Fun and Games - why online FPS games are interesting to network engineers

(a very shallow overview)

18 May 2007

Grenville Armitage

garmitage@swin.edu.au

Overview

- What are First Person Shooter (FPS) games?
- Reasons to care about online games
- Evidence of human-driven traffic patterns
- Player tolerance to network degradation
- The background noise of server discovery
- Game traffic characterisation
- Challenges facing game developers and network operators

What are First Person Shooters?

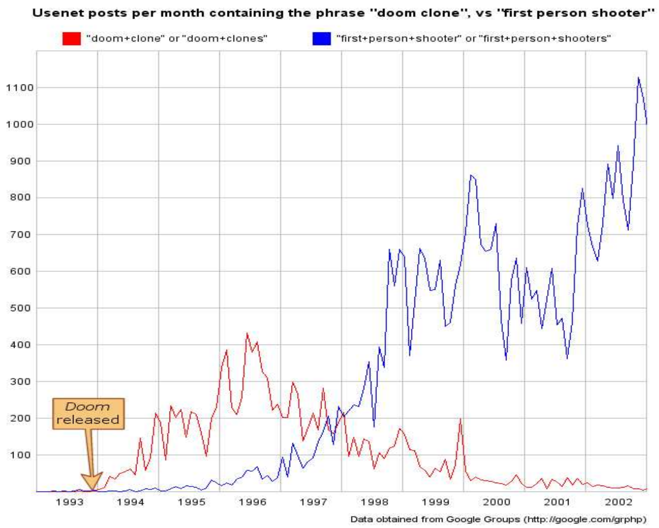
- Game genres are often named to:
  - differentiate between *styles* of game play
  - the *scale* of technology used to deliver the game
  - the *game play experience* or even the *intended impact* of the game experience on the participants
- First Person Shooters
  - Primarily combat based (team or individual)
  - 3D virtual environment from 1st person visual perspective
  - Real-time, highly interactive

First person view...

Quake III Arena

Wolfenstein Enemy Territory

FPS orginally called “Doom clone”



http://en.wikipedia.org/wiki/Image:Doom_clone_vs_first_person_shooter.png

Other game types exist

- RTS: Real time strategy
  - participants build and control a large number of troops fighting enemy players
- RPG: Role playing game
  - participants play a detailed character solving quests and interacting with other player and non-player characters
- MMOG: Massively multiplayer online game
  - Games where thousands to hundreds of thousands of players interact simultaneously in the game world
  - MMORPG: Massively multiplayer online RPG

(Other game types, cont'd)

- Simulation games
  - where the focus is to control an accurate simulation of a specific real-world environment (such as flight and racing simulators)
- Augmented reality / pervasive games
  - Synthetically blend virtual and real-world stimuli and immersion experiences
    - For example, where the game is played in a real environment but with virtual artefacts
      - http://wearables.unisa.edu.au/projects/ARQuake/www/
      - Wayne Piekarski and Bruce Thomas, ARQuake: The Outdoor Augmented Reality Gaming System, Communications of the ACM,2002 Vol 45. No 1, pp 36-38
      - A. Cheok, K. Goh, W. Liu, F. Farbiz, S. Teo, H. Teo, S. Lee, S. Fong, X. Yang, “Human Pacman: A mobile wide-area entertainment system based on physical, social, and ubiquitous computing”, Proceedings of Advances in Computer Entertainment (ACE2004), Singapore, 3-5 June 2004

Multiplayer vs online

- Multiplayer does not imply online or vice-versa
- Multiplayer
  - More than one concurrent player on the server
  - More than one player, not concurrent
  - 2+ players interacting in the same game at same time
- Online?
  - Loosest interpretation - using digital network to connect geographically separated players
- We're interested today in online multiplayer FPS

  (such as Quake III Arena, Half-Life Counterstrike, Wolfenstein Enemy Territory, and Half-Life 2)

Overview

- What are First Person Shooter (FPS) games?
- Reasons to care about online games
- Evidence of human-driven traffic patterns
- Player tolerance to network degradation
- The background noise of server discovery
- Game traffic characterisation
- Challenges facing game developers and network operators

Why care about online FPS games?

- Entertainment industry is big money
- Games are big part of entertainment
- FPS games have tight interactivity requirements
- Most FPS games run over IP
- Gamers are not as budget-conscious as other QoS-sensitive applications (such as consumer VoIP)

So we have entertainment-seeking consumers, with money, demanding a *better* IP networking experience

Evidence of caring....

- Marketing has identified gamers as a target niche
- ISPs and game hosting companies are increasingly marketing 'gamer friendly' services
  - “We host local game servers...”
  - “We provide more bandwidth... lower lag...”
- CPE vendors (home routers, modems) offering user-controlled 'game traffic prioritisation' as a product differentiator

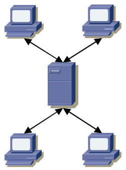
So, network engineers and software developers will need to 'understand' the online game segment....

