Rapid Synch for RTP Multicast Sessions

draft-versteeg-avt-rapid-synchronization-for-rtp-02

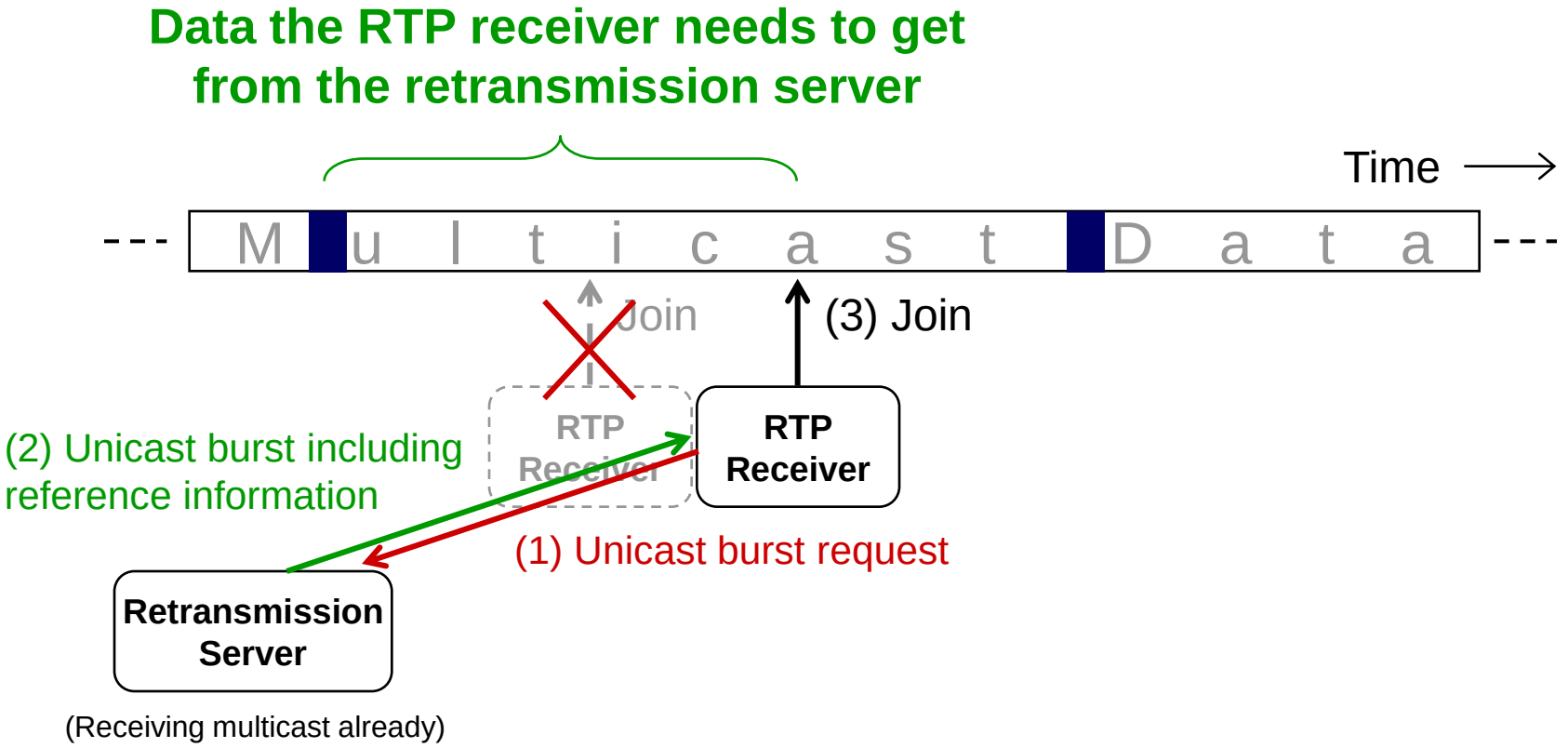
IETF 74 – March 2009

Bill Ver Steeg, Ali C. Begen, Tom Van Caenegem and Zeev Vax

{billvs, abegen}@cisco.com, Tom.Van_Caenegem@alcatel-lucent.be, zeevvax@microsoft.com

Recap: Proposed Approach

- Prior to join, receiver requests a unicast burst from a server caching the recent data



Summary

- RTP receiver says to the retransmission server:

