Clock synchronization

(a Residential Ethernet SG presentation)

David V James	JGG
Alexei Beliaev	Gibson
George Claseman	Micrel
Geoff Garner	Samsung

Categories of work

- Service discovery (out of scope)
  - Identify/control “talkers” and their available “plugs”
- Subscription (802.1 centric)
  - Establish conversation between talker and listener(s)
  - Reject unless: *linkBandwidth < linkCapacity*
- *Clock synchronization*
  - *Synchronous reception, forwarding, and presentation*
- Prioritized queues
  - Talkers and 100Mb bridge ports must be gated
- Formats
  - Frame formats and content (stream IDs, time stamps)
  - Time aware service interfaces

Overview

- What?
  - The clock slaves time-of-day tracks the grand master
  - No requirement for slaves to be clock-synchronous
- How?
  - Periodic exchanges of small messages
- Why?
  - Bridges: synchronized 125us cycles
  - Applications: accurate presentation times

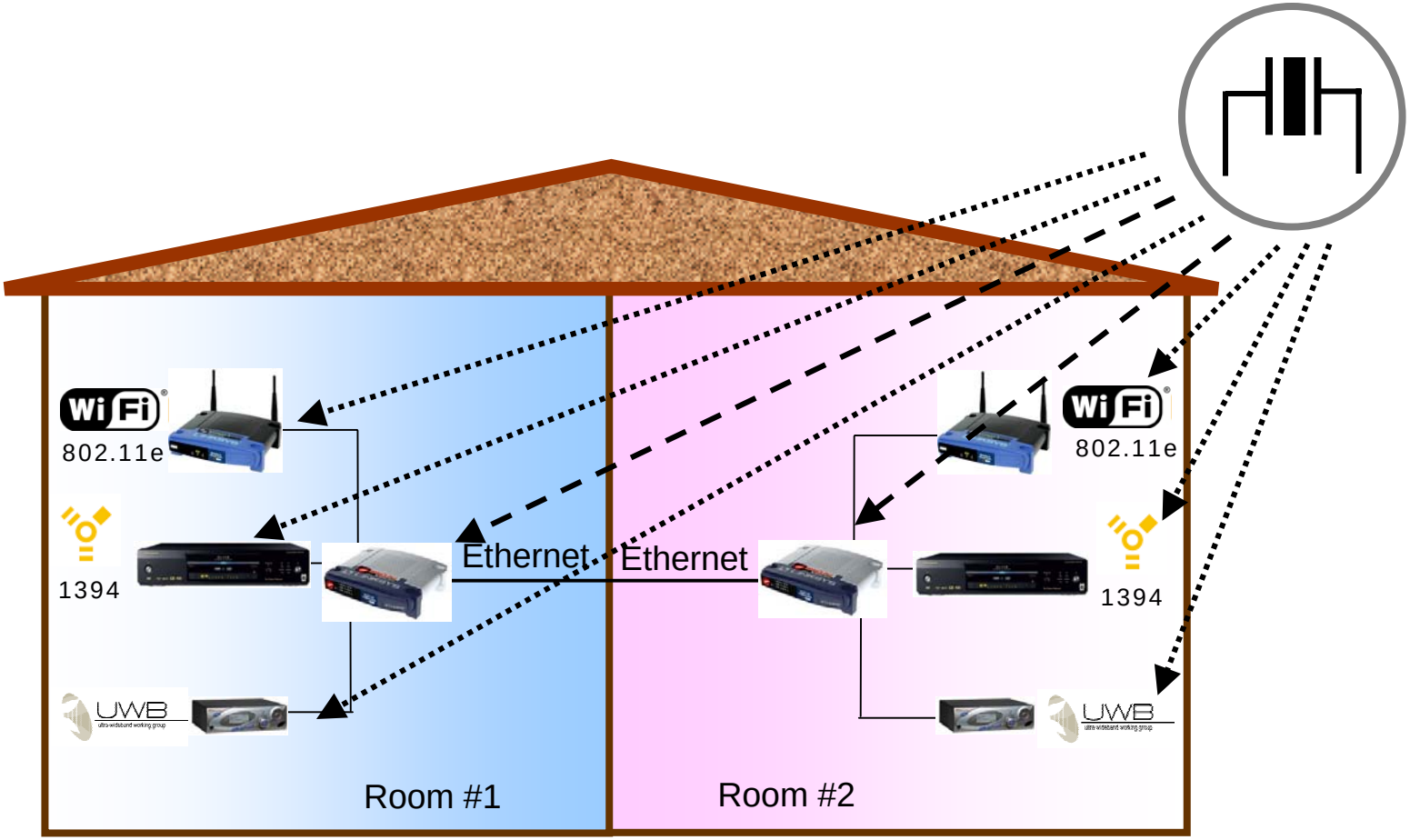
Leveraged protocols

- Spanning tree protocol (STP)
  - Defines the grand-master precedence format
  - But, we use a distinct value and distribution protocol (The STP root and grand master could be distinct!)
- NTP (RFC-1305) and SNTP (RFC-2030)
  - Definition of the 64-bit time-of-day value
- IEEE 1588-2002
  - Techniques for delayed-sampling synchronization

Clock synchronization

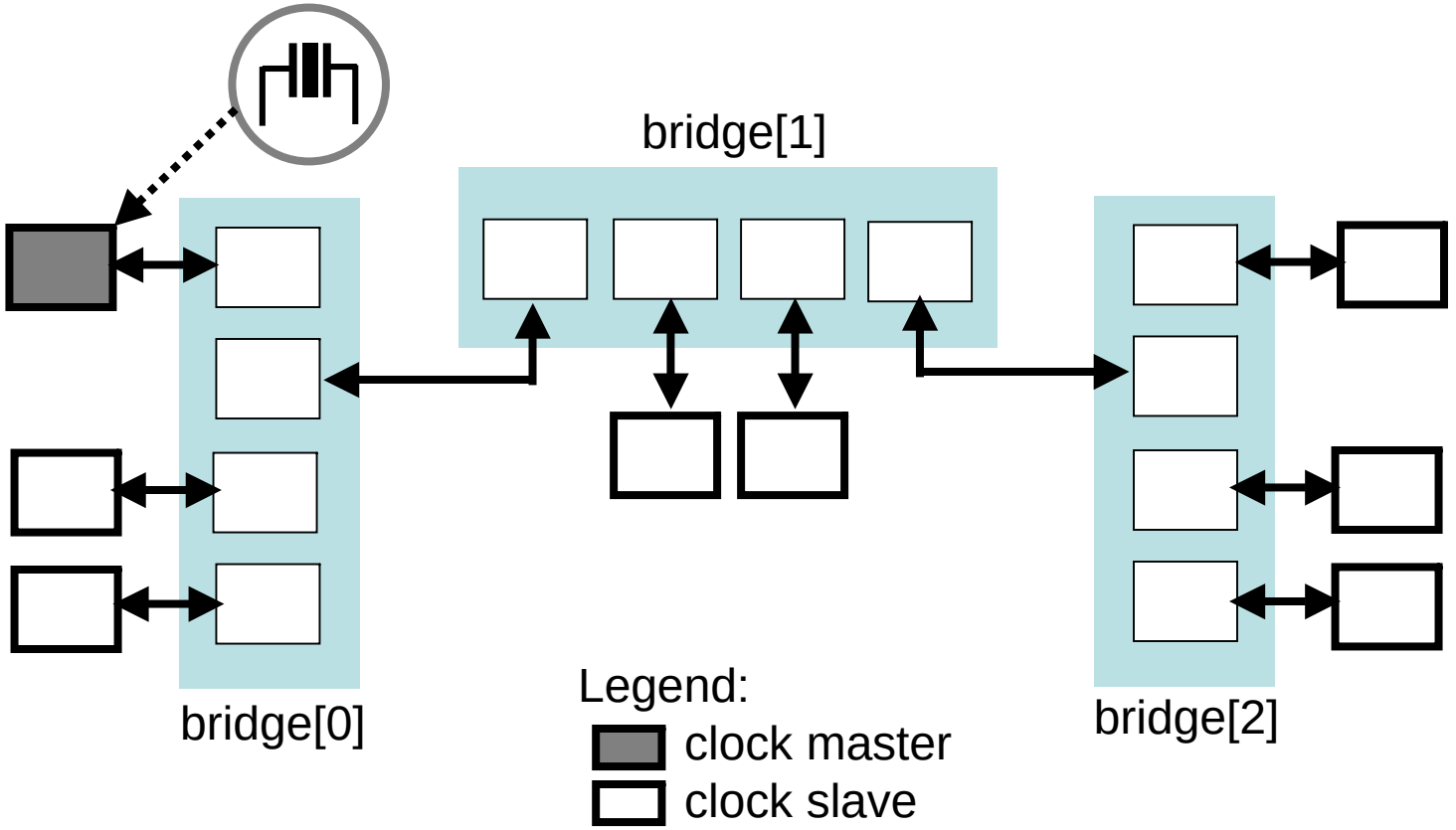
What?

House reference clock



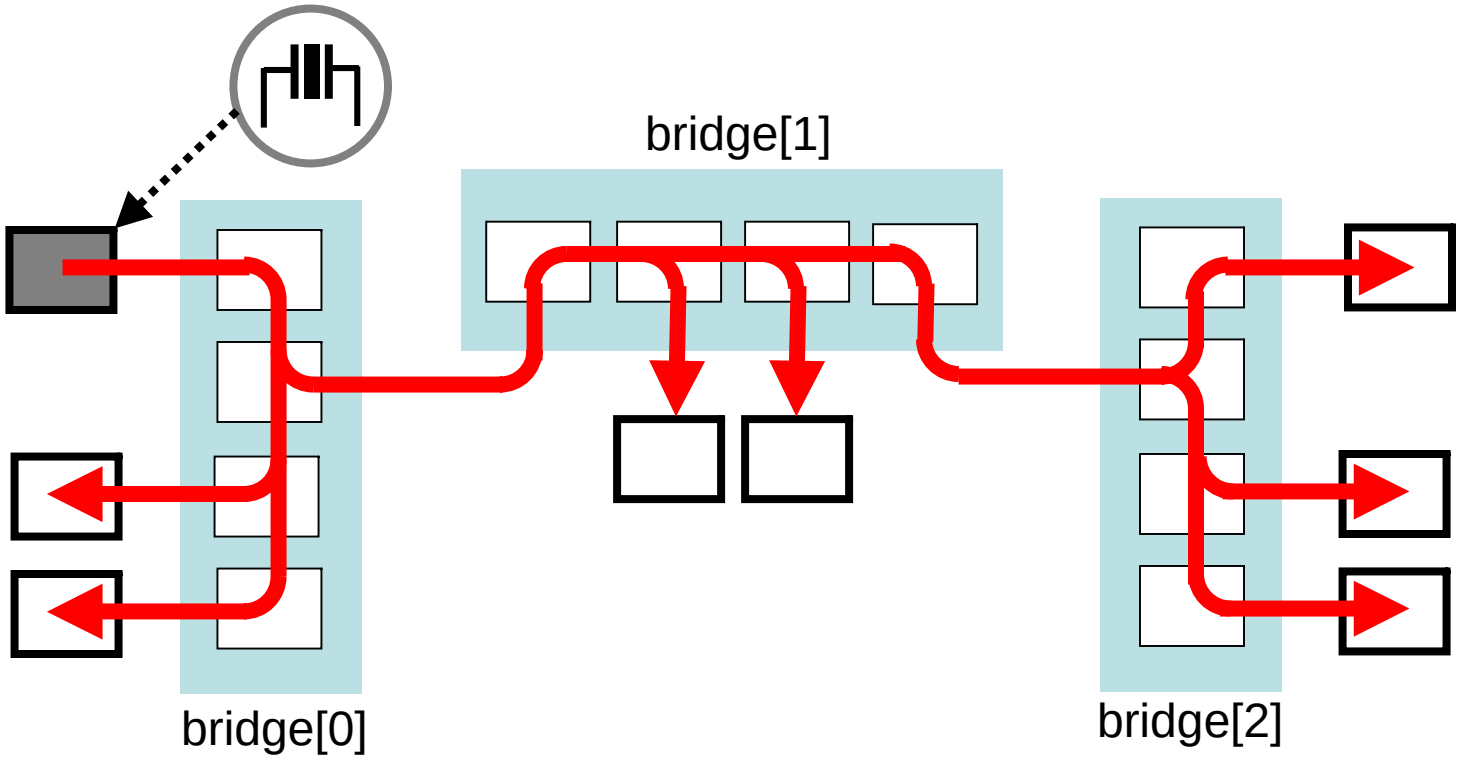
Cascaded TOD synchronization

Physical topology constraints



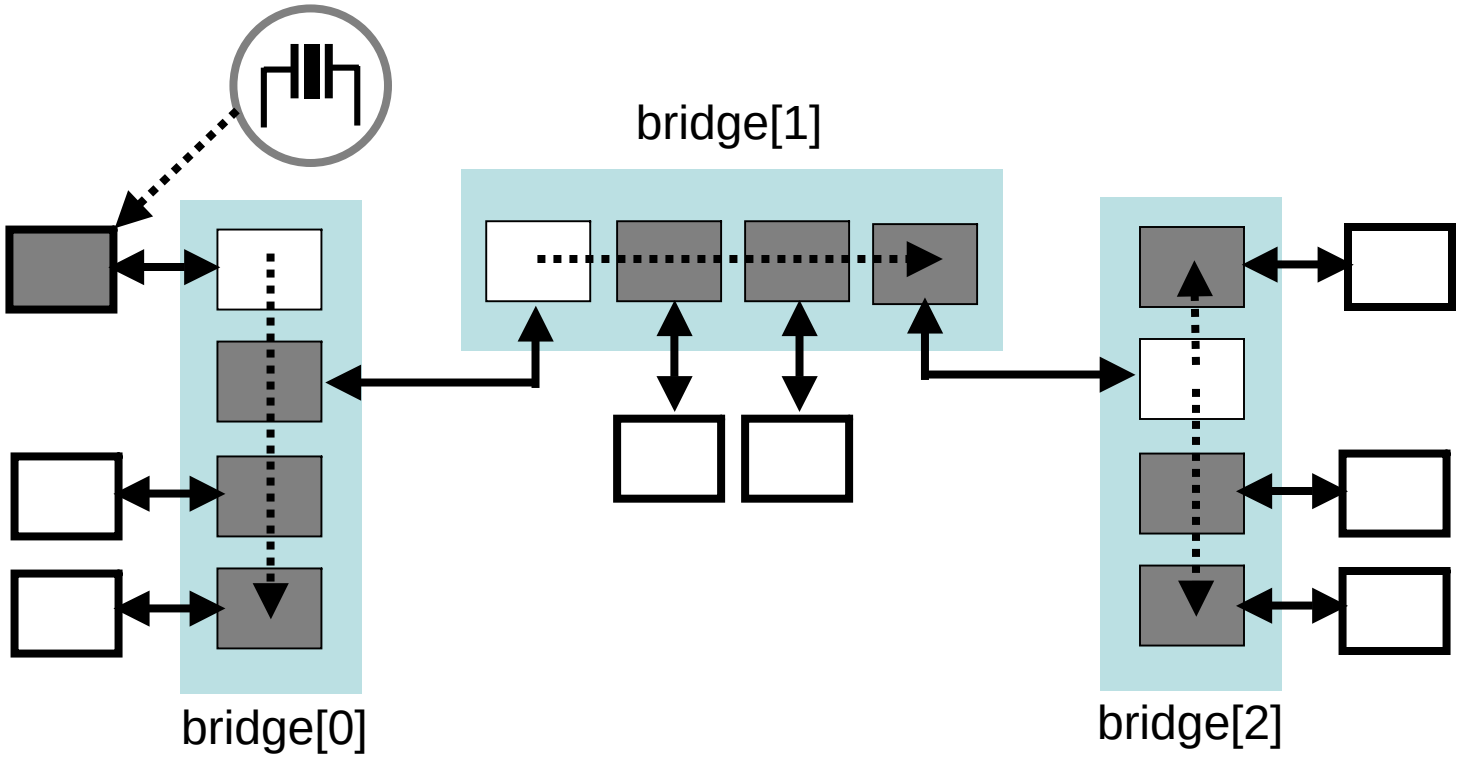
Cascaded TOD synchronization

Wall-clock distribution model



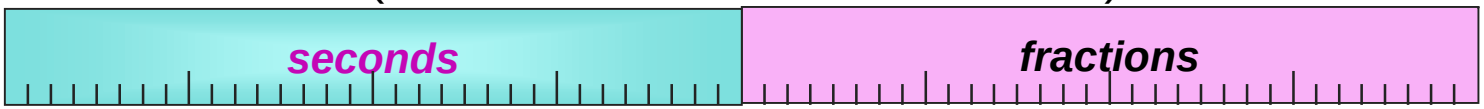
Cascaded TOD synchronization

Cascaded adjacent-synchronization hierarchy



Time-of-day format options

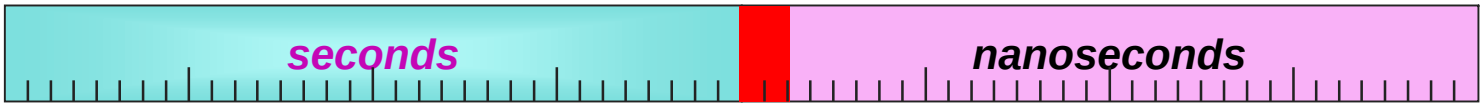
(NTP RFC-1305, SNTP RFC-2030)



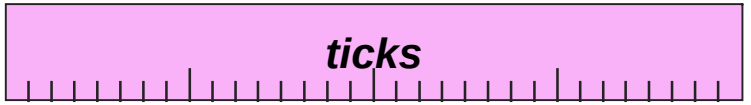
>150 years <250 ps

(IEEE 1588)

(+epoc)



(EPON)



