AV Bridge Summary: an early view

Michael Johas Teener

Mikejt@broadcom.com

Preface and warnings

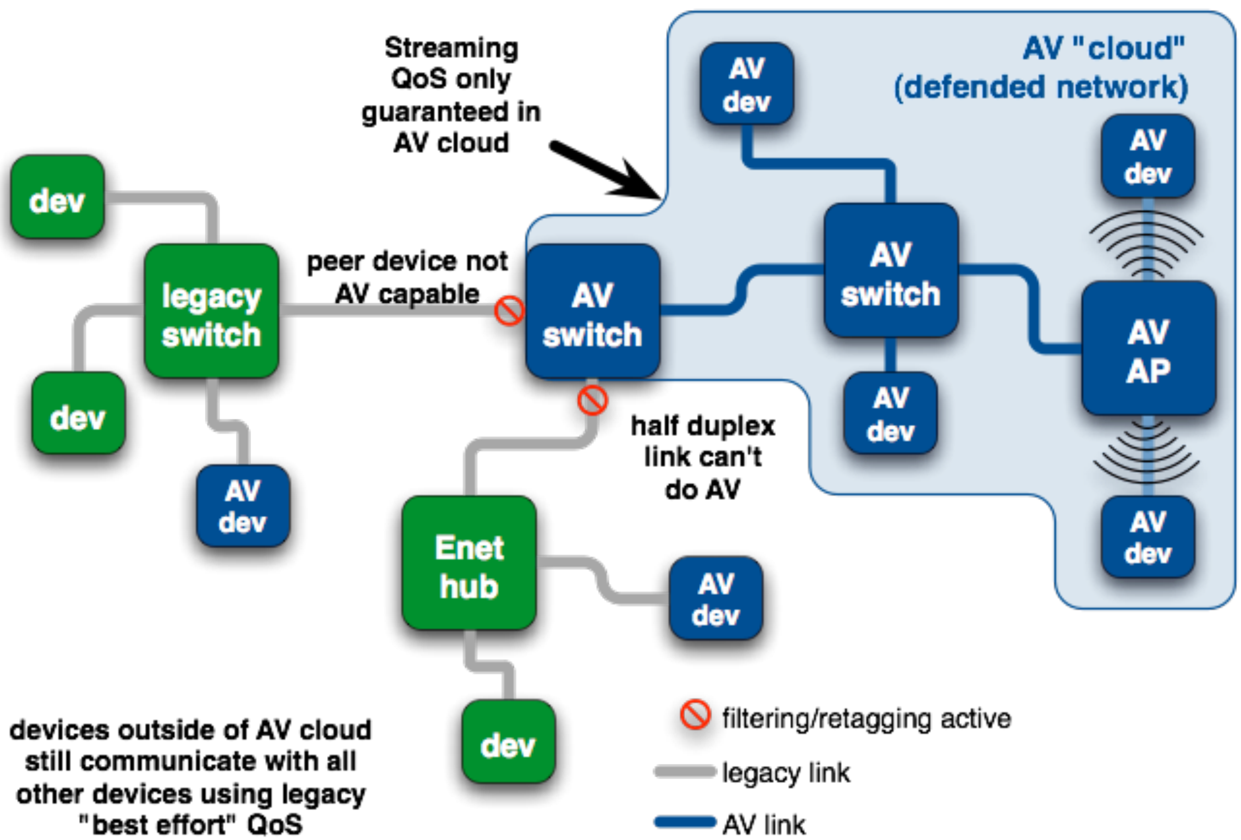
- Outline of bridge (and DTE) operation
  - To be used as the very start of a “Ethernet AV” recommended practice
    - (there needs to be an “802 AV” recommended practice that includes .11/.15/.16/.etc … but that is later work)
- Personal point of view
  - No one else has reviewed this
- Very preliminary!

Contents

- Structure of an 802 AV network
- 802-only time synchronization
- Reservation protocol
  - What is being reserved?
- AV stream frames
- Setting up a defended network
- Traffic shaping

Structure of an 802 AV network

- Directly connected participating devices … any intervening non-participating device defines a boundary to AV QOS



802-only time synchronization

- Assume something between a 1588 profile and a timing domain that appears to be a 1588 subdomain
  - Bridging between 1588-2002 and 802.1(time-synch) is very simple
- Bridges (and 802.1 layer of endpoints) have “pretty good” idea of time, and very accurate measurement of delay to attached peers
  - “pretty good” means within a microsecond or so
- Specification TBD, but all current proposals have adequate performance
  - Cost/complexity/interoperability/robustness may be best ways to choose a method
  - Assume unique Ethertype with packet type to uniquely identify packets that must have timestamps saved
- Separate PAR (perhaps 802.1at?)

Reservation protocol

- Assume something similar to "SRP" as described by Felix Feng
- Reserves bandwidth at a defined class of service for a specific stream
  - Bandwidth is defined as "bytes per class interval"
  - Class of service is "highly interactive" (with 125us class interval) or "normal interactive" (with 1ms class interval)
  - Stream is identified by destination address (multicast)

AV stream frames

- 802.1Q tagged frames with PCP 5 for highly interactive and PCP 4 for normal interactive
- How is multicast address chosen?
  - Should this be an 802 problem?
  - If so, default 802 operation could be same as “auto IP”: pick an address, probe using it to see if anyone responds … repeat for “n” times

Setting up a defended network

- Need to prevent interference from non-participating traffic
- If all bridges were managed ones (participated in spanning tree), we could do this with existing protocols
  - Oops … unmanaged bridges are *the most likely* type to find in the home
- So … since we require time synch on AV networks, we can use that to determine if an unmanaged bridge is attached to a port
  - If cable delay between peers is “unreasonable” (>> 100 m), we can assume that the link cannot be used for AV streaming
- If cable delay is OK, and peer responds to enhanced link discovery with correct attributes, then peer can be assumed to be a part of the AV cloud

Traffic shaping

- Source device must do traffic shaping to match reservation (launch no more than x bytes per class interval)
  - Right? No arguments?
- Advantages for shaping traffic at bridges
  - Allows better best-effort performance
    - Nothing to be gained if stream data arrives early … all streaming applications are built around worst case latency and provide appropriate buffering
    - Delaying “early” stream data allows best effort traffic to use unneeded transmission opportunities … best effort QoS is typically improved with reduced latency
  - Network scalability
    - Buffers within bridges and endpoints can have fixed “small” size regardless of network topology and never drop packets with streaming QoS
- Two methods: transparent and explicit

Transparent traffic shaping

- SRP results in bridges knowing the maximum traffic that can be transmitted and received on each port
- Time synch protocol guarantees that bridges agree on common time reference (+/- some small delta … e.g., 1us)
- These two pieces of information allow a bridge to make a good guess whether a particular frame should be transmitted during a particular class interval
- Done “transparently” … i.e., bridges do not communicate class interval boundary information
  - Implementation/specification a bit subtle … algorithm correctness TBD
  - Delivery jitter greatly reduced, but bounds increase with number of hops

Explicit traffic shaping

- Each device transmits a “start of interval” frame between all frames sent within a particular class interval
- Receiving bridge internally labels frames with class interval and uses for scheduling transmission in a following class interval
- Implementation/specification is very simple … algorithm correctness easy to prove, but rather unusual behavior for a bridge
- Delivery jitter bounded to no more that 2x class interval regardless of network topology