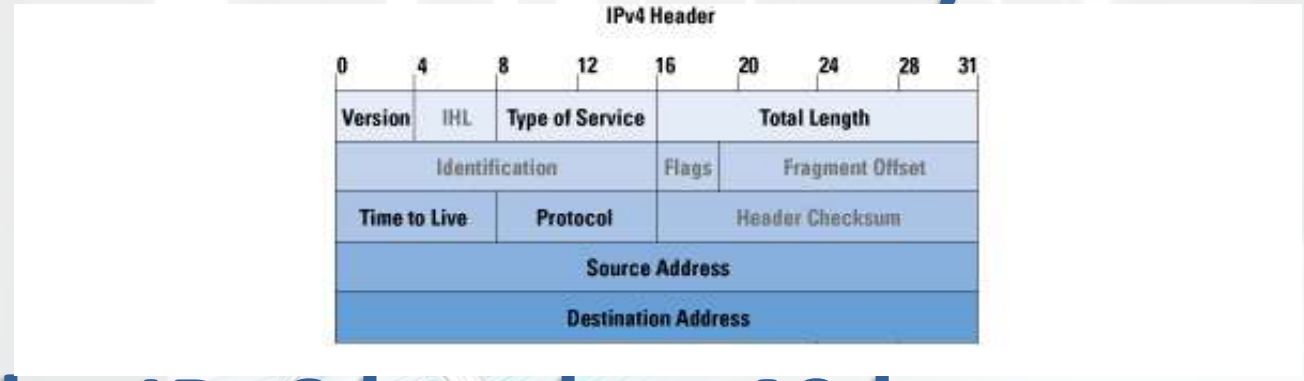


The two problems

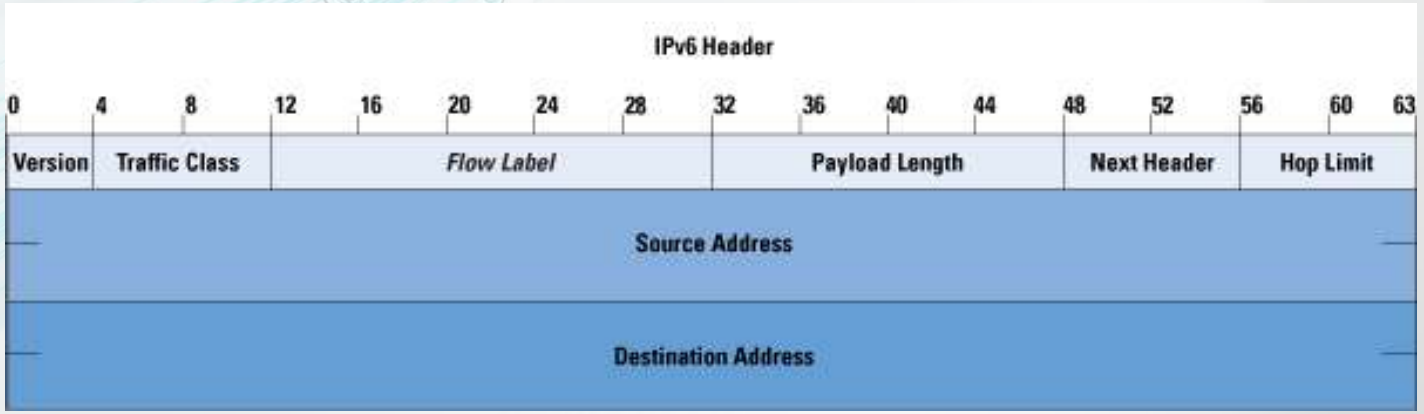
- Efficiency problem
  - In the Internet, every packet needs a header, including its origin, destination, port, etc.
  - It is not my information: it is the *overhead*, which is removed at the destination (envelope)
  - The maximum size is 1,500 bytes.

The two problems

- The IPv4 header: 20 bytes

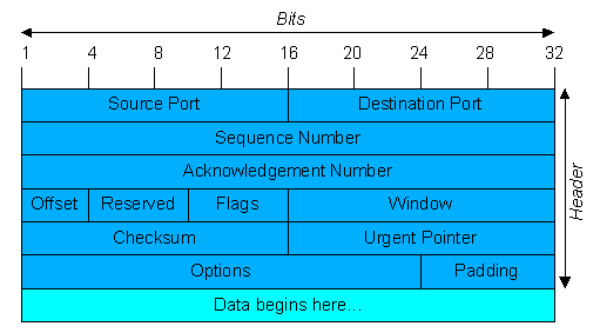


- The IPv6 header: 40 bytes

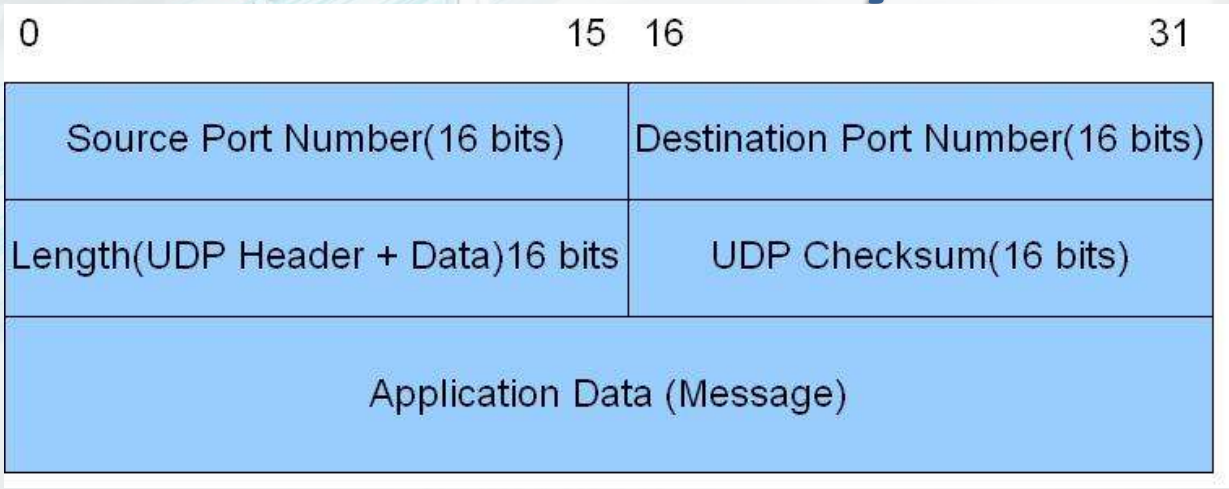


The two problems

- The TCP header: 20 bytes



- The UDP header: 8 bytes



The two problems

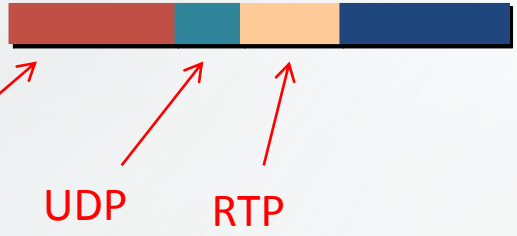
- Efficiency problem
  - E.g. If I have to send a big file, I divide it in **chunks** of 1500 bytes, and I have an efficiency of 97%

The two problems

- Efficiency problem
  - But real-time services cannot wait. They need information every 20 ms, so efficiency gets really bad (33%):

One IPv4/UDP/RTP VoIP packet with two samples of 10 bytes

η=20/60=33%



codec G.729a, 2 samples

The two problems

- Efficiency problem
  - If I use IPv6 (the next version of the protocol, with a huge number of different addresses), the efficiency only drops to 96%

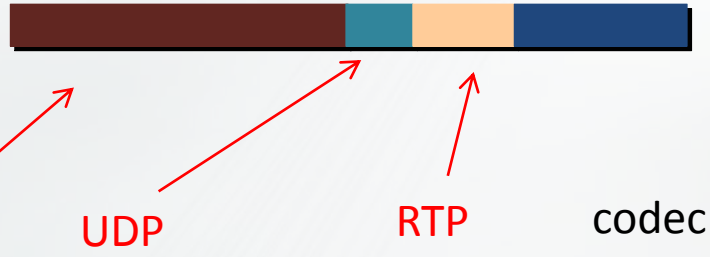
One IPv6/TCP packet 1500 bytes
η=1440/1500=96%

The two problems

- Efficiency problem
  - But for real-time services it becomes even worse: only one byte out of four is useful information!

One IPv6/UDP/RTP packet of VoIP with two samples of 10 bytes

η=20/80=25%



The two problems

But…

Weren't you supposed to talk about online games?

Index

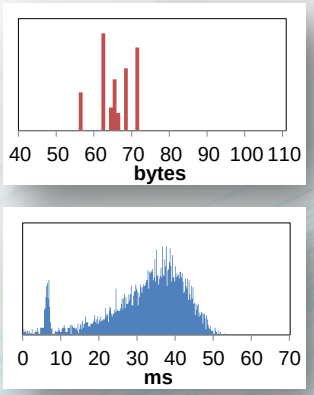
- I. The two problems
- II. Online games
- III. Multiplexing FPS
- IV. MMORPG adaptation

Online games

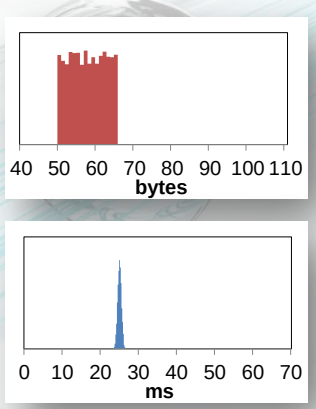
- In fact, online games also have these two problems:
  - Quality problem: Very stringent real-time requirements: Players are really difficult to satisfy

Online games

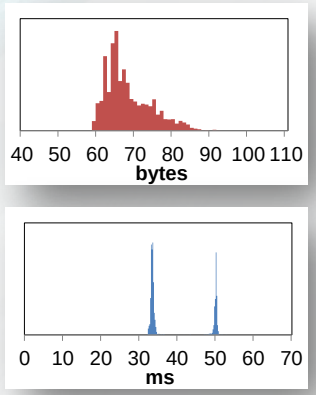
- They generate small packets, so the efficiency problem is also present



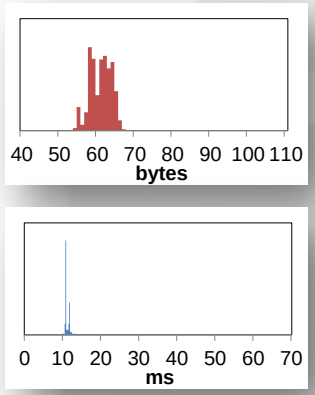
Quake II



Unreal Tournament



Counter Strike I



Quake III

Online games: genres

Real-time strategy

Sports

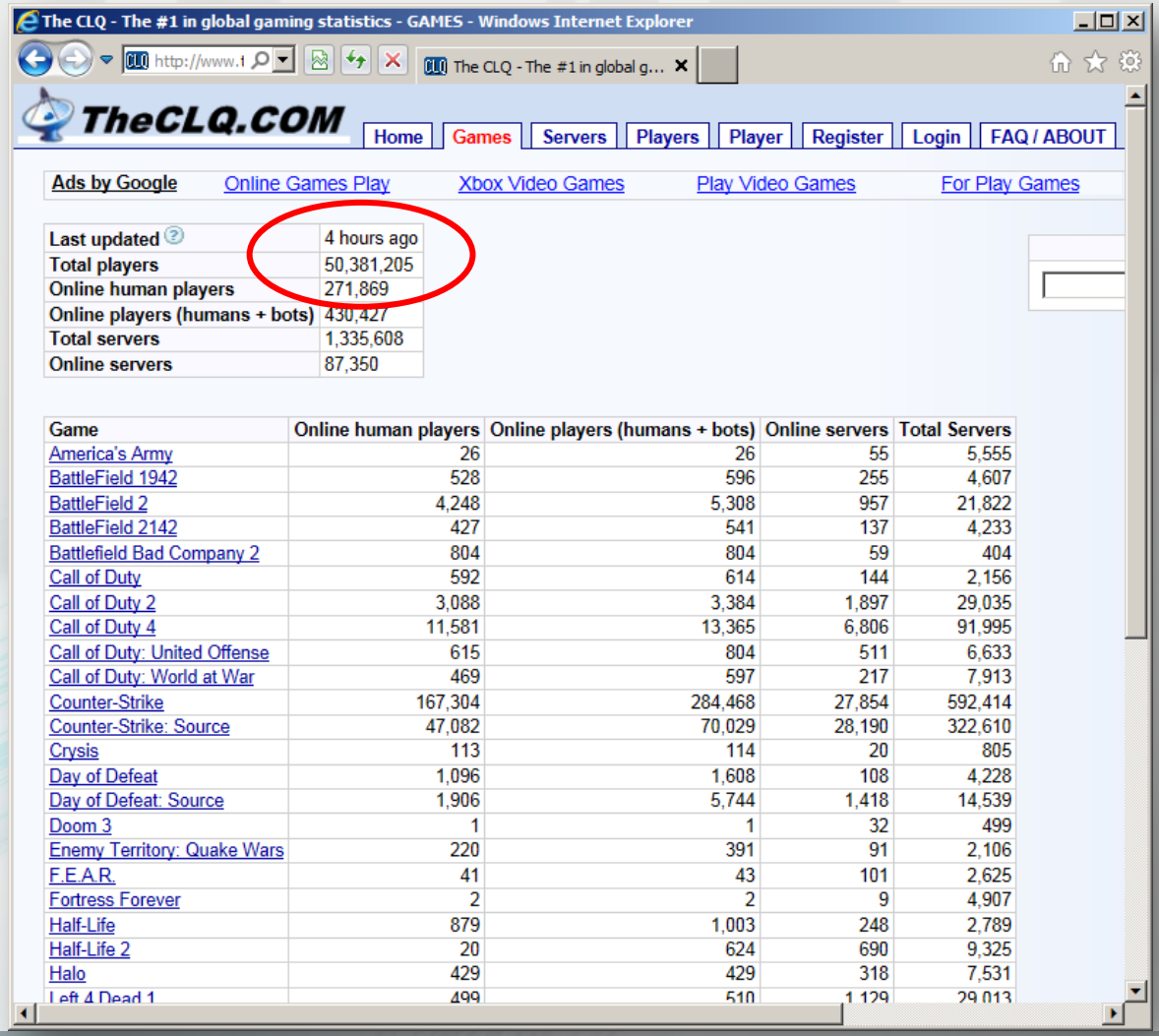
MMORPG

FPS

FPS online games

- First Person Shooters are the ones with the tightest real-time requirements (video)