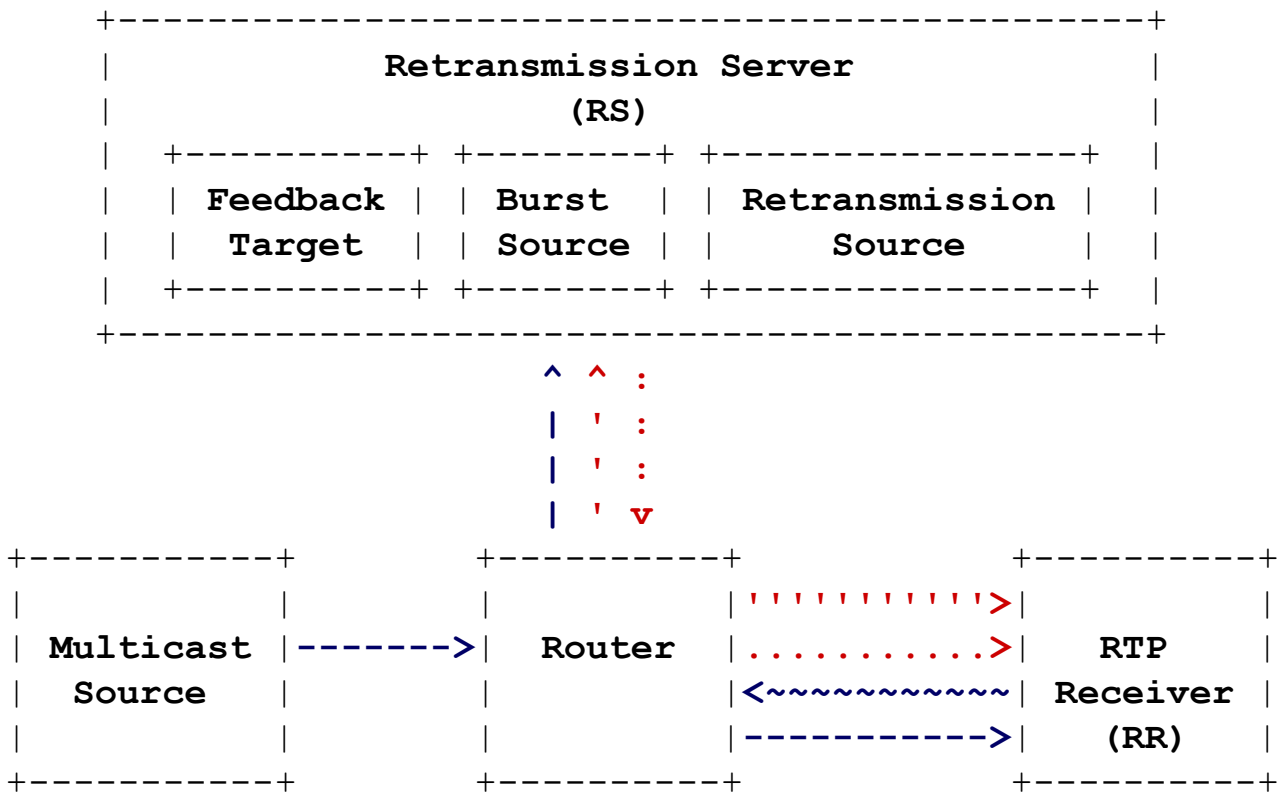
  "I have no synch with the stream. Send me a repair burst that will get me on the track with the multicast session"

- Differences compared to conventional retransmission:
  - Receiver does not know exactly what it is missing
  - Retransmission server
    - May need to parse data from earlier in the stream than it is needed for retransmission (Reference information may be dispersed)
    - May need to burst faster than real time
- We define a method that enables a joining receiver to acquire and process a multicast flow quickly
- The method is applicable to any RTP-encapsulated multicast flow