16 ns

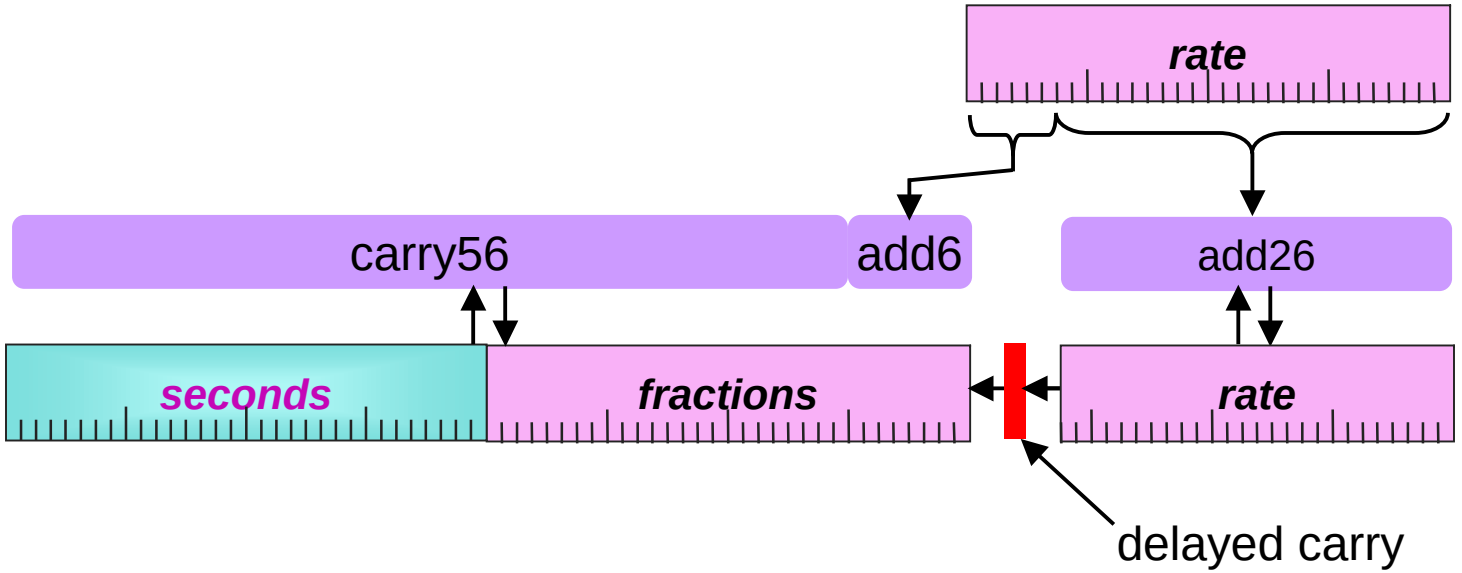
OR

(…)

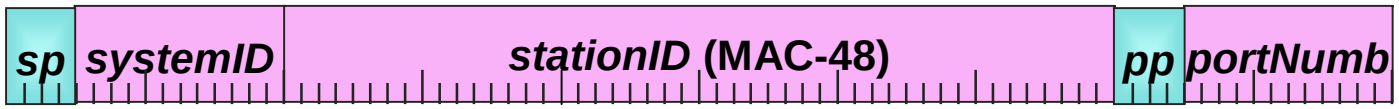
Format selection criteria

- Highest possible precision
  - Binary number (not BCD)
- Complete solution
  - 64-bit number
- Consistent with 1588, etc.
  - 32-bit seconds component
- Simple computations
  - 64-bit: seconds and fractions-of-second
- Client interface?
  - A logical interface (doesn't really matter)

Time-of-day rate adjustments



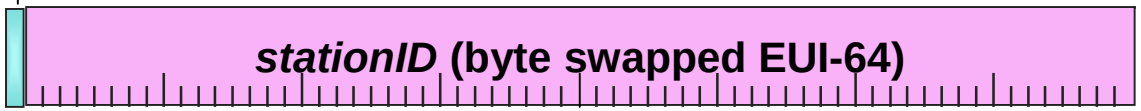
Time-of-day precedence



smaller

802.1 STP precedence
(IEEE Std 802.1D-2004)

preferred



larger

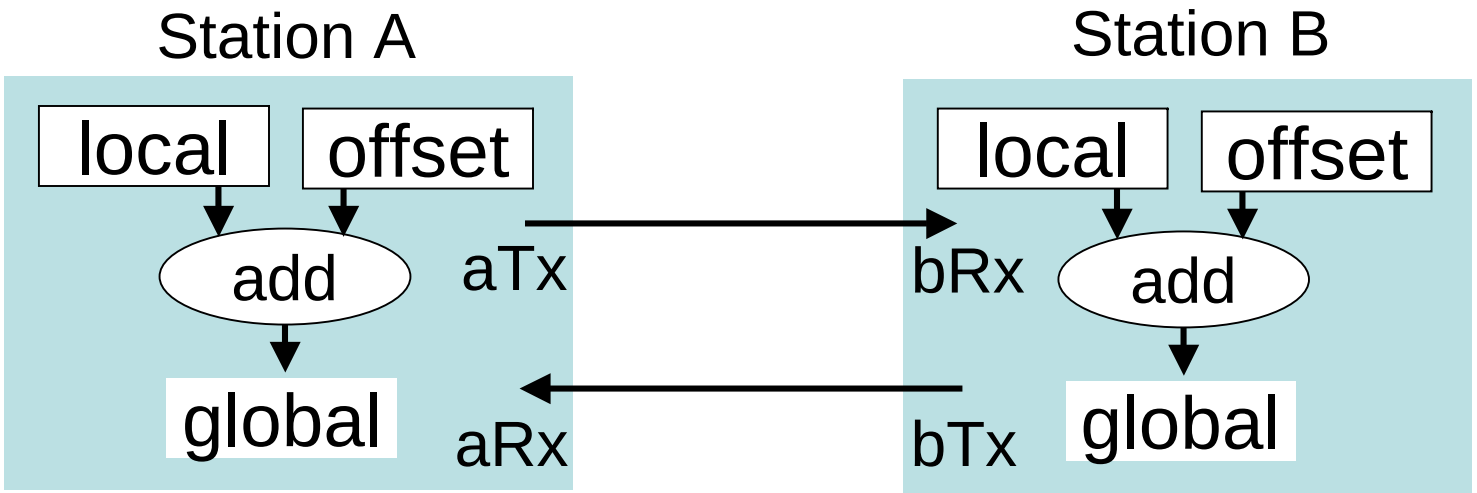
1394 precedence

Synchronized time-of-day clocks

How?

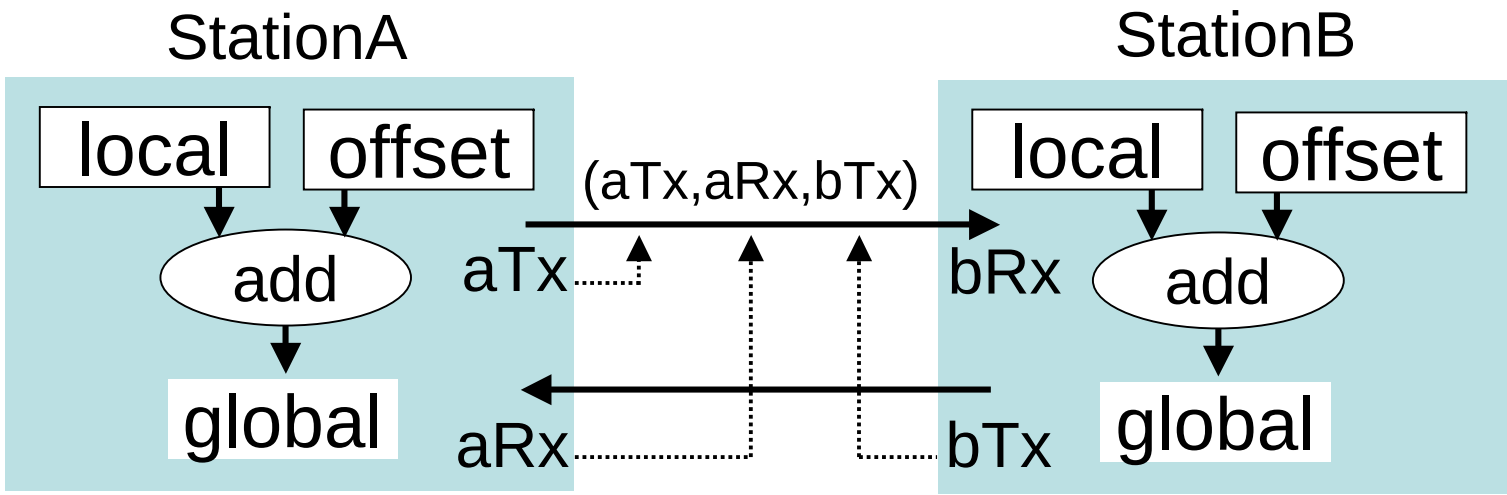
Adjacent-station synchronization

Timing snapshots



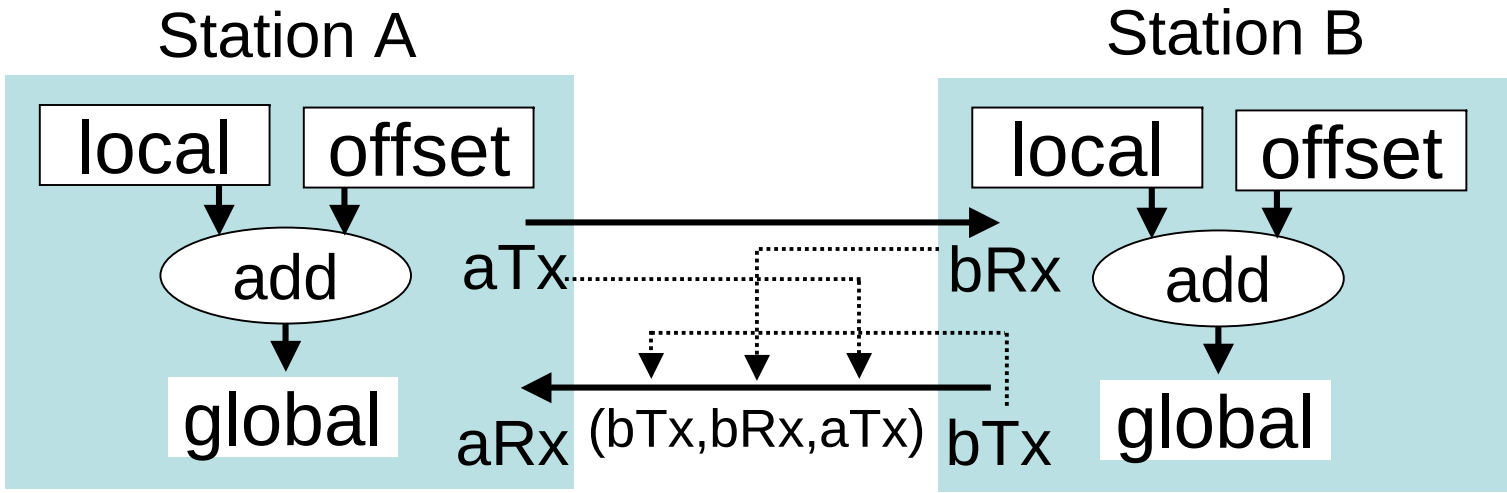
Adjacent-station synchronization

Snapshot value distribution
(information for stationB)

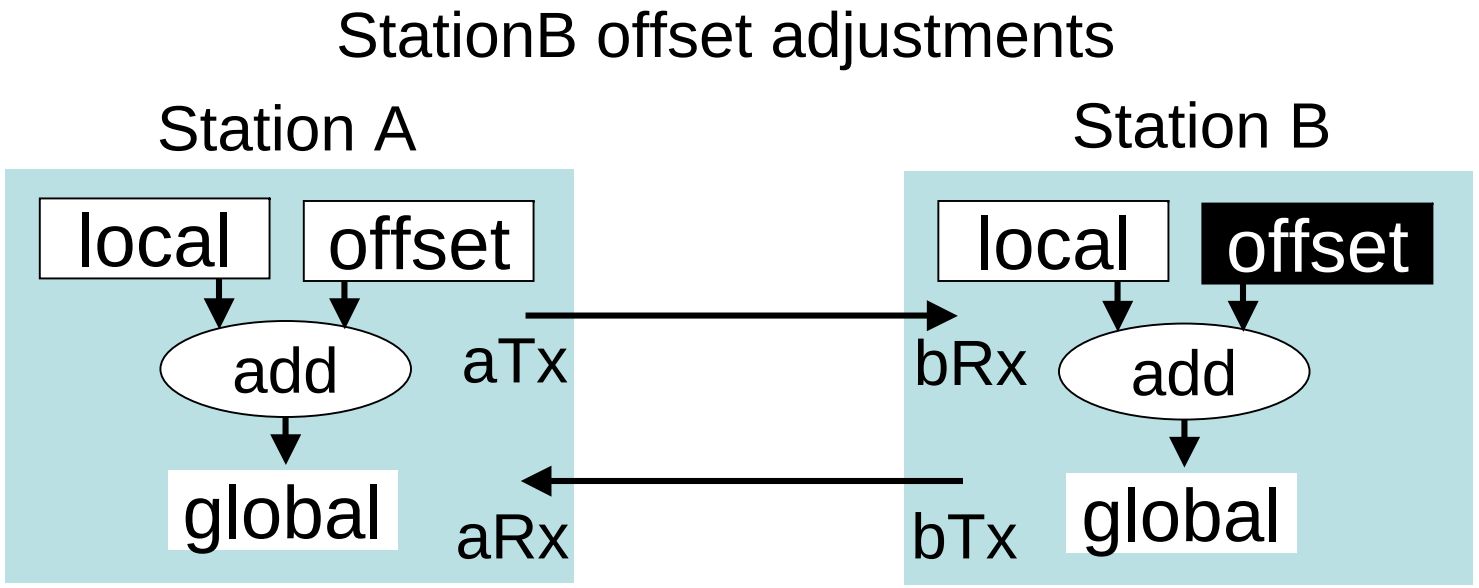


Adjacent-station synchronization

Snapshot value distribution
(information for stationA)

