

# PtsCase20

## Problem in aggregating data streams

### Background

As part of radio astronomy applications testing, 512 Mbps real-time data were transmitted (streamed) from Jodrell Bank, UK and Onsala, Sweden to Metshovi, Finland. The streams were simultaneous and over the normal Internet; no tuning or jumbo frames were used. One week later, the astronomers tried a 1 Gbit/s Ethernet to transfer two data streams with an aggregate speed 896 Mbit/s (one 512Mbps and one 338Mbps). They observed some problems: from point-to-point they could transfer 940 Mbps test traffic without errors. Same goes for two data streams that don't share ports. But when they tried to aggregate two data streams to the same link, they saw significant packet loss. The equipment they use is a Extreme Networks Summit 450 and their computers were cheap nForce4-based or Dell office computers. They are using a modified realtime version of the tsumani protocol that is a rate-based udp protocol.

### Outcome

The problem is well known, and described at [NetworkBufferSizing](#). In brief, when you transmit large amounts of data over a link with a large bandwidth-delay product, you must have big buffers along the path to accommodate the possibility of temporary congestion. Current router-based networks are designed that way, but in the access sites often this is forgotten. The exact buffer dimension is still being discussed, anyway it is generally agreed that they should be as big