FPS online games



Last updated	4 hours ago
Total players	50,381,205
Online human players	271,869
Online players (humans + bots)	430,427
Total servers	1,335,608
Online servers	87,350

Game	Online human players	Online players (humans + bots)	Online servers	Total Servers
America's Army	26	26	55	5,555
BattleField 1942	528	596	255	4,607
BattleField 2	4,248	5,308	957	21,822
BattleField 2142	427	541	137	4,233
Battlefield Bad Company 2	804	804	59	404
Call of Duty	592	614	144	2,156
Call of Duty 2	3,088	3,384	1,897	29,035
Call of Duty 4	11,581	13,365	6,806	91,995
Call of Duty: United Offense	615	804	511	6,633
Call of Duty: World at War	469	597	217	7,913
Counter-Strike	167,304	284,468	27,854	592,414
Counter-Strike: Source	47,082	70,029	28,190	322,610
Crysis	113	114	20	805
Day of Defeat	1,096	1,608	108	4,228
Day of Defeat: Source	1,906	5,744	1,418	14,539
Doom 3	1	1	32	499
Enemy Territory: Quake Wars	220	391	91	2,106
F.E.A.R.	41	43	101	2,625
Fortress Forever	2	2	9	4,907
Half-Life	879	1,003	248	2,789
Half-Life 2	20	624	690	9,325
Halo	429	429	318	7,531
Left 4 Dead 1	499	510	1,129	29,013

FPS online games

Delay: Very important

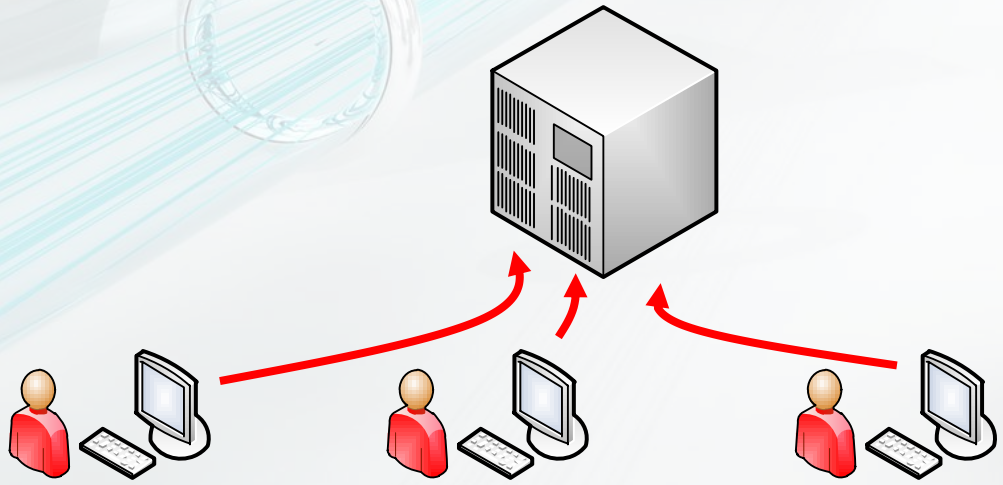
FPS online games

- Traffic characteristics
  - UDP
  - Small packets (100 bytes maximum)
  - High frequency (25 to 85 pps)
    (very similar to Voice over IP)
- A universal MOS does not exist
  - Some games are more sensitive to delay, or packet loss, or jitter, etc.

FPS online games

FPS use a client-server architecture

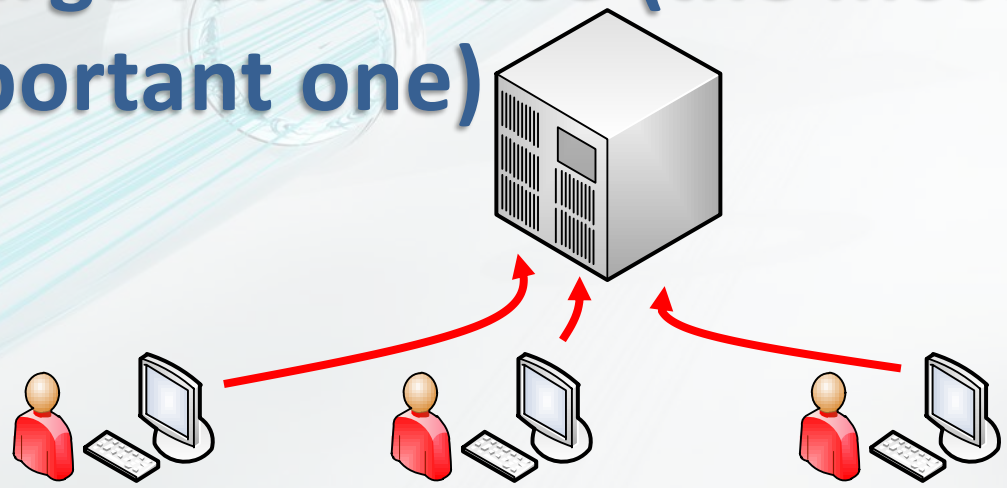
- Consistency of the game
- Avoiding cheating



FPS online games

FPS use a client-server architecture

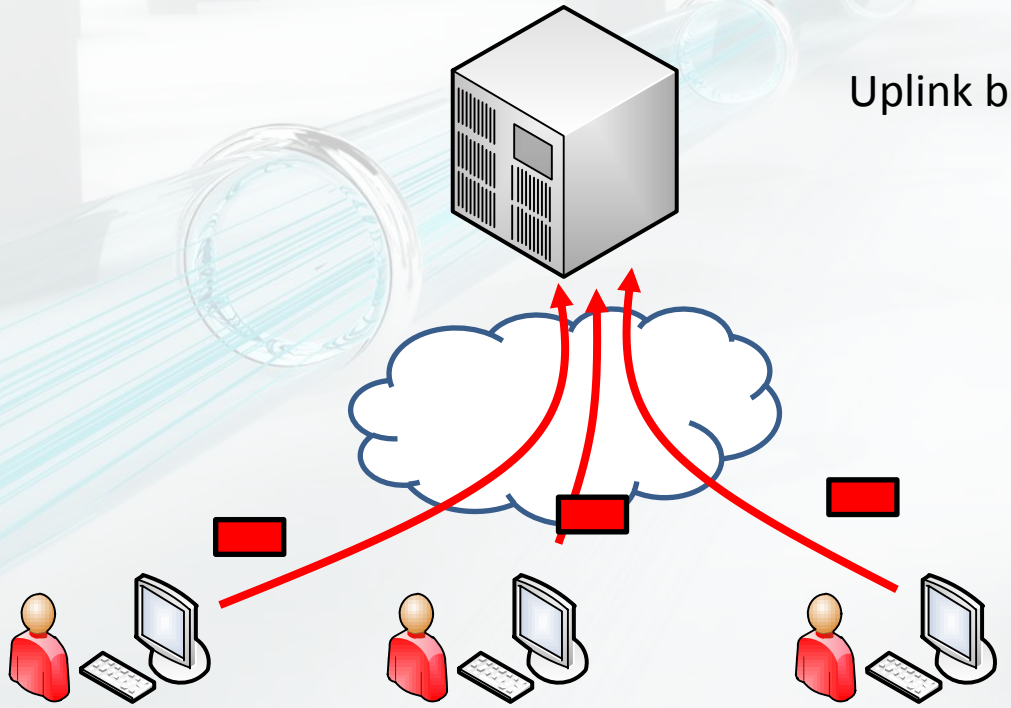
- Consistency of the game
- Avoiding cheating
- Charge for the use (the most important one)



FPS online games

FPS use a client-server architecture

- Three limitations

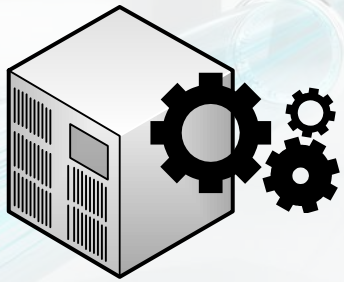


FPS online games

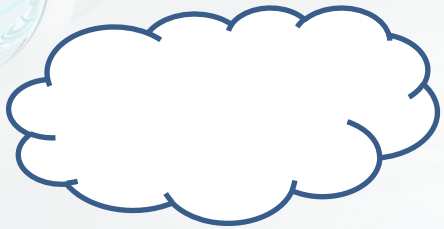
FPS online games

FPS use a client-server architecture

- Three limitations



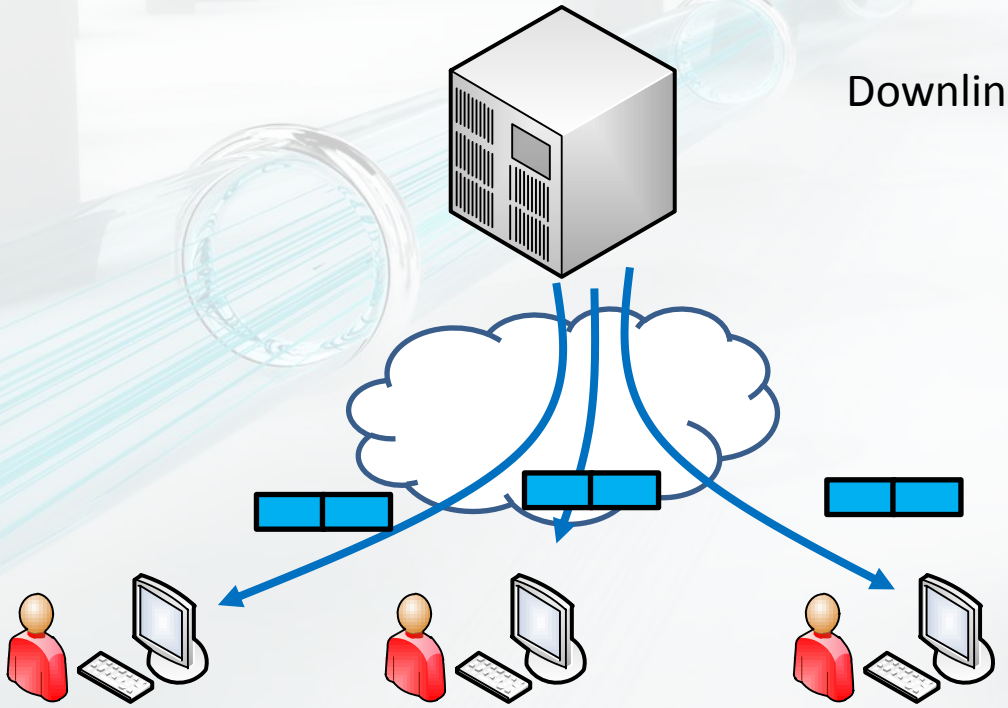
Server processing capacity limit



FPS online games

FPS use a client-server architecture

- Three limitations



FPS online games

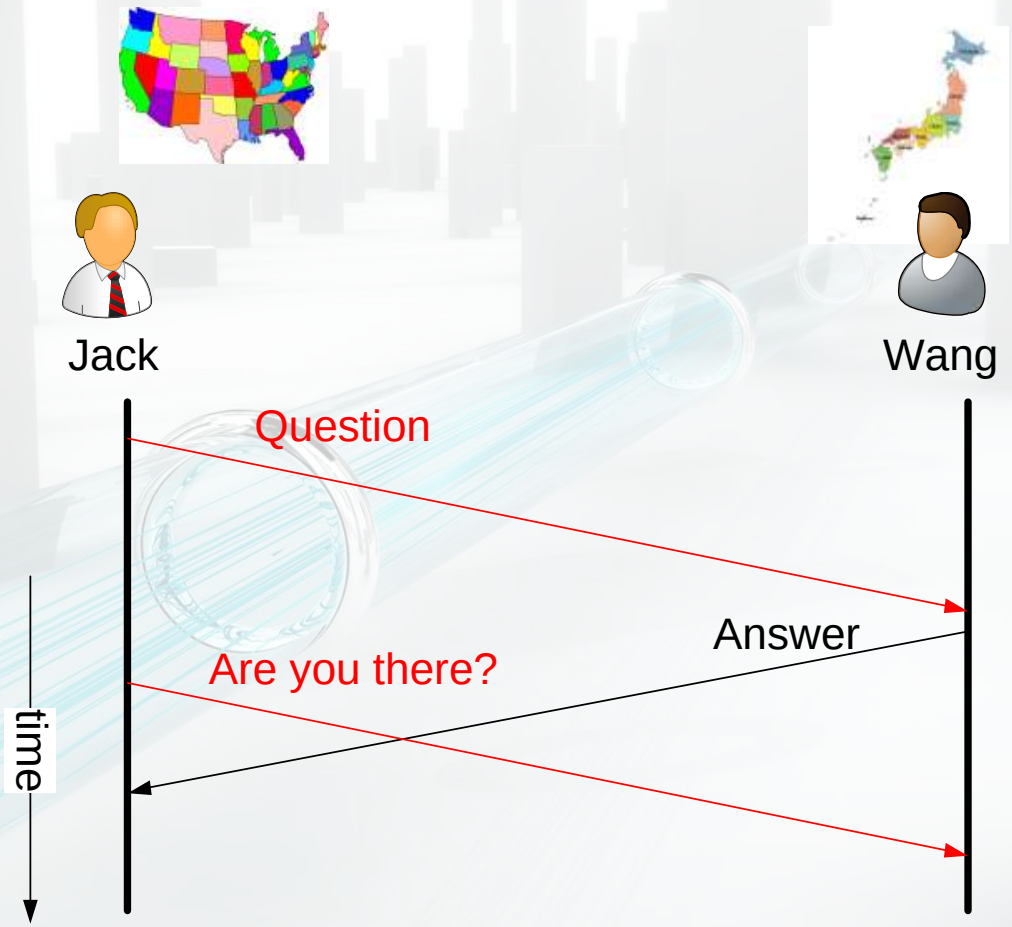
The network is the part that I do not control

- Big amount of data in my hard disk (size of WoW folder)
- Small packets
- Frequent actualizations for interactivity

FPS online games

- The same problems we can find in VoIP.
- Instead of having two users talking at the same time, we have the "shooting around the corner" problem.