Changes since Version -01

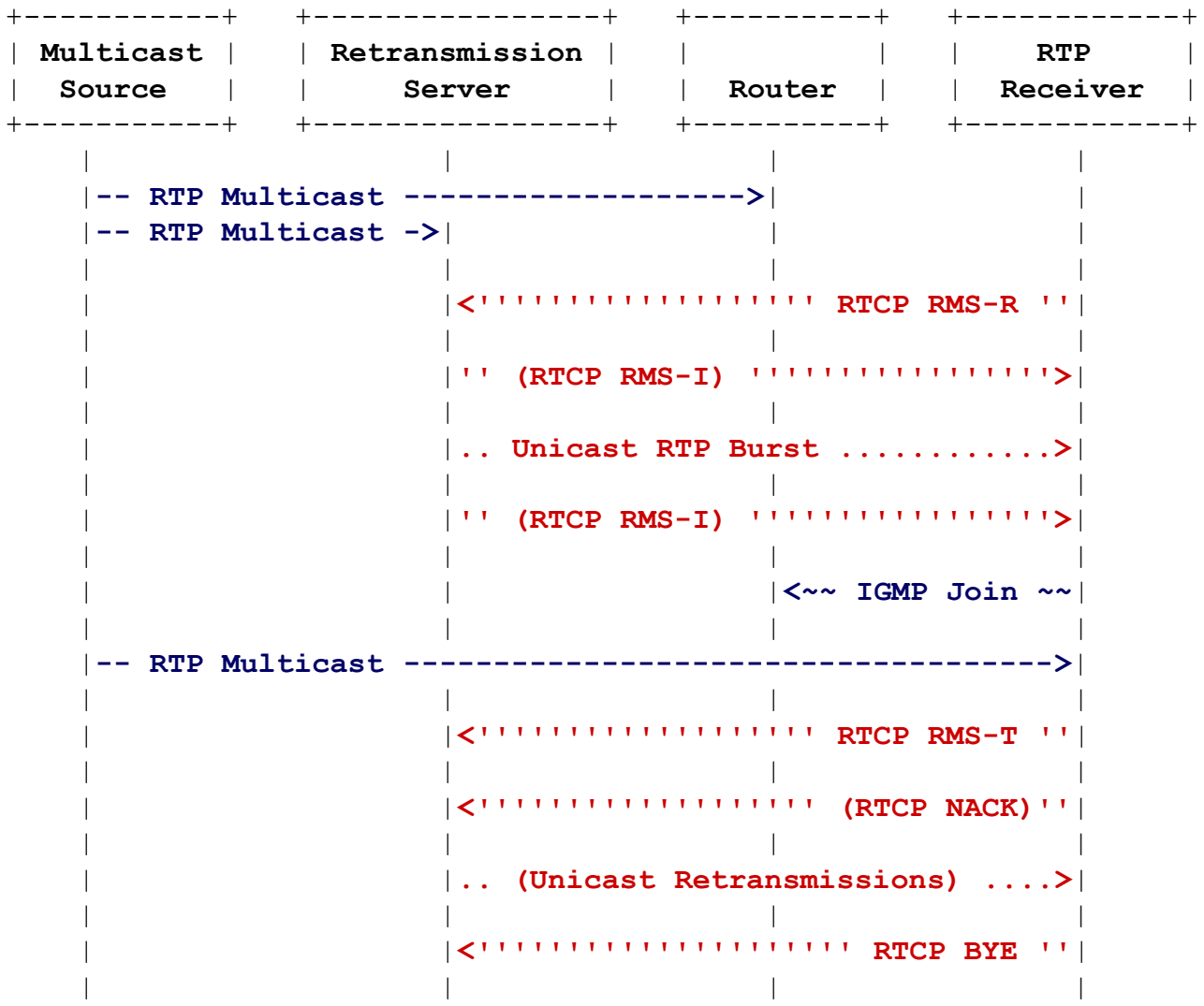
- Two drafts have been combined together:
  - draft-versteeg-avt-rapid-synchronization-for-rtp-01
  - draft-levin-avt-rtcp-burst-00
- A new section on protocol design considerations has been added
- The draft is no more MPEG2-TS specific
- The video-specific discussions have been moved to:
  - draft-begen-avt-rtp-mpeg2ts-preamble-00
- The RMS-R, RMS-I and RMS-T messages have been modified
- The discussion of RTCP XR report has been moved to:
  - draft-begen-avt-rapid-sync-rtcp-xr-00

