Localized Optimizations over Path Segments

IETF 107+, LOOPS design team side meeting 2020-05-07

Note Well

- You will be recorded
- Be nice, and be professional
- The IPR guidelines of the IETF apply: see **http://ietf.org/ipr** for details.

Repo: https://github.com/loops-wg/ietf107
Notes: https://hackmd.io/3rfz-NO2RcKmxZ98QbRMHw

Agenda

- 1504Z: Quick overview (10)
- 1514Z: Technical discussion about specific items (50)
- 1604Z: Quick recap (10)
- 1614Z: Relationship to other WGs and RGs (40)
- 1654Z: Wrap-up, next meeting

Agenda

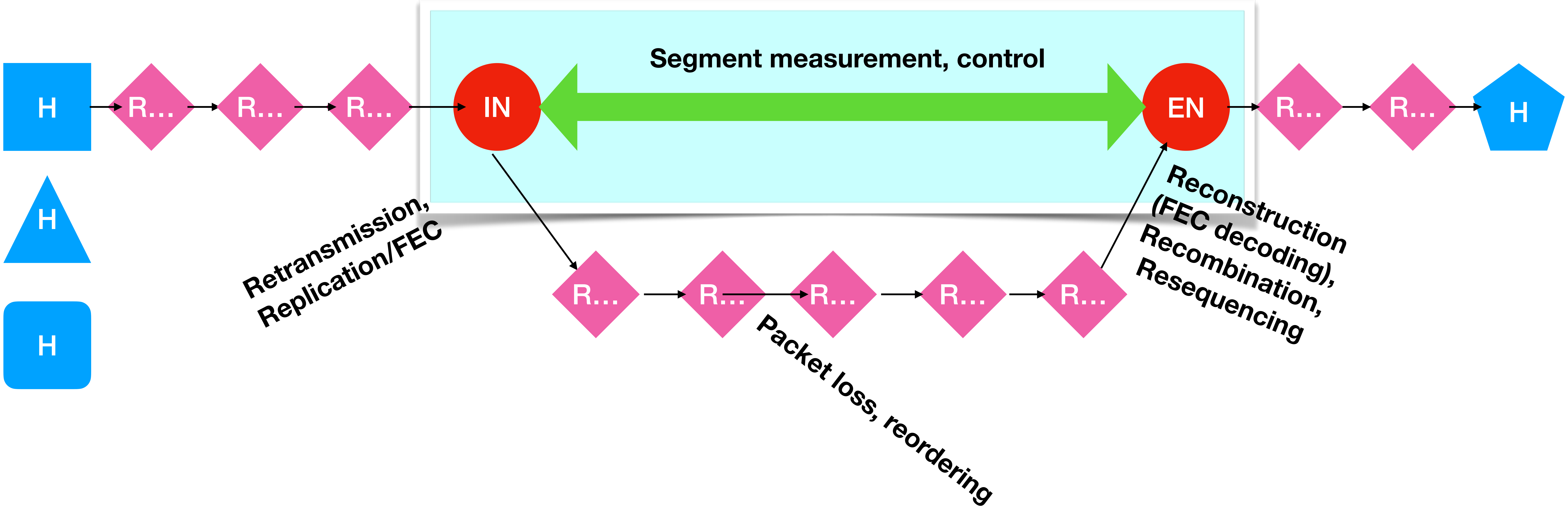
- **1504Z: Quick overview (10)**
- 1514Z: Technical discussion about specific items (50)
- 1604Z: Quick recap (10)
- 1614Z: Relationship to other WGs and RGs (40)
- 1654Z: Wrap-up, next meeting

What is LOOPS?

Localized Optimizations over Path Segments

IETF 106, LOOPS side meeting 2019-11-19
(Slides compiled by Carsten Bormann)

LOOPS Opportunity



Recover Packets Locally

Reduce end-to-end packet loss

Recover locally, **where needed**, with low latency

In the **network**

Host participation not required

Don’t look
Don’t touch

Works with any kind of IP packets

How to recover?