FPS online games



Network delay scheme

FPS online games

"Shooting around the corner"

FPS online games

"Shooting around the corner"

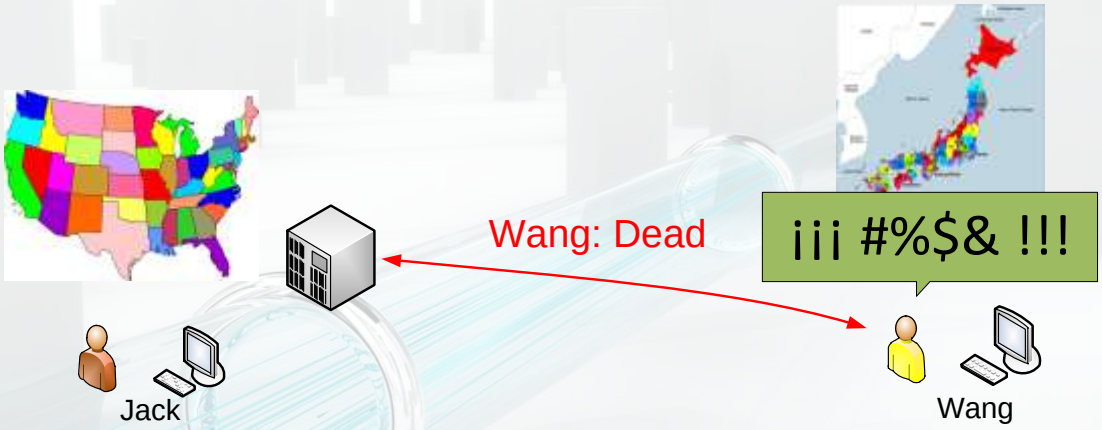
Wang: Dead

Jack

Wang

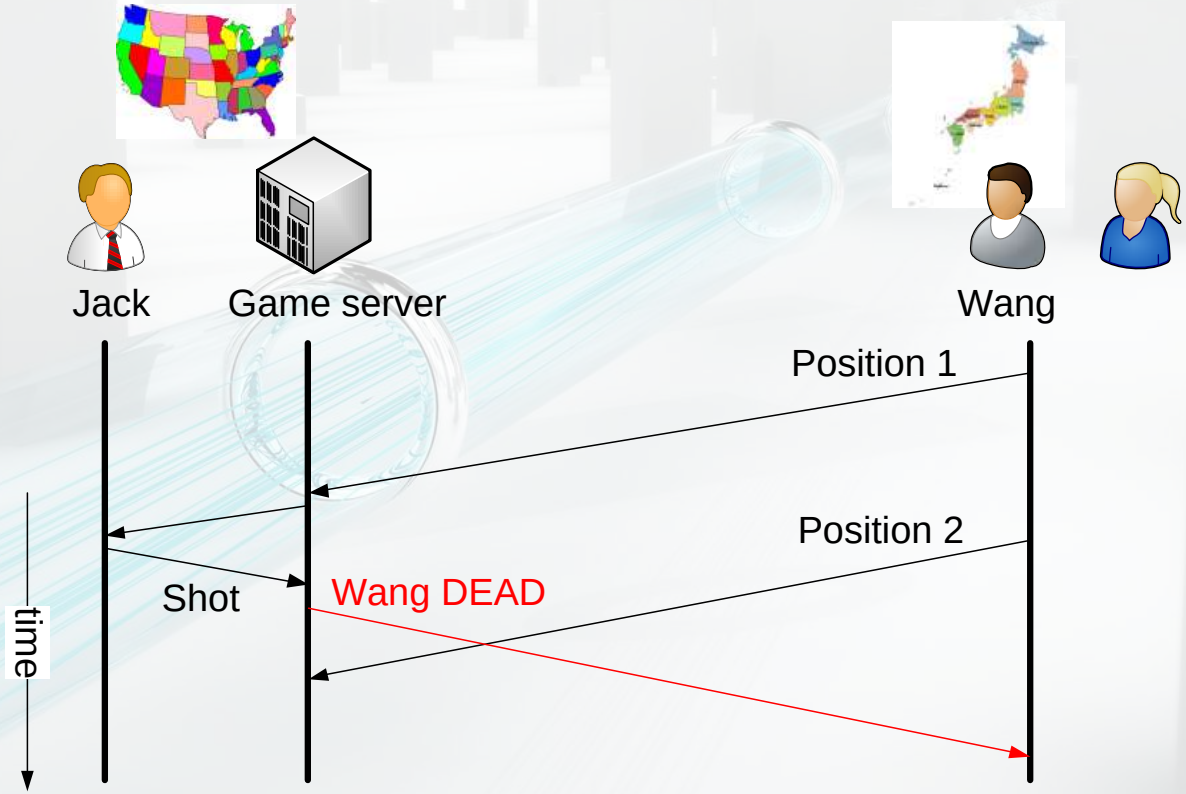
FPS online games

"Shooting around the corner"



FPS online games

"Shooting around the corner"



Network delay scheme

FPS online games

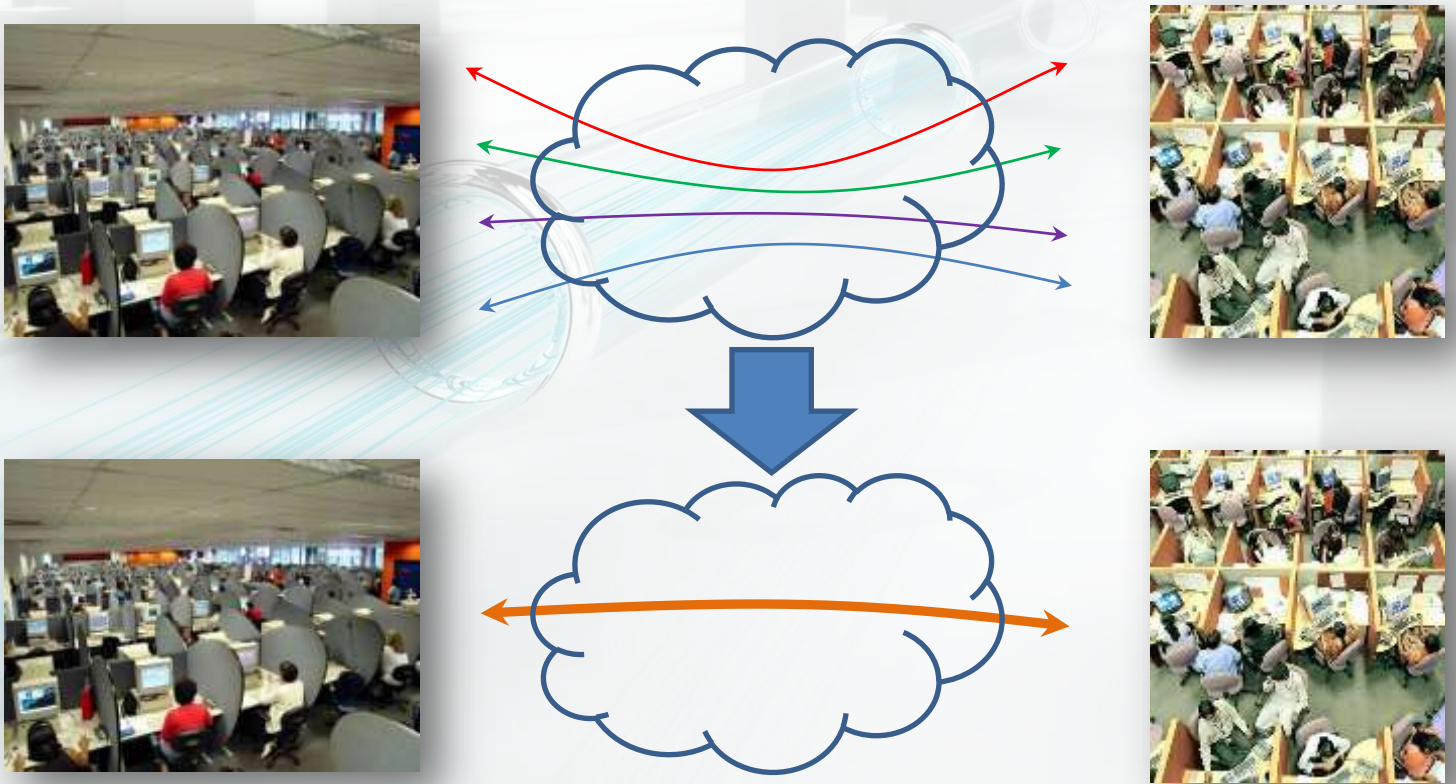
- VoIP and FPS are very similar services: we have the same two problems:
  - Quality problem (delay, packet loss, etc)
  - Efficiency problem
- Let's use the same solutions!

Index

- I. The two problems
- II. Online games
- III. Multiplexing FPS
- IV. MMORPG adaptation

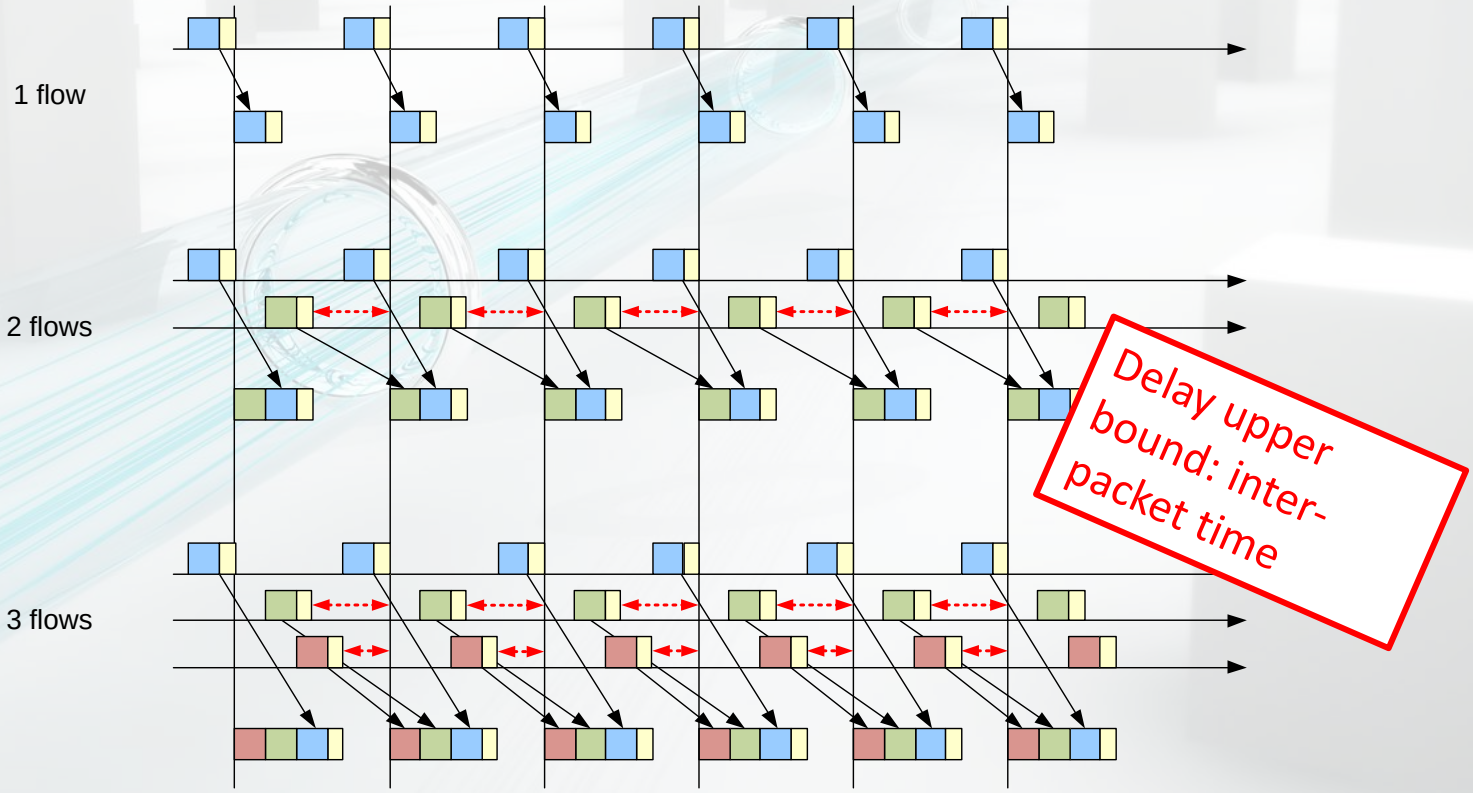
Multiplexing

- Voice trunking between two offices:



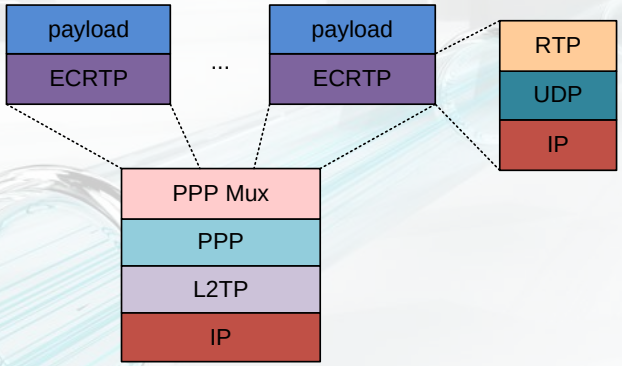
Multiplexing

- We merge the packets and make them share the header:



Multiplexing

- RFC 4170 (2005) deploys this, and also compresses the header

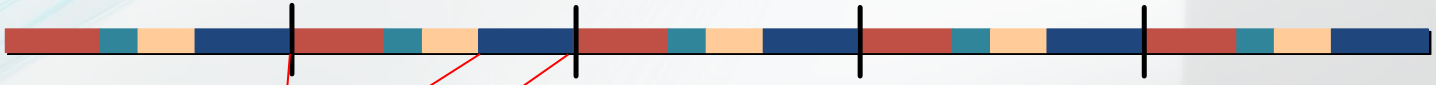


VoIP

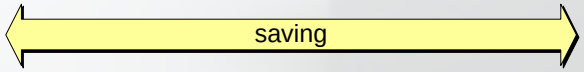
One IPv4/UDP/RTP VoIP packet with two samples of 10 bytes
η=20/60=33%

40 to 6-8 bytes compression

Five IPv4/UDP/RTP VoIP packets with two samples of 10 bytes
η=20/60=33%

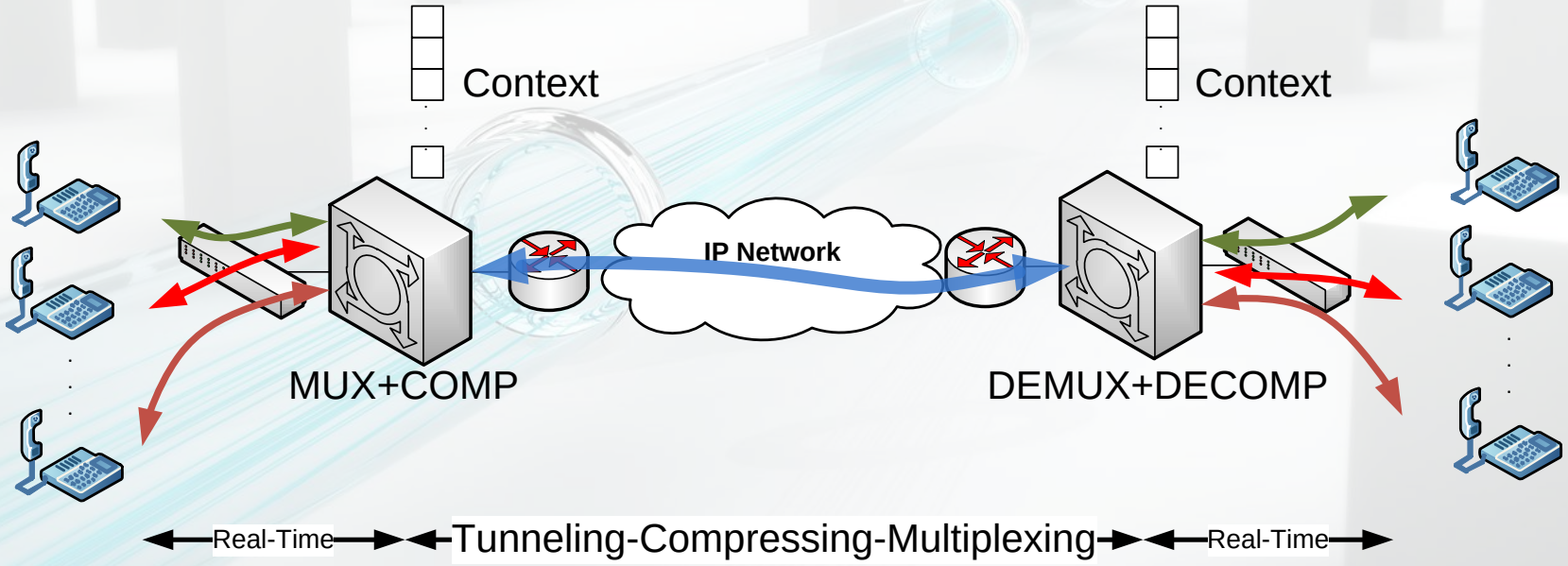


One IPv4 TCMTF Packet multiplexing **five** two sample packets
η=100/161=62%



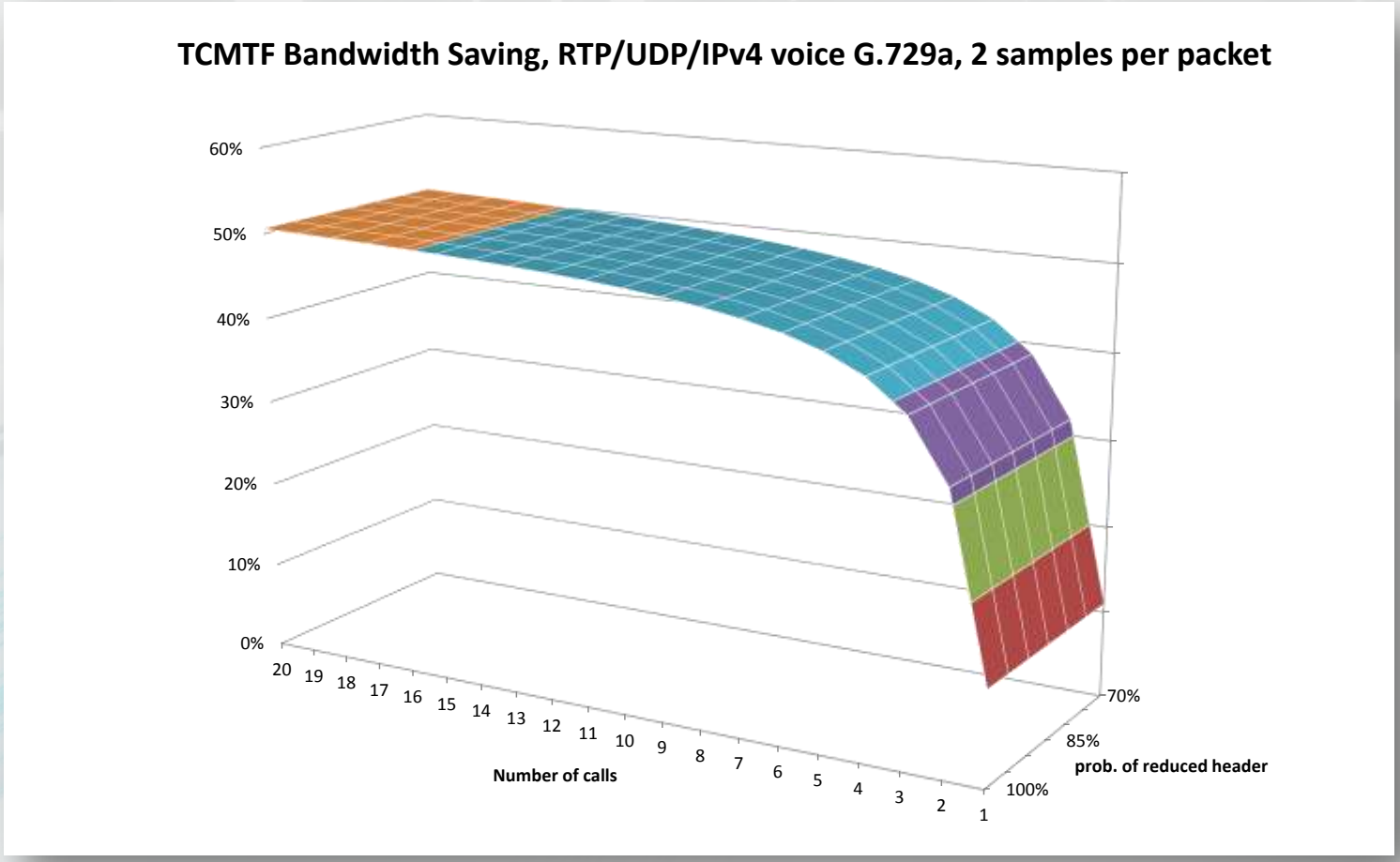
Multiplexing

- RFC 4170 (2005) deploys this, and also compresses the header



Multiplexing

- RFC 4170 results:



Multiplexing

- A trade-off appears:
  - Quality problem: We are adding an additional delay
  - Efficiency problem: we are improving it, since we are saving bandwidth

Multiplexing

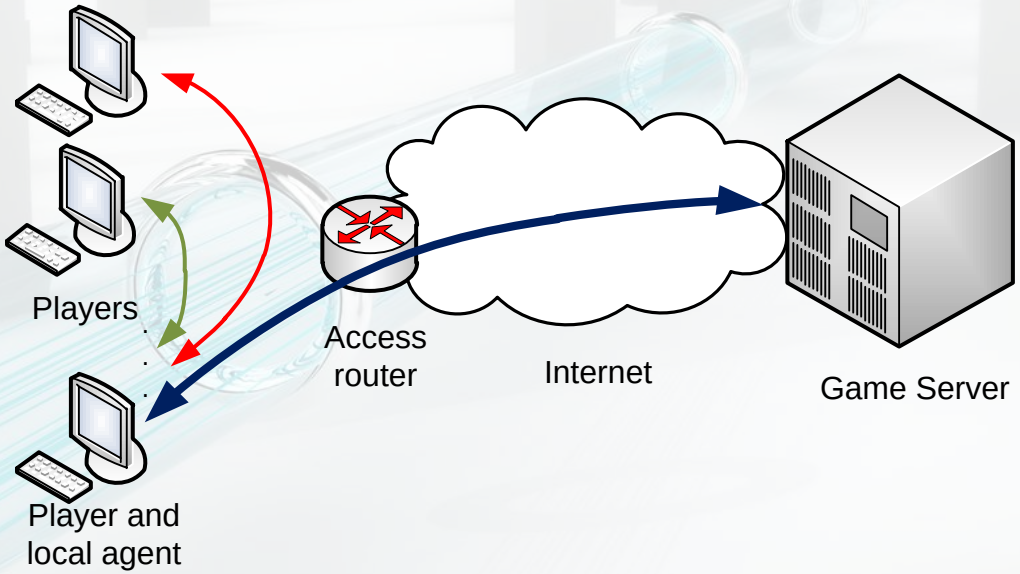
- Is there any scenario where a number of game traffic flows share the same path?