Overview

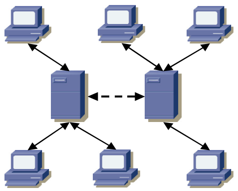
- What are First Person Shooter (FPS) games?
- Reasons to care about online games
- Human-driven, client-server traffic patterns
- Player tolerance to network degradation
- The background noise of server discovery
- Game traffic characterisation
- Challenges facing game developers and network operators

Network communication models

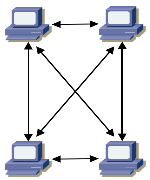
- All participants must experience the 'same' shared perspective of game-state at all times
  - 'Same' view requires prompt and timely transmission of updates to all participants
  - Network transmission is never instantaneous, so players (and game software) must adapt to 'approximately same time' views



Central game server (client-server)



Synchronised, distributed game server(s)



Peer to peer

FPS are usually client-server

- Many benefits revolve around cheat-minimisation
  - Central servers provide (quasi-)neutral mediator
  - Confirm timing of 'kills' (who shot whom and when)
  - Confirm player behaviour conforms to virtual world's laws of physics
  - Only one person (the server operator) needs to be trusted
- Game servers act as a rendezvous point
  - Hierarchy of 'master servers', 'game servers' and players scales reasonably using loose control mechanisms

FPS games use UDP because....

- Highly interactive games require timeliness
  - Neither TCP nor UDP have timeliness as a goal, but at least UDP doesn't do worse than 'best effort' IP
- There's not much point retransmitting data that's “too late” to be of any use
  - TCP is focused on reliability at expense of timeliness
- Flow control isn't an option
  - Games need state updates at whatever rate is required to maintain in-game immersion experience
    - TCP's *windowed flow control* would impose performance fluctuations unrelated to game need, and thus negatively impact game experience

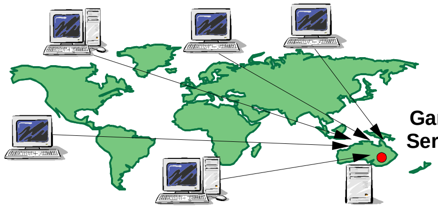
FPS games: Client-server UDP

- Three main in-game components using UDP
  - Probes to discover servers
    - 'Well known' master servers list active public game servers
  - Connecting to server
  - Actual game-play
    - Tight latency / jitter / loss requirements during this phase
  - Downloading maps, skins, patches
    - May occur concurrently with game play or map changes
    - (May also use FTP or HTTP to download maps, etc, in an “out of band” fashion, which both use TCP)

Usage patterns on a 24-hour cycle

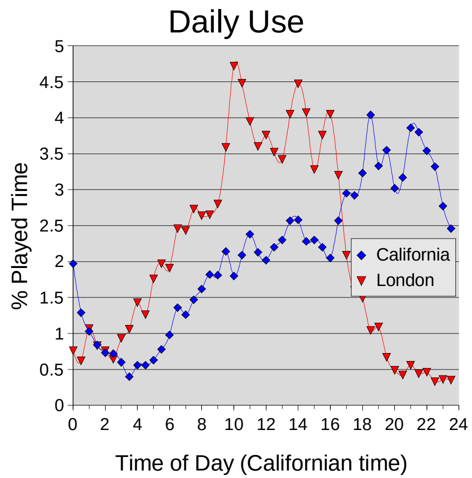
- One might reasonably presume that people tend to play in the late afternoon and evening
  - So network traffic would rise and fall on 24-hour cycles
- 'Server discovery' and 'game play' traffic not in phase
  - Different regions contribute traffic at different times

Clients from all over the world



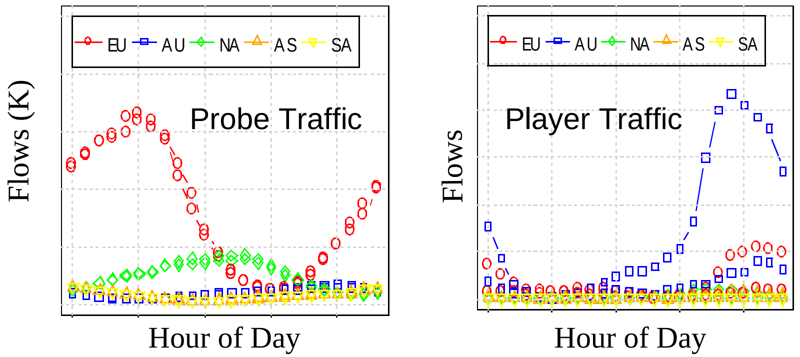
Quake III Arena in 2001

- Usage peaks around afternoon/evening in their respective time zones
  - (London 8 hours ahead of Palo Alto, California)
- Servers attract regional players
  - Seems that clients prefer 'closer' server, all other things being equal