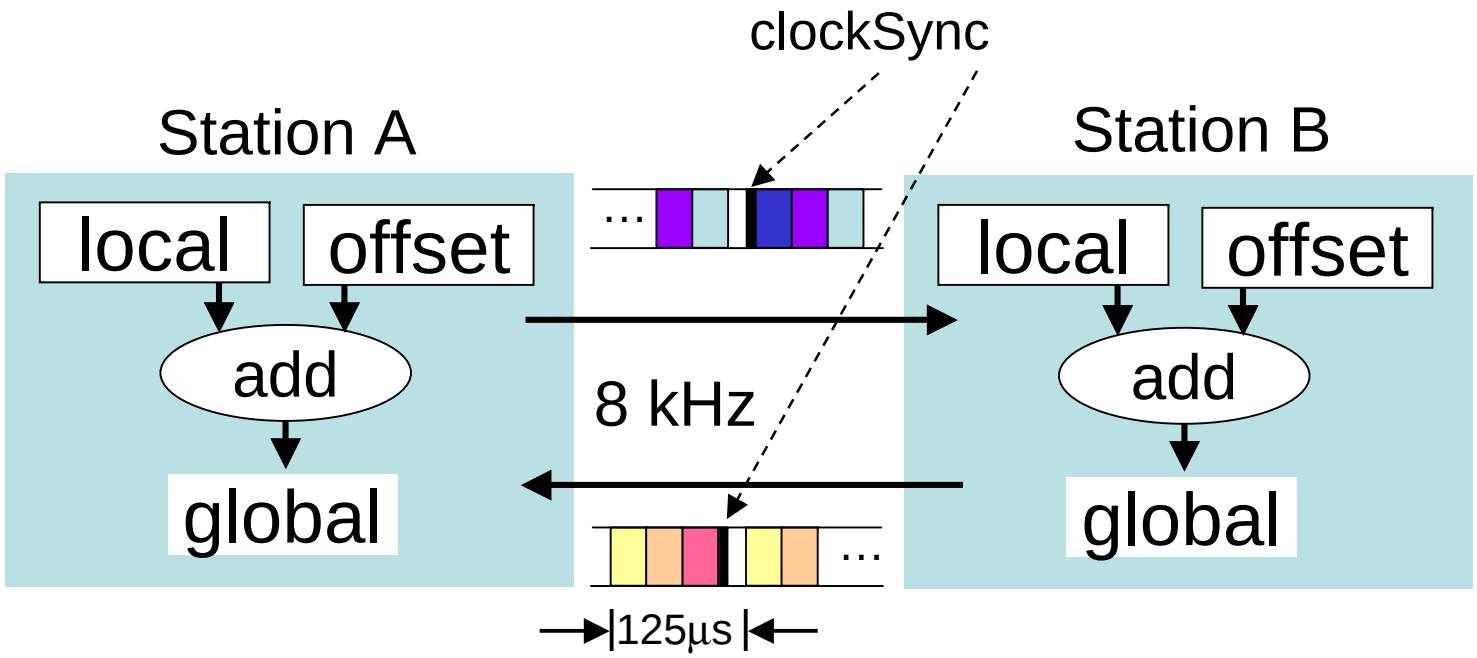


Adjacent-station synchronization

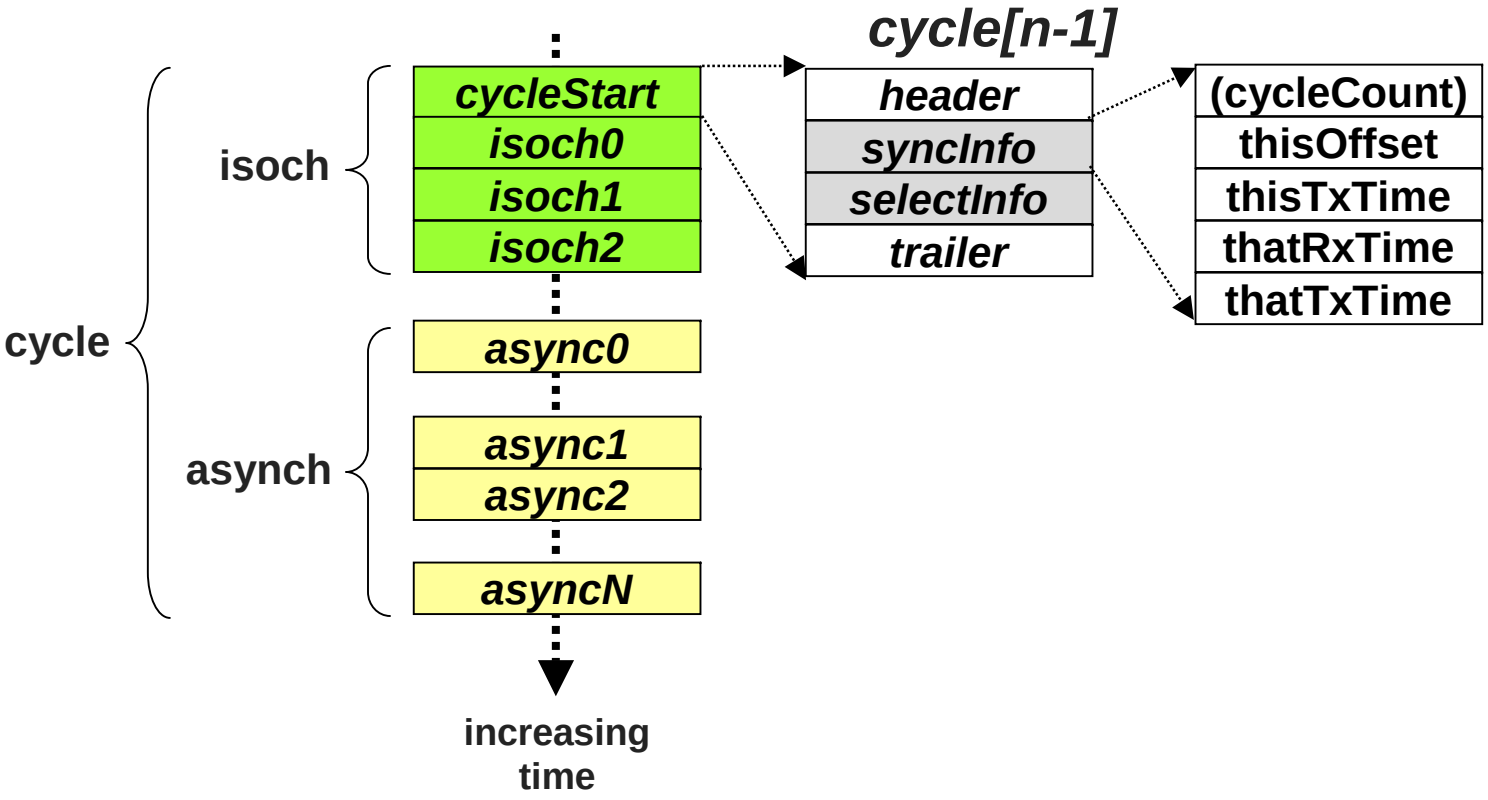


- rxDelta = (bRx – aTx);
- txDelta = (bTx – aRx);
- clockDelta = (rxDelta – txDelta) / 2;
- cableDelay = (rxDelta + txDelta) / 2;
- offsetB = offsetA – clockDelta;

Adjacent station synchronization



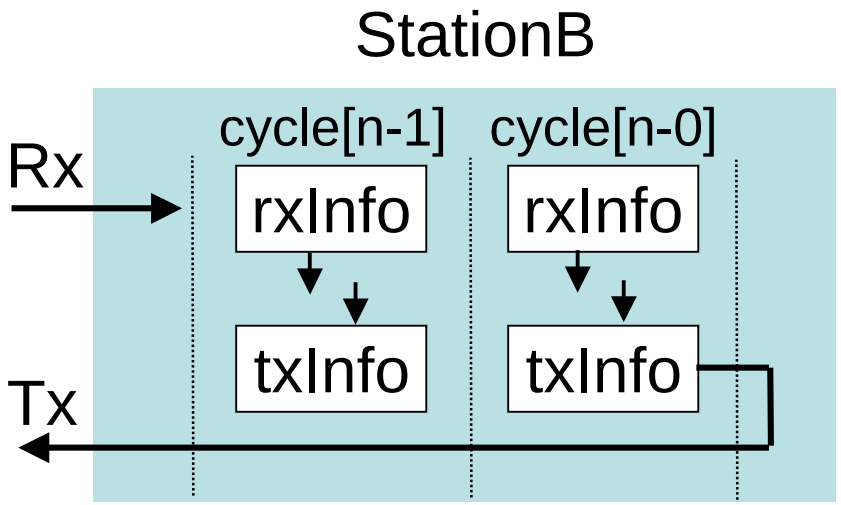
clockSync frame format…



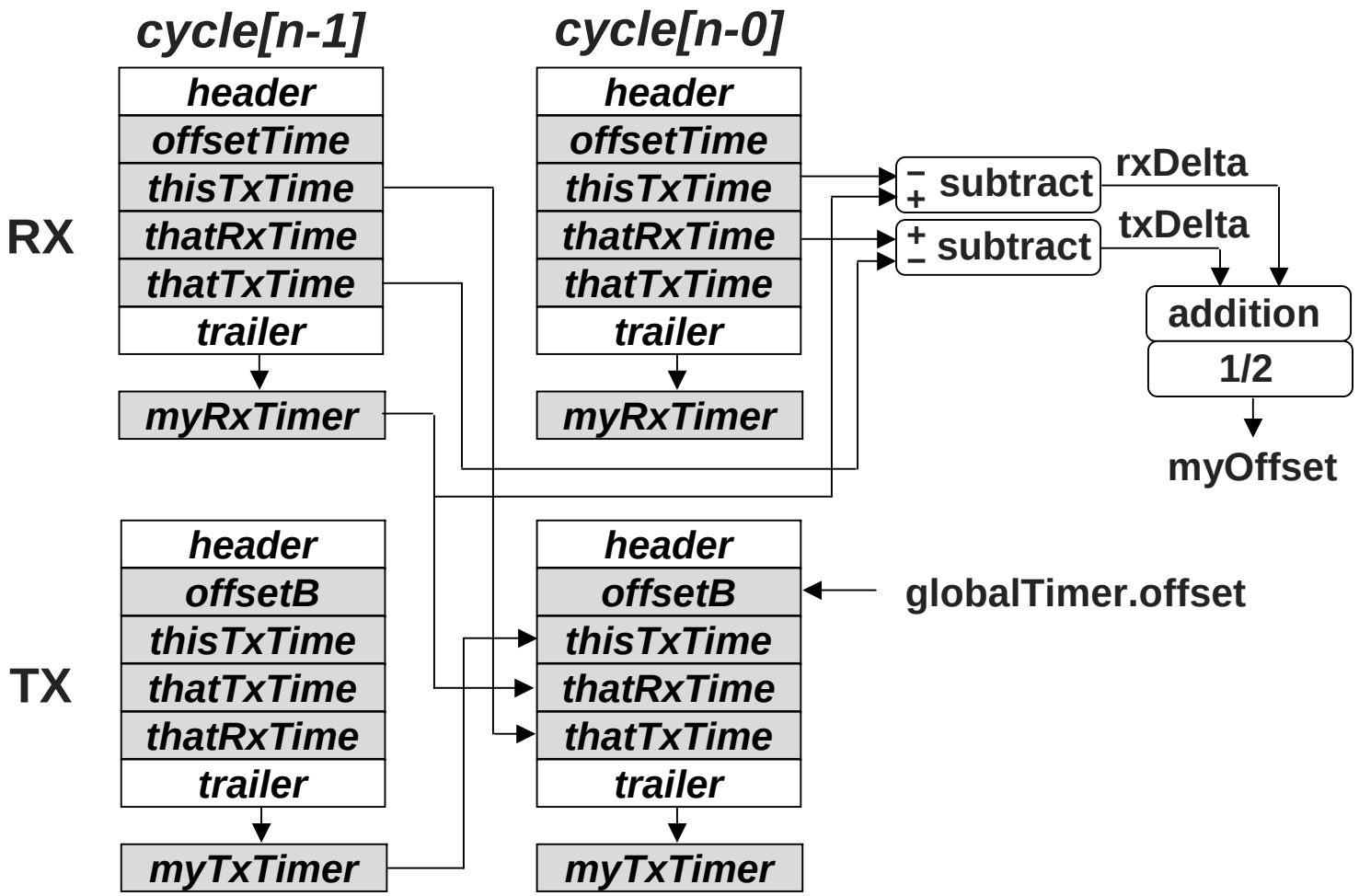
Why synchronous clock-syncs?

- Simple
  - Uses existing isochronous transmission state machine
  - Like IEEE 1588, requires only frame-sent snapshots
- Timely
  - Quasi-periodic transmissions
- Responsive
  - Sampling-to-adjustment delays are minimized
- Efficient
  - *sequenceNumber* is the *cycleCount*
- Consistent
  - Just another isochronous frame…

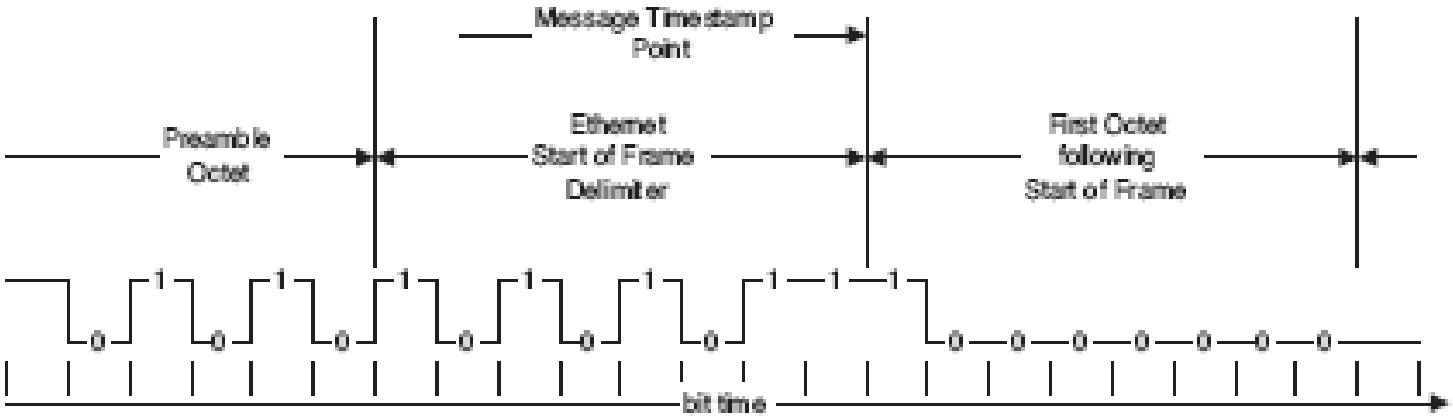
Clock slave details (1)



Clock-slave details (2)

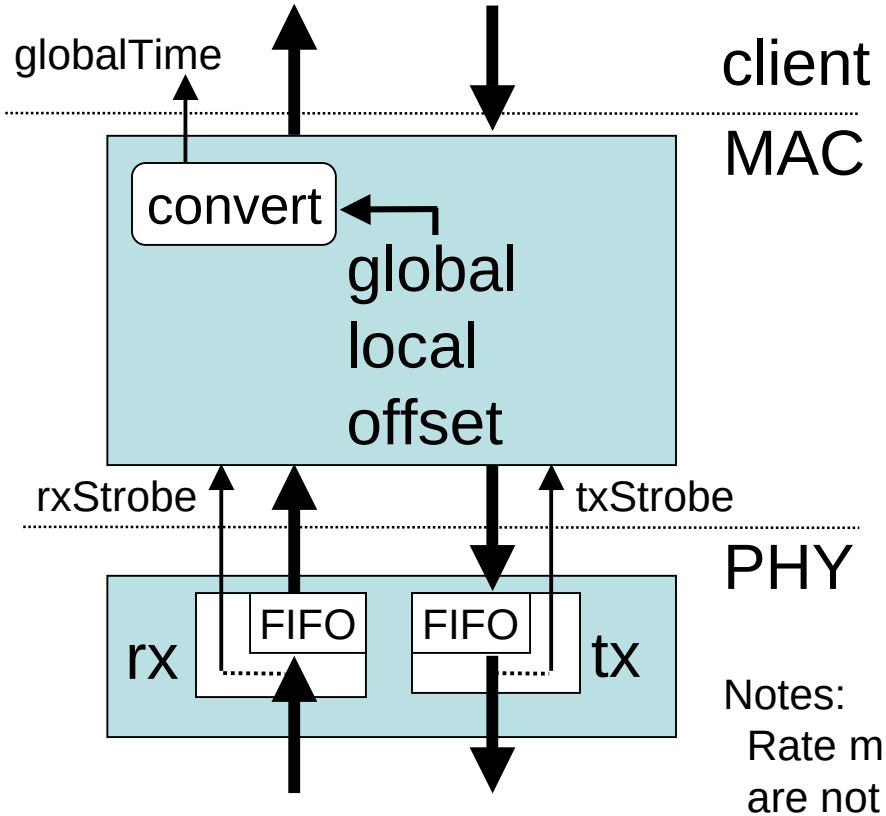


Timing specifics…



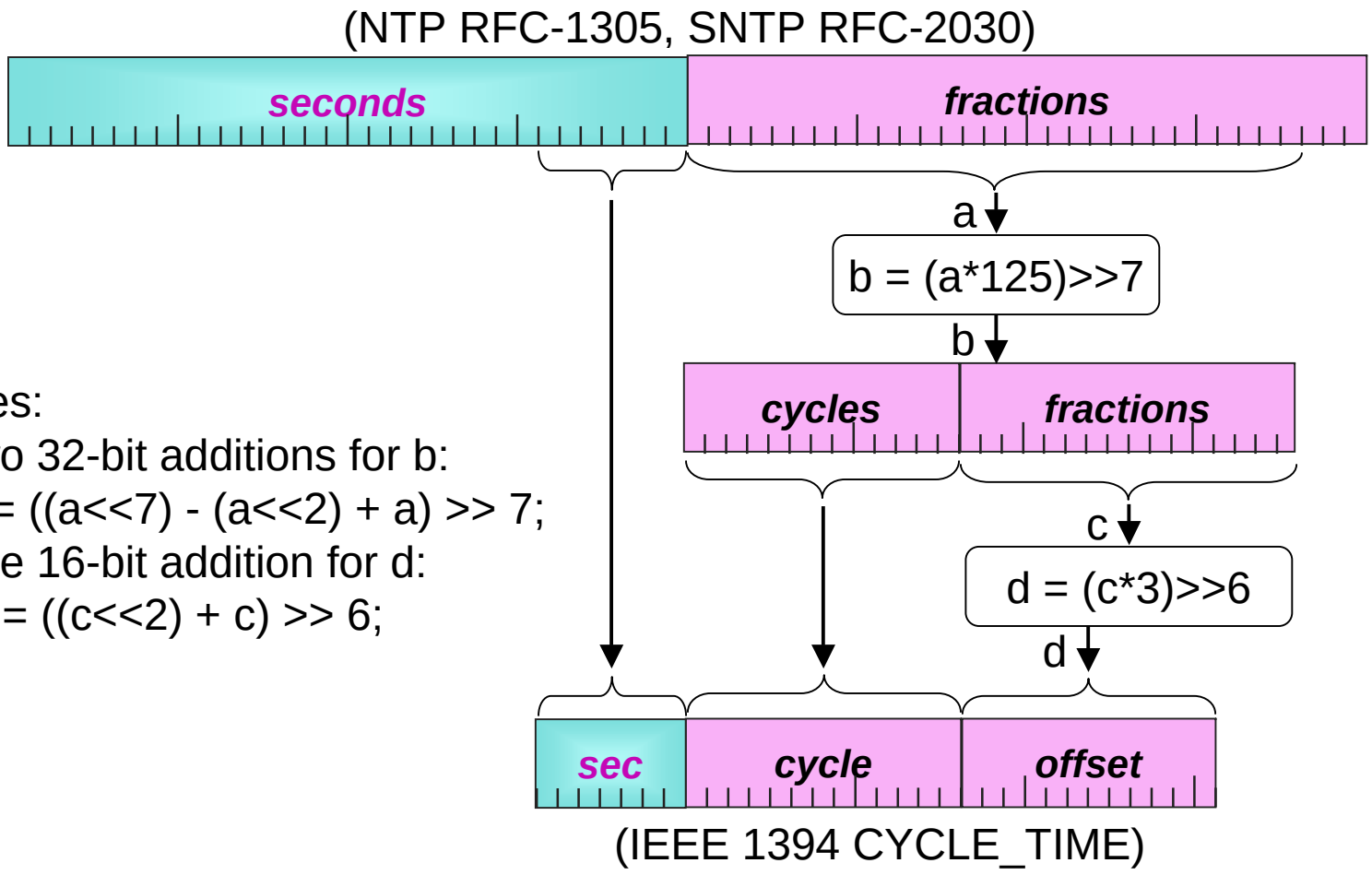
(from IEEE 1588-2002, subclause D.1.1, page 127)

A viable design model



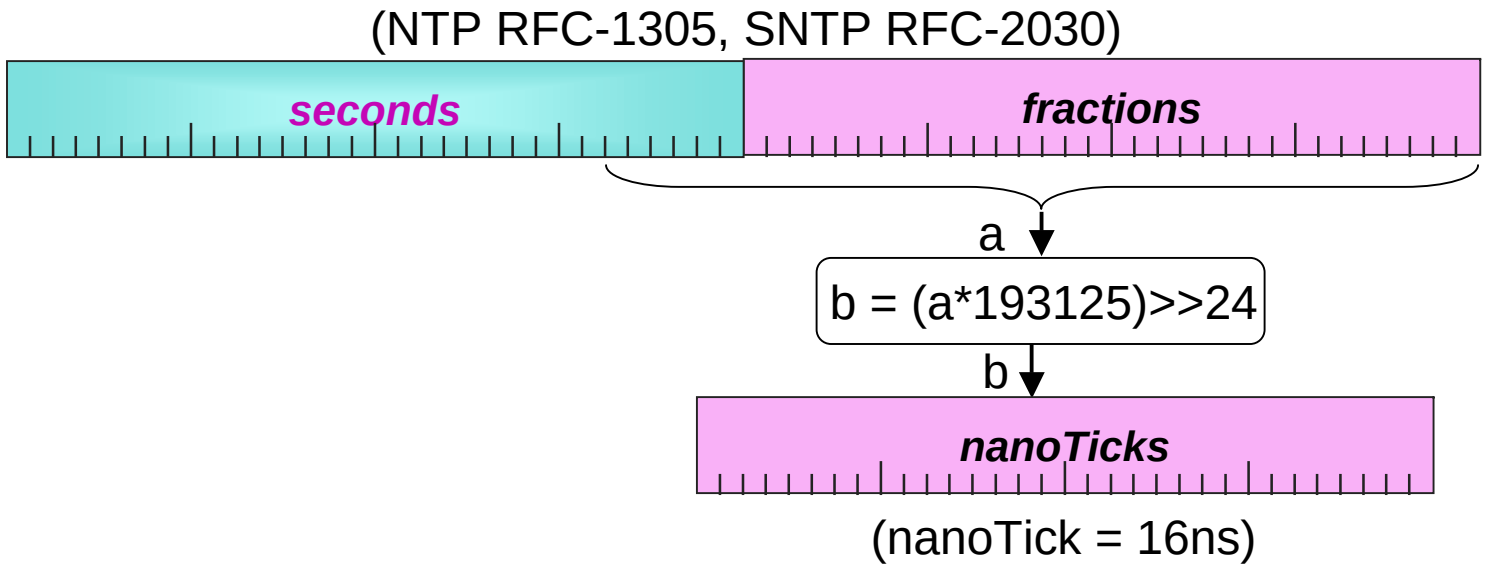
Notes:
Rate matching FIFOs
are not within our scope.

