CHIANTI

Towards More Adaptive End-to-End Applications

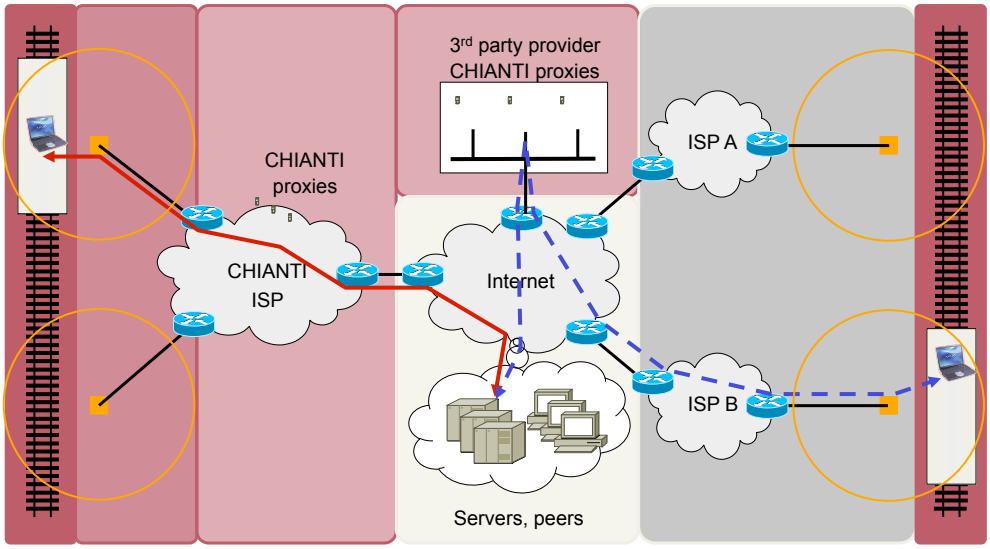
Jörg Ott jo@comnet.tkk.fi

9 June 2009

Euro NF Workshop Future Internet 2009

CHIANTI

Background: Challenged Mobile Internet Access



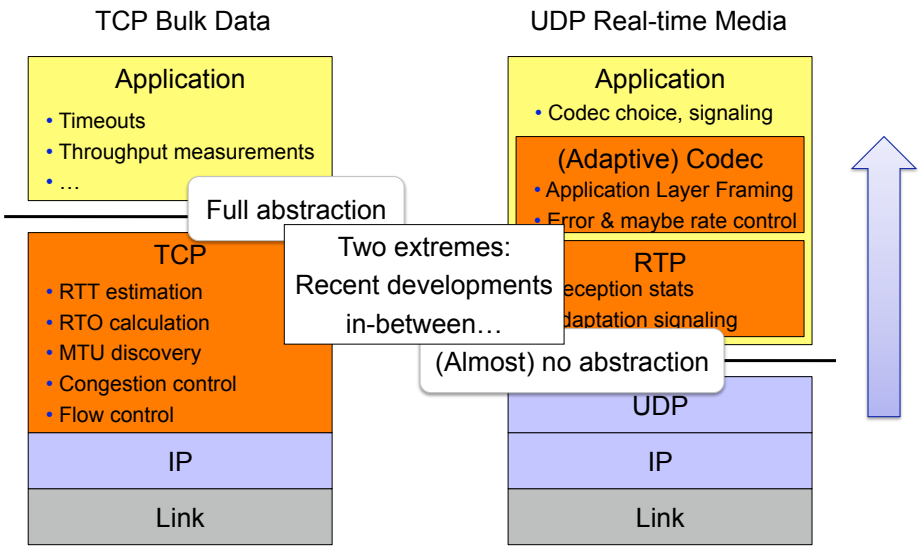
Adaptive Applications

- Varying radio link conditions and coverage/connectivity
  - Often less predictable changes
  - Congestion vs. errors
- Varying path characteristics in the Internet
  - Variable load
  - Route changes

Delay **Loss** **Date Rate** **MTU**

- "Fair" sharing of communication resources
  - Utilize available resources effectively, but do not overload
- Obtain sufficient application performance in spite of the above