G.J. Armitage, "An Experimental Estimation of Latency Sensitivity In Multiplayer Quake 3", 11th IEEE International Conference on Networks (ICON 2003), Sydney, Australia, September 2003

Enemy Territory in 2004



- Europe dominated server-discovery probe traffic (the most clients)
- Australia dominated game-play traffic (international players didn't much like our Melbourne-based server)

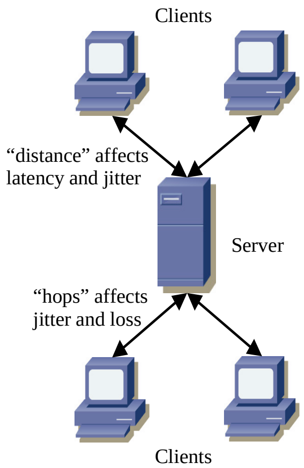
S. Zander, D. Kennedy, G. Armitage, "Dissecting Server-Discovery Traffic Patterns Generated By Multiplayer First Person Shooter Games," ACM NetGames 2005, NY, USA, October 2005

Overview

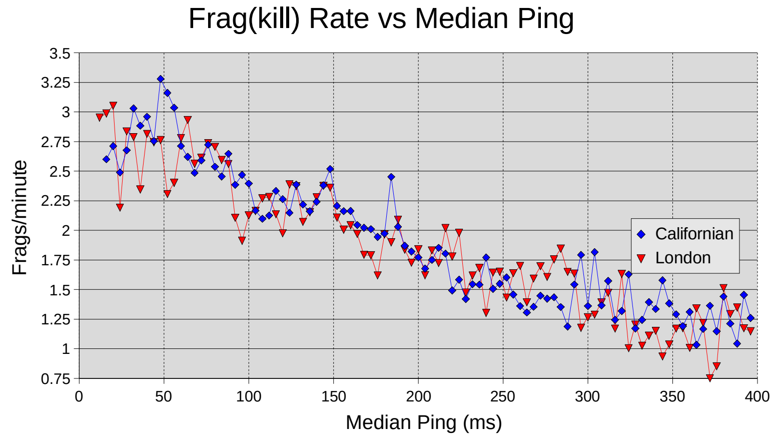
- What are First Person Shooter (FPS) games?
- Reasons to care about online games
- Evidence of human-driven traffic patterns
- Player tolerance to network degradation
- The background noise of server discovery
- Game traffic characterisation
- Challenges facing game developers and network operators

The network influences game play

- Game style typically bounds network behaviour associated with successful/happy players
  - Latency
    - Path length (km) and hops
  - Loss
    - ~Congested paths (or wireless)
  - Jitter
    - Hops (~router congestion points)



Latency impact on 'success' in FPS

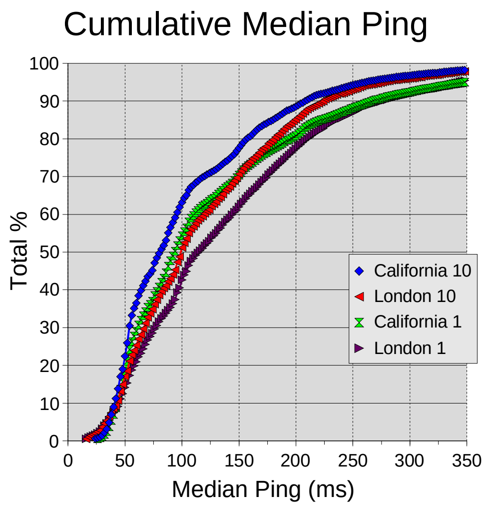


(public Quake III Arena servers in 2001)

G.J. Armitage, "An Experimental Estimation of Latency Sensitivity In Multiplayer Quake 3", 11th IEEE International Conference on Networks (ICON 2003), Sydney, Australia, September 2003

Preference for low RTT

- Each player's 'ping' sampled > 10 times per game
- Median values per player per game
- Cumulative plot reflects most frequently appearing median ping values
- So.... we don't know where geographically, but we know they're almost all under 200ms away!



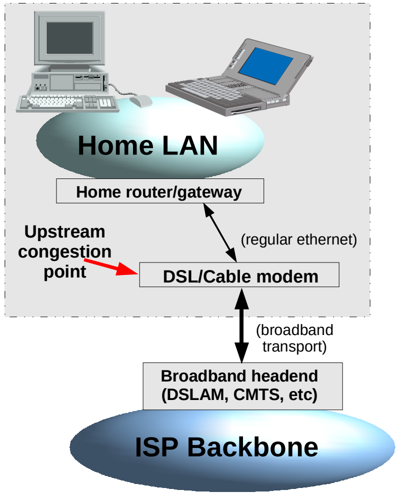
G.J. Armitage, "An Experimental Estimation of Latency Sensitivity In Multiplayer Quake 3", 11th IEEE International Conference on Networks (ICON 2003), Sydney, Australia, September 2003

Where does latency come from?

