Service Traffic Models: Email, Online Gaming and PTT

VU 389.134
Video and Multimedia Transmissions over Cellular Networks

25.01.2010
(http://www.nt.tuwien.ac.at/teaching/courses/winter-term/389134)
Questions from the last time

**EXAM:**
- DATE: Fr. 29.1.2010 (Duration ca. 30 min)
- Room: Sem 118 (here)
- Please register via mail (so we can make up a schedule)
  - psvoboda@nt.tuwien.ac.at and/or mrupp@nt.tuwien.ac.at
- Till now: J. Bartos, G. Maier, registered

- What does the control flow in FTP do
- What is the impact of scripts to HTTP modeling
- How to build a video stream for network traffic modeling?
- Transfer of a web-page
  - Connection pipelining?
  - Reading time
  - Main, inline objects?
Overview of this lecture

- Service level traffic models
- Email
  - Impact of large RTT values
- Online Gaming
  - Type
  - Setup
  - Impacts
- Push to Talk
  - Protocols
  - Models
Service Traffic Models for new Services

- New service models
  - Need for mobile broadband
  - New demanding service classes
  - Long term goals for mobile internet access technologies

- UDP based online games (FPS, RTS)
  - No interaction on the protocol layer
  - Modeling based on
    - Packet size
    - Packet rate

- TCP based online games (MMOG)
  - Interaction on the protocol layer (TCP)
  - Modeling based on derived input parameters

- Push to talk over mobile
  - Matlab model based on ITU model for artificial voice communications

FPS... First Person Shooter
RTS...Real Time Strategy
MMOG... Massively Multiplayer Online Games

Philipp Svoboda
Repetition of the last lecture

2G

3G

Video and Multimedia Transmissions over Cellular Networks 389.134
Modeling Network Traffic: Service Level

- User access services
  - Reproduce service per user
  - Allows to measure QoS

- Definition of a service
  - What is HTTP?

- Common service models
  - Video, HTTP, E-Mail, ...

- Service classification
  - Detect service on user level
Service Models for 3G Networks
EMail
Service Level Network Models: High RTT

- Every model simplify
  - One email size for all users
  - Home/Standard/Heavy user

- Assumptions are the starting point for any model
  - Same client applications for the same service
  - Low latency network

- Example Authentication
  - Assumption: Low delay network
    • Authentication like a small burst of packets
    • No need to include in a model
  - Assumption: High delay network
    • Authentication spread over long period
    • Important part for the model
Example: Email Authentication

- U: Data from the user
- S: Data from the server
- # Server starts to give some information "variable number of Strings"
  - S: "Connected to mobilkom.at"
  - S: "Escape character is '\^\]'."
  - S: "+OK QPOP (version 2.4) at mobilkom.at starting."

- # Client sends its ID
  - U: user hugo@nowhere.at

- # Server checks user, requests pass if needed
  - S: +OK USER hugo@nowhere.at set, pass?
- # Client sends the Pass, Server verifies
  - U: PASS xxxxxxx
  - S: +OK pop3 80

- # Client asks for mail
  - U: STAT
  - S: +OK 7(mails) 60000(bytes)
Email: Extended Service Models

- Common email models (Paxson, FUNET, Stuckmann)
  - Server pushes email according to a CDF
  - No client requests
  - No login process
  - Less TCP ACKs

- Adaptation to the model
  - 10-12 packets as login process
  - Similar size (40-60 Bytes)
  - CDF from METAWIN traces
  - Enhancement
    - High RTT is visible in the simulation.
    - Empty mail checks can be separated.