Implementing Adaptivity: Examples



Implicit Assumptions

1. Applications are capable of adapting across a sufficiently wide range of communication characteristics

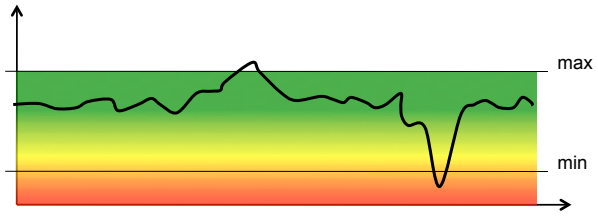
OR

2. The best effort service delivered will just be good enough for the applications to work well

Adaptive Protocols and Applications

Delay **Loss** **Date Rate** **MTU**

- Limitations in the operational range
  - Minimum performance requirements needed for acceptable operation
  - Maximum they are (practically) able to utilize (mostly data rate)
  - Capability to “cancel out” over- and underperforming over time



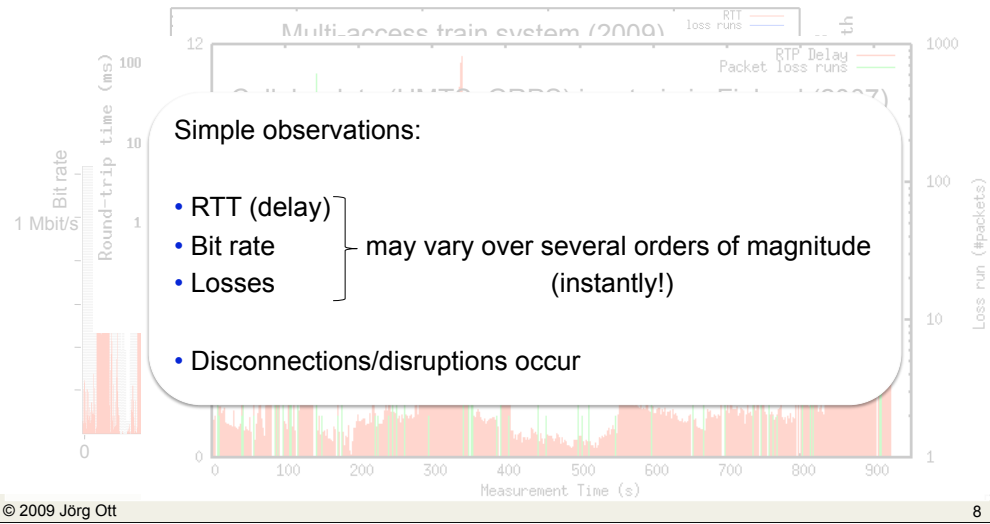
- With insufficient performance, users may get annoyed, give up, need to retry (later), …

Some Examples…

	Delay	Loss	Data rate	MTU
Bulk data	Don't care as long as TCP does not stall or disconnect… (P2P even better)			
Interactive web	RTT<300ms	Low	100 kbit/s – 1 Mbit/s	1500 bytes ok
	Interactivity = f(loss, delay) needs to be sufficient			
Streaming	seconds	low	100 kbit/s – 100 Mbit/s	1500 bytes ok (could be larger)
	Data rate = f(loss, delay) needs to be sufficient			
VoIP	< 200ms	< 5%	4 kbits – 100+ kbit/s	< 100s bytes

When Best Effort is Not Enough…

- Mobile Internet access and wireless networks



Simple observations:

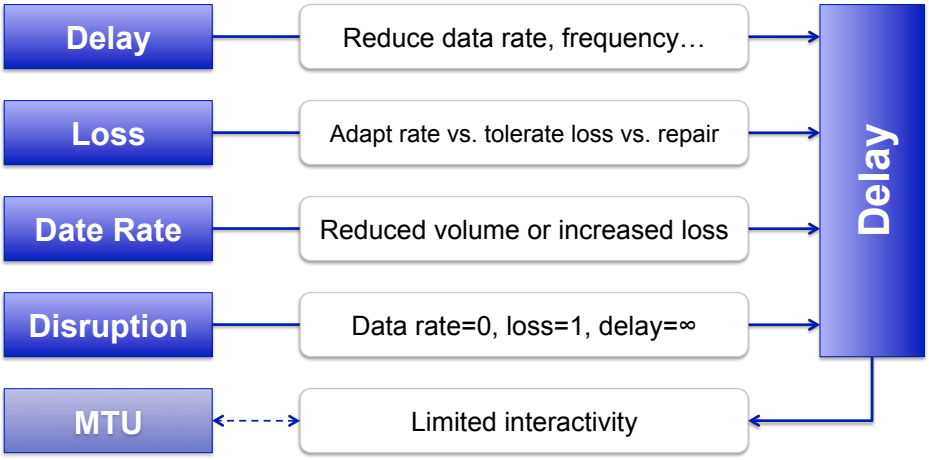
- RTT (delay)
- Bit rate
- Losses

} may vary over several orders of magnitude (instantly!)