- Speed of light
  - Physical distance, Topology (path length)
- Serialisation delay
  - Link speed (or virtual speed if rate capping in use)
- Queuing delay
  - Transient congestion on outbound links, queue builds up, packets transmitted at finite rate (see serialisation delay)
- Processing or link access protocol delay
  - Router pipeline architectures, link media access control (e.g. waiting for link to go idle in shared infrastructures)

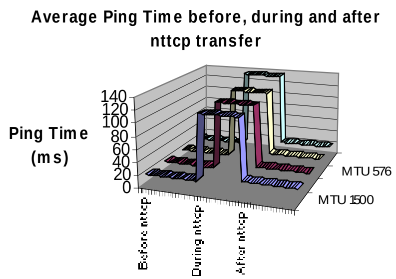
Latency in the home access link

- Consider home LAN sharing DSL or Cable
- Game traffic can be jittered by large packets upstream
- Consider an ISP uplink capped at 128Kbit/sec
  - 1.5K pkt adds ~ 94ms
  - e.g. sending email with attachments



For gamers over cable modem....

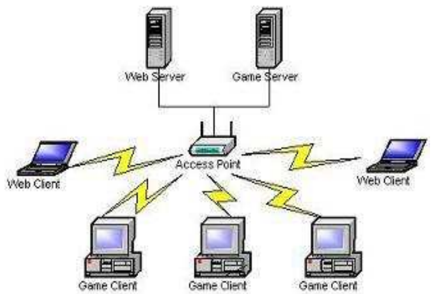
- There's a significant spike in RTT when sharing a DOCSIS link with other traffic (e.g. file download)

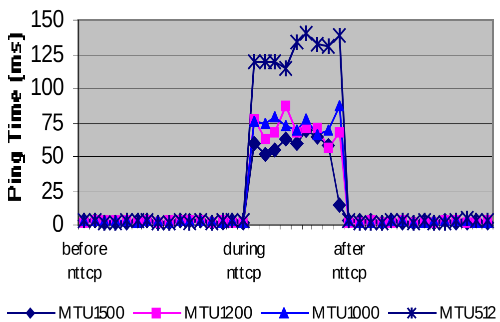


T.T.T.Nguyen, G.Armitage. "Experimentally derived interactions between TCP traffic and service quality over DOCSIS cable links," Global Internet and Next Generation Networks Symposium, IEEE Globecomm 2004, Texas, USA, November 2004

Gamers over shared 802.11b...

- Typical shared scenario, bulk TCP transfers significantly influence RTT through Access Point for all clients





Source: T.T.T.Nguyen, G.Armitage "Quantitative Assessment of IP Service Quality in 802.11b Networks", Australian Telecommunications Networks & Applications Conference 2004 (ATNAC2004), Sydney, Australia December 8-10 2004

Overview

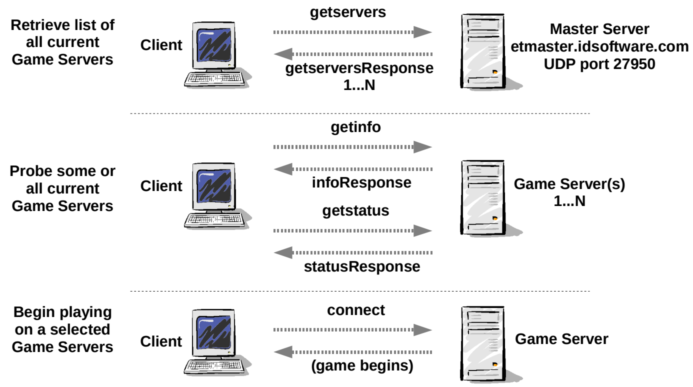
- What are First Person Shooter (FPS) games?
- Reasons to care about online games
- Evidence of human-driven traffic patterns
- Player tolerance to network degradation
- The background noise of server discovery
- Game traffic characterisation
- Challenges facing game developers and network operators

Server discovery - two stage process

- Master servers hold lists of active game servers
  - Master server addresses preconfigured into clients
  - Clients query master server, then query the list of game servers returned by master server
- Creates 'background radiation' on the Internet
  - If your server is public, it will 'experience' probe traffic
  - Traffic at a game server due to server discovery is independent of the game server's game-play popularity

Thousands of probes, only a few people will play.....

Wolfenstein Enemy Territory example



Picture from: G.Armitage, M. Claypool, P. Branch, "Networking and Online Games - Understanding and Engineering Multiplayer Internet Games," John Wiley & Sons, UK, April 2006 (ISBN:0470018577)

Same sequence for Quake III Arena.
Similar probe-response behavior for other FPS games such as Half-Life, Half-Life 2, etc....