Multiplexing

Internet café

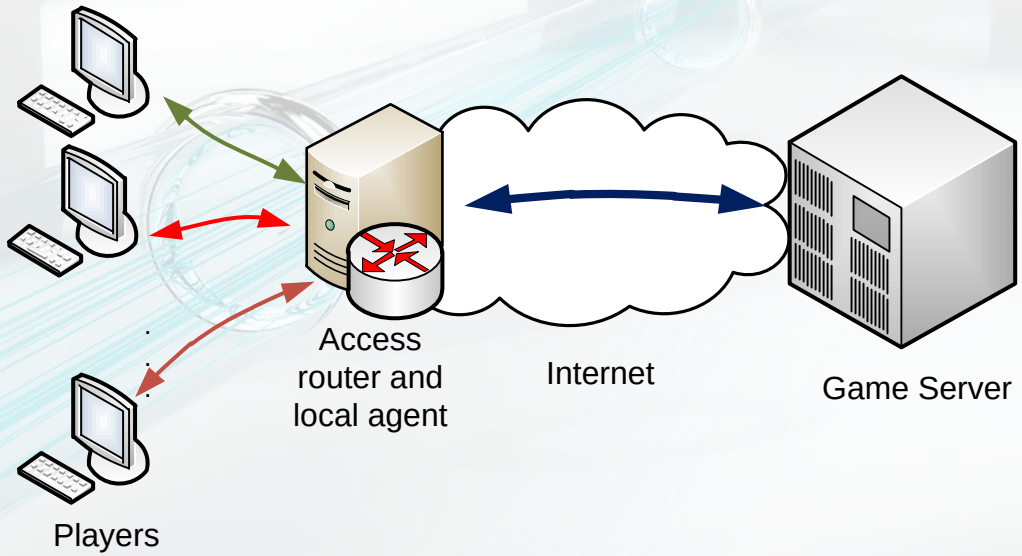
Multiplexing

- Local agent in the computer of a player



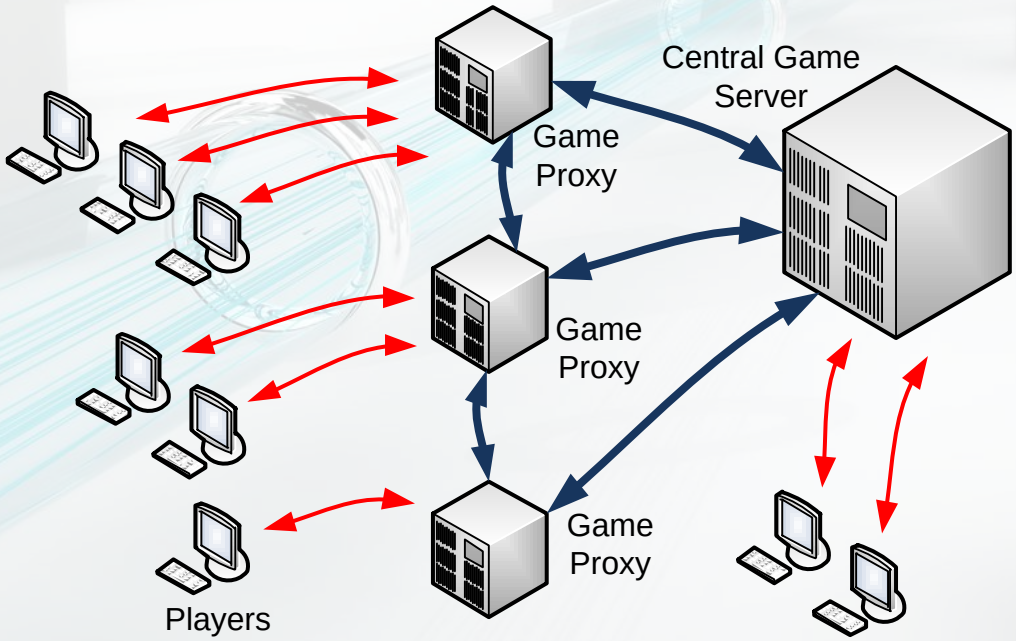
Multiplexing

- Local agent embedded in the router



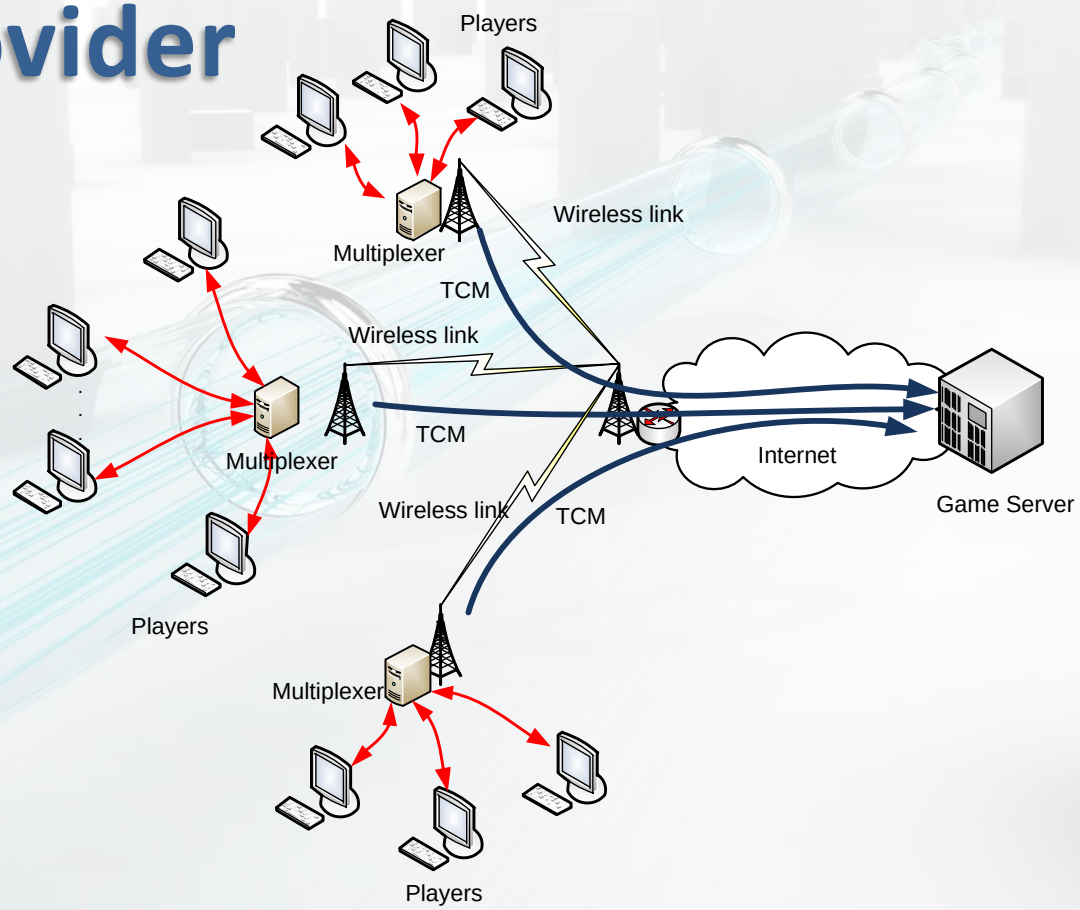
Multiplexing

- Proxies managed by the game provider



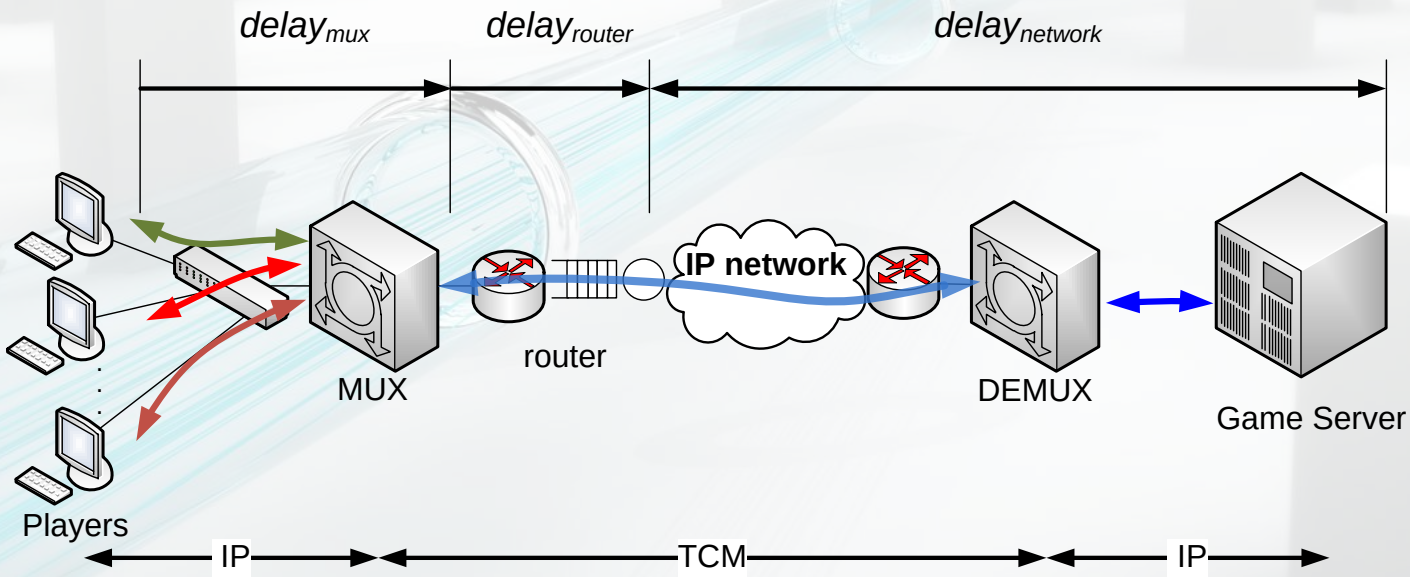
Multiplexing

- Proxies managed by the game provider



Multiplexing

A multiplexer is introduced, and it also compresses headers



Multiplexing

In these scenarios, we can adapt the "trunking" scheme in order to save bandwidth and packets per second

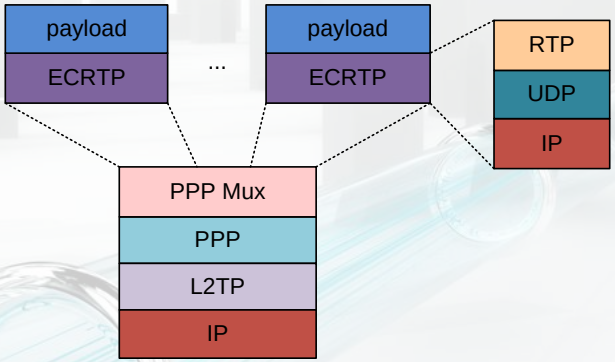
Multiplexing

In these scenarios, we can adapt the "trunking" scheme in order to save bandwidth and packets per second

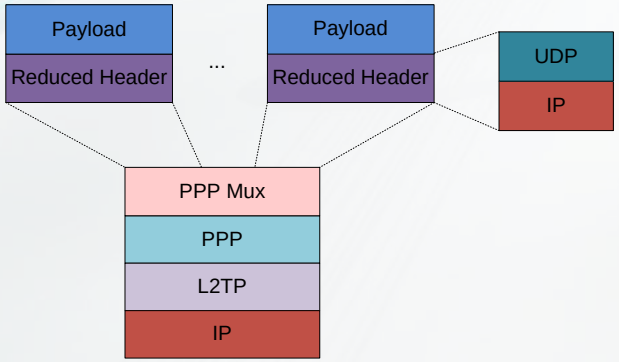
… at the cost of adding delay and Jitter

Multiplexing

First done for VoIP (RFC4170, “TCRTP”):

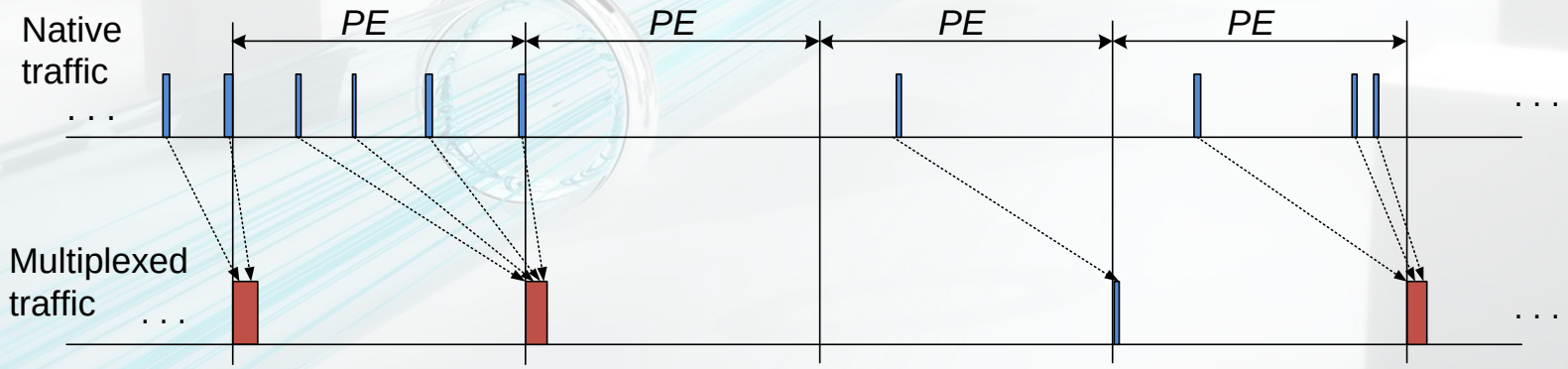


Adapted for non-RTP flows:



Multiplexing

A period is defined, and all the packets arrived are compressed and multiplexed



Multiplexing

Efficiency improvement IPv4

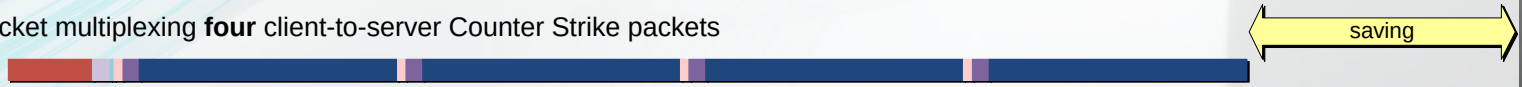
One IPv4/TCP packet 1500 bytes
η=1460/1500=97%

One IPv4/UDP/RTP packet of VoIP with two samples of 10 bytes
η=20/60=33%

One IPv4/UDP server-to-client packet of Counter Strike with 9 players
η=160/188=85%

Four IPv4/UDP client-to-server packets of Counter Strike
η=61/89=68%

One IPv4/TCM packet multiplexing **four** client-to-server Counter Strike packets
η=244/293=83%

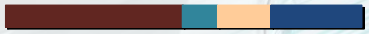


Multiplexing

Efficiency improvement IPv6

One IPv6/TCP packet 1500 bytes
η=1440/1500=96%

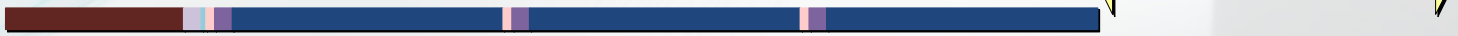
One IPv6/UDP/RTP packet of VoIP with two samples of 10 bytes
η=20/80=25%



One IPv6/UDP server-to-client packet of Counter Strike with 9 players
η=160/208=77%