- Disconnections/disruptions occur

Adaptation continued…

Given a path and the need to send a certain amount of information…



Exploring sample applications…

1. Disruption-tolerant SIP
2. Asynchronous Voice

Example 1: DT-SIP

- Looking at a well-known issue for mobile phone users
  - Dropped calls due to coverage gaps
- The usual remedy: THHP (The Human HELLO Protocol)

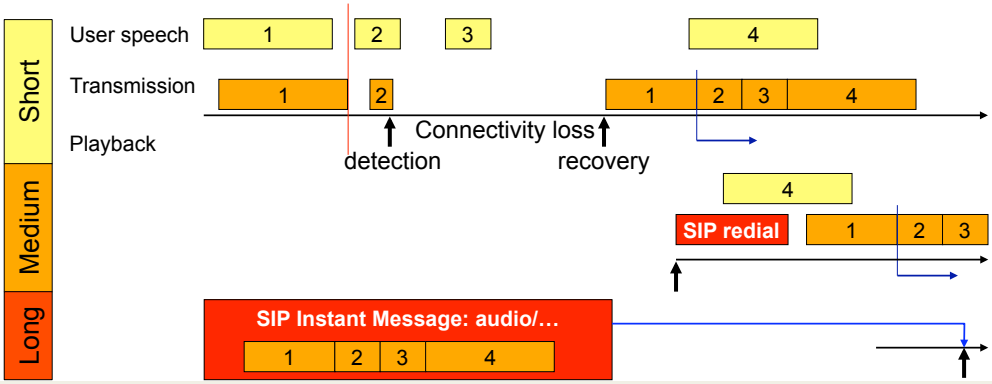
Simple case study:
Adding disconnection tolerance to SIP-based VoIP

- 3 Phases
  - Be prepared for connectivity loss: keep a copy of a window of voice
  - Detect disconnections
  - Handle disconnections

Jörg Ott and Lu Xiaojun: **Disconnection Tolerance for SIP-based Real-time Media Sessions.** International Conference on Mobile Ubiquitous Multimedia (MUM'07), Oulu, Finland, December 2007.

Disconnection Handling

- Sender: records, stops sending, listens for reconnections
  - Assumption: users stop talking after a moment
- Receiver: duplicate filtering, resynchronization
  - Both skip silence to catch up

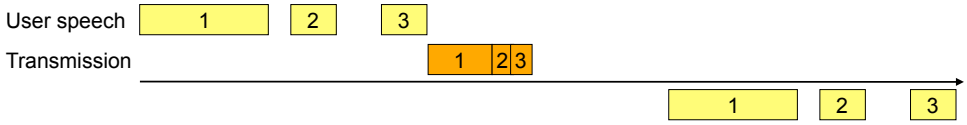


Example 2: DT-Talkie

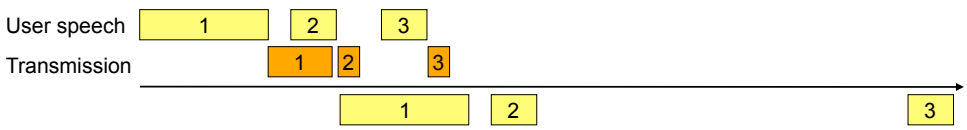
- Asynchronous audio in delay-tolerant (ad-hoc) networks
  - Experimentally driven development
  - Targeting voice interaction without (too) tight time bounds
  - Walkie-talkie-style interactions between individuals and in groups
- Push-to-talk communication without infrastructure
  - Single-hop or multi-hop ad-hoc networks
    - Bluetooth and WLAN
    - Might use APs for connectivity as well
  - Reliable hop-by-hop store-and-forward to deal with packet losses
    - Speech quality is not impacted, only delay is
  - Delay tolerance: decoupling sender and receiver
    - Asynchronous interaction without dedicated mediator
  - Store-carry-and-forward with replication for extending reachability
    - Bounded message lifetimes

DTN-based Voice

- Plain and simple: record – send – forward – receive – playback
  - Based upon user-indicated (button press) statements