Potential impact of probe traffic (2004)

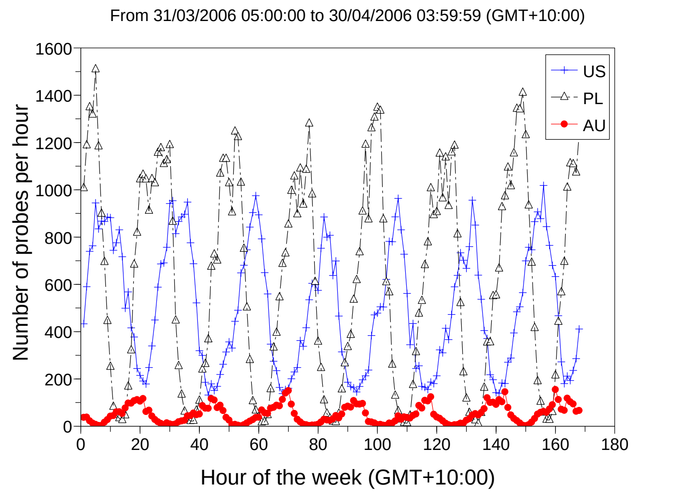
	CAIA		GrangeNet	
	Probe	Non-Probe	Probe	Non-probe
Flows	16.18e6 (99.95%)	7993 (0.05%)	16.93e6 (99.99%)	1757 (0.01%)
Mpackets	36.46 (4.61%)	755.13 (95.39%)	36.94 (25.01%)	110.74 (74.99%)
GBytes	8.18 (6.56%)	116.58 (93.44%)	8.10 (35.75%)	14.56 (64.25%)

- "CAIA" and "Grangenet" servers both east coast Australia over 20 weeks
- CAIA vastly more popular with players (116 Gbytes vs 14Gbytes of traffic)
- Yet both servers see ~8Gytes of probe traffic
- Even worse: 16 million distinct IP flows due to probe traffic (vs < 8000 for game play)
  - An issue if you're keeping per-flow statistics on your local network link(s)

S. Zander, D. Kennedy, G. Armitage, "Dissecting Server-Discovery Traffic Patterns Generated By Multiplayer First Person Shooter Games," NetGames 2005, NY, USA, October 2005 http://researchweb.watson.ibm.com/netgames2005/papers/zander.pdf

Enemy Territory in 2006

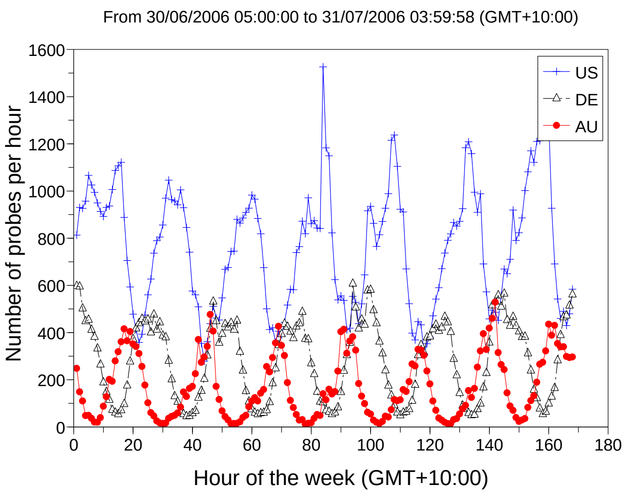
- ET server hosted in Melbourne
- '*getinfo*' probes gathered over a month
- Average probes per hour vs 'hour of the week'
- Broken out by source region, phase-shifted 24-hr cycles evident



G. Armitage, "A Packet Arrival Model for Wolfenstein Enemy Territory Online Server Discovery Traffic" (under review, 2007)

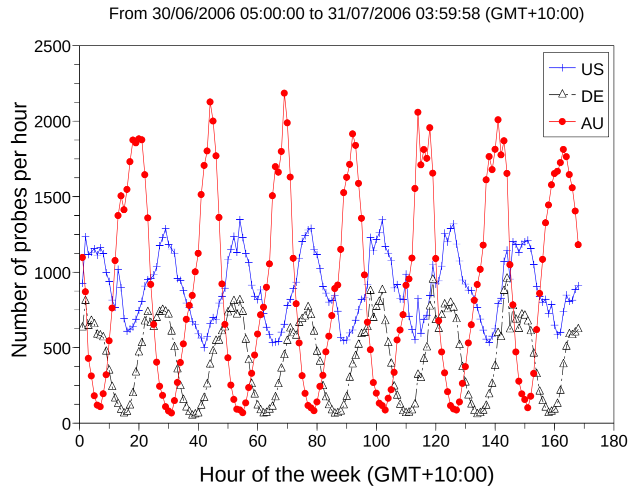
Half-Life 2 Deathmatch in 2006

- HL2 DM server hosted in Melbourne
- '*Tsource Engine Query*' probes over a month
- Average probes per hour vs 'hour of the week'
- Broken out by source region, phase-shifted 24-hr cycles evident