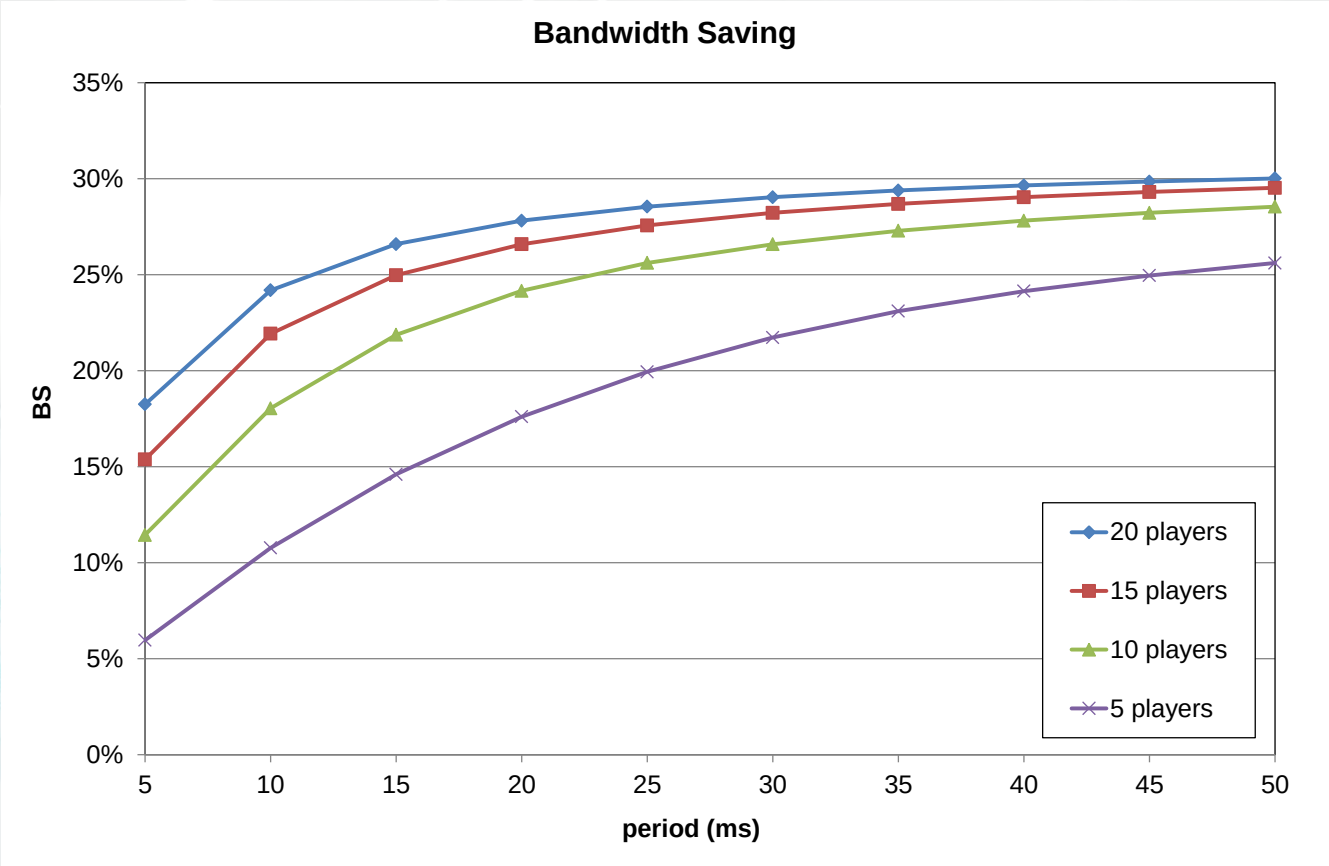
Three IPv6/UDP client-to-server packets of Counter Strike
η=61/109=56%

One IPv6/TCM packet multiplexing **three** client-to-server Counter Strike packets
η=183/246=74%



