

**BR076R00 (submitted for p1394.1 committee vote):
Peak bandwidth averaging**

Dr. David V. James, Sony
3300 Zanker Road, MS SJ3C1
San Jose, CA 95134-4591
Phone: 408 955-6295
FAX: 408 955-4591
Email: dvj@alum.mit.edu

December 4, 1999

This contribution is one of several, presented for review and incorporation. An overall contribution, which provides an overall context for this and other contributions, is presented in BR047R10.

Philips has presented techniques for more effectively utilizing limited bus-bridge bandwidth by specifying isochronous bandwidth requirements in terms of peak, peak-duration, and average bandwidth parameters.

This submission attempts to refine the Philips proposal, and eliminates the statistical components, as suggested by the p1394.1 working group.

Also, techniques for partitioning surplus delays (to reduce bandwidth requirements) has also been provided. The original concept dealt with the handling of surplus delays in one bridge, but not the distribution of delays to multiple bus bridges.

6.3.5 Bandwidth allocation

6.3.5.1 Isochronous bandwidth parameters

Isochronous connection management uses *maximumBandwidth*, *averageBandwidth*, and *creditLimit* parameters to specify traffic parameters of an isochronous connection. Only the *maximumBandwidth* parameter is used for Serial Bus allocations; the *averageBandwidth*, and *creditLimit* parameters are provided for the benefit of other bandwidth-constrained Surreal interconnects. The rationale for providing additional parameters is twofold:

- 1) Bursty traffic. The type of multimedia traffic expected to pass through bridges may be bursty and the bridge may be able to average the effects of a burst over multiple cycles.
- 2) Limited bandwidth. Some Surreal interconnects may not be able to operate at the full Serial Bus rate, due to physical media or other limitations. Possible examples of limited-bandwidth Surreal interconnects include twisted-pair phone lines as well as wireless RF and IR transmissions.

The *maximumBandwidth* parameter specifies the maximum number of bytes transferred in an isochronous cycle. Specifying the number of bytes (as opposed to quadlets) allows other Surreal interconnects to specify alternate packet-padding schemes.

The *averageBandwidth* parameter specifies the average number of bytes transferred in multiple isochronous cycles. The averaging period is effectively defined by the *bandwidthCredits* parameter, as described in the remainder of this subclause.

The *creditLimit* parameter specifies the extent (in bytes) to which isochronous traffic can exceed the specified average value. Isochronous traffic may have peaks above and valleys below the specified average value, as illustrated in figure 1.

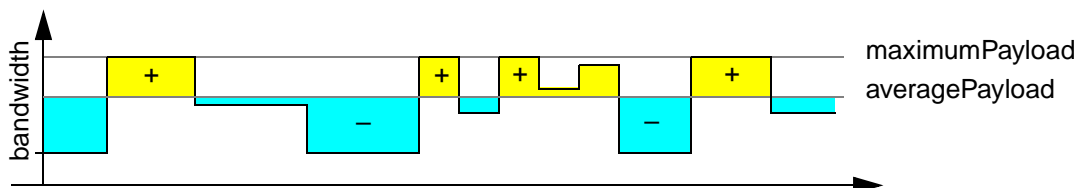


Figure 1—Nonoverlaid connection acquisition

A *runningCredits* value is (in concept) increased or decreased in each cycle, based on the difference between the actual and average bandwidths. The *creditLimit* value specifies a maximum *runningCredits* value that is never exceeded. This is more accurately defined by equation 1, computations that could (in theory) be performed in each isochronous cycle.

```
runningCredits=Minimum(0,runningCredits-(payload-averagePayload));  
assert(payload<maximumPayload && runningCredits<creditLimit);
```

 (1)

Not all implementations are not expected to compute a *runningCredits* value, but many are expected to allocate isochronous bandwidth based on knowledge of their averaging buffer size and the traffic pattern's *creditLimit* parameter.

With these parameters, the isochronous bandwidth allocation on some Surreal interconnects may depend on the number of delays introduced by the bridge. For this reason, connection protocols allow the application to

specify acceptable delay values and device the excess delays among the bridges, as described in the following subclauses.

6.3.5.2 Delay sampling

The connection is established by sending messages in the listener-to-talker direction, as illustrated in figure 2. At each step, pilot proxies allocate bandwidth based on the shortest-possible delay, in steps (2) and (5).

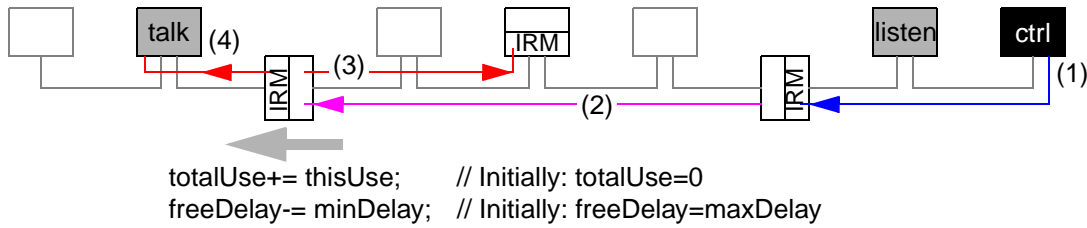


Figure 2—Accumulation of delay preferences

In addition to allocating their local local isochronous resources, bridges increase the message's *totalUse* value by their delay-use. The delay-use value equals the number of internal bandwidth units saved if an additional delay were to be provided. The intent is to allocate residual delays in a weighted fashion, where a bridge's preference is proportional to the benefits derived from additional delay allocations.

When the connection message reaches the talker-proxy, the message has accumulated the *totalUse* values from intermediate bridges. The connection confirmation flows in the talker-to-listener direction, as illustrated in figure 3.

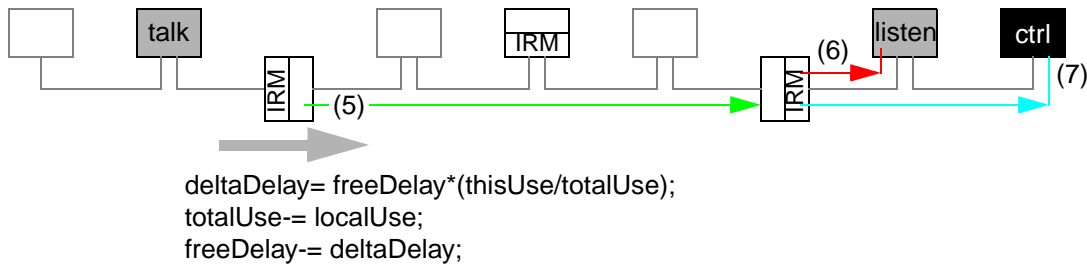


Figure 3—Delay allowance adjustments

Confirmation messages distribute the excess delay allowances, which can enable releases of internal isochronous resources in intermediate bridges (not illustrated). The additional delay allocation, *deltaDelay*, is allocated based on the bridges delay-use parameter. Before the message is forwarded, its *totalUse* and *freeDelay* parameters are adjusted to be reflect this bridge's contribution.