Counter-Strike: Source in 2006

- CS:S server hosted in Melbourne
- '*Tsource Engine Query*' probes over a month
- Average probes per hour vs 'hour of the week'
- Broken out by source region, phase-shifted 24-hr cycles evident

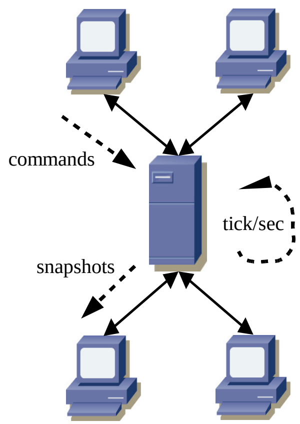


Overview

- What are First Person Shooter (FPS) games?
- Reasons to care about online games
- Evidence of human-driven traffic patterns
- Player tolerance to network degradation
- The background noise of server discovery
- Game traffic characterisation
- Challenges facing game developers and network operators

Basic model of FPS communication

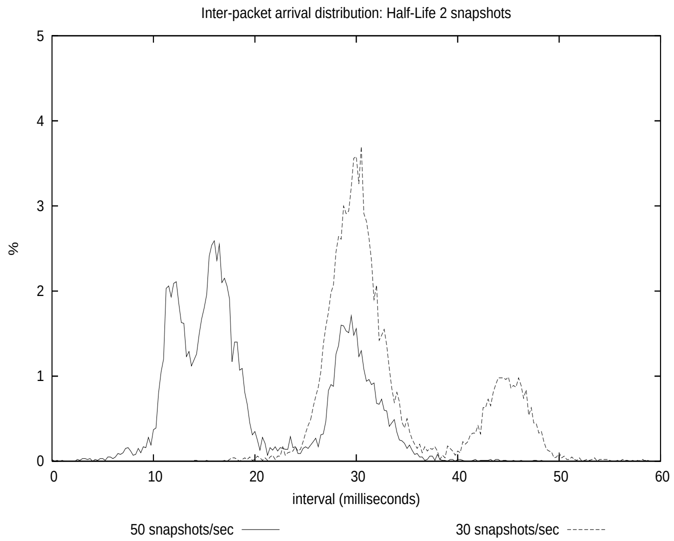
- Servers have internal *tick* rate
  - Game state updates every tick
- Clients receive *snapshots* and send *commands*
  - Snapshots convey game state to clients, at client-specified rate
  - Server will send snapshots every tick or multiple of tick intervals
    - May round up to next multiple of tick time (e.g. Q3, ET) or average over many ticks (e.g. HL2), depending on server design



http://developer.valvesoftware.com/wiki/Source_Multiplayer_Networking

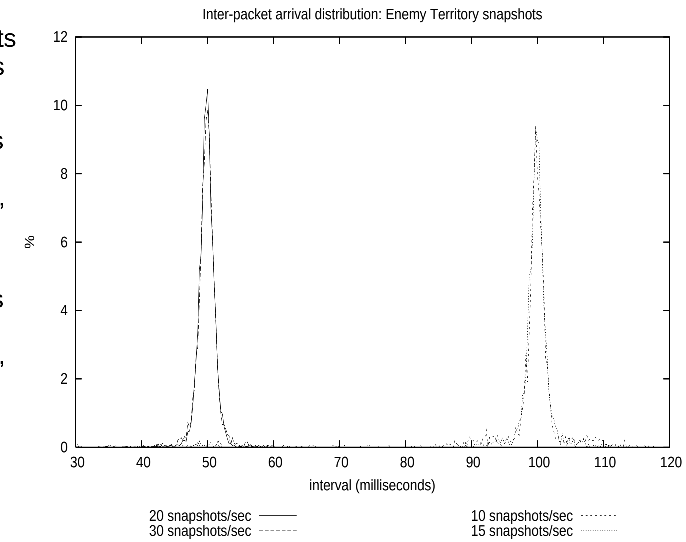
Half-Life 2 snapshot intervals

- Server 'ticks' 66 times/sec, snapshots at multiples of 15ms
- Clients asks for 30 snapshots/sec, gets a mix of 30ms and 45ms intervals between snapshots, average 30 snaps/sec
- Clients asks for 50 snapshots/sec, gets a mix of 15ms and 30ms intervals between snapshots, average 50 snaps/sec



Enemy Territory snapshot intervals

- Server 'ticks' 20 times/sec, snapshots at multiples of 50ms
- Clients asks for 30 snapshots/sec, gets 50ms intervals between snapshots, for 20 snaps/sec
- Clients asks for 15 snapshots/sec, gets 100ms intervals between snapshots, for 10 snaps/sec