Rapid Synchronization



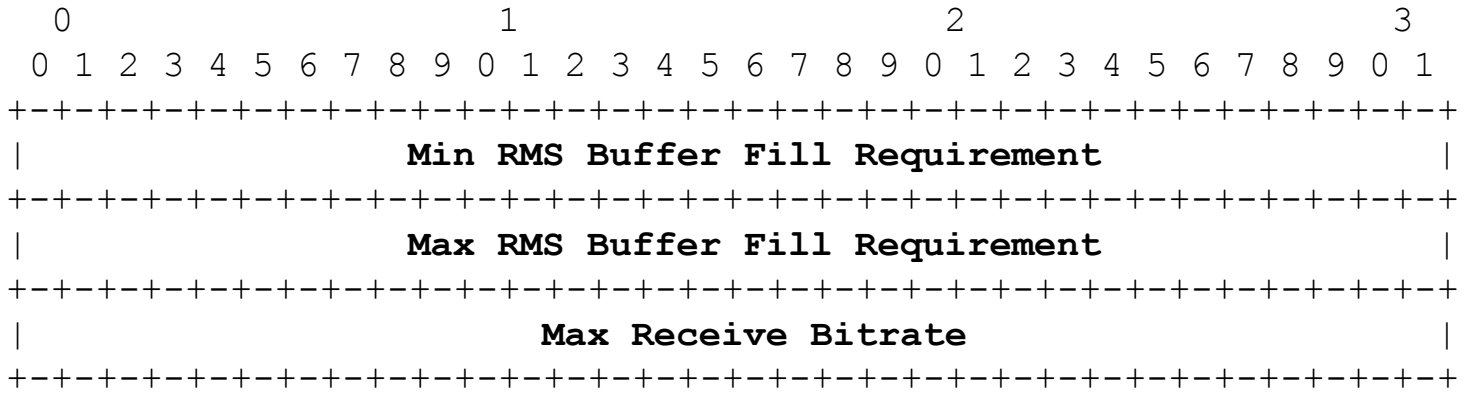
'''> Unicast RTCP Messages
~~~> IGMP Messages
...> Unicast RTP Flow
---> Multicast RTP Flow

Open Issue:
Any need to explicitly address other topologies where FT, Burst and Retransmission Sources are not co-located?
The Feedback:
Such topologies are for further study
~~~

Rapid Synchronization



RMS Request (RR → RS)

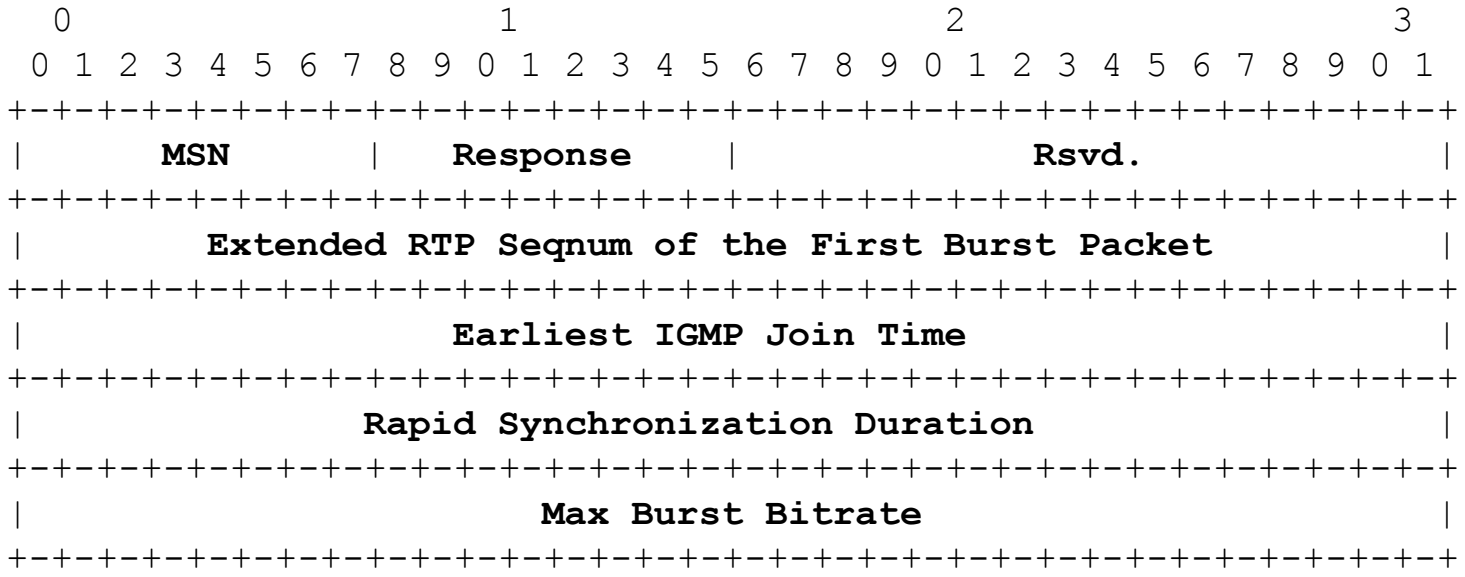
(Payload-Independent) Transport-Layer Feedback (PT=RTPFB, FMT=5)