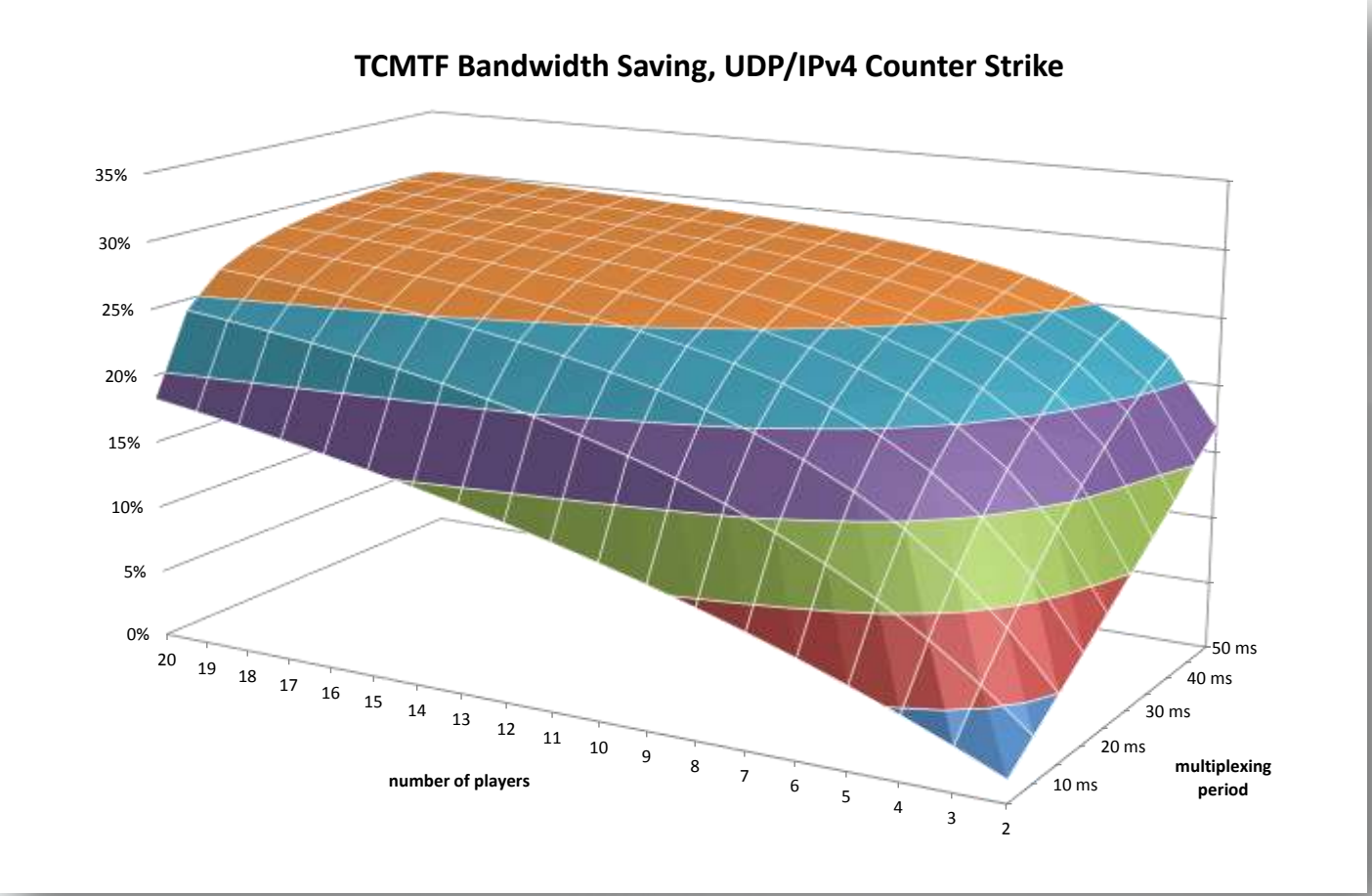
Multiplexing

Significant savings *(Counter Strike)*



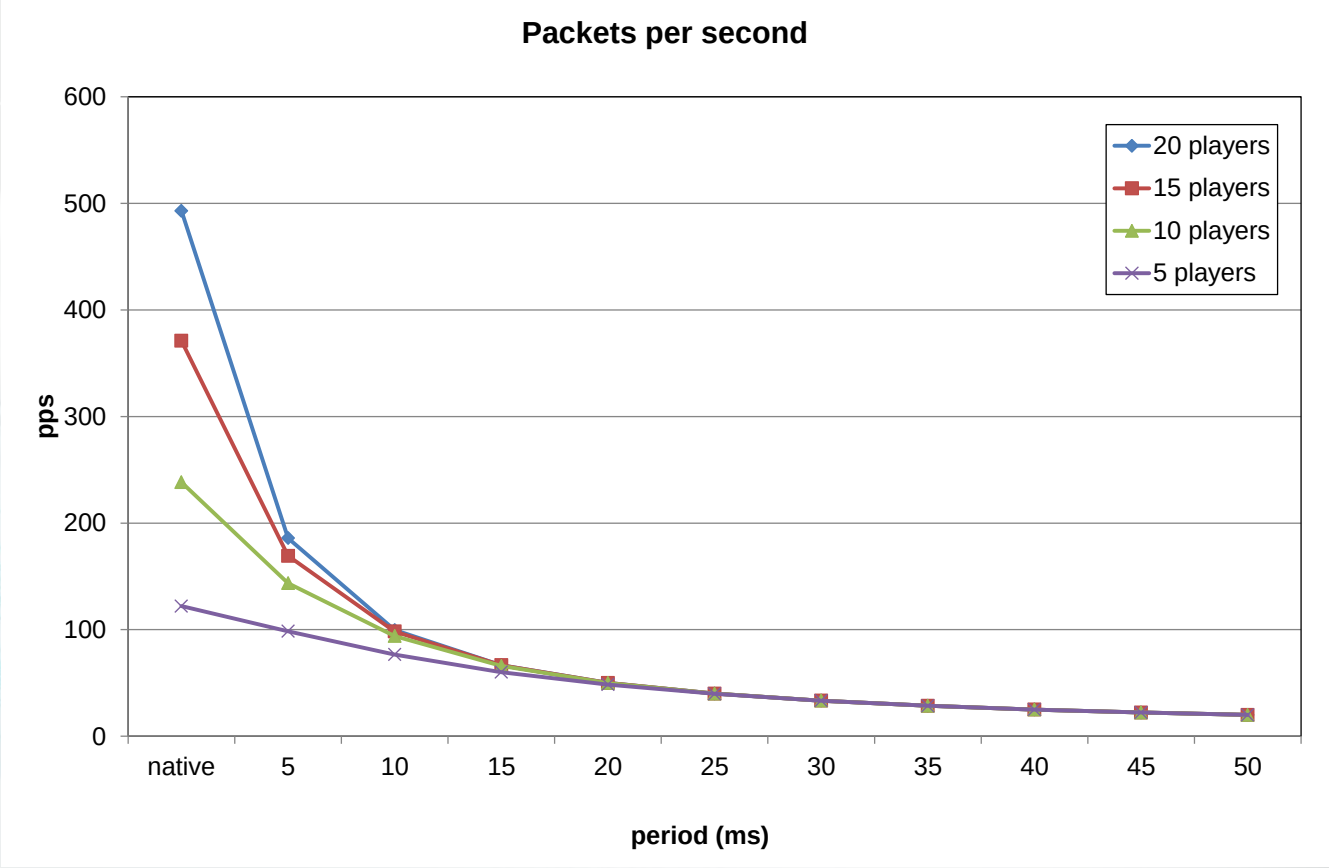
Multiplexing

Significant savings *(Counter Strike)*

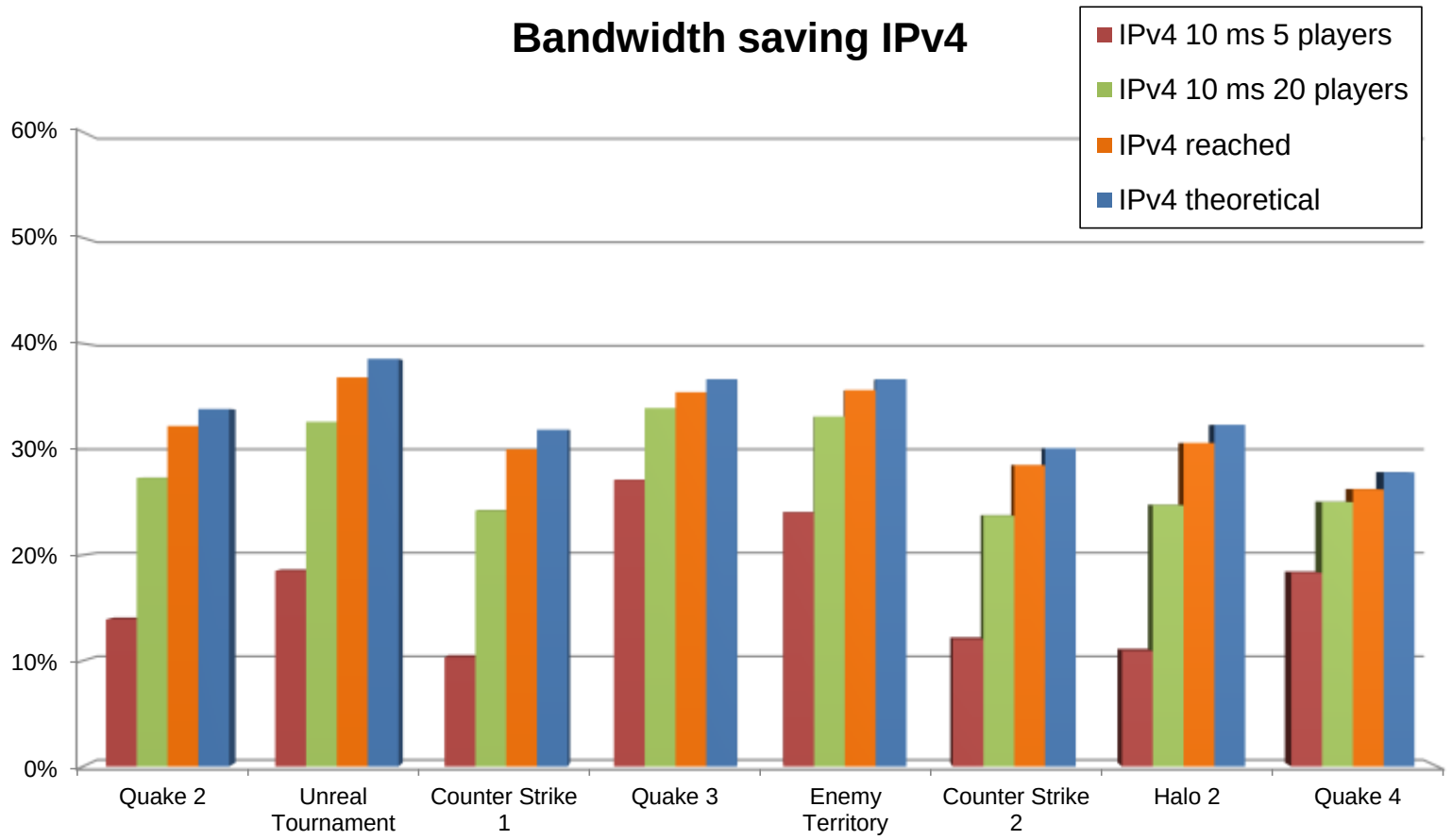


Multiplexing

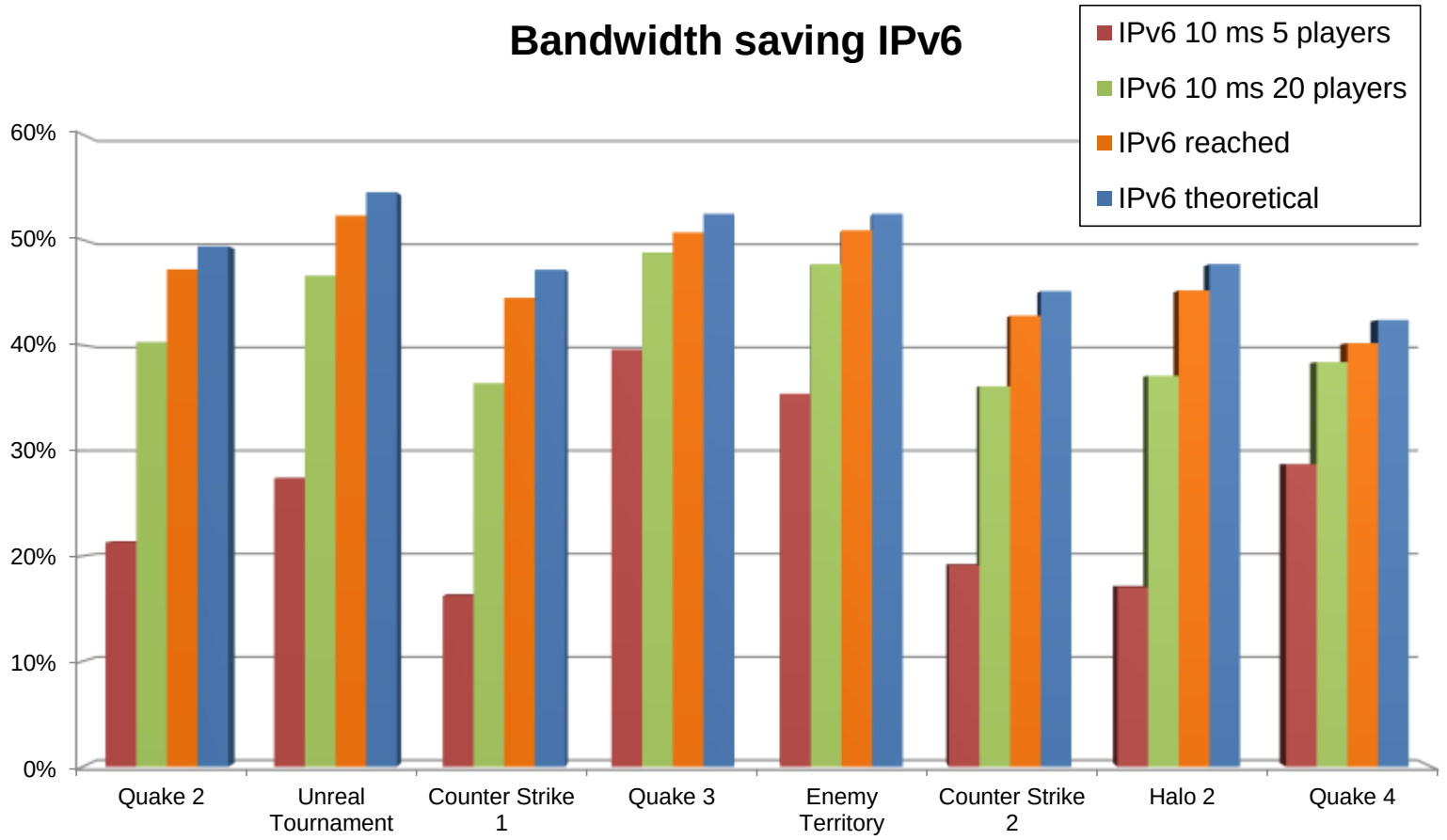
Significant savings *(Counter Strike)*



Multiplexing



Multiplexing

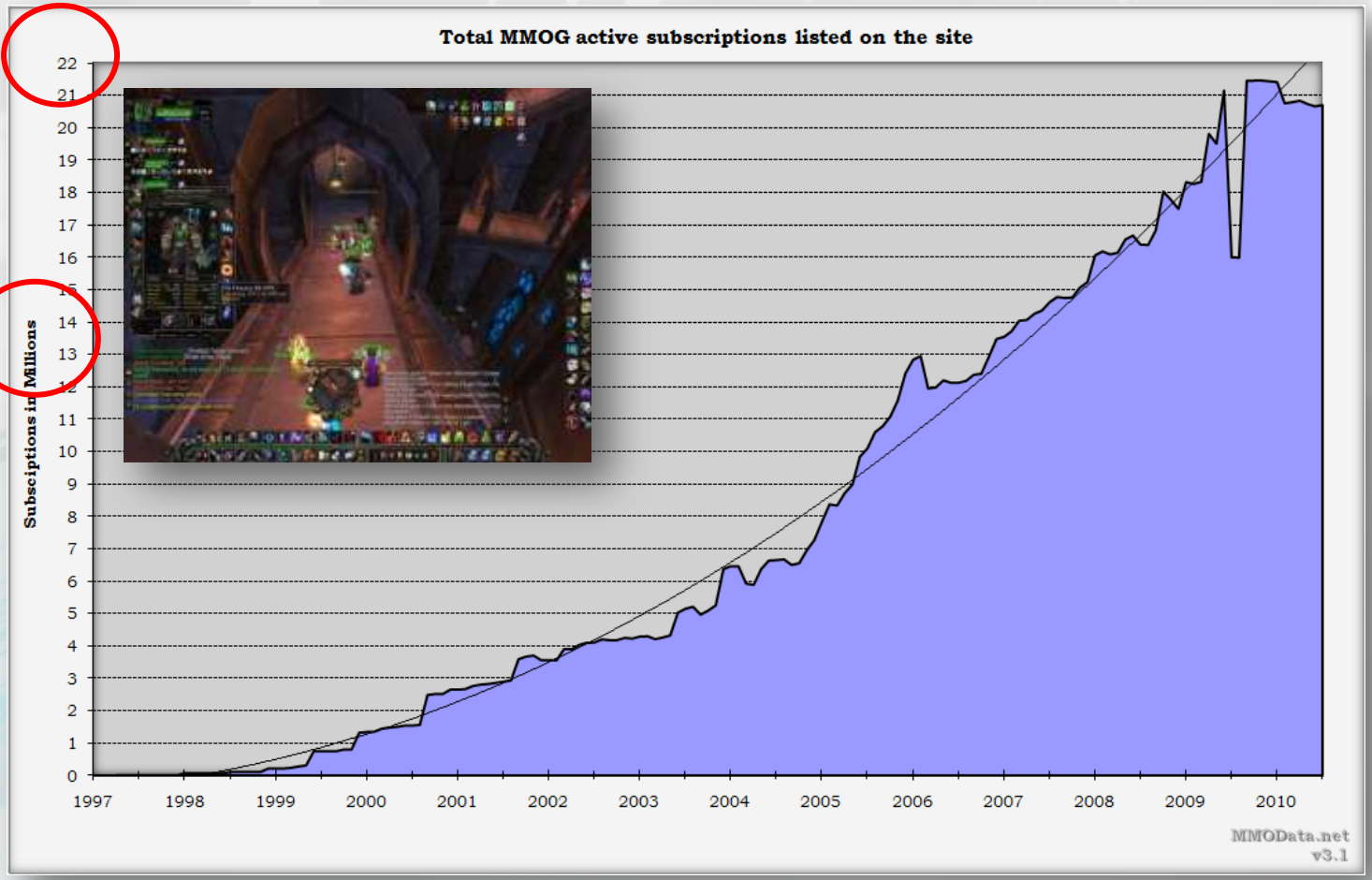


Index

- I. The two problems
- II. Online games
- III. Multiplexing FPS
- IV. MMORPG adaptation

MMORPG

Can this be adapted to MMORPGs?

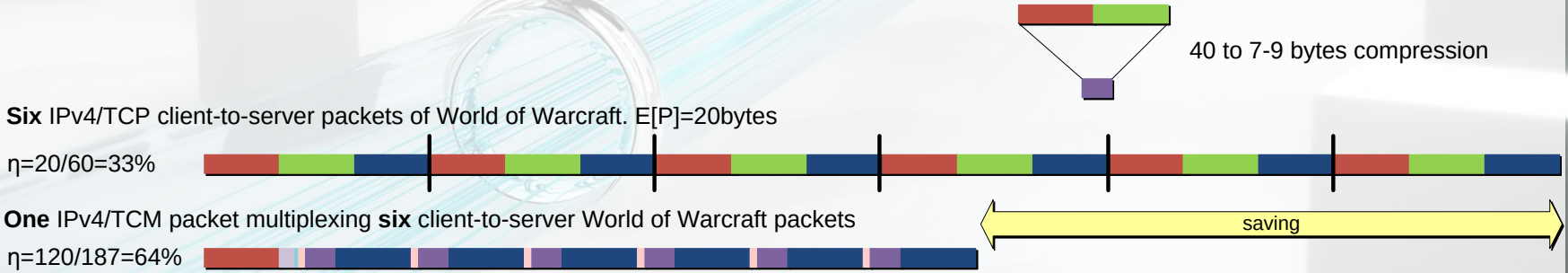


http://designcult.org/designcult/2010/08/mmo-subscription-charts.html

MMORPG

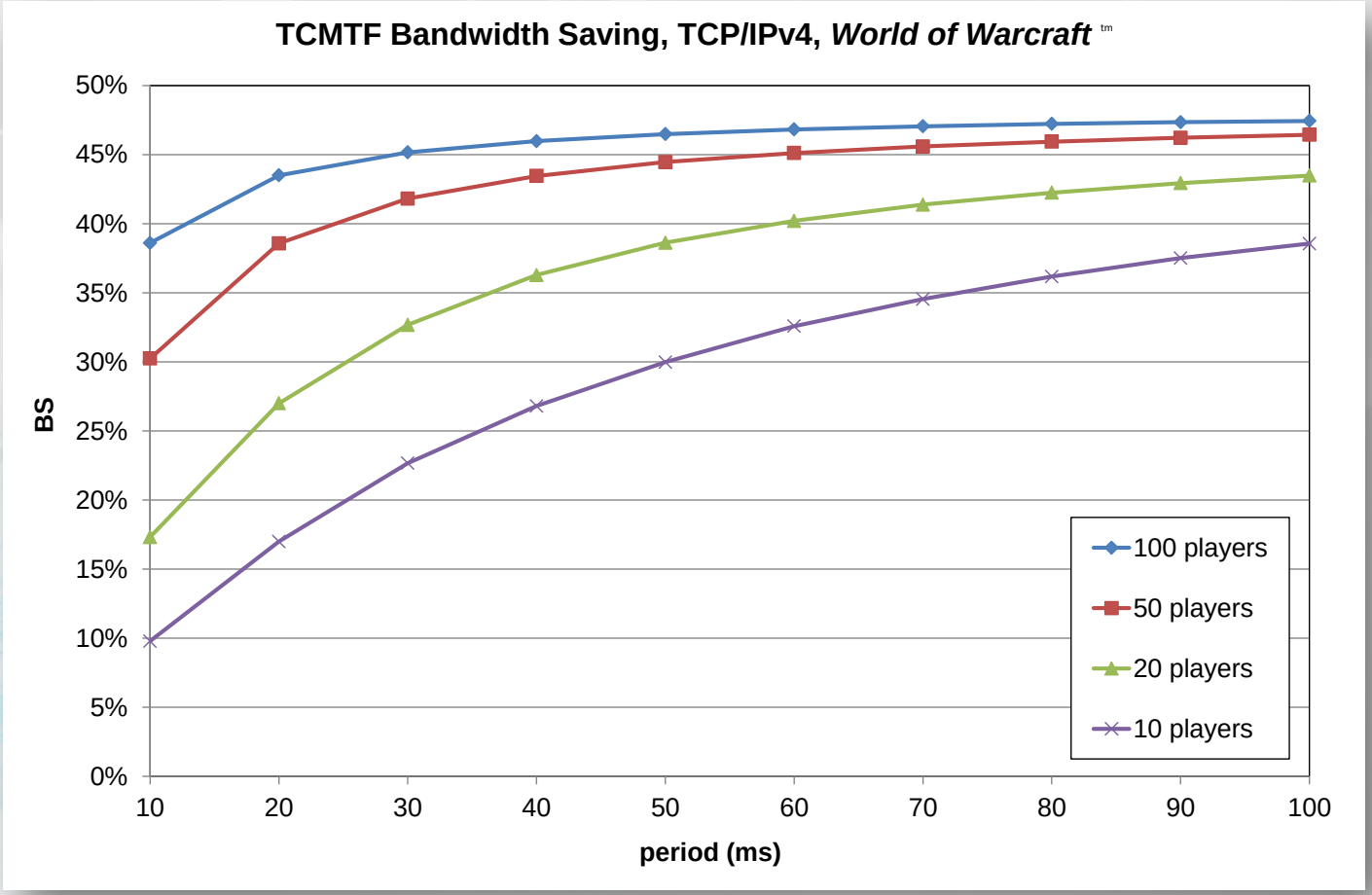
Can this be adapted to MMORPGs?