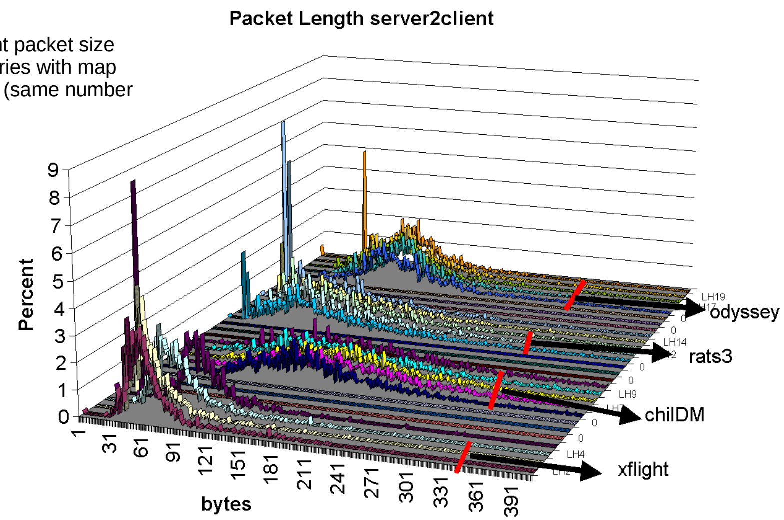


Estimating impact on queuing....

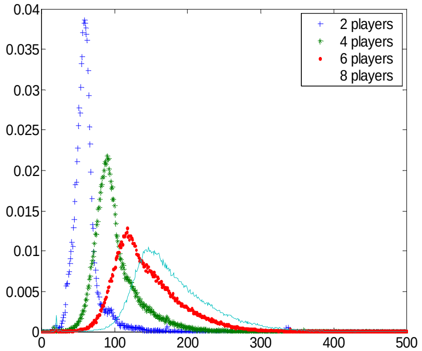
- Knowing the inter-packet intervals is part of it
- We ideally would like to model/predict packet size distributions
- Packet sizes from client->server are generally quite small (50 – 80 bytes) and constrained range
- Packet sizes from server->client see large variation
  - Due to number of players
  - Due to map style/layout
  - We can extrapolate from distributions captured using small numbers of players...

Half Life packet size distributions

Server to client packet size distribution varies with map type and style (same number of players)

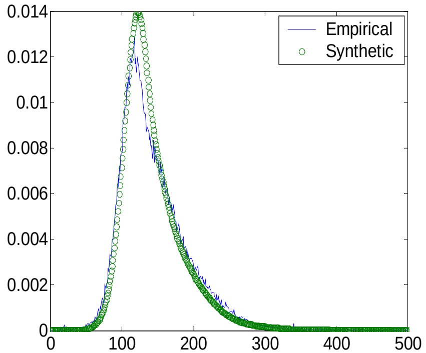


Extrapolating packet size distributions



Actual (measured) Quake III Arena packet size distributions for 2 to 8 players on same map

Turns out we can create a synthetic N-player distribution by (N/2) convolutions of an empirical 2-player distribution

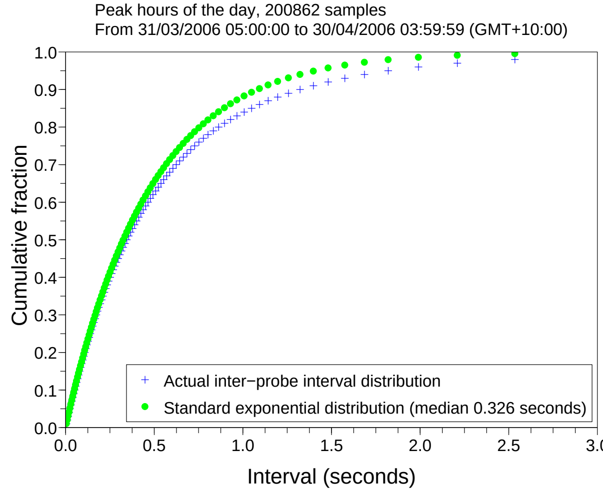


Real and synthetic packet size distributions for 6 player scenario

P. Branch, G. Armitage, "Extrapolating Server To Client IP traffic From Empirical Measurements of First Person Shooter games," 5th Workshop on Network System Support for Games 2006 (Netgames 2006) Singapore, 30-31 October 2006.