- Sending one RMS-R is required prior to RMS
  - Min RMS Buffer Fill Req: RR's min data req (in ms) from the burst
    - A zero value means it is not specified
  - Max RMS Buffer Fill Req: Max data (in ms) RR can accept from the burst
    - A zero value means it is not specified
  - Max Receive Bitrate: Maximum bitrate (in bps) that RR can receive
    - A zero value means it is not specified

RMS Information (RS → RR)

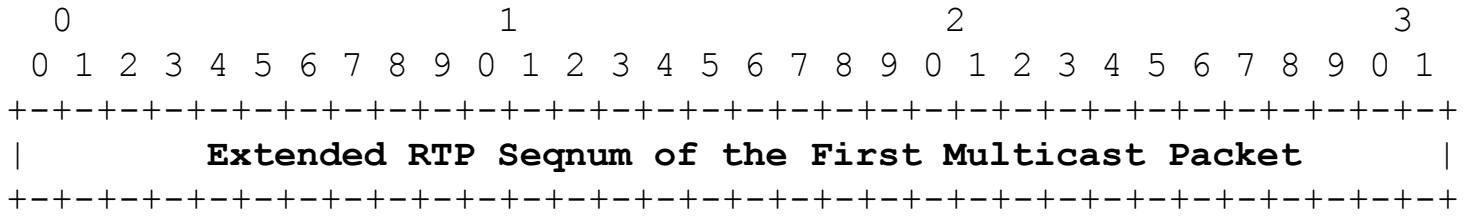
(Payload-Independent) Transport-Layer Feedback (PT=RTPFB, FMT=6)



- Sending one RMS-I is required before or during the burst
  - Response may indicate whether RMS request has been accepted or not
  - Response may be used to signal RR to join immediately or at an indicated time
- Further RMS-I messages may be sent to update any information
  - MSN indicates message seqnum (useful to identify reordered messages)

RMS Termination (RR → RS)

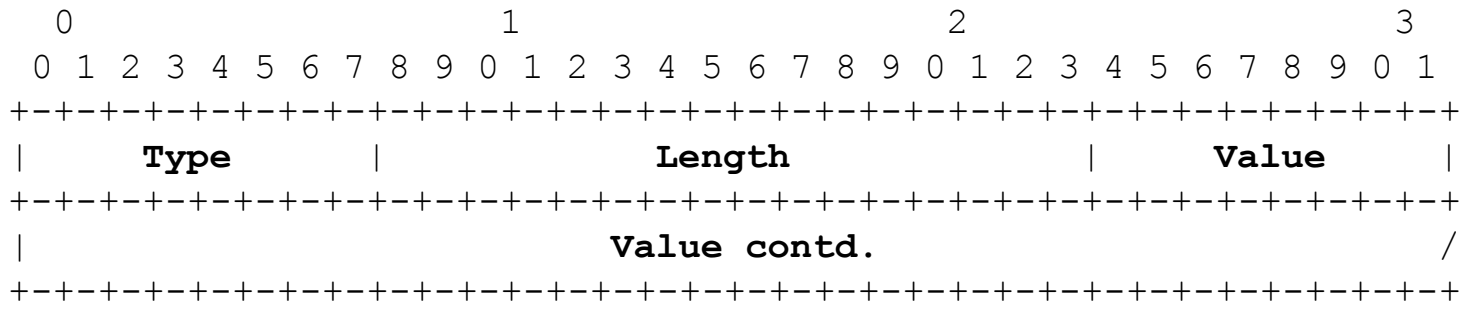
(Payload-Independent) Transport-Layer Feedback (PT=RTPFB, FMT=7)



- Although RS may end the burst proactively, sending RMS-T at least once is required
- If RR has not joined the multicast session or has not started receiving multicast packets
  - RR sends an empty RMS-T message (w/o an RTP seqnum)
  - RS must stop the burst upon receipt
- If RR has started receiving multicast packets
  - RR sends an RMS-T message with the RTP seqnum of the first multicast packet
  - RS should continue bursting until the reported seqnum
- RS may continue bursting if RMS-T message gets lost
  - RMS-T messages may be repeated (by following the rules of RFC 4585)
  - RS should eventually end the burst at some point (e.g., after a timeout)
- If RR needs to cancel an active/pending unicast session, RR sends a BYE