Conversion example: 1394



Notes:
- Two 32-bit additions for b:
  - b= ((a<<7) - (a<<2) + a) >> 7;
- One 16-bit addition for d:
  - d = ((c<<2) + c) >> 6;

Conversion example: EPON

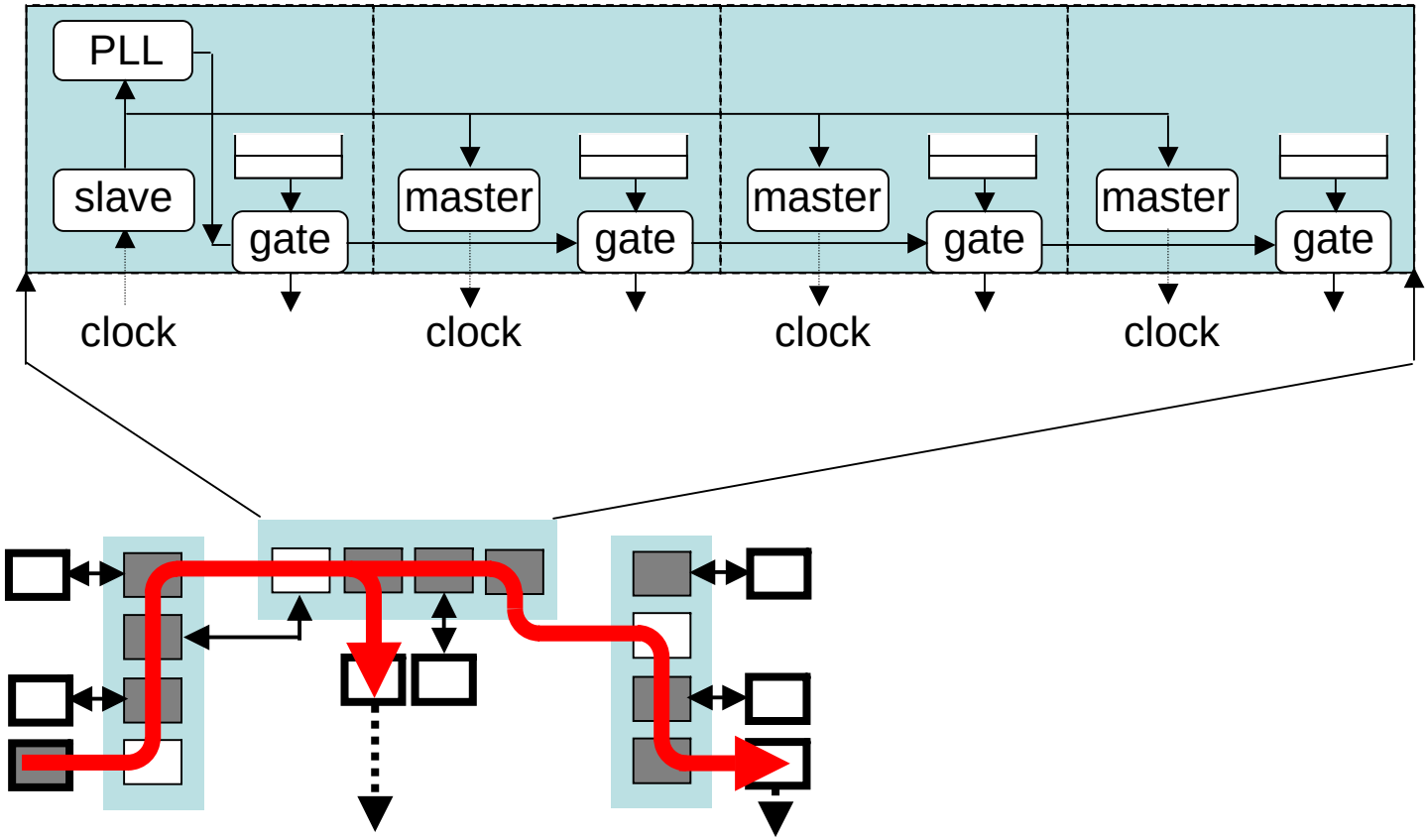


Notes:

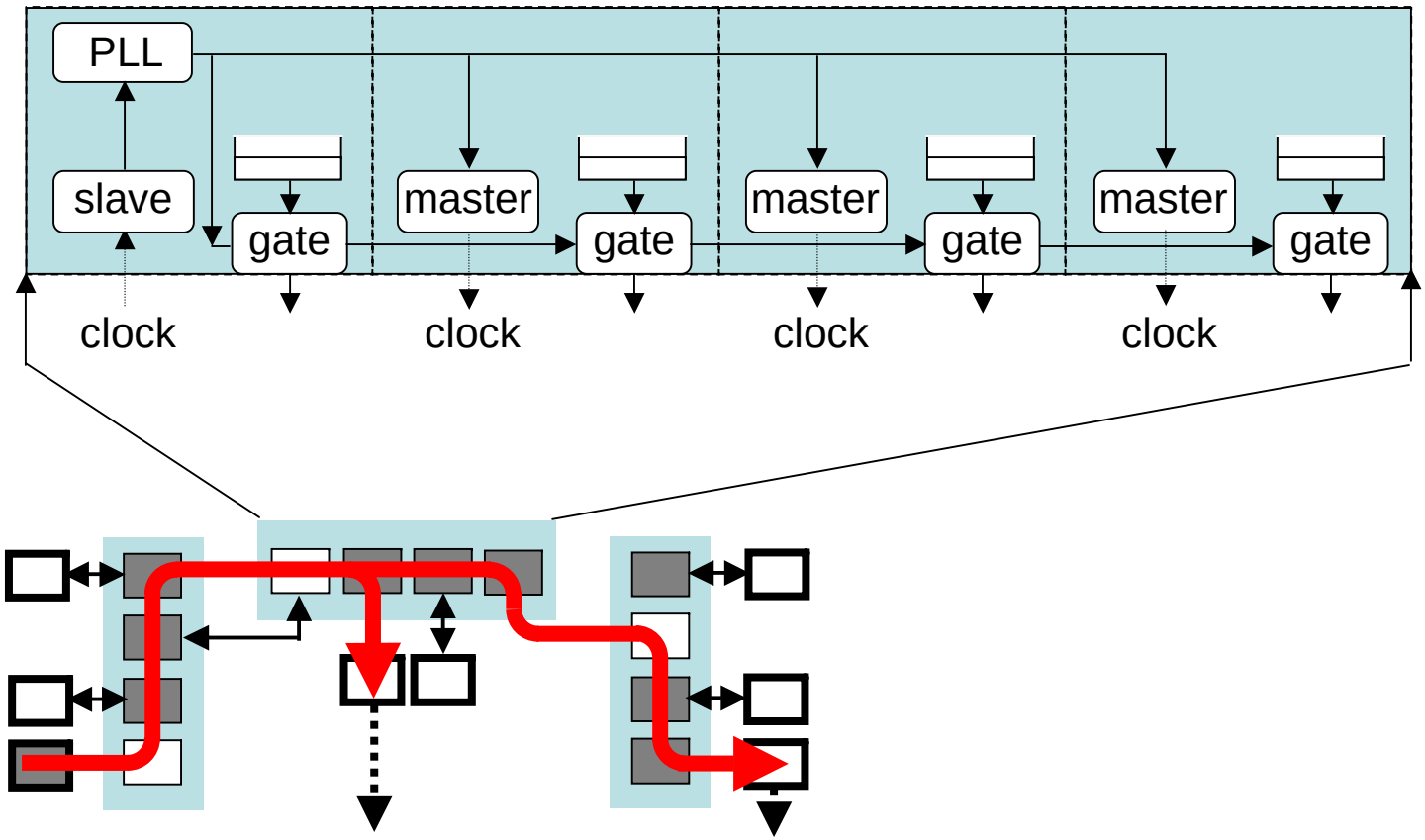
Thus, eight 36-bit additions compute a very precise b value:

b= ((a<<17)+(a<<16)-(a<<12)+(a<<9)+(a<<6)+(a<<5)+(a<<2)+a) >>24;

Passby PLLs (proposal 1)



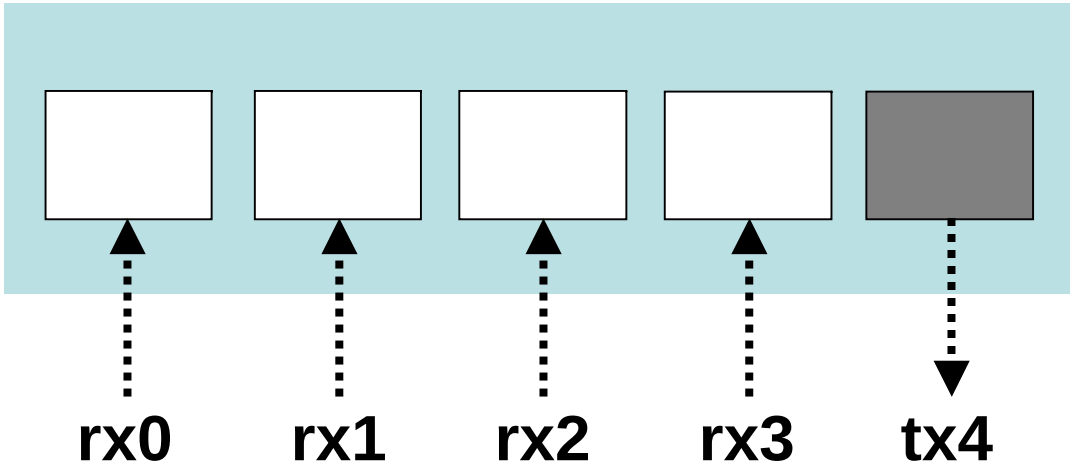
Passthrough PLLs (option 2)



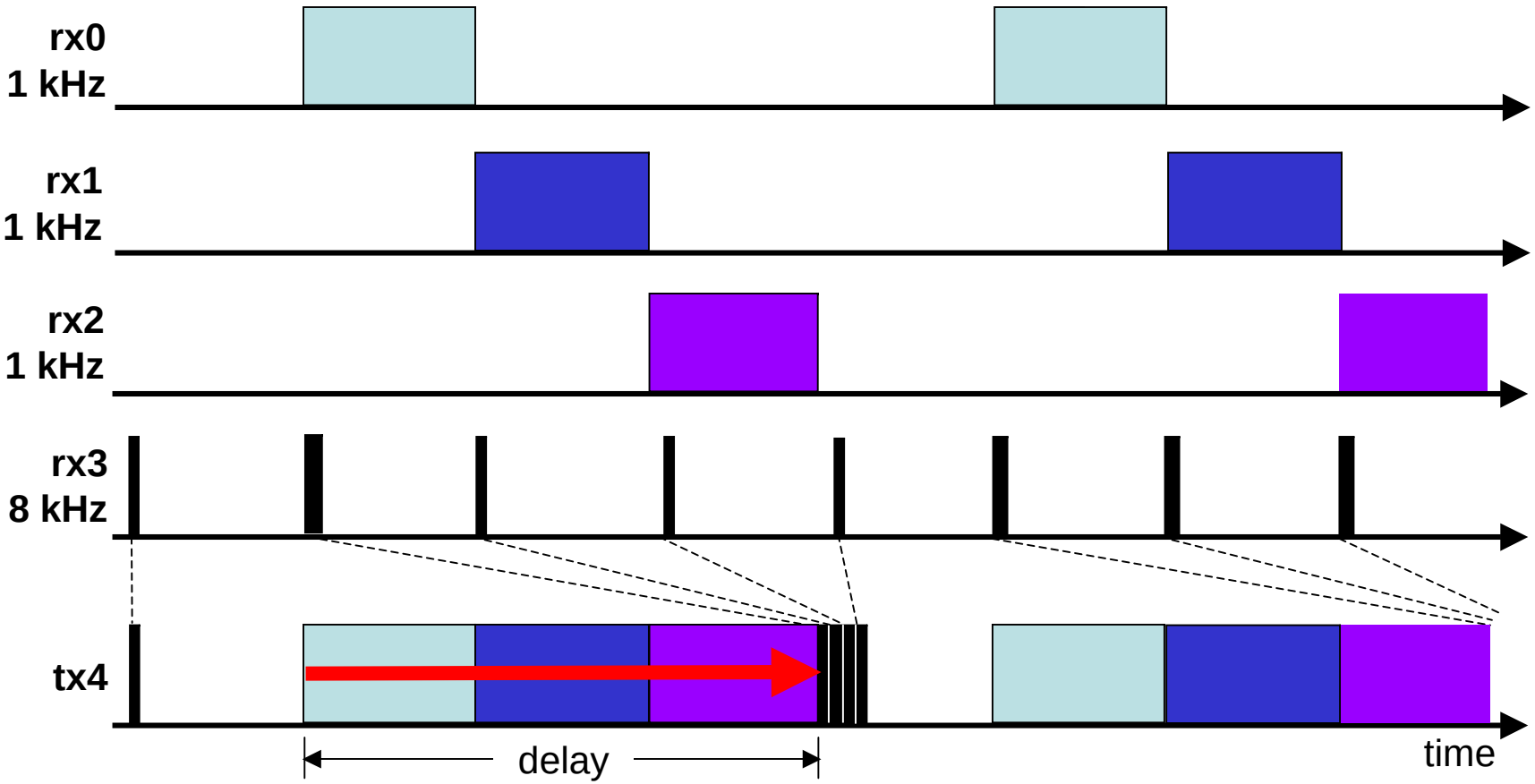
Synchronized time-of-day clocks

Why?