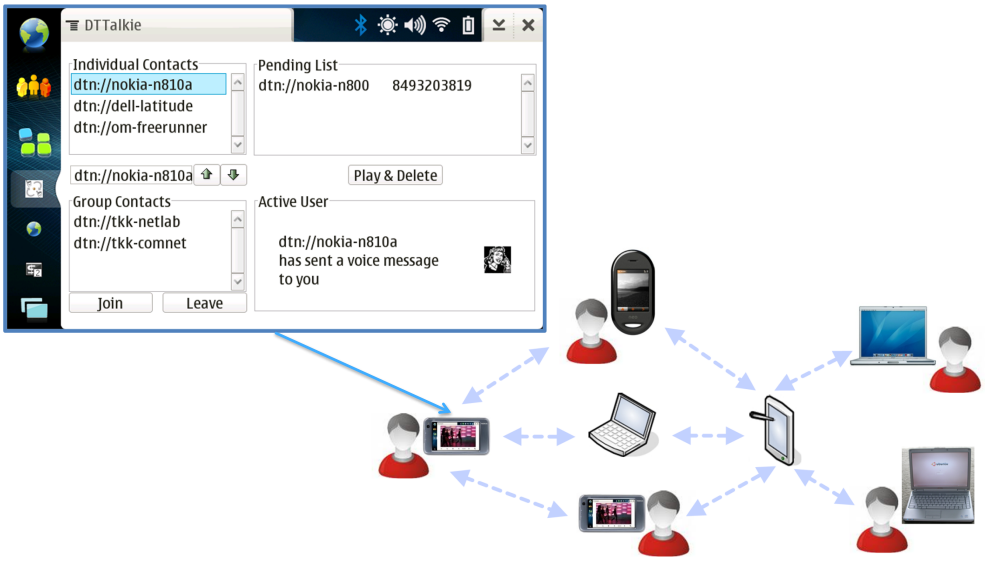


- Subtleties: message size?
  - Semantic fragmentation (Application Layer Framing)
  - Keep talkspurts together (“MTU”)
  - Good connectivity and short messages: interactive communication workable



- Subtleties: codec interoperability (no negotiation signaling)

Sample Operation



Synthesizing…

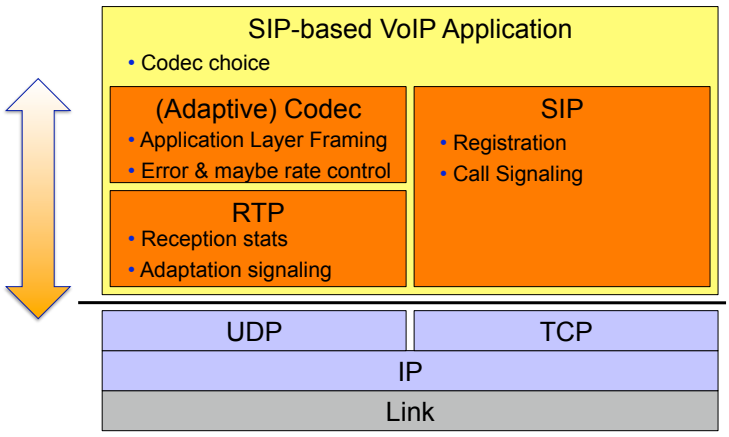
A Case Study for Adaptive Audio

What do we have so far...?

- Extending the “playout delay” allows a wider range of adaptation
- DT-SIP: Integrating synchronous and asynchronous voice
  - But why just for disconnections – and what is a disconnection?
  - How many lost packets in a row start getting disturbing?
  - What is in the space between packet- and dialog-based repair?
- DT-SIP: Artificial division into discrete repair cases
  - Isn’t this more a continuum?
  - What timeouts to use? ➔ more adaptivity (+ learning?) needed
- DT-Talkie: Quality audio in multihop wireless networks
  - Trading interactivity for flawless speech quality
  - Controlling message size via segmentation into talk spurts (“hard-coded”)
  - Can still be “fairly” interactive