Model Parameters: Mail Size

- Analyze TCP streams per service request
- 2 Paths
  - Mail retrieval (20.5%/25.4%)
  - Login & check (79.5%/74.6%)
Parameterize new Mail Model

- How many mails?
  - Mails per Users
  - Service calls per time
ns-2 Simulation Setup

- 2 Nodes
- TCP application over FullTCP
- Login process
- Variable bandwidth and delay

Link bandwidth
Delay

Output:
BW, Time
ns-2 simulation with login

- Mean e-mail size: 18.12 kB
- Login: 8 Pkt + processing time
Mail Traffic (Measured vs. Simulation)

Service Footprint (April 2007)

Measured

Simulated

- Without Login
- With Login
Service Models for Online Gaming
Service Level Network Models: Online Gaming

- Game played over some kind of computer network
  - Real time application
  - Broad field of different types
  - Online only games

- Low data-rate, real-time service
  - Different service class than video streaming

- Client
  - Web-Browser
    - Java, Flash
    - Flexible platform
  - Game client
    - Higher performance possible
  - Realtime $\rightarrow$ UDP transport protocol
  - Security
Types of Online Gaming

- Solo- games with **shared Results**
  - Download of JAVA games to the mobile
  - YETI SPORTS, ....

- Strategy:
  - Real Time
  - Sessions

- Massive Multiplayer Games

- First Person Shooter

http://www.yetisports.org/
http://www.battle.net/
http://www.eve-online.com/screenshots/
http://www.unrealtournament.com
Network Setup

Flat Topology
- Equal clients
- Synchronized games
- Single failure stops the session
- Simple logic

Tree Topology
- Server sets the rules
- Users can join after session start
- Delay tolerant
- Lose coupled
UDP based Online Games: Unreal Tournament (UT)

- UT Model
  - Parameters measured on wired networks
  - Uplink: data dependent on the client input
  - Downlink: regular updates from the server
  - Session times extracted from measurements

---

**P. Svoboda, M. Rupp: “Online Gaming Models for Wireless Networks”, IMSA 05, Switzerland, 2005.**
TCP based Online Games: World of Warcraft

- Data transmission over TCP
  - Bytes not packets
  - Nagle algorithm
  - PSH-flag reassemble

- Simulation for World of Warcraft

Quality Considerations for Online Gaming
Delay

- Time is relative...

- Counteractive measures:
  - Prediction of client input and objects
  - Suitable for delays < 100ms

- Impacts: FPS, Racing
- No Problem: Strategy, MMOG
• A picture tells more than 1000 words:

- Latencybuffer → does not work for interactive applications
- Interpolation → can increase your problems

• Counteractive measures:

• Online Gaming QoS ≠ Video QoS
  - Less tolerance and bandwidth!

Latency Compensating Methods in Client/Server In-game Protocol Design and Optimization
Audio in a Packet Switched Environment (Push to Talk)
Introduction to Push to Talk (PTT)

- Uni-cast (near real-time) audio service over packet switched networks
  - Uni-cast: one station sending at a time
    - Walky Talky like communication
  - Audio-service: convert analog audio into packets
    - low data-rate, small packets, AMR codec
  - Over PS: no fixed connection between partners
    - Delay in packet delivery

- Shift high quality audio CS to low quality audio PS
  - Less data-rate per minute voice
  - Less load to signaling plane
Introduction to Push to Talk (PTT)

• PTT Protocol details
  - AMR coded payload
  - RTP protocol
  - UDP transport

• Signaling
  - Session setup (ringing)
  - Heartbeats
  - ...

• Conversation
  - Model for voice: ITU P.59.03
  - Defines “heat” of a communication
    • Hot = intense/strong

• Higher delay?
Push to Talk over Mobile

- Simulation results
  - GPRS time slot configuration
  - Medium user intensity
  - 99% parallel active users → number of lines needed
Summary of this Lecture

• Design models with respect to the network in use

• Email traffic model
  - Need for emulation of an authentication procedure (3G case)
  - Service patterns change in time

• Online gaming models
  - UDP / TCP oriented services
  - Service states
  - Delay / Jitter

• Push to talk
  - Conversation model
  - Considerable savings possible
Some space for your input

- Good stuff you liked?

- Bad stuff you liked?
Thank you for your attention

Questions?
psvoboda@nt.tuwien.ac.at