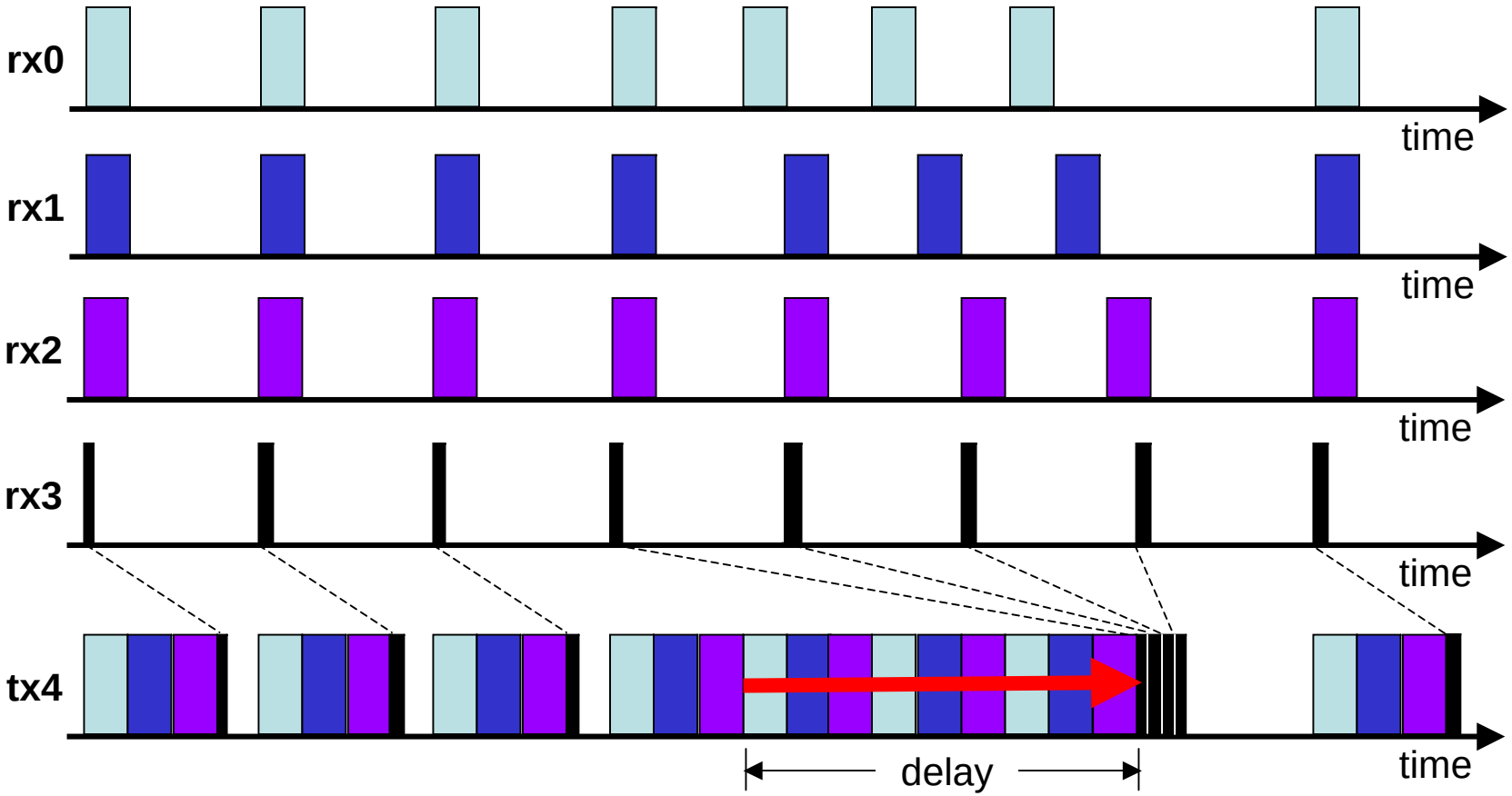
Consider possible congestion…



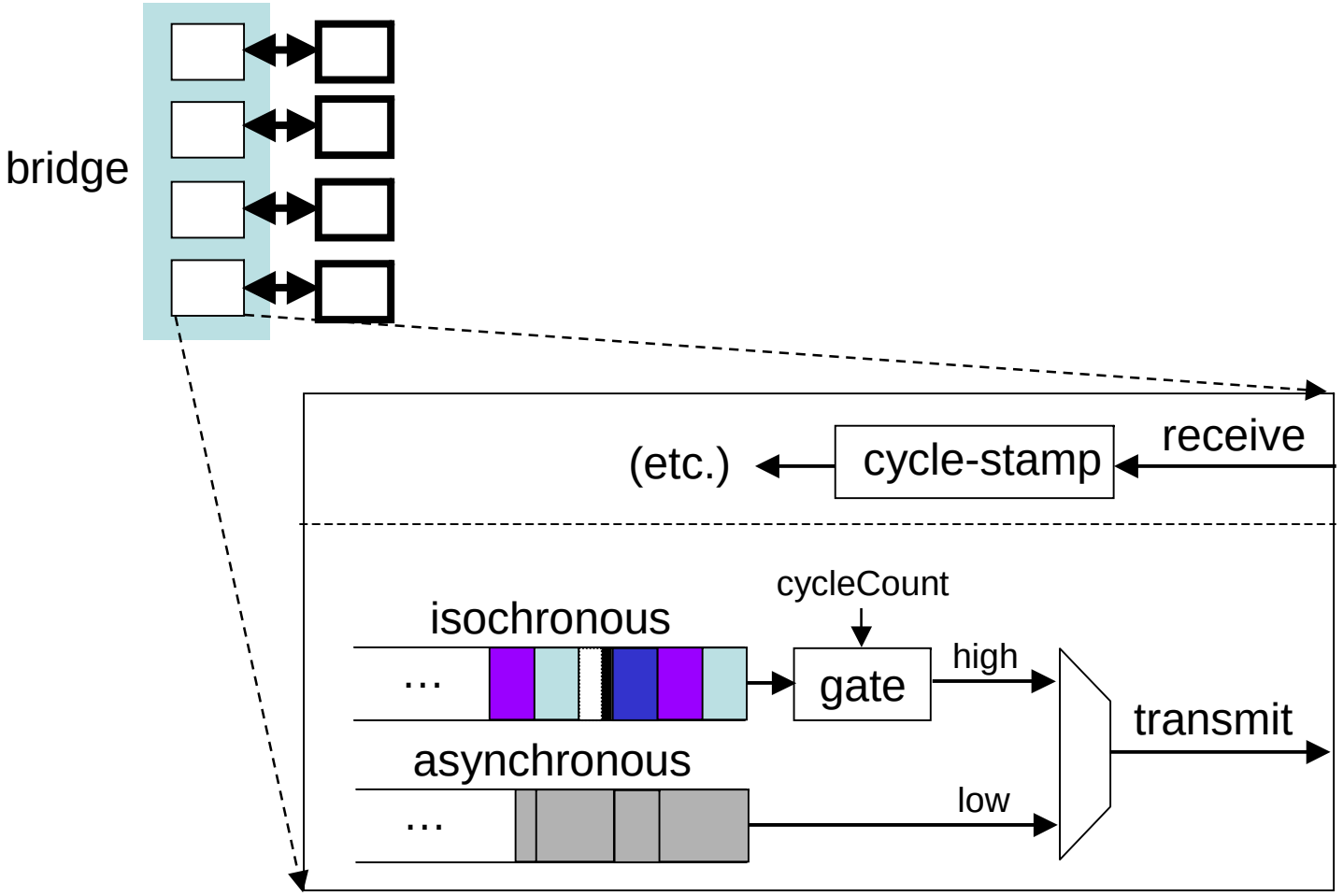
Bursting causes jitter



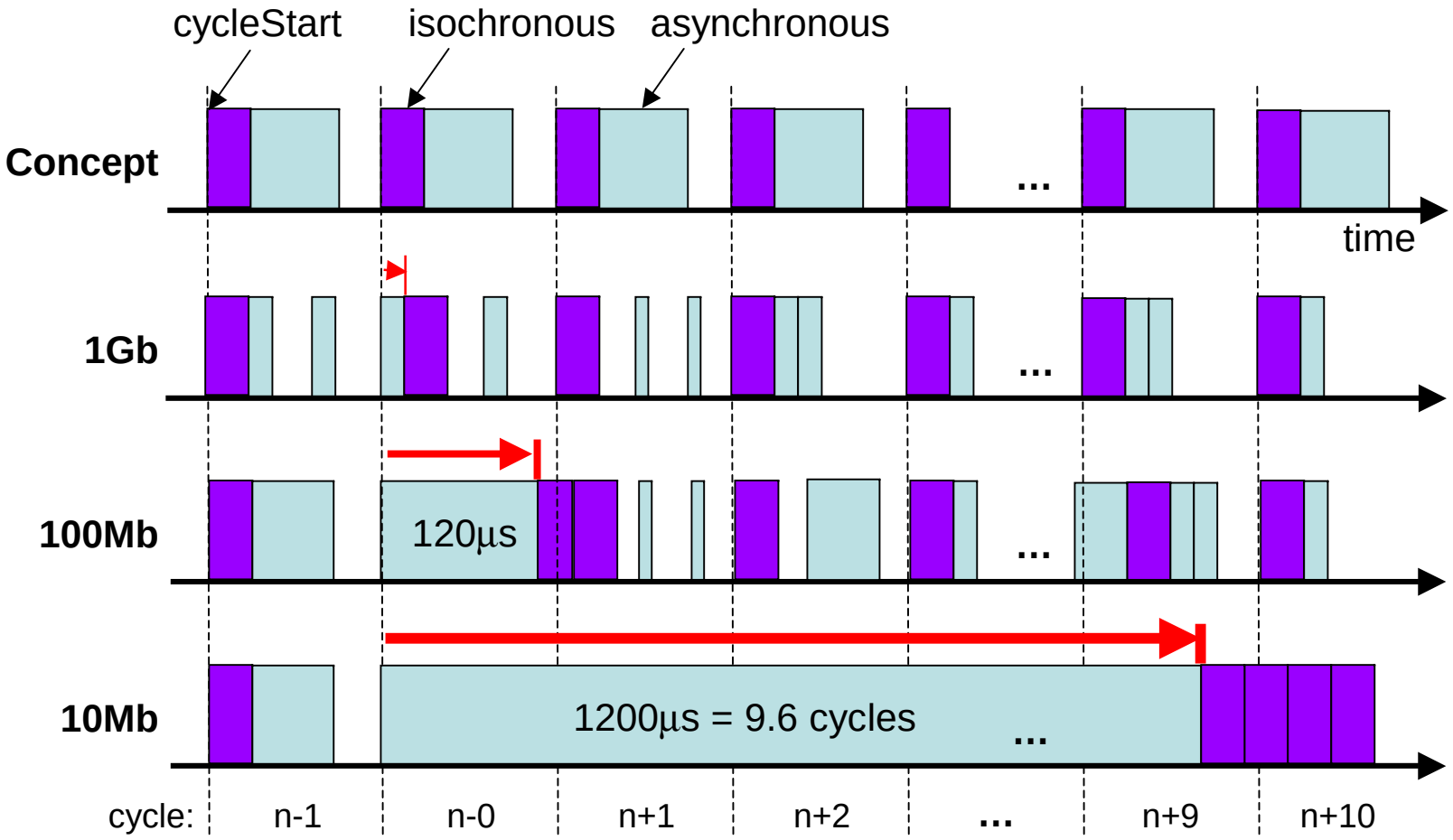
Bunching causes jitter



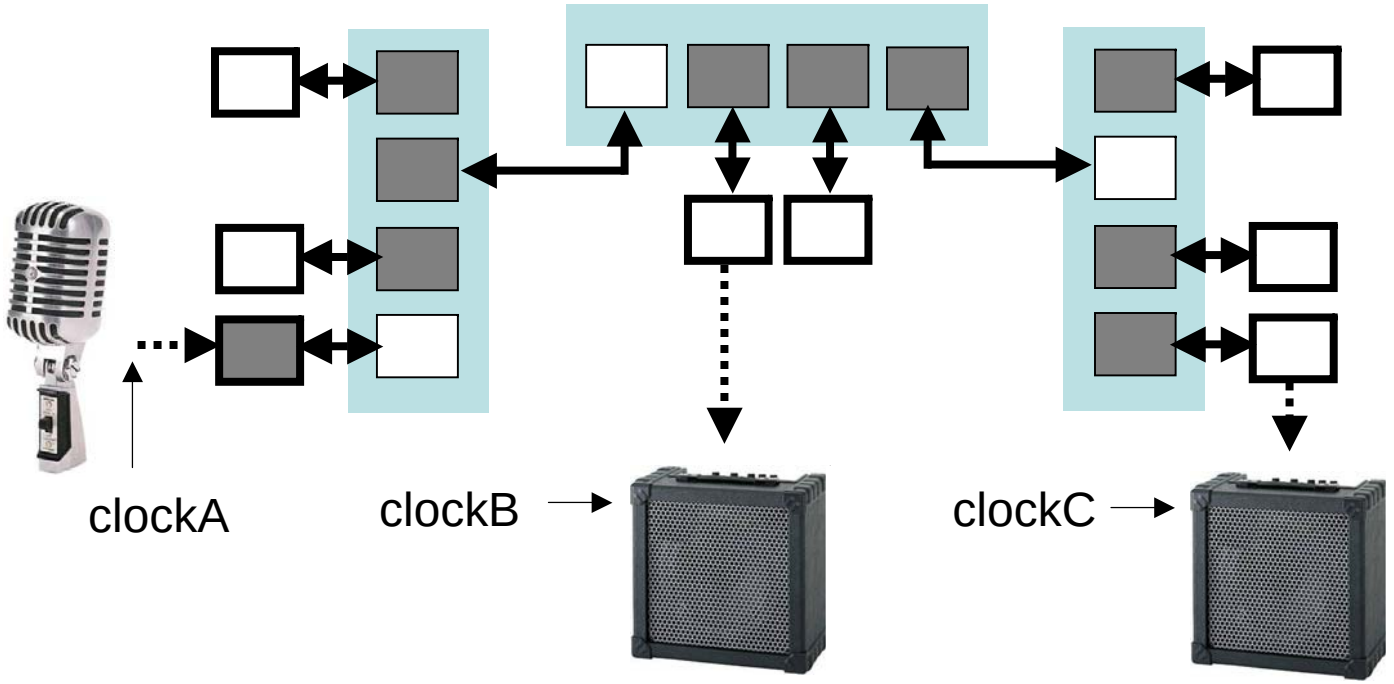
Bridge re-clocking contains jitter



Frame transmission timings



Synchronized reception/presentation



No long-term drift: clockA, clockB, clockC
Clock jitter: sub nanosecond (after PLL)

In summary

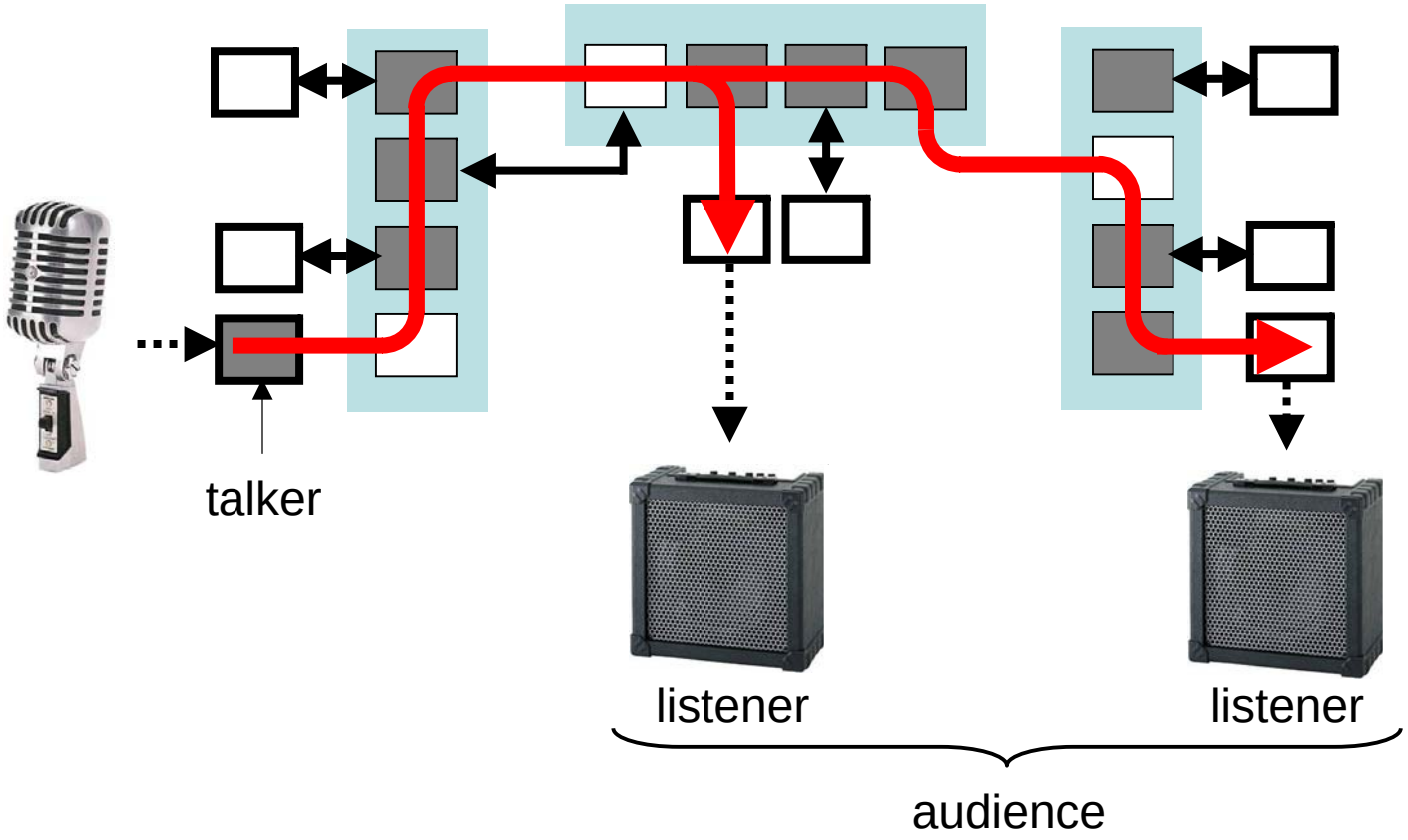
- Time-of-day synchronization (house clock)
  - Global synchronization is required
  - Implemented as cascaded adjacent synchronizations
- Time synchronization formats
  - Binary time is accurate with simple add/subtract
  - Clock-master voting: 48+ or 64+ selection priorities
- Time-of-day applications
  - Synchronous reception and presentation, within applications
  - Synchronous re-clocking within bridges
- Time-of-day distribution
  - Pipelined sampling for highest accuracies
  - Cable delays can be derived, based on the same information

Synchronized time-of-day clocks

Questions?