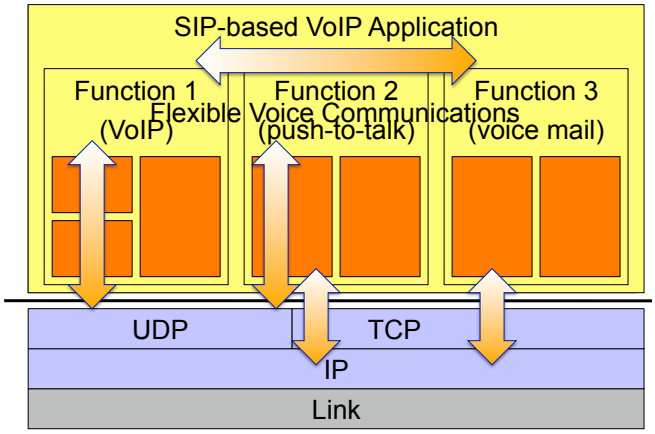
A Simple SIP Application

- Vertical adaptivity for VoIP yields a certain operational range



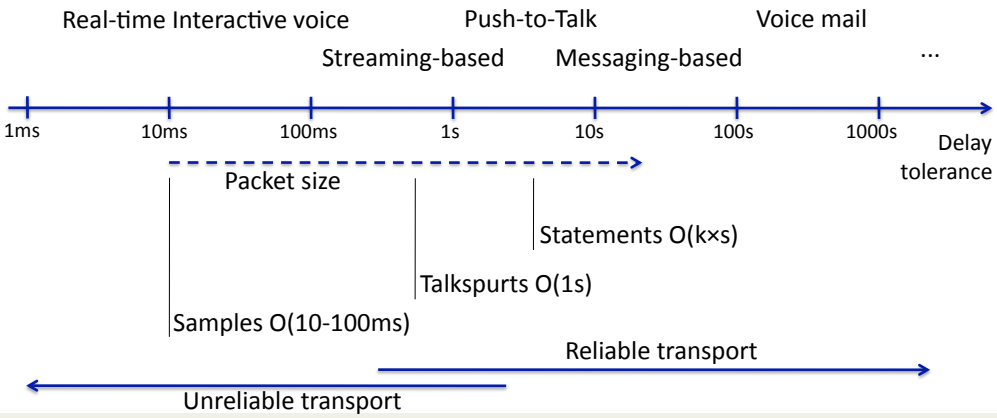
Taking a step back looking at the semantics…

- Advancing individual (vertical) adaptation per function…
  …to integrate them across different ones (horinzontal adaptation)



…to fully adaptive Voice

- Macro adaptation: data unit size and reliability as a function of path properties and delay tolerance
- Micro adaptation: error and rate control



Case Study Summary

- Going back to the purpose of the application (protocol)…
  - Voice-based communications
  - May serve many different purposes with different demands
- …may open up wider adaptation possibilities
  - Within applications functions (RTP over UDP vs. RTP over TCP)
  - Across application functions (do these actually need to stay separate?)
  - In some cases, horizontal adaptation is fine, in others hanging up is better
- Endpoints matter
  - Enable easy transition back to synchronous communications
  - User interface should support ping-ponging voice messages
  - Presence indications?
  - Usability is really the key issue here!

Random Thoughts: Further Applications

- Is voice communication just the only one?
  - Seems conceivable to straightforward to just add video…
- Other application classes:
  - Don't care: file transfer, etc.
    - Peer-to-peer systems are already perfectly adaptive
  - Related kind: media streaming
    - Straightforward simple adaptation mechanisms
    - Reduce quality, still images, remove video: all heavily content-dependent
    - Option to exploit multi-path connectivity via p2p overlays
  - A little but not entirely unlike: Interactive web, etc.
    - RSS feeds vs. pull/push-based pre-load/caching vs. interactive retrieval vs. download
    - Different degrees of interaction
- How many ends available to adapt?

More Random Thoughts…

- How is adaptivity supposed to work?
  - Specific vs. generic monitoring mechanisms?
  - Time-scale?
- How about fairness…?
  - Is additional delay another dimension to consider?
  - E.g., more data in return for less urgent data?
- And complexity?
  - Can protocol and application designers get this right?
  - Can some common abstraction be provided?
- Finally, interoperability…?
  - Hard to get with a single mode of operation, now using multiple?

And the Future Internet…?

- Can we derive a generalizable model…?
  - And a common abstraction?
  - Split between transport and application layer
  - How much of an abstraction is tolerable (or implementable) to begin with?
- What would we expect from the network – if anything?
  - Targeting an end-to-end approach surely helps incremental deployment
  - Delay-tolerant delivery?
  - Tailorable packet sequence reliability (all-or-nothing semantics)?
- What about BSD Sockets…?
  - Always cursed, but kind of workable and people got used to them…

Conclusion

- Challenges in (mobile) communications won't go away
- Allowing for delay tolerance may extend adaptation capabilities
- Requires looking at the intention and semantics
- Case study has shown that some applications may be viable
- Applicability and adaptivity limited by application and "content"

CHIANTI project:
http://www.chianti-ict.org/

Finnish ICT-SHOK Future Internet project:
http://www.future-internet.fi/