Modeling (ET) server-discovery traffic

- Complements our knowledge of game-play traffic patterns
- ET inter-probe intervals during busiest 'typical hour' seem Exponentially distributed
- (Same for least-busy hour)



G. Armitage, "A Packet Arrival Model for Wolfenstein Enemy Territory Online Server Discovery Traffic" (under review, 2007)

Overview

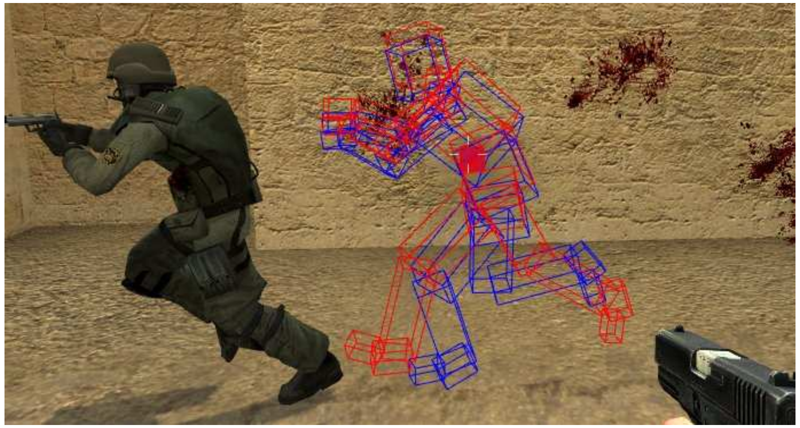
- What are First Person Shooter (FPS) games?
- Reasons to care about online games
- Evidence of human-driven traffic patterns
- Player tolerance to network degradation
- The background noise of server discovery
- Game traffic characterisation
- Challenges facing game developers and network operators

Lag (latency) and loss compensation

- FPS games have evolved techniques to compensate for network level latency and loss
- Latency introduces timeline discrepancy between each client's view of the virtual world
  - Loss is an extreme form of discrepancy
- Excellent summary by Valve
  - *http://developer.valvesoftware.com/wiki/Source_Multiplayer_Networking*
- Client-side vs server-side techniques
  - Client can interpolate for smooth viewing, but only server is trusted to decide 'official' events (e.g. hits)

Lag compensation (server side)

- *http://developer.valvesoftware.com/wiki/Source_Multiplayer_Networking*



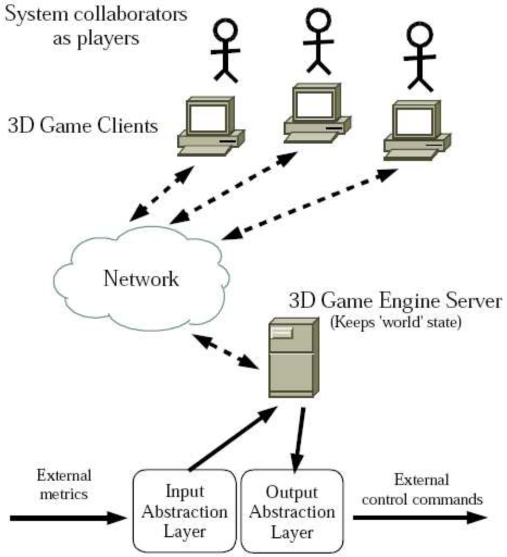
A challenge to control latency

- Given installed base of hosts, network and CPE (consumer routers, modems, Access points, etc)....
  - How do you incorporate control of IP QoS mechanisms into operational networks?
    - Require operating system support in end-hosts?
    - Require game engine redesigns to use signalling?
    - Provide ISP-side automagic detection of game traffic, and 'suggestions' to CPE equipment?
  - Can we do intelligent network planning, knowing the packet size distributions and arrival patterns?

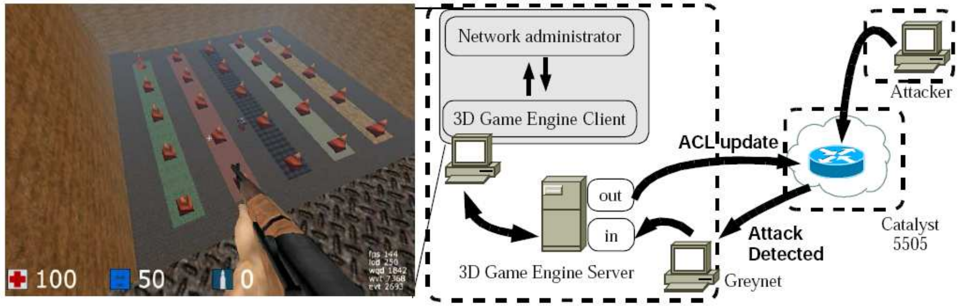
.......or do we just make links faster ?

And for something a little odd...

- Using 3D game engines as interactive network management tools
- Dynamic network state represented by animated avatars in virtual world
- Administrators 'view' activity, interact with avatars
- Interactions translate into network control events



Running network as a virtual world



- First prototype used CUBE game engine
  - W. Harrop, G. Armitage, "Real-Time Collaborative Network Monitoring and Control Using 3D Game Engines for Representation and Interaction," VizSEC'06 Workshop on Visualization for Computer Security, Virginia, USA, October-November 2006.
- Current L3DGEWorld based on Quake III Arena
  - http://caia.swin.edu.au/urp/l3dge

Thank you for listening to this cursory overview of one of the areas currently holding my interest....