- **Retransmission**
  - Reverse information needed: ACK/NACK ⬋ **Piggyback (Tunnel), separate Packets**
  - Forward information: sequence numbering (if needed) ⬋ **Tunnel**
- **Reconstruction** (**Forward Error Correction,** redundancy)
  - Can use dynamic selection of block size/rate: measurement input
  - “Retransmission” also possible by adding FEC

- Aim for low setup overhead
- Keep most setup out of protocol (“controller model”)

How not to blow up the Internet

- Concealing losses removes important congestion signal
  - End-hosts would ramp up to higher rates, increase congestion

- Need **congestion feedback**
  - Preferred: ECN
  - Fallback: Selective dropping (selective recovery, actually)
- Host transport protocol improvements will help improve LOOPS performance, but are not prerequisite to obtaining benefit

Elements of LOOPS

- Information model for local **recovery**: in-network retransmission/FEC
    - Can be encapsulated in a variety of formats; define some of those
- Local **measurement**: e.g. segment forward delay/variation
    - To set recovery parameters
    - To determine if loss was caused by congestion
- Congestion **feedback**:
  ECN (or drops) to inform end hosts about congestion loss

LOOPS vs. transport protocols

- LOOPS is separate from the end-to-end transport protocol
  - Hands-off approach: don’t meddle
  - Do not assume the end-to-end protocol is out to help us, either
  - No direct control over sending rate (cc feedback only)
- LOOPS should not just be a classical transport protocol
  - Residual loss is OK
  - More choices: Tight interaction with the path segment being optimized

Where “transport protocol” intuition may not even work

- Relatively controlled/managed environment;
  setup mechanism assumed (can supply parameters so not everything needs to be high dynamic range)
- No full reliability intended; remaining gaps are OK (and at some point must leave the focus of attention)
  - Setup might set upper bound for overhead volume (e.g., 10 %), can well be “risky” in the way that this is used
- Tunnels usually have packets in flight (possibly a large number); tail processing rarely invoked (but may still be desired); don’t need overly conservative RTO

Documents out there

- Use cases and problem statement: “LOOPS (Localized Optimizations on Path Segments) Problem Statement and Opportunities for Network-Assisted Performance Enhancement” <draft-li-tsvwg-loops-problem-opportunities>
- Protocol: “LOOPS Generic Information Set” <draft-welzl-loops-gen-info>
- Encapsulations: “Embedding LOOPS in Geneve” <draft-bormann-loops-geneve-binding>, <draft-wang-loops-srv6-binding>

- Charter proposal for a LOOPS WG <https://github.com/loops-wg/charter>
- LOOPS mailing list loops@ietf.org

Agenda

- 1504Z: Quick overview (10)
- **1514Z: Technical discussion about specific items (50)**
- 1604Z: Quick recap (10)
- 1614Z: Relationship to other WGs and RGs (40)
- 1654Z: Wrap-up, next meeting