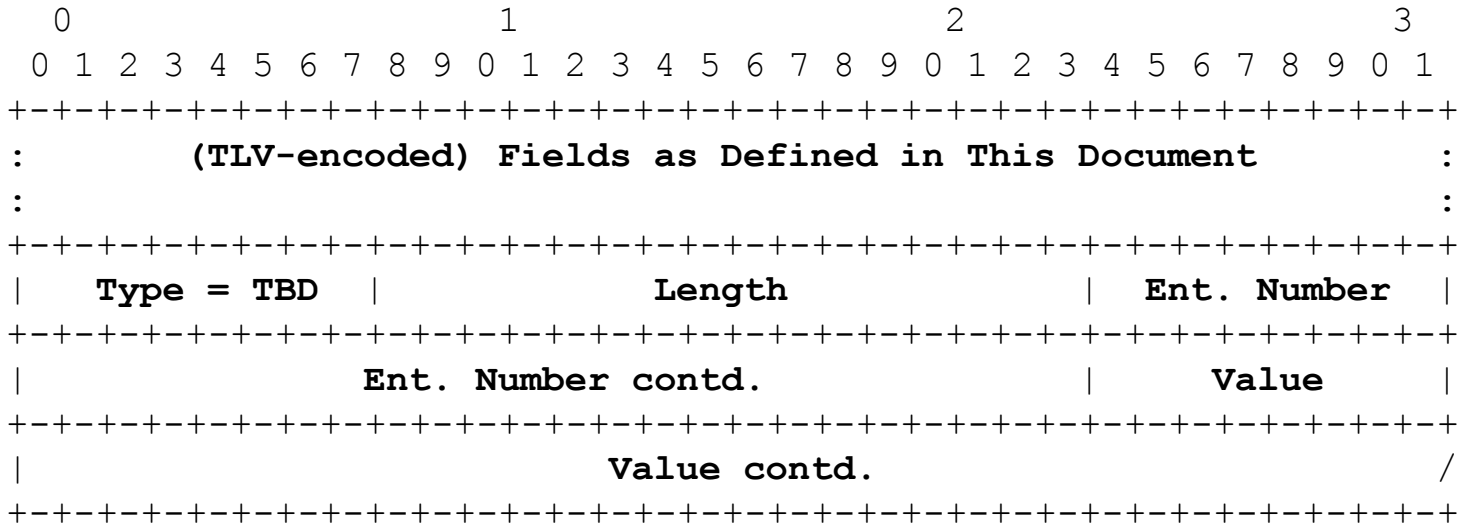
TLV Encoding in Control Plane

- Almost all fields in RMS control messages are optional
  - Not every implementation needs every field
  - An implementation may not need every field all the time
- Then, why not make every field TLV encoded?
  - Unused fields will not be encoded
    - Saves bandwidth
  - Eliminates the need for special values and ambiguity



Extensions for Control Plane

- New TLV elements may be defined later
  - These extend the protocol in a vendor-neutral manner
  - These should be accompanied by informational RFCs
- Vendors may need vendor-specific extensions
  - For interoperability, such extensions MUST NOT collide
  - Use numbers from http://www.iana.org/assignments/enterprise-numbers



Known Implementations

- **Open Source RTP Receiver Implementation by Cisco**

  Documentation:

  http://www.cisco.com/en/US/docs/video/cds/cda/vqe/3_0/user/guide/ch1_over.html

  FTP Access:

  ftp://ftpeng.cisco.com/ftp/vqec/

  Preliminary Results:

  See the references

- **IPTV Commercial Implementation by Microsoft**

  Information:

  http://www.microsoft.com/mediaroom

  http://informitv.com/articles/2008/10/13/channelchangetimes/

Other Open Issues

- Name confusion with draft-perkins-avt-rapid-rtp-sync
  - We propose to update our title as:

    Unicast-Based Rapid Acquisition of Multicast RTP Sessions
  - Any other proposals?
- Collision in FMT numbering space
- Using extended RTP seqnums in RMS-I and RMS-T
- Discussion of burst shaping, NAT and security issues

Next Steps

- Shall we add a milestone to AVT's charter to produce an RFC on rapid acquisition of multicast RTP sessions?
- Breakout Session:
  - Tomorrow at 9am (till 10:30am)
  - Room: Yosemite A
  - Bridge: See AVT mailing list