Massively Multiplayer Online Role Playing Game (TCP)



MMORPG

Can this be adapted to MMORPGs?

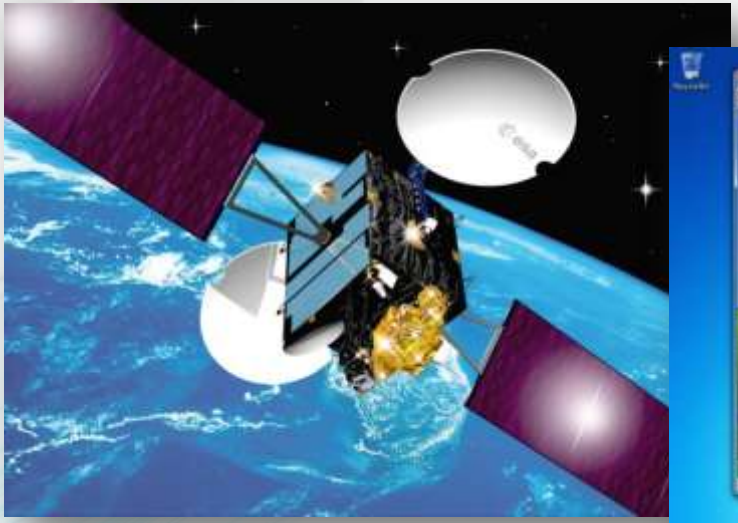


"*Widening the Scope of a Standard: Real Time Flows Tunneling, Compressing and Multiplexing*," IEEE ICC 2012, Workshop on Telecommunications: from Research to Standards, June 10-11, 2012, Ottawa, Canada. In press

…and beyond

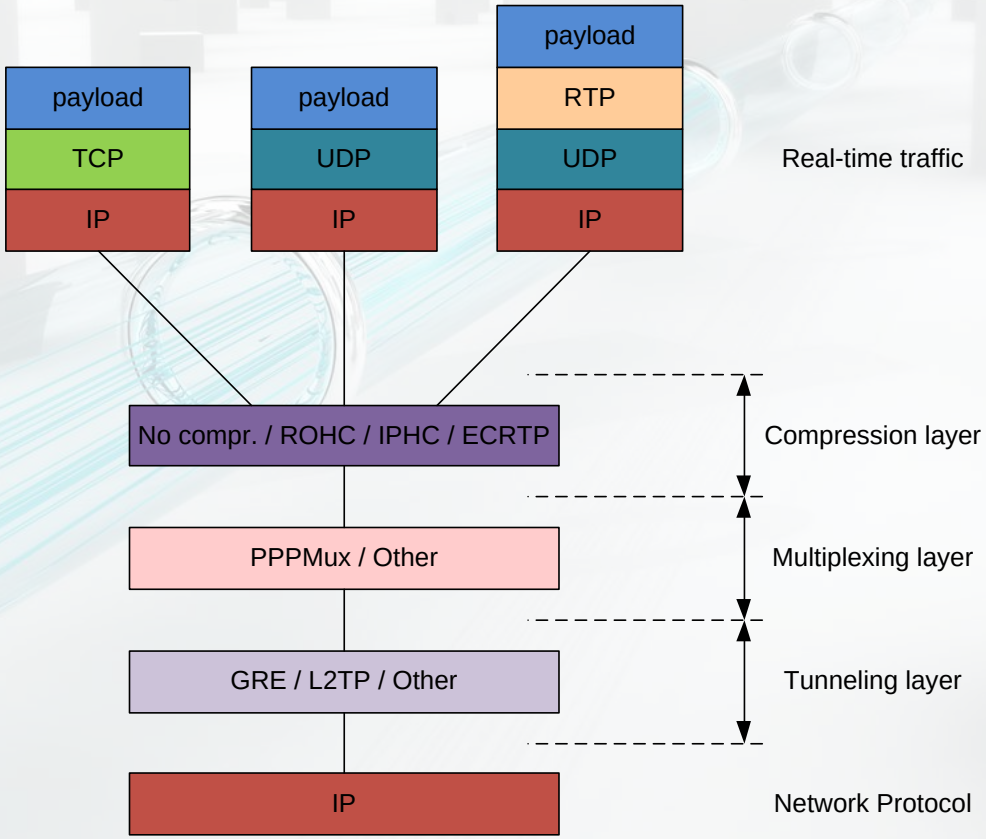
And to other services?

- Merging ACKs in satellite links
- Remote desktop applications



IETF draft proposal

TCMTF (Tunneling, Compressing and Multiplexing Traffic Flows)

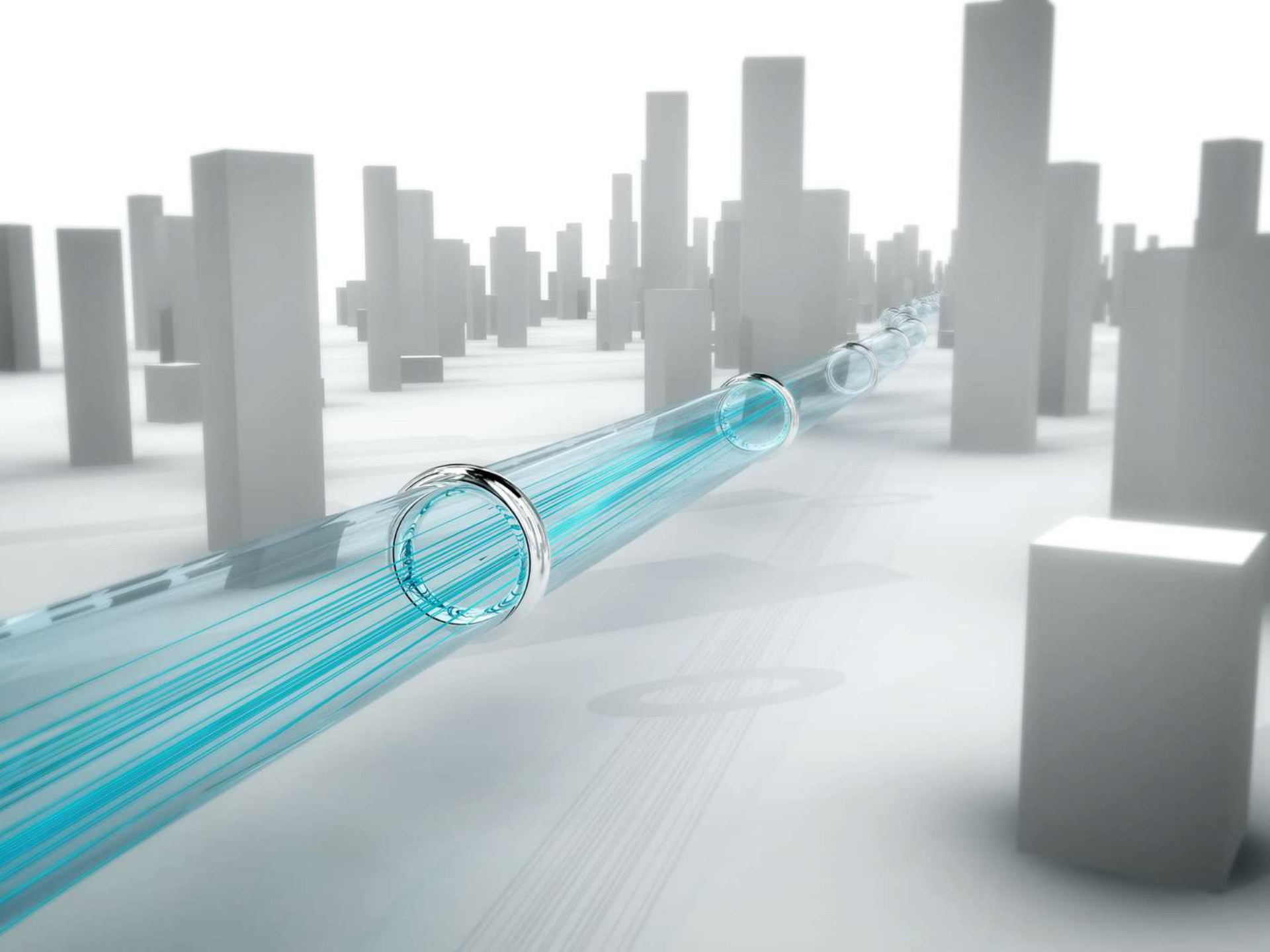


IETF draft proposal

TCMTF (Tunneling, Compressing and Multiplexing Traffic Flows

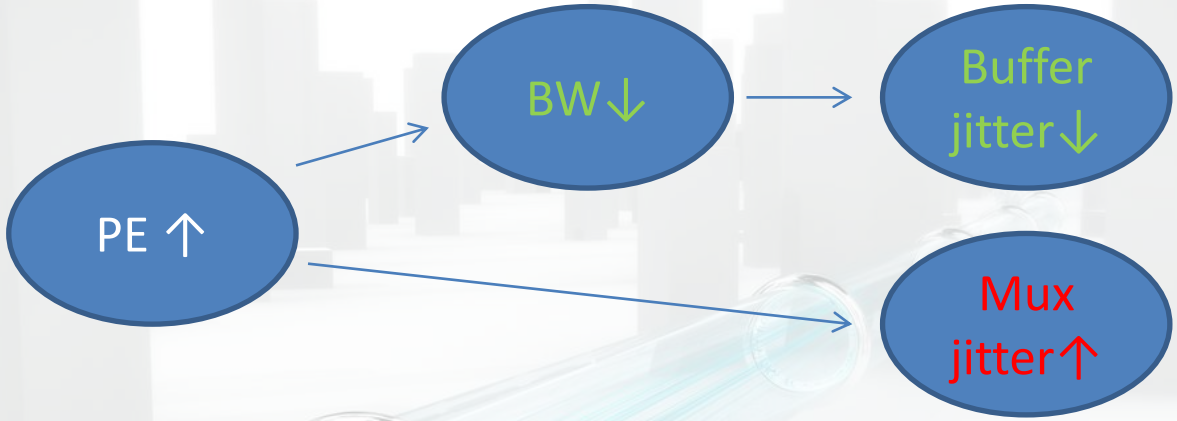
- Presentation in IETF83, Paris, Apr. 2012

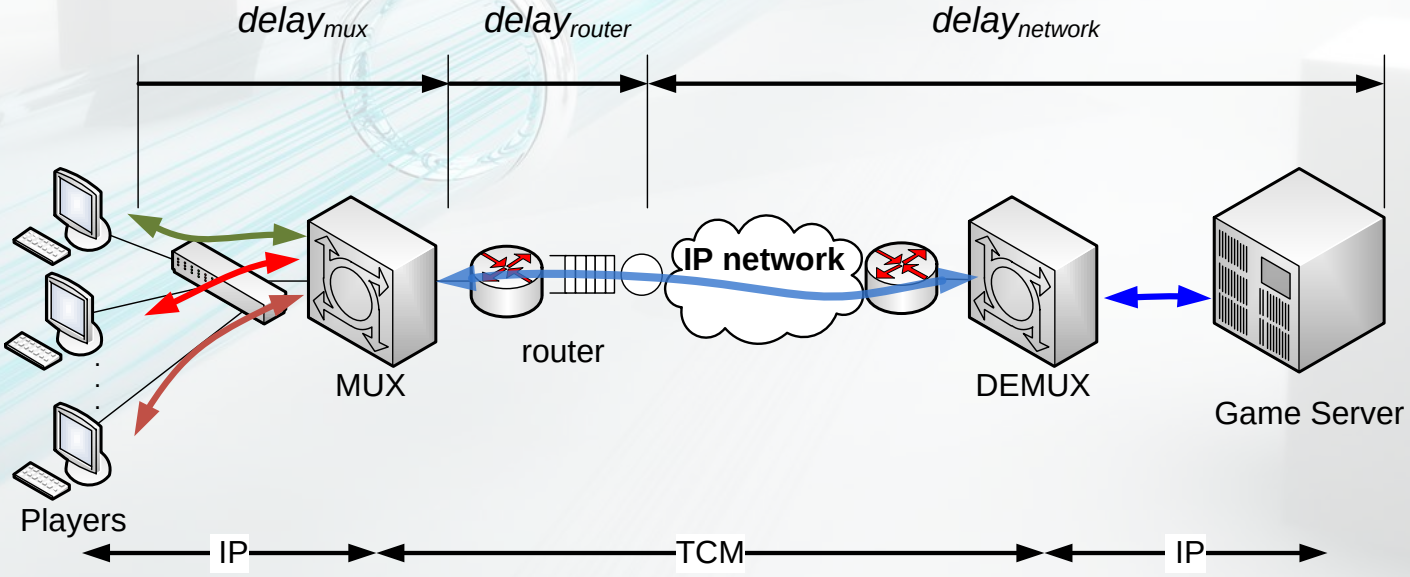

THANK YOU
GTC
Communication
Technologies Group
jsaldana@unizar.es



Extra Slides

Tradeoff


BW↓
Buffer jitter↓
PE ↑
Mux jitter↑


delay mux
delay router
delay network
IP network
MUX
router
DEMUX
Game Server
Players
IP
TCM
IP