Can Wireless Preserve the E2E Argument?

- Dumb vs. Flow-Adaptive Link Layers (LL)
- Low vs. High LL ARQ Persistency for TCP

Reiner Ludwig
Ericsson Research
Link Layer Design Philosophies

The “Dumb” Link Layer
(aka transport-blind, one-size-fits-all, …)

- Streaming
- Bulk Data (TCP)
- Interactive (TCP)

The “Flow-Adaptive” Link Layer
(aka transport-aware, …)

- Streaming
- Bulk Data (TCP)
- Interactive (TCP)
Wireless Link Layers SHOULD be Flow-Adaptive

- Flow-Adaptive Makes Little Sense for Wireline
  ⇒ Because:
  Wireline Link Layers have No Knobs for Tuning (not needed!)

- Flow-Adaptive Makes Lots of Sense for Wireless
  ⇒ Because:
  Wireless Link Layers have Many Knobs for Tuning:
  FEC, Interleaving, ARQ, Power Control, ...
  ⇒ Allows to Adapt Knobs to Flow’s QoS Requirements
  ⇒ Spectrum Efficiency
  ⇒ Power (Battery) Efficiency
How to Implement a Flow-Adaptive Link Layer

**Link Layer Sniffing**
(Not Maintaining Transport Layer State !!!)

- Streaming
- Bulk Data (TCP)
- Interactive (TCP)

**Clean Layered Design**
(a la E2E Argument)

- Streaming
- Bulk Data (TCP)
- Interactive (TCP)
BUT: E2E Argument Promotes Dumb LL ...

- “Everything should be done at the end-points. The network including the link layers should remain dumb.”

- E2E Argument:
  “[Link layer error control is] an incomplete version of the function provided by the communication system [that] may be useful as a performance enhancement”.
BUT: LL Sniffing is Layer Violation ...

True! On the other hand ...

   ⇒ If LL Sniffing (Layer Violation) was such a Concern:
     “Call the Layer-Police to Put the ROHCers into Jail” :-)

2. We have an alternative:
   ⇒ Extended IP/LL API
   New PILC Work Item?
Partly True …

1. People that are so Paranoid to use IPsec Gladly Trade Performance for Security.
   ⇒ People who are less Paranoid Should Use TLS.

2. DS-field is unencrypted

3. IPsec-friendly Solution Possible (unencrypted TOS IP-Option?)
Link Layer ARQ Persistency for Reliable Flows (TCP)

- Assume Flow-Adaptive LL, i.e., TCP flows are separated

- Assume LL ARQ is Possible
  ⇒ Not the case on uni-directional links (e.g., some satellite links)

- LL ARQ Persistency for TCP?

- **Definition of “LL ARQ Persistency”:**
  The Time (in milliseconds) the LL Delays a Single IP Packet in an Attempt to Successfully Transmit it Across the Link.
BUT: We do Not Need LL ARQ …

“Simply set the MTU too small and use TCP-SACK” … Does Not Work!

⇒ Optimal Frame Size on some Wireless Links is less than 100 bytes (e.g., GSM, IS95, GPRS, UMTS)

⇒ IPv6’s Minimum MTU is 1280 Bytes!

⇒ Might Work for Satellite Links: Optimal Frame Size >> 1280 Bytes
Use Highly Persistent LL ARQ for TCP

- More Precisely, LL ARQ SHOULD try for up to 64 seconds (TCP’s MAX-RTO) to Transmit a TCP Packet!

- This is NOT Saying: Unbounded Queues!
  - Queues Need to Remain Small (Active Queue Management)
  - If Queue Beyond Threshold ⇒ “Drop From Front”
  - Early Congestion Signal

- This is NOT Saying: Hop-By-Hop Instead of E2E Reliability!
  - E2E Argument:
    “[Link layer error control is] an incomplete version of the function provided by the communication system [that] may be useful as a performance enhancement”.

Why Such a High LL ARQ Persistency?

- First of all, High Delays Due to LL ARQ are Rare
  - Typically < 1 second (excluding transmission delay)
  - Mainly Occurs During Transient Link Outages

- Most Spectrum & Energy (Battery) Efficient
  - If the LL Can’t Do it, TCP can’t Do it!
  - Discarding a Packet that Already Made it 90% Across the Link Makes No Sense!
  - Measurements over GSM with LL ARQ Disabled and an MTU of 1500 Bytes Show up to 18% Undelivered Packets (Discarded by PPP due to CRC Error)
    - RFC2914: “Congestion Collapse Due to Undelivered Packets”

- Robustness Against Link Outages
  - No Need for an “ICMP-Link-Outage Agent” at the Basestation
Link Outage & High LL ARQ Persistency

- Packets Queued at the LL Restart E2E Flow
- Spurious Timeouts
- Spurious Retransmit of Entire Flight (Go-Back-N) !!!
- 70 seconds

Time of Day (s)

Sequence Number

Packets Queued at the LL Restart E2E Flow

Spurious Timeouts

Spurious Retransmit of Entire Flight (Go-Back-N) !!!
Link Outage & Low LL ARQ Persistency

- No Packets Queued at the LL to Restart E2E Flow
- Spurious Timeouts
- 30 s IDLE
- 95 seconds

Sequence Number vs. Time of Day (s)
BUT: Spurious Timeouts ...

True, they Force TCP into Go-Back-N . On the other hand ...

1. Likely to be Solved in TSV WG
   ⇒ Eifel Algorithm

2. Go-Back-N Often Less Harmful than Waiting for Long RTO
   ⇒ See Last 2 Slides
BUT: Inflated RTO ...

True! On the other hand ...

1. RTO Decays Quickly after an RTT Spike; especially when Timing Every Packet (Timestamp Option).

2. If the Path’s RTT Varies Largely, RTO should be Inflated, i.e., should be conservative.
BUT: Head of Line Blocking ...

Not True, as long as ...

... we Allow the LL to Perform Out-Of-Order Delivery Between Flows.

⇒ Requires LL Per-Flow Operations (Not Per-Flow State!)
⇒ However, No Scaling Concern on Last/First-Hop Links!
A Word on TCP Proxies

- TCP-Throughput = \( \frac{1}{RTT} \times \frac{1}{\sqrt{p}} \times C \)

- Flow-Adaptive LL + Highly Persistent LL ARQ for TCP
  - Eliminates Non-Congestion Packet Losses on Wireless Link
  - No Need for a Proxy to Avoid Influence on \( p \)
  - For High Latency Links, a Proxy Might be Needed to Avoid Influence on \( RTT \)
A Word on Robust TCP/IP Header Compression

- Flow-Adaptive LL + Highly Persistent LL ARQ for TCP
  - Eliminates Non-Congestion Packet Losses on Wireless Link
  - No Losses Between Compressor & Decompressor
  - No Need for Robustness in TCP/IP Header Compression Scheme!
  - Only Things Left to do for ROHC WG: Compression of SYN, FINs & TCP Option Fields (Timestamp, SACK, …)
The Message

1. Wireless Link Layers SHOULD be Flow-Adaptive

2. Highly Persistent LL ARQ for TCP (all fully-reliable flows)

3. If 1. not feasible, e.g., due to IPsec, Low Persistent LL ARQ (< 100 ms ?) SHOULD be Operated for All Flows
Can Wireless Preserve the E2E Argument?

YES!

The E2E Argument is (Still) THE Guideline

Leading to Well Designed Wireless Link Layers