Subscription
(some possibilities)

Vocabulary terms (1)



Vocabulary terms (2)

- audience
  - The set of listeners associated with a talker
- clock master
  - A bridge or end station that provides a link clock reference
- clock slave
  - A bridge or end station that tracks the clock reference
- grand clock master (grand master)
  - The clock master that provides the network time reference
- listener
  - A sink of a stream, such as a television or acoustic speaker
- path
  - A logical concatenation of links and bridges for a stream

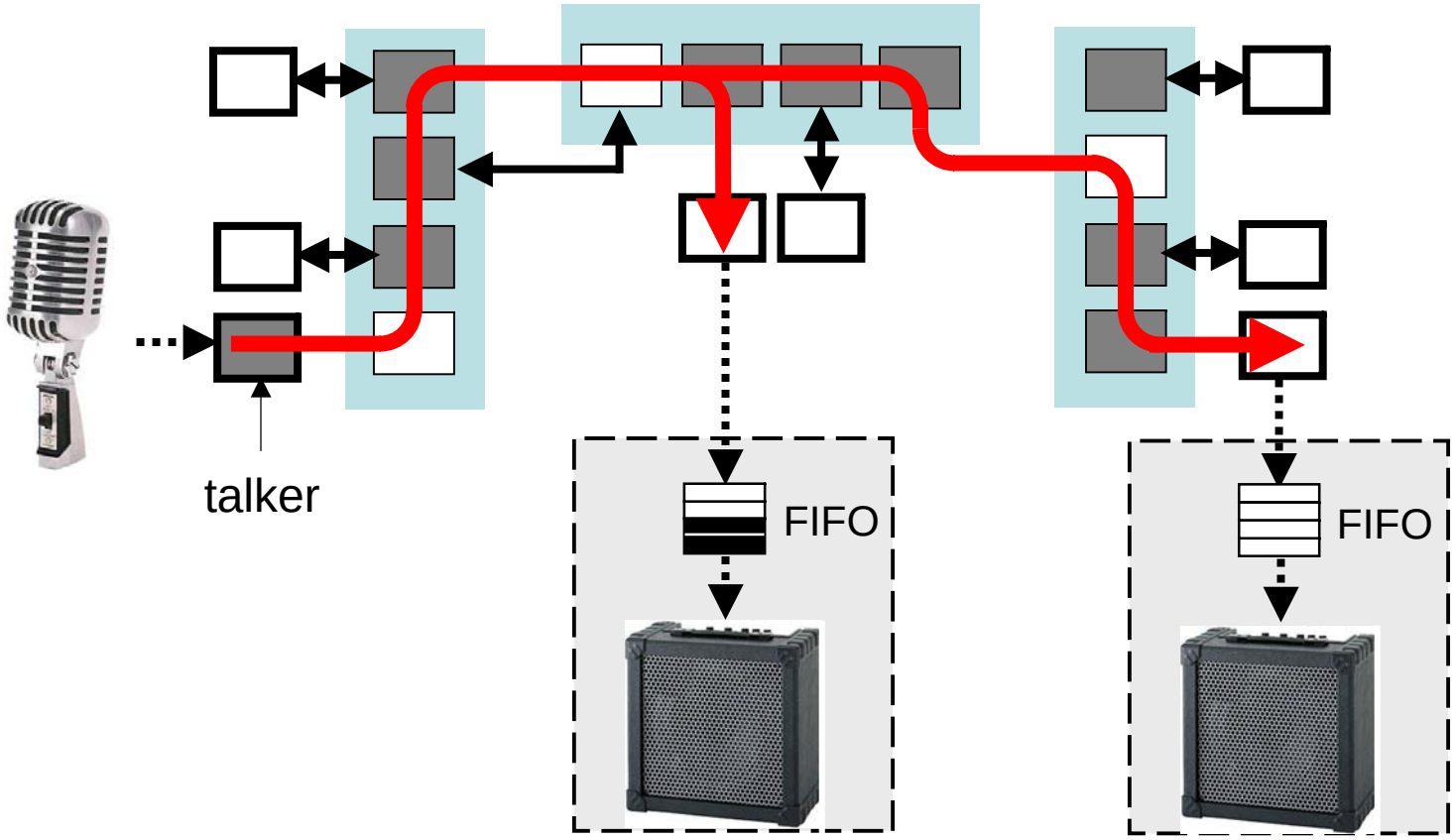
Vocabulary terms (3)

- service discovery*
  - The process used to identify/control/configure talkers
- stream
  - An RE frame sequence with a distinct streamID
- subscription**
  - Establishing committed paths between talker and listener(s)
- talker
  - A source of a stream, such as a cable box or microphone

* A complete solution/vocabulary includes out-of-802 scope activities

** A complete solution/vocabulary includes out-of-802.3 scope activities

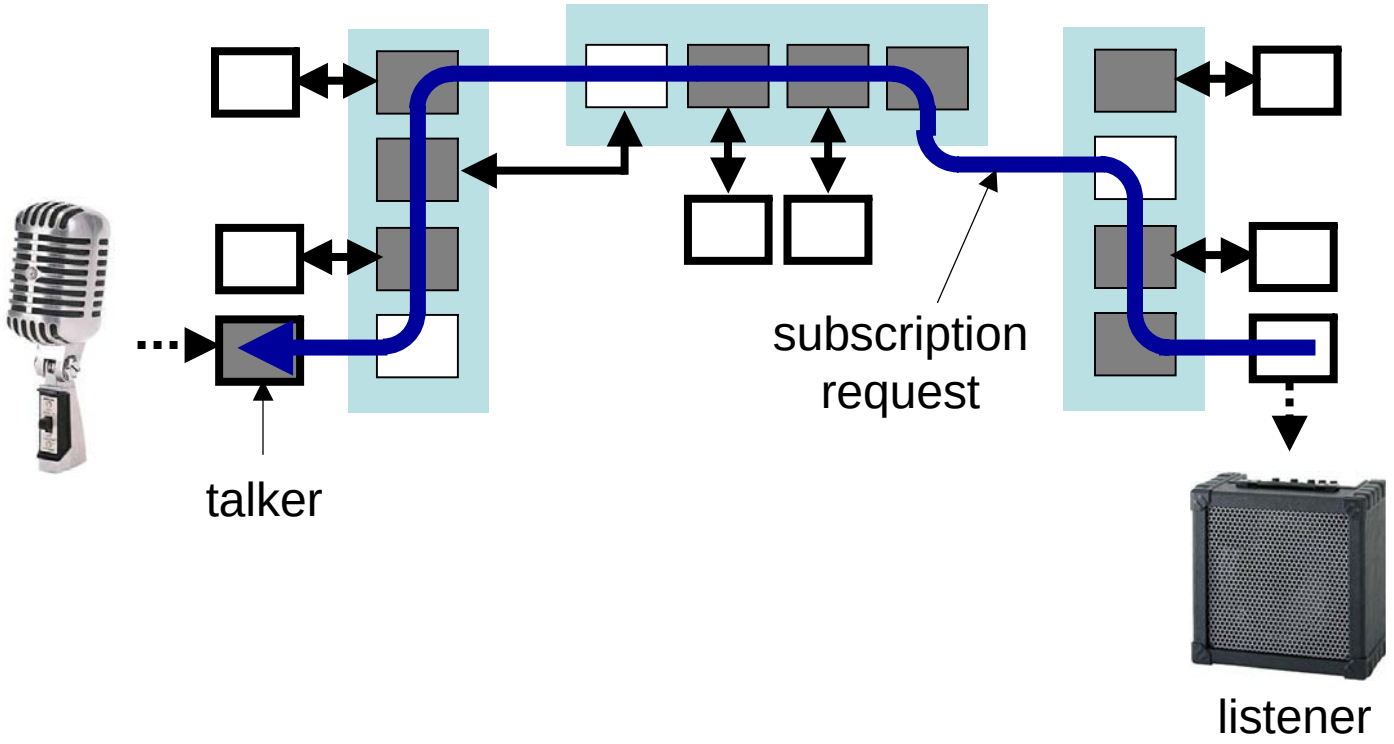
Delay-matching listener FIFOs



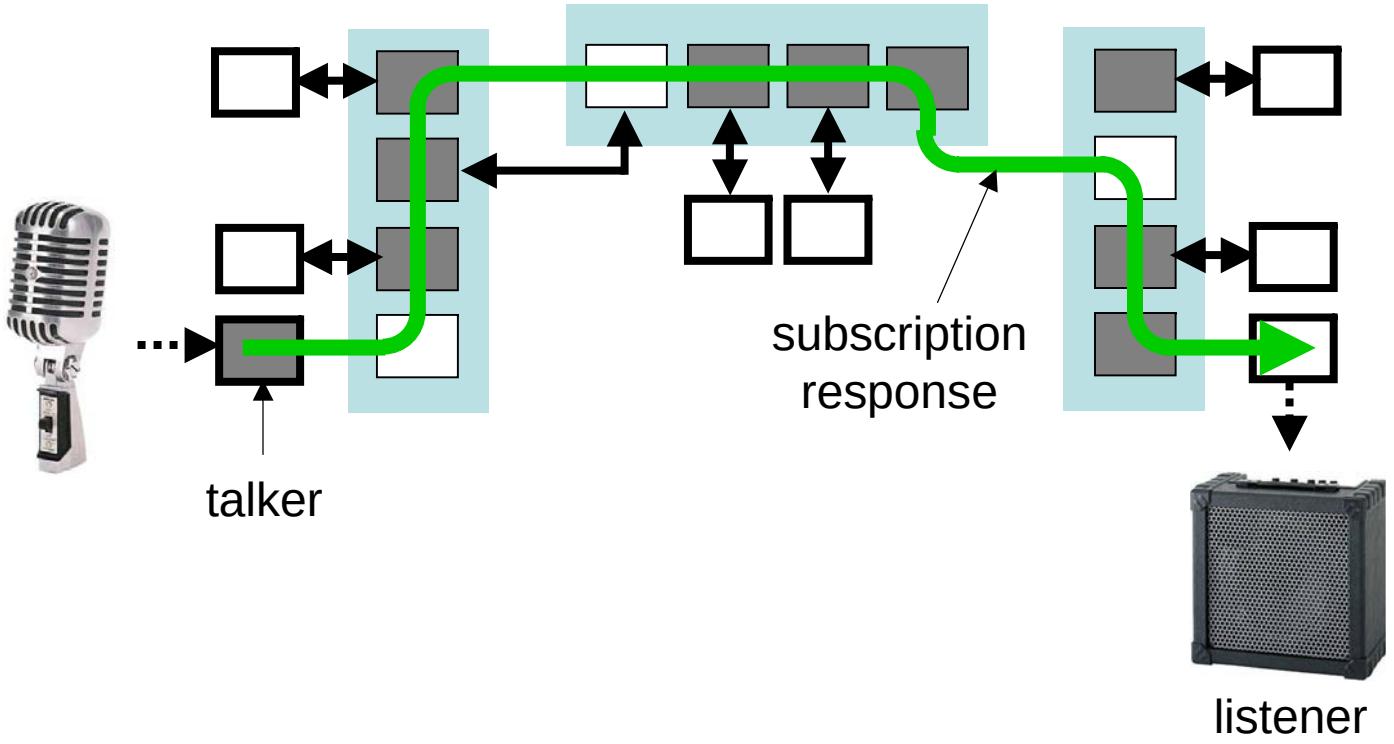
Multicast addresses!

- Multicasts are 1-to-N, not N-to-N
  - However, listener stations can listen to more than one talker
- Unicast
  - Unicast is just a special case of 1-to-N (N is 1)
  - 1-to-N is an architectural/temporal extension
- Allocation of multicast addresses?
  - Multicasts addresses come from a server
  - Multicast address assignments are “distributed”
  - Multicast addresses == streamId == {source, plug}

