Multiplayer Game Programming

Lecture 9: The Real World

ITP 484

Review Time!

- Why is it a good reason for the server to process more than one incoming packet per frame?
- Give a reasonable method for determining how many packets to process per frame
- What's the difference between a Remote Procedure Call and a Remote Method Invocation?
- Explain how a Stub or Proxy RPC system works.
- Explain a good way to pass dynamically allocated game objects as parameters to RPCs

Issues for multiplayer games in the real world

- Collaboration between distant hosts
- Must be consistent in real time
- Large Interactive World with Complex State
- Simulation, Rendering and Network sync can all run at different speeds
- Second order interactions between objects owned by different hosts

Kinds of Consistency across hosts

- Absolute Consistency
  - Everything is the same
- Absolute View Consistency
  - Everything looks the same to the user
- Approximate View Consistency
  - Off by few ms due to display latency, simulation/ render sync, etc.
- Good Enough Consistency
  - Time-shifted by network latency, as consistent as necessary

Good Enough Consistency

(From Steed)

- State changes at each host replicated to other hosts
- Each host's state converges over time
- Causality preserved
- Temporal characteristics of events preserved

Good Enough Consistency (More Subjective Edition)

- Events "jointly plausible" to multiple viewers
  - Local plausability vs. joint plausability
- Outcome feels fair
  - Fair access: equal access to interaction by hosts
  - Fair outcome: event outcome independent of host
- Intensions of users are preserved
  - More than just input

Consistency is our job

Latency is our enemy

What is Latency?

- No shared formal definition of latency
- "Network Latency" sometimes used to describe half RTT or half "ping time"
- But it's not that simple

Where does Latency come from?

- "End to end graphics latency"
  - Input Sampling Latency
    - Input only sampled once per simulation frame
    - Value comes from OS sampling of input!
  - Render Pipeline Latency
    - Render pipe a frame or more behind simulation
  - VSync delay
    - To avoid tearing, have to wait for refresh
  - Monitor Latency
    - Deinterlacing, adjustment to native resolution, HDCP, motion interpolation
    - Often up to 3 frames of buffering required!
  - LCD pixel transition time
    - 10 ms on a good display!

“It’s faster to send a packet to Europe than a pixel to your screen”

-sorta John Carmack

http://www.geek.com/chips/john-carmack-explains-why-its-faster-to-send-a-packet-to-europe-than-a-pixel-to-your-screen-1487079/

http://www.altdevblogaday.com/2013/02/22/latency-mitigation-strategies/

But wait there's more!

- "Network Latency"
  - Time taken from network layer of source host to network layer of destination host