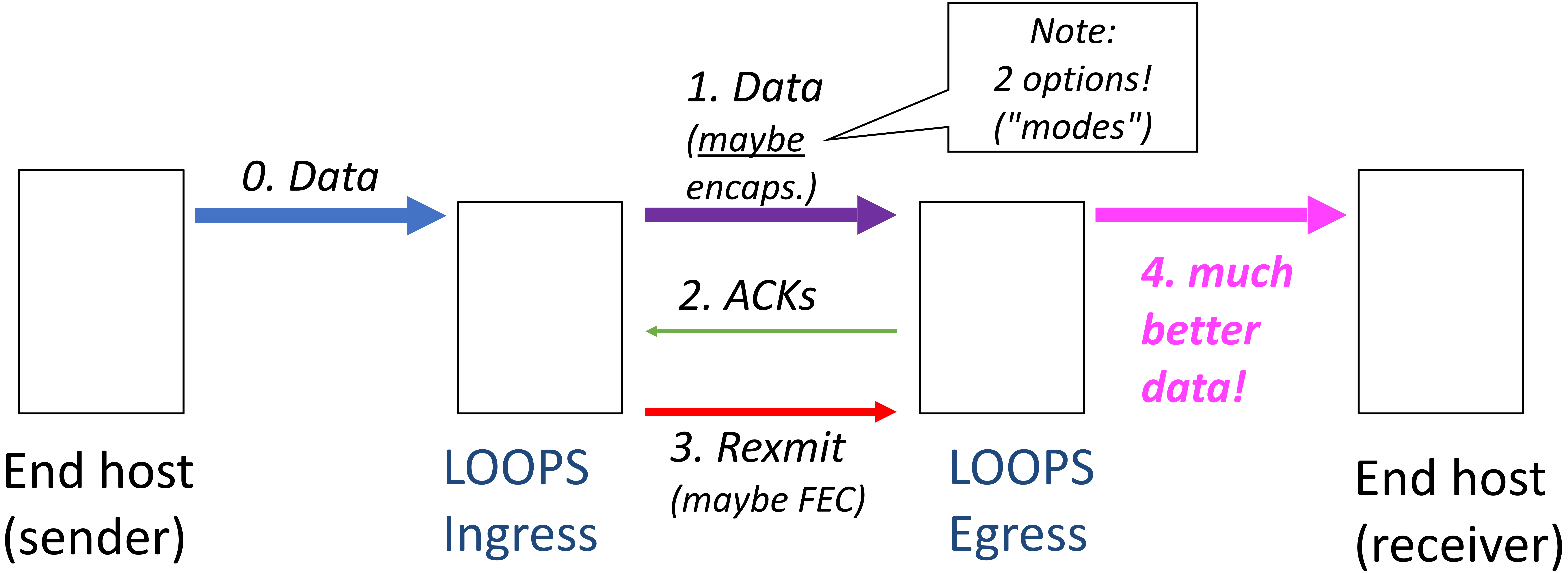
Technical items

(A) ACK management, partial reliability, garbage collection,
“connection” setup [(1)]

(B) “Streams” (forward, reverse, repair), encapsulation principles [(4), (5), (6)]

(C) Measurement, congestion feedback [(3), some (2)]

Context


0. Data
1. Data
(maybe
encaps.)
Note:
2 options!
("modes")
2. ACKs
3. Rexmit
(maybe FEC)
4. much
better
data!
End host
(sender)
LOOPS
Ingress
LOOPS
Egress
End host
(receiver)

Summary: information exchanged

- Forward: encapsulation, containing...
  - Packet Sequence Number (PSN)
  - Tunnel type
  - "ACK desirable" flag (asks for feedback block 1, next slide)
  - Anything needed by FEC

- Backward: optional blocks type 1 (timely) and 2 (eventual)…
  (can be piggybacked, aggregated, interspersed, repeated, ...)
  - Block 1 (optional, only upon "ACK desirable” — timely)
    - PSN being acknowledged
    - Absolute time of reception for the packet acked (PSN)
  - Block 2 (optional — eventual)
    - an ACK bitmap (based on PSN)
    - a delta indicating the end PSN of the bitmap

Refine the streams:

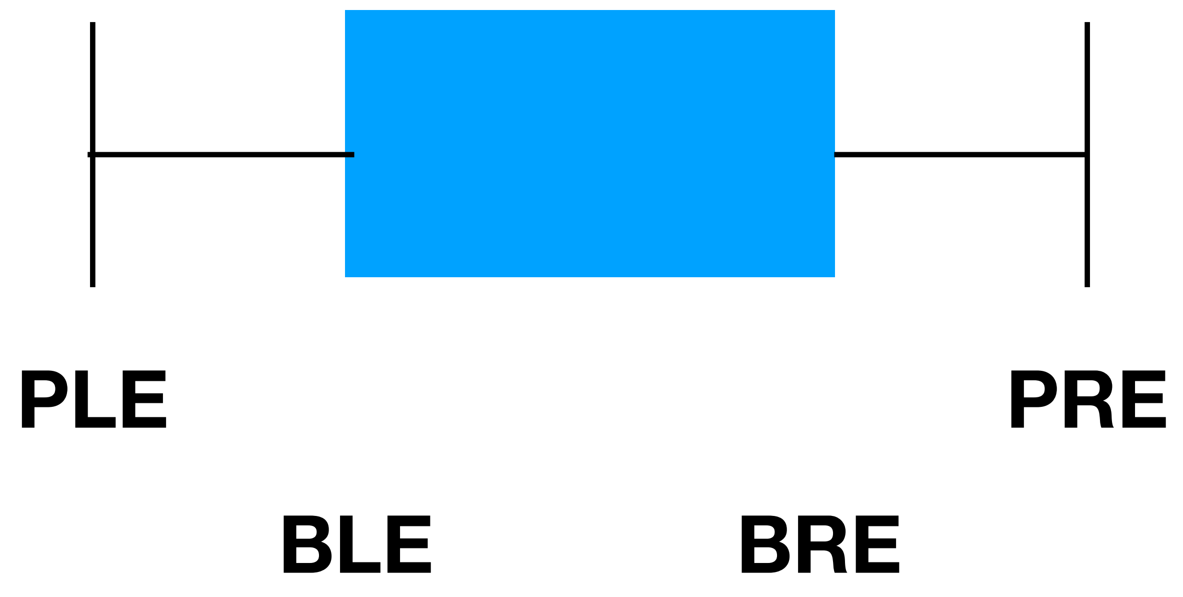
- → Data (fast path, maximize encapsulation fidelity)
  - packet + PSN (can we get away with that minimality?); timestamp? ACK desired?
  - including retransmissions (look the same way)
- → Control/Repair
  - beacons (next slide)
  - FEC info
  - measurement forward
- ← Return
  - ACK/NACK
  - measurement backward

Beacons

- Regular control frame
- Delimits active range of packet sequence numbers (PSN)
- Serves as keep-alive (infrequent, N per second/minute)
- Serves as tail-loss detector (frequent during activity, N per RTT)
- Content:
  - PSN left edge “PLE” (retransmission horizon, for garbage collection of ACK state)
  - PSN right edge “PRE” (most recent PSN; tail loss probe) [== PSN of beacon]
  - Some forward measurement
    - Trigger return (e.g., ACK desired [???no longer on data?])
    - Report (e.g., current RTT estimate/variance)

ACKs (“Block2”)

- Each ACK covers subrange within receiver’s view of PLE..PRE
- Bitfield (from Bitfield left edge to Bitfield right edge)
- No statement is made about range outside subrange
- ?? Optimization: all-ones range left or right of bitfield



BLE, BRE, PRE Delta-encoded

- No NACK format: A holey ACK does the job
  - But NACKs may be triggered differently from ACKs
- ACK trigger: every RTT/N while active
  - beacon provides minimal rhythm

Acks and retransmission

- LOOPS defines a protocol: What information do the two ends make available to each other when
- Retransmission decision can be local matter, if protocol supplies needed information
  - Traditional DUPACK-style: Any (small number of) packets that are ahead of the current order indicate a loss
  - Time-based: tolerate reordering; use more information than just most recent ACKs
- **Can the *protocol* work with both *implementation* strategies?**

Encapsulation

- LOOPS focuses on the “generic information” interchanged
- Need to map to specific encapsulation protocols.
  These have:
  - Required fields (that may map to LOOPS information)
  - Existing options (that may map to LOOPS information)
  - Extension points (TLVs, flag-based)
- **Observation: Lots of commonality**
  - Factor out and move into generic information document?

Beacon encapsulation

- Make sure beacon travels with data
  - Encapsulate as data (say, UDP), directed to egress itself?
    - Consumes a PSN, ackable!
    - No confusion which beacon is most recent
    - Replaces setup exchange (ACK of Beacon PSN = SYN-ACK)
  - (Discuss interaction with control plane protection)

- Beacon structure is LOOPS-specific; no interaction with encapsulation

Return encapsulation

- Two cases:
  - Standalone
    - Carry as simple UDP-based protocol from egress to ingress
  - Piggybacked in/on data of a reverse channel
    - Carry in option defined by encapsulation scheme