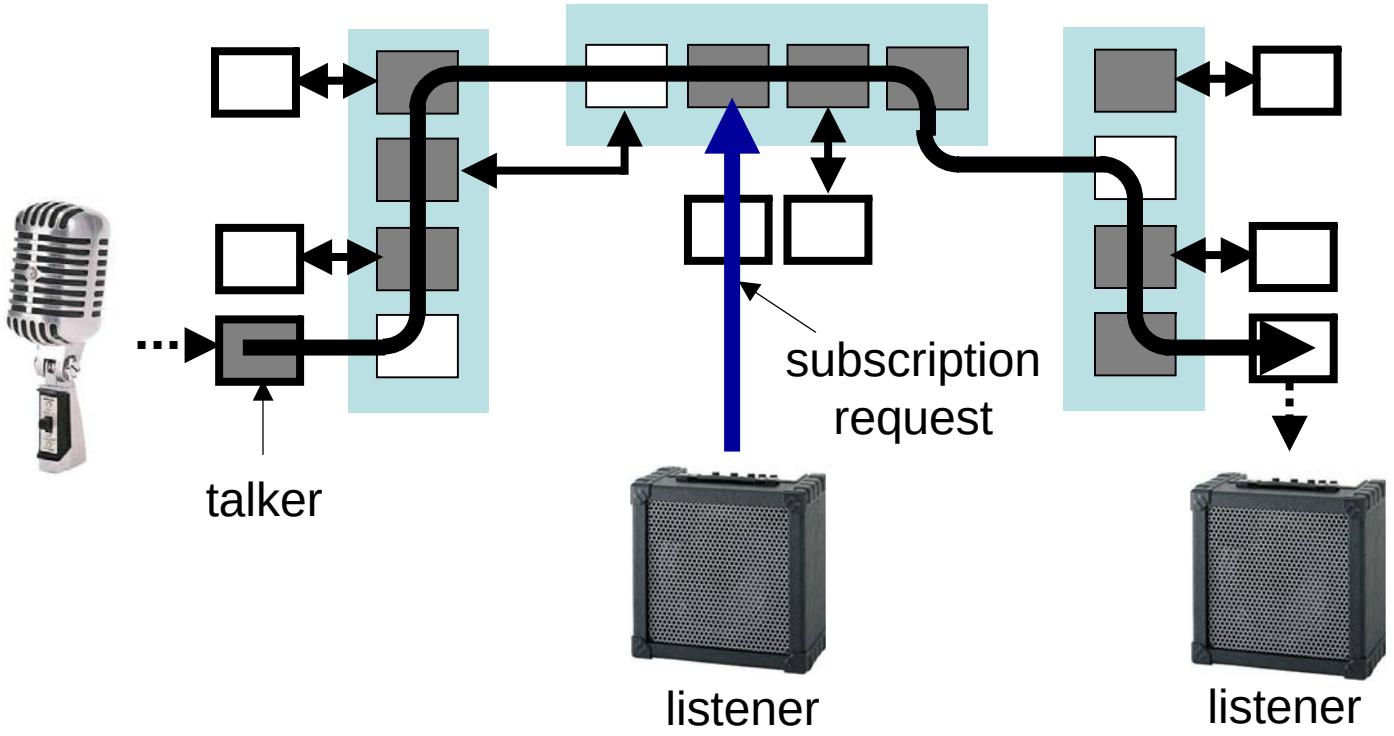
First access request



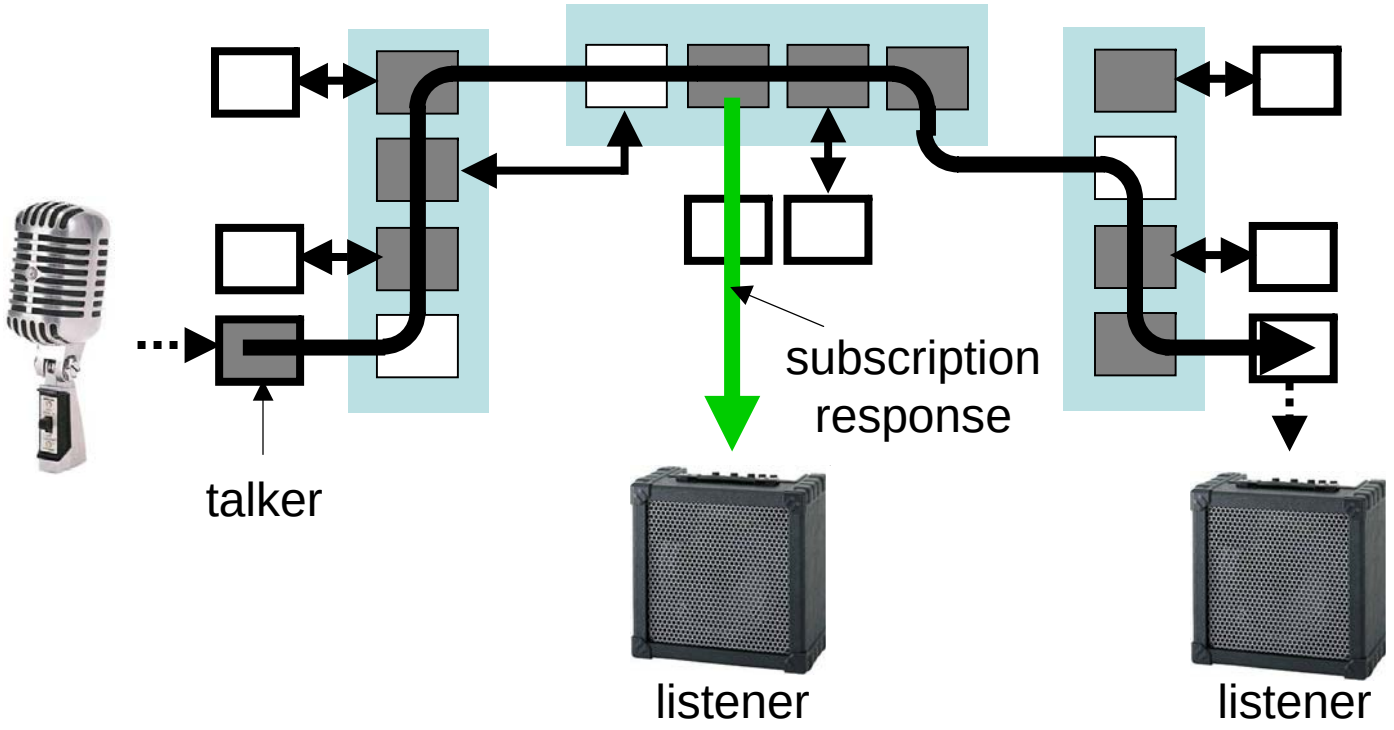
First access response



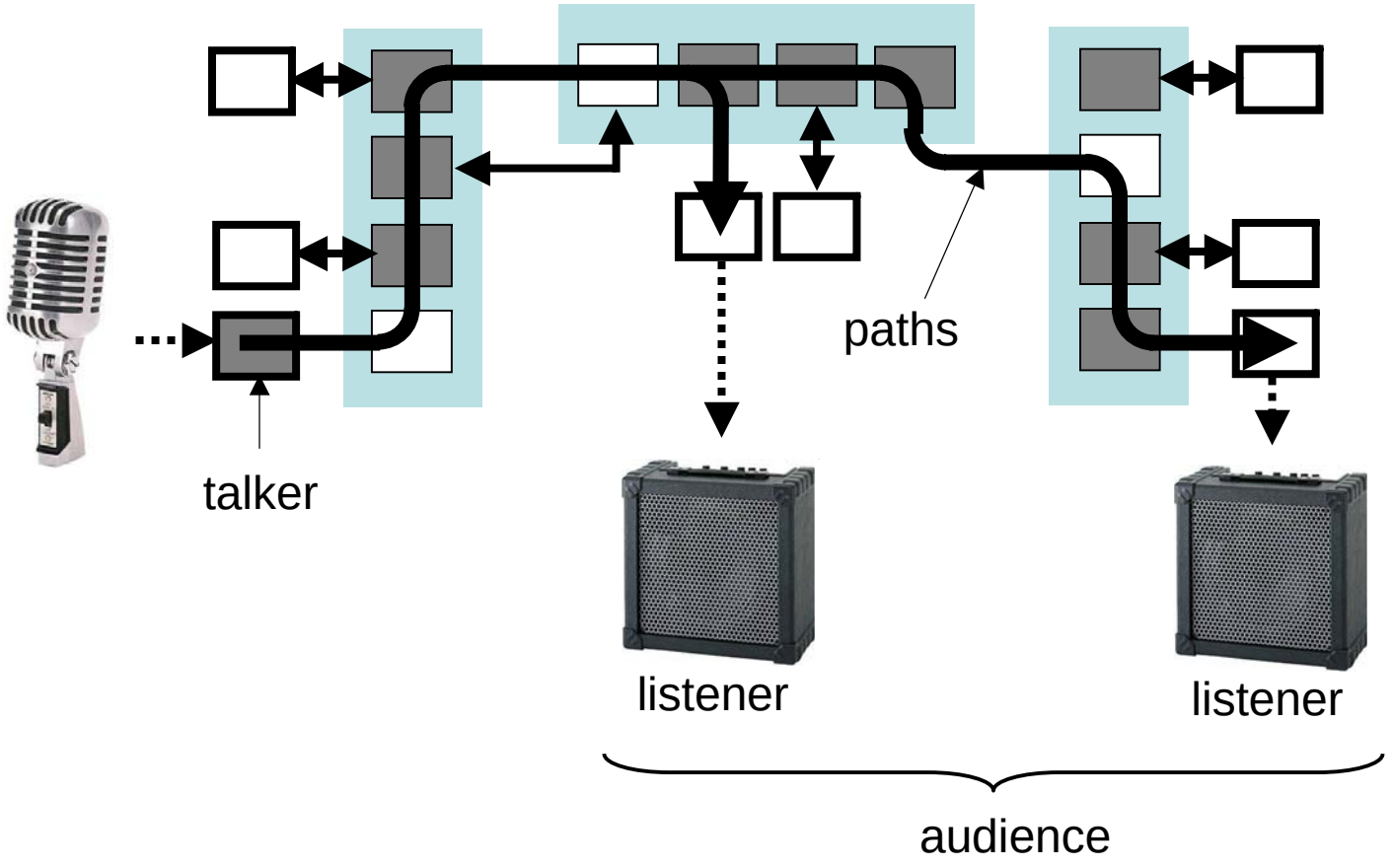
Second access request



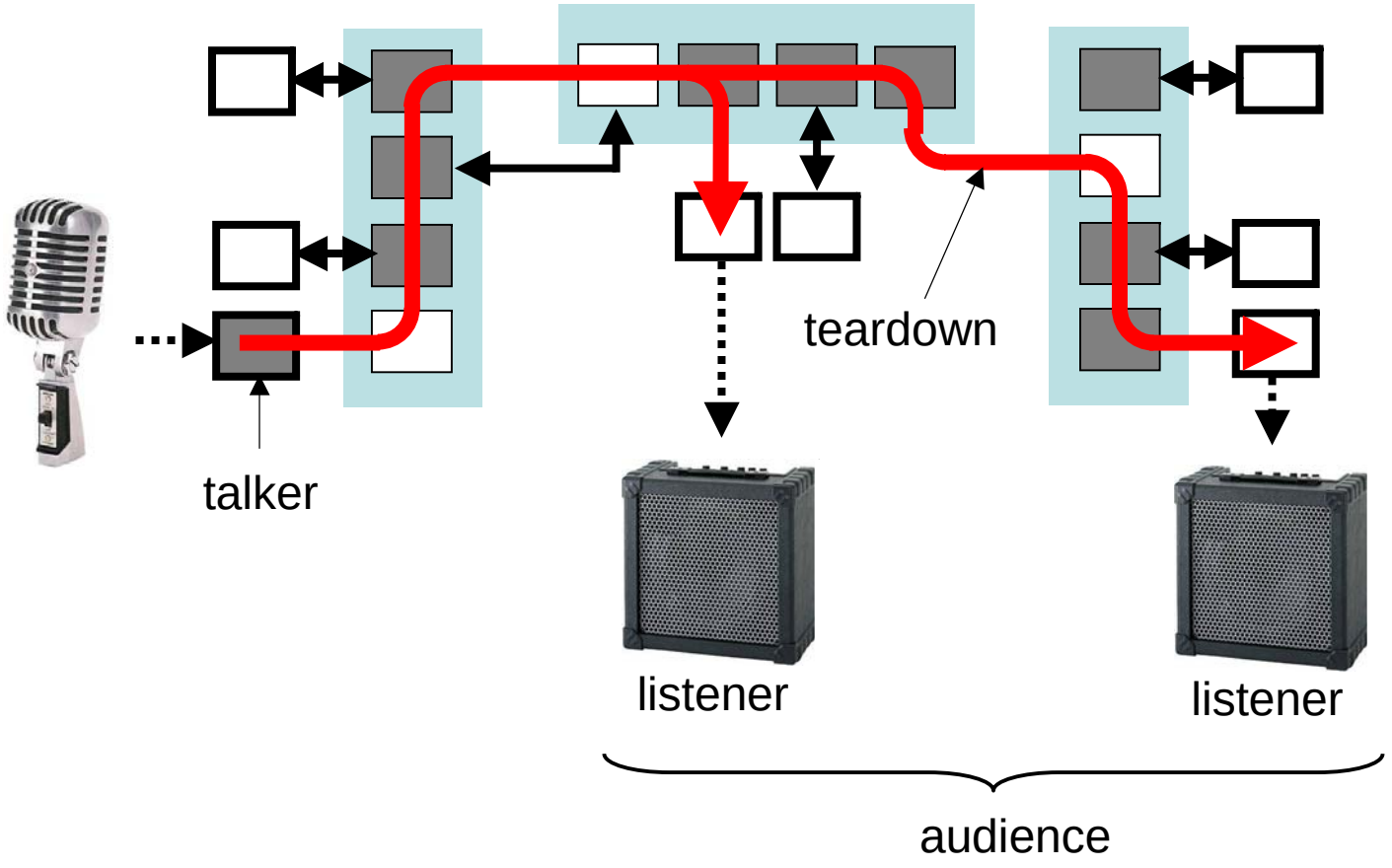
Second access response



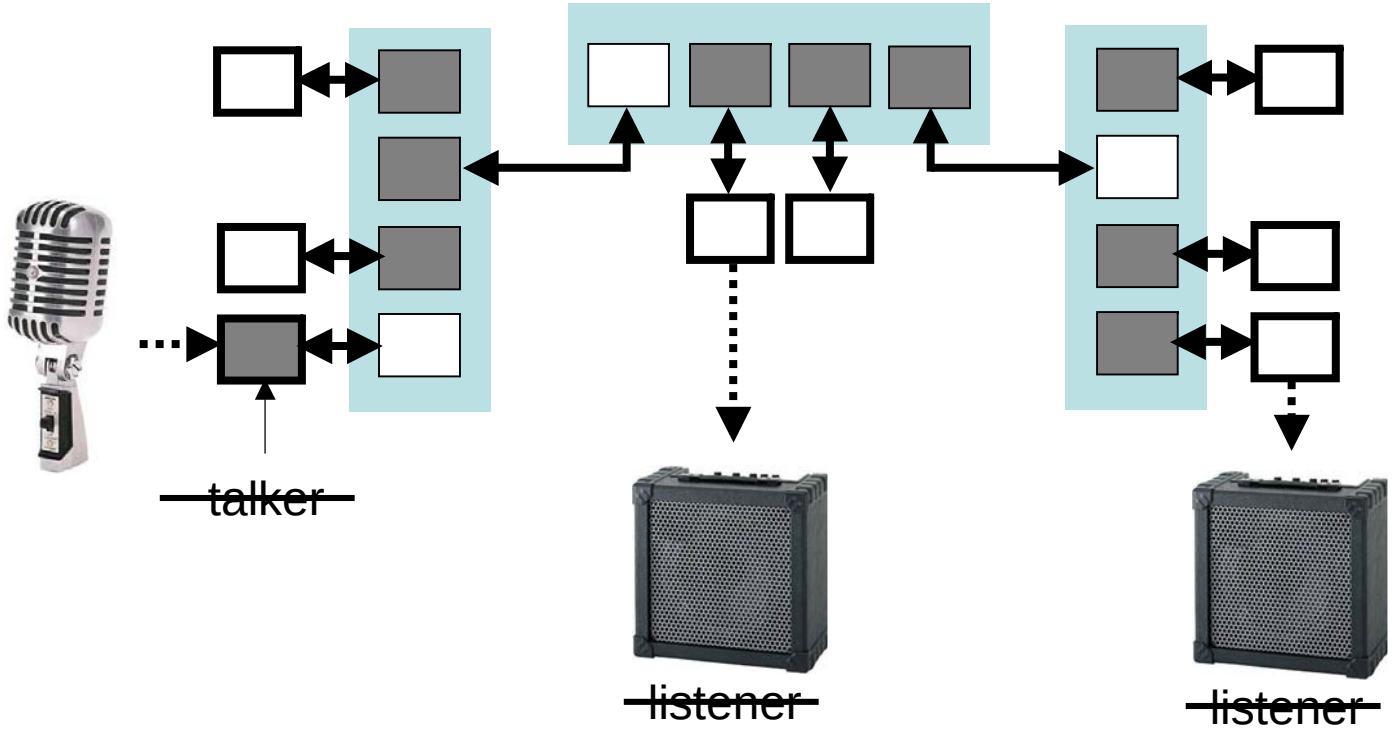
Established paths



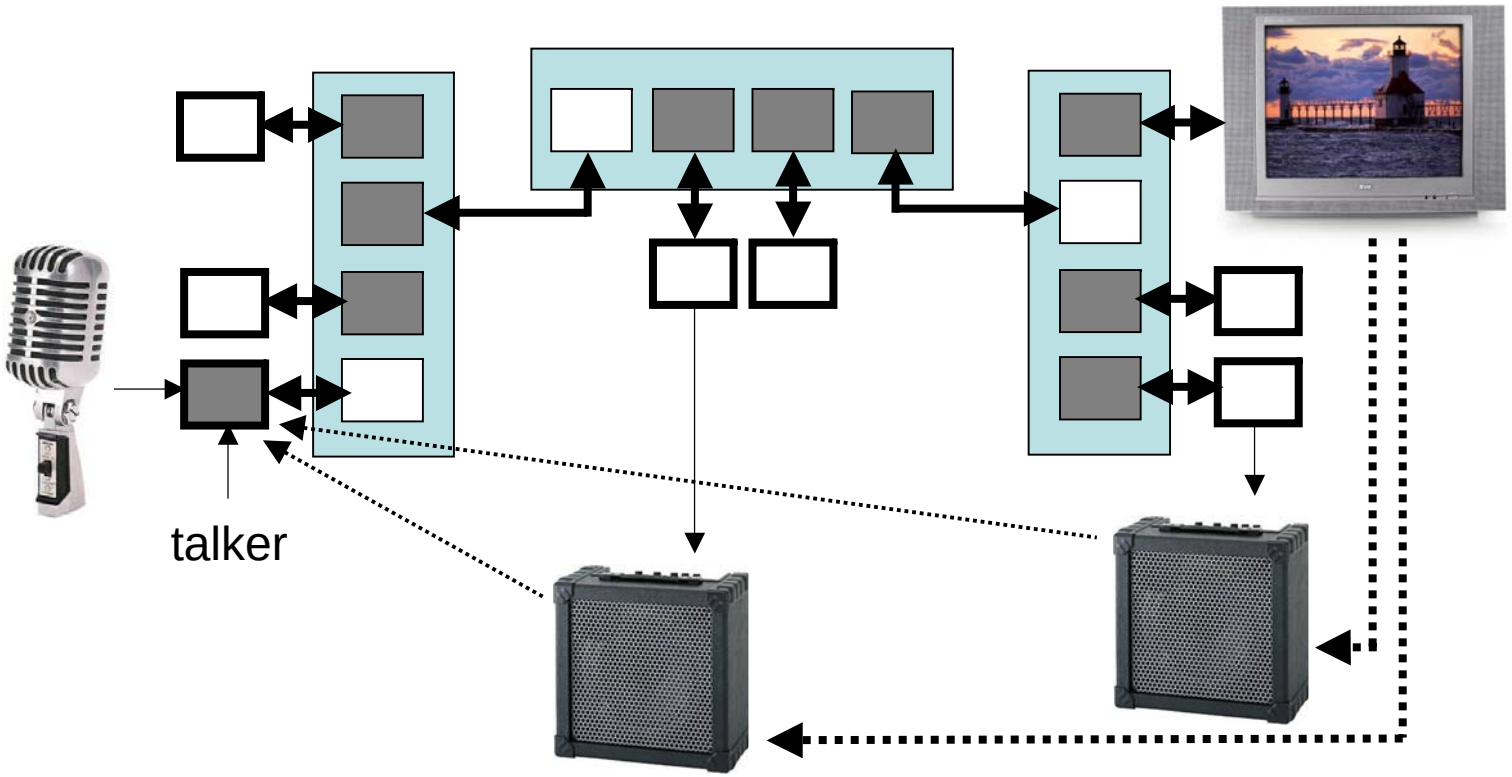
Teardown: talker➔audience



Teardown completed



Third-party activations



Legend:

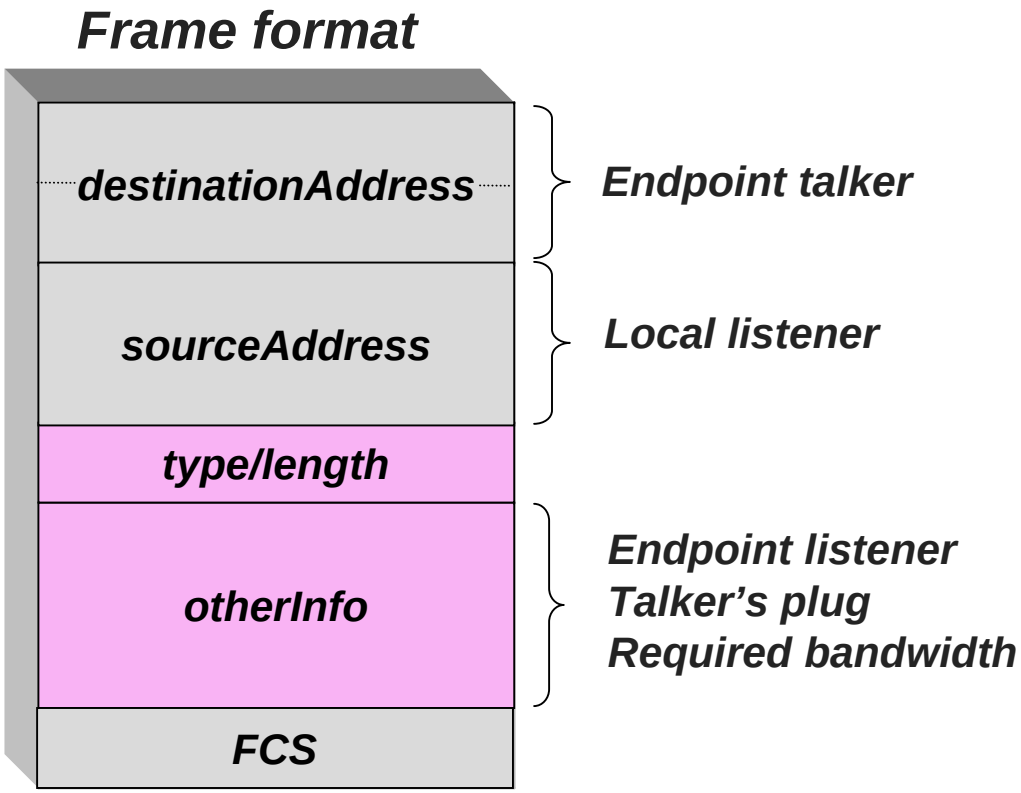
Third-party activates the listener, provides talker's streamID

Listener subscribes to the talker's streamID

Listener-initiated heartbeats (RSVP)

- RSVP based timeout strategy
  - Periodic local listener confirmations
  - Confirmations are really just subscription requests
  - Tolerate single-frame losses
  - Tolerates configuration changes
- Talker “leaves” if no requests observed
- Listener “leaves” if no responses returns
- Talker “responses”
  - Is the flow of isochronous traffic sufficient?