Network Latency

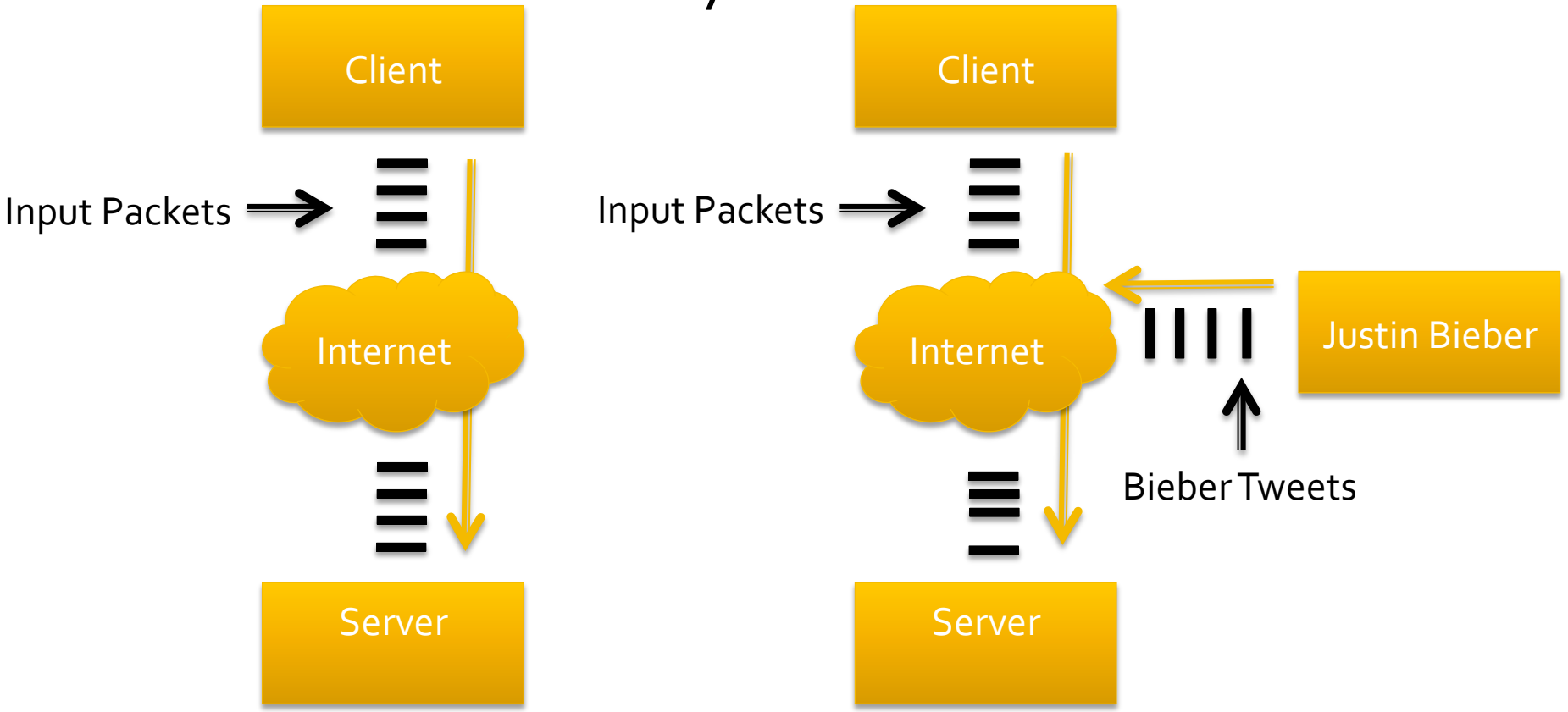
- Processing delay: router processing
  - Choosing next gateway
  - Rewriting for NAT, etc.
- Queuing delay: when route or router is busy
- Transmission delay: sending a packet to a link
  - Based on speed / protocol of link
    - 10/100/1000 Mbps Ethernet, 3Mbs ADLS
- Propagation delay: Through the physical medium
  - Usually near the speed of light in the medium
    - Roughly between 3 and 5 µs/km

It's not over yet!

- Unnamed latency
  - Network to Transport layer processing by OS
  - Sampling of Transport layer data by Application
    - Similar to problems with Input sampling

Jitter

- Variation in latency over time:



Jitter problems

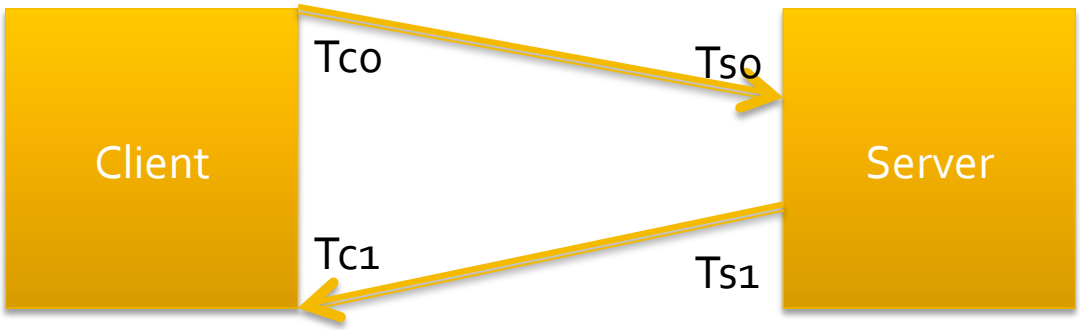
- Packets can be delayed more than expected
  - Two inputs arrive on same frame
  - Two state updates for same object on same frame
  - Packets arrive out of order!

How to measure latency

- Naïve approach:
  - Every computer has a clock!
  - Send the time in each packet
  - When a packet arrives, the difference between the time in the packet and the current time is the latency!
- Problem: Clocks probably aren't synced
  - Almost definitely not to the ms

Pragmatic Approach

- Rough Estimate: RTT / 2
- Slightly less rough estimate
  - Factor in processing time in the middle



$$RTT = (Tc_1 - Tc_0)$$

$$LatencyFromNetwork = \frac{(Ts_0 - Tc_0) + (Tc_1 - Ts_1)}{2}$$

- Less useful for game simulation, more useful for strictly measuring network conditions

We can measure latency!

But so what?

Latency is like overcooked spinach:
There's no way to get rid of it, but if you spread it around on your plate enough, your mom might not notice it's still there, and you can have dessert.