Beacons revisited

- Idea: simplify protocol, create natural “rhythm”
- Regular communication of ingress’s idea of RTT estimates
- Enables garbage regular collection at egress (PLE)
- Automatic tail loss probe (beacon consumes PSN and is ackable)
- Keepalive (also replaces SYN), triggers the return keepalive (as ACK)
  - Exercises encapsulation; quick health signal

- ~ Rate in the single-digit percents of data traffic
- ~ Period exponentially doubles during idle times up to say 30 s

Carsten’s answers to Yizhou’s input/questions

- (1) Should we simply assume that reordering in the LOOPS segment is unlikely, so we only have to specify the DUPACK style of loss detection within LOOPS?
  ➔ enable RACK style **and** DUPACK style
- (2) What can we find out about the end-to-end transports contributing to a LOOPS-enhanced flow (e.g., are they RACK—based?), so we can adapt behavior appropriately? (This assumes some minority traffic might be negatively impacted. It also violates the “don’t peek” principle.)
  ➔ That is maybe material for a second step?
- (3) Re congestion control: LOOPS ingress does not do sending rate adjustment. The control is at the end host sender. LOOPS tries to do the minimum set of work in participating in the congestion control, e.g., via CE-marking. ➔ Yes

Carsten’s answers to Yizhou’s input/questions

- (4) IPPM-defined ioam format intends to help network oam. It puts a lot of effort in its trace option (pre-allocated trace and incremental trace). Trace option is not appropriate for LOOPS since LOOPS would work on tunnel endpoints. No intermediate node’s information is required. If ioam edge-to-edge mode is to be used, then ioam edges have to happen to be LOOPS segment ingress and egress. ➔ Yes
- (5) Timestamp formats are one thing we can copy from IPPM (for where ingress and egress agree about the format). Any use for data formats defined for IPFIX, and/or measurement protocols such as OWAMP, TWAMP? ➔ TO DO
- (6) Examine GRE as another encapsulation format ➔ Yes

Agenda

- 1504Z: Quick overview (10)
- 1514Z: Technical discussion about specific items (50)
- **1604Z: Quick recap (10)**
- 1614Z: Relationship to other WGs and RGs (40)
- 1654Z: Wrap-up, next meeting

Agenda

- 1504Z: Quick overview (10)
- 1514Z: Technical discussion about specific items (50)
- 1604Z: Quick recap (10)
- **1614Z: Relationship to other WGs and RGs (40)**
- 1654Z: Wrap-up, next meeting

Charter text on other WGs/RGs

- LOOPS will work closely with developments in TSVWG, TCPM, and in particular with QUIC as an example of a transport protocol that may more readily absorb features of interaction with LOOPS segments. However, there is no dependency — LOOPS is designed to optimize already in the presence of legacy transport protocols. LOOPS will interact with the homes of tunneling protocols to which bindings are being defined, which depending on the choices of the WG may include NVO3 (Geneve), intarea (GUE), spring/ 6man (SRv6).

- (No other **SDOs** mentioned.)

- Refine this (see following slides); derive updated charter text.

https://github.com/loops-wg/charter/blob/master/wg-rg-relations.md

- "watch": shared interests, but no specific exchange or mutual dependency envisioned now; could develop, though.
- "import": specs that the other WG works on will be used by ours.
- "export": we could do some work that would benefit that other WG.

Transport protocol innovations

- Name: TCPM, TSVWG, QUIC
- Issue in other group: What new transport protocol mechanisms provide benefits that cannot be had with a classical transport
- Issue in LOOPS: Develop the retransmission part of LOOPS (as well as possibly the reconstruction/FEC part) based on these innovations
- Type: Watch, potentially Import
- Issue in LOOPS: Are there new interactions between the LOOPS "transport" elements and the innovations in the upper layer transport protocols?
- Type: Watch

RACK

- Name: TCPM

- Issue in other group: (from draft-ietf-tcpm-rack-08:) ... develop a new TCP loss detection algorithm called RACK ("Recent ACKnowledgment"). RACK uses the notion of time, instead of packet or sequence counts, to detect losses, for modern TCP implementations that can support per-packet timestamps and the selective acknowledgment (SACK) option. It is intended to be an alternative to the DUPACK threshold approach [RFC6675] ...