Subscription requests

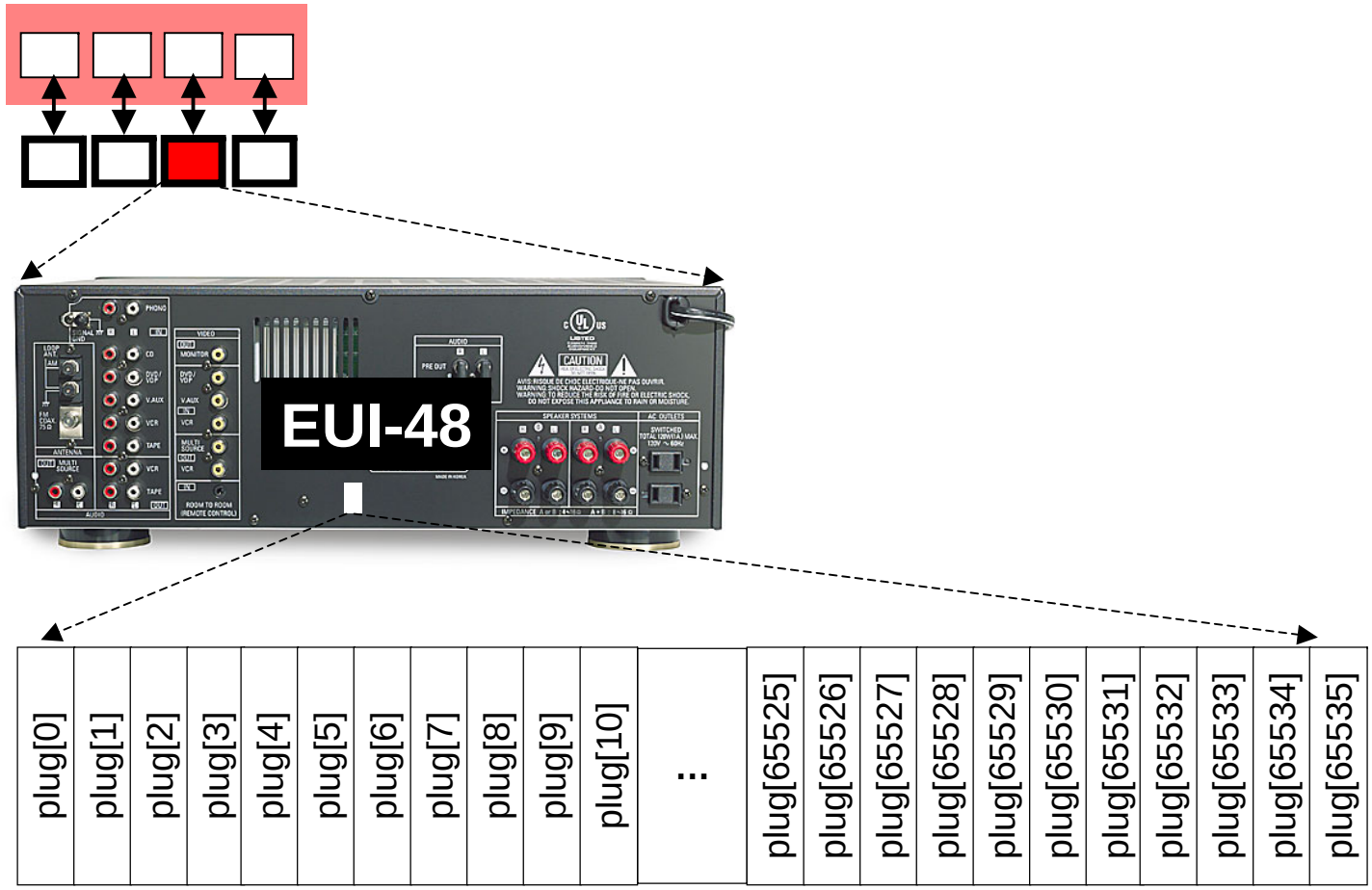


Stream addressing?

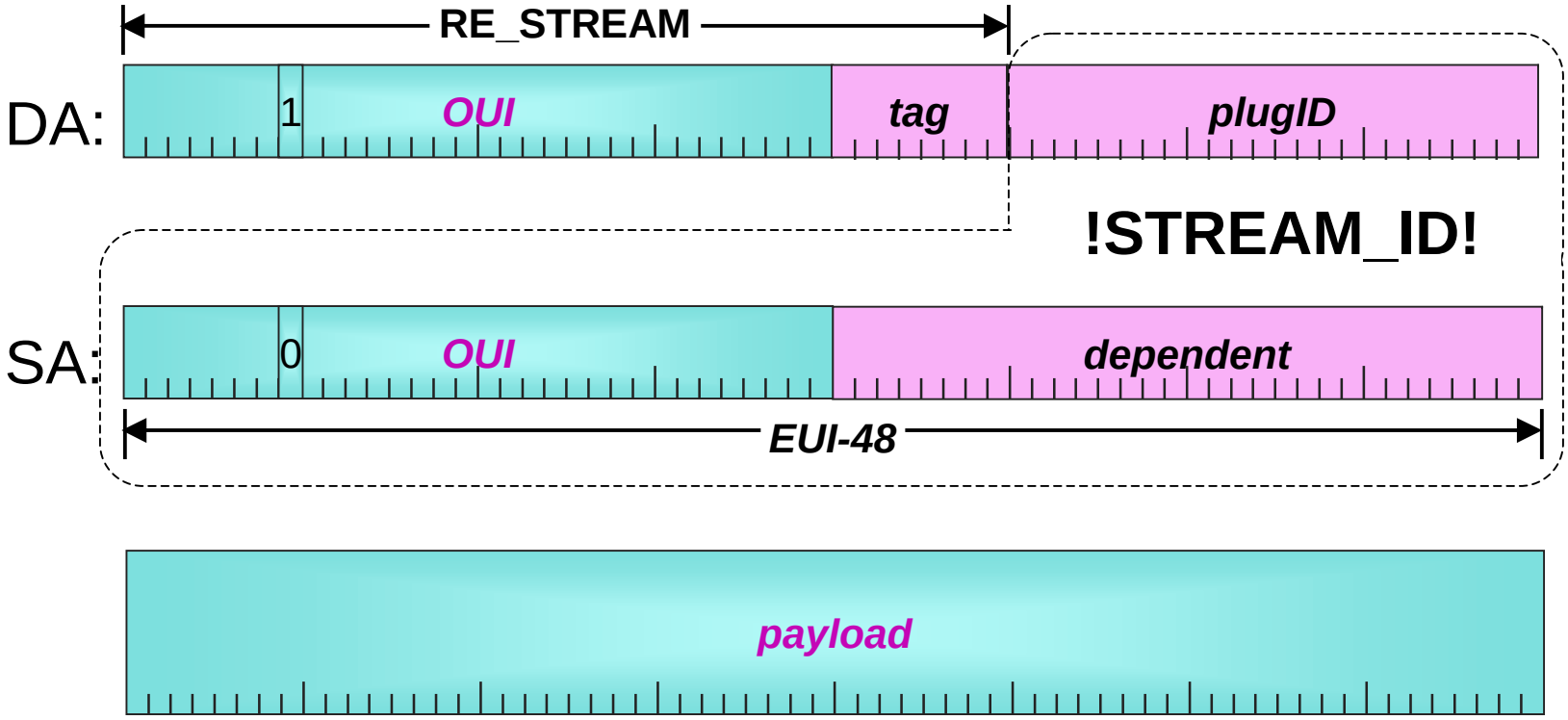
- A New Resource ReSerVation Protocol (RSVP)

  (Again, in the Internet context a data source can be specified by the source host address plus source port number. We only refer to the source host address here.)

What is a streamID?



What is the talker's *streamID*?



Are bridge changes necessary?

- End-station throttling (assuming highest priority)
  - With 1Gb/s switches, this may be sufficient
- Source device spreads transmissions evenly
  - Bunching may be tolerable within the home (needs study)
  - Must ensure that nothing else uses the highest priority
- But, some access control changes needed anyway
  - Bridges are naturally encountered along the path
  - Central topology database is thus unnecessary

Summary

- We need an RSVP-like lower level protocol
  - Restricted to 1-to-N traffic
  - Restricted to “no-filter” streams
- The IP admission control alternative
  - Out of scope
  - Layering violations (non-IP synchronous traffic?)
  - Not generally supported by residential bridges
  - Multiple components (and synchronized use) required:
    - A multicast address server?
    - IGMP/snooping for multicast setup?
    - RSVP/snooping for bandwidth negotiation?

Synchronized time-of-day clocks

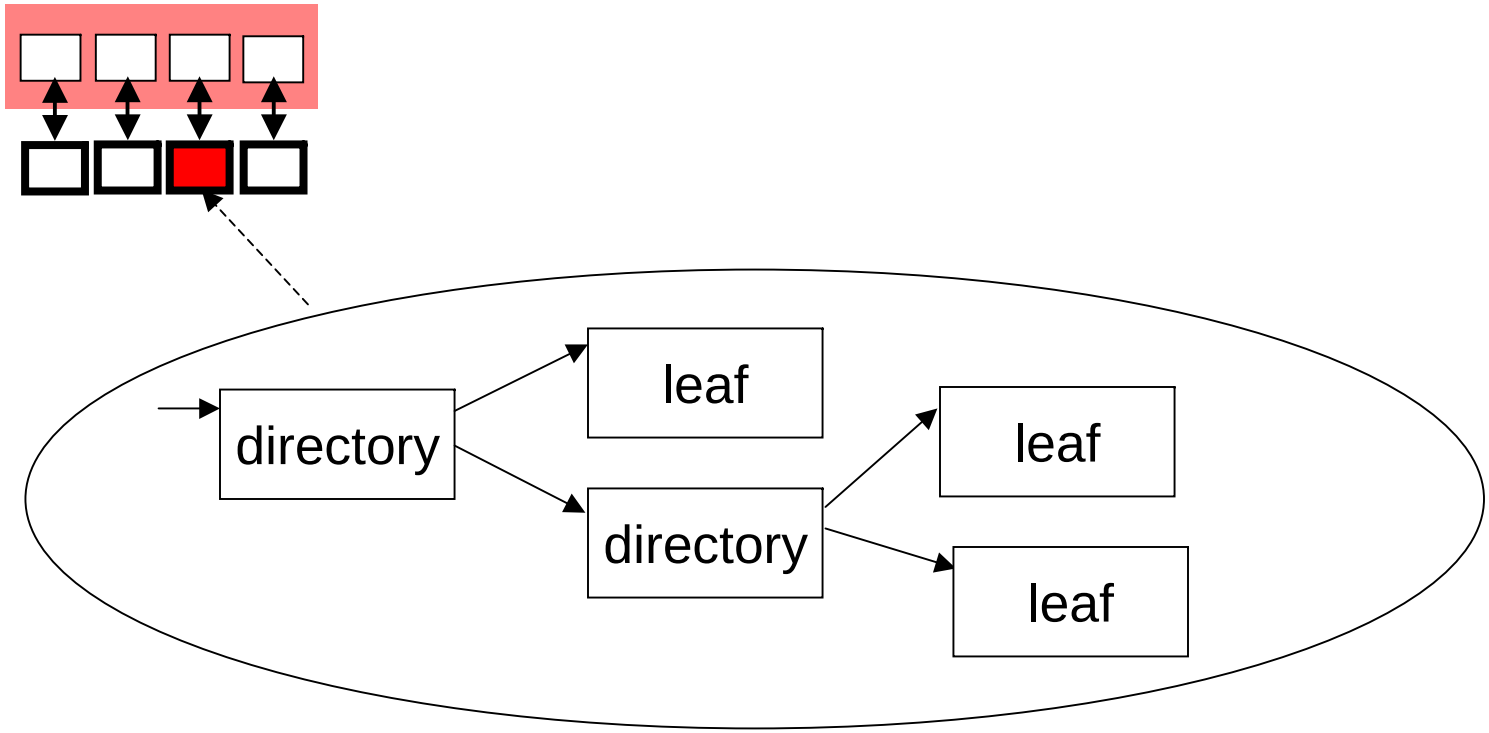
Questions?

Service discovery
(some possibilities)

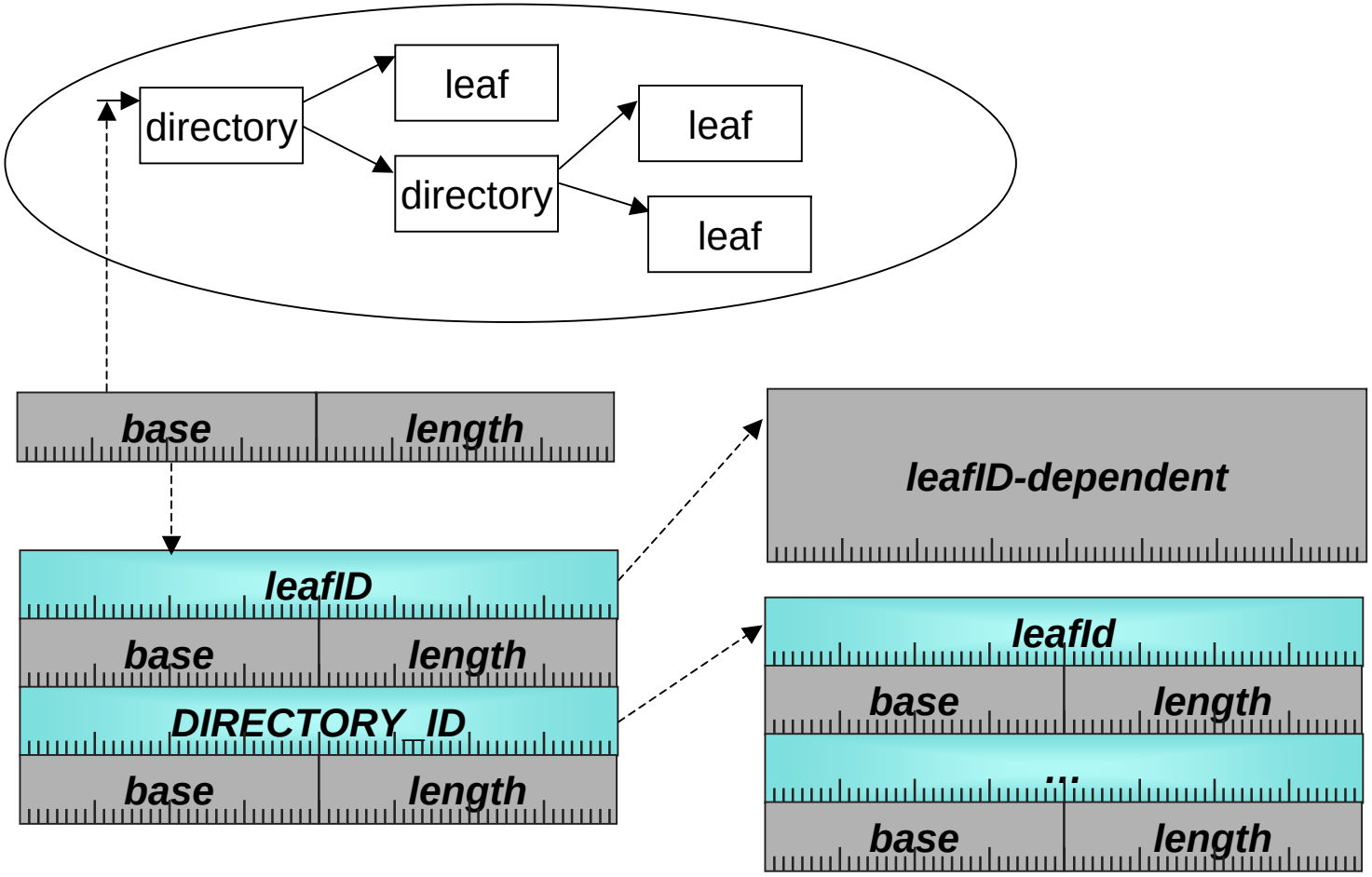
Disclaimer

- "Service discovery" email has been "vocal".
  - But, what is service discovery at Layer 2?
- This presentation clarifies a possible meaning
  - To stimulate and/or resolved internal discussions
- This is _not_ an advocated position!
  - Service discovery is not normally an L2 protocol
  - Higher level protocols have better flexibility&consensus

What is service data?



What is service data?



Summary

- Service discovery could be simple
  - Single frame of information keeps protocol simple
- Service discovery could be elegant
  - Directory structure supports organized data
- Service discovery could be flexible
  - 64-bit identifiers are easily self administered!
  - proprietary information well supported.
  - stealth protocols are easily defined.

Synchronized time-of-day clocks

Questions?

Synchronized time-of-day clocks

Backup slides…

Residential Ethernet

(an unofficial cumulative slide set; 2005Mar11)

Maintained by David V James

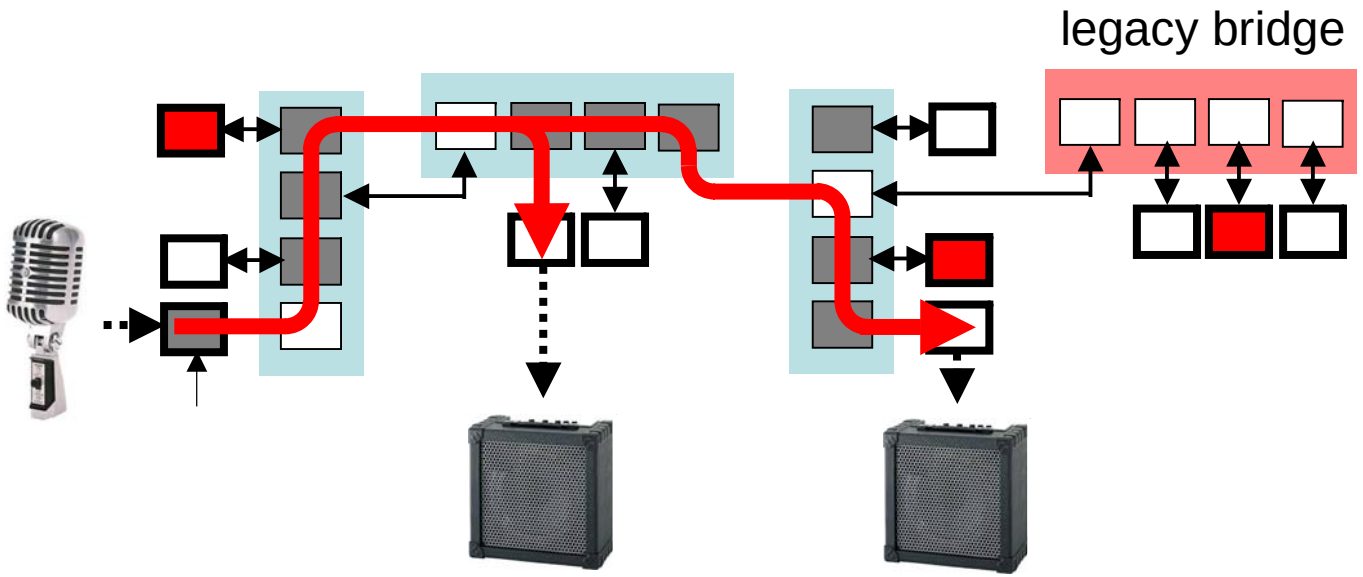
For most recent copy:

http://groups.yahoo.com/group/REInterestGroup

Categories of work

- Service discovery (out of scope)
  - Identify/control “talkers” and their available “plugs”
- Subscription (802.1 centric)
  - Establish conversation between talker and listener(s)
  - Reject unless: *linkBandwidth* < *linkCapacity*
- Clock synchronization
  - Synchronous reception, forwarding, and presentation
- Prioritized queues
  - Talkers and 100Mb bridge ports must be gated
- Formats
  - Frame formats and content (stream IDs, time stamps)
  - Time aware service interfaces

Ethernet compatibility (yes!)



Legend:

legacy bridge

legacy endpoint

Isochronous addressing?