- Issue in LOOPS: Does LOOPS need to rely on a sequence-preserving path for efficient loss detection, or can we use some of the RACK work for more time-based loss detection? How do we translate the TCP features used by RACK into features that are available in LOOPS?

- Type: Watch, potentially Import

Embedding proxied flows into QUIC and HTTP/3

- Name: MASQUE (WG to be, recent BOF 2020-03-24)

- Issue in other group: "Congestion Control over Congestion Control in case of UDP or IP proxying", i.e., when the proxied flows are UDP or IP (~ VPN), the interaction between end-to-end transport protocol control loops (congestion control) and the control loops run by QUIC

- Issue in LOOPS: Does LOOPS need to perform specific work to mitigate issues stemming from nested congestion control? Note that the LOOPS segment is typically, but not always, significantly shorter than the end-to-end path, which might be similar in a MASQUE situation.

- Type: Watch

Congestion control for media flows

- Name: RMCAT
- Issue in LOOPS: Interactions between upper layer congestion control schemes in use for media and those in use on the LOOPS path segment
- Type: Watch

Formats and algorithms for measurement

- Name: IPPM, TSVWG
- Issue in LOOPS: What IPPM-defined formats and algorithms for measurement can we usefully import into the LOOPS measurement system
- Type: potentially Import
- To do: What came out of the draft-ietf-tsvwg-tunnel-congestion-feedback work?

Tunnel encapsulation formats

- Name: NVO3, INTAREA, SPRING/6MAN
- Issue in LOOPS: We need to follow the homes of tunneling protocols to which bindings are being defined, which depending on the choices of the WG may include NVO3 (Geneve), intarea (GUE), spring/6man (SRv6).
- Type: Import
- To do: There is no obvious home for GRE at this point, but GRE is another obvious choice for a tunneling protocol.

Advanced ECN

- Name: TSVWG

- Issue in other group: Agree on an approach for advanced ECN

- Issue in LOOPS: LOOPS can benefit from ECN both in the end-to-end traffic and in the LOOPS segment. The protocol may need to anticipate the developments on advanced ECN and adjust its usage of both classic and advanced ECN. This may also influence how LOOPS makes use of RFC 6040 (Tunnelling of Explicit Congestion Notification).

- Type: Watch

DETNET data plane, RAW, PREOF

- Name: DETNET, RAW
- Issue in other group: Define a data plane (encapsulation) for replicating, eliminating duplicates, and potentially reordering traffic (PREOF). RAW extends this work to wireless networks.
- Issue in LOOPS: Can we integrate some of this work for retransmission and reconstruction?
- Type: Watch, potentially import
- Issue in LOOPS: Potentially (long term): Can we make some of the LOOPS components available in a way that is useful for DETNET and RAW?
- Type: Export

FEC formats and protocols

- Name: NWCRG (IRTF), TSVWG
- Issue in other group: Define forward error correction (FEC) formats that provide good robustness, low reconstruction latency, low implementation complexity, low CPU requirements, low overhead
- Issue in LOOPS: How does the structure of these new formats and protocols influence the way LOOPS handles reconstruction?
- To do: Watch the patent claim thickets in this space
- Type: Watch, Import/Reference (specific schemes)

Measurement, Congestion control

- Name: MAPRG (IRTF), ICCRG (IRTF)
- Issue in other group: Do leading edge work in this space
- Issue in LOOPS: Benefit from that leading edge work
- Type: Watch

PANRG?

- New item proposed in: https://github.com/loops-wg/charter/issues/2
- _______?

Relationship to other SDOs?

- None identified so far
- _______?

Next Steps

While we are not a WG…

- Continue on, *working* like a WG
  - Explore design space, maybe holding back on tough decisions for now
- Continue improving the set of documents, possibly adding FEC document
  - Identify authors and reviewers
- Employ github.com/loops-wg and loops@ietf.org for coordination
- Next meeting: LOOPS side meeting, 2020-05-26, 1500Z–1700Z
  - Continue and confirm results of technical discussion
  - Update charter proposal based on technical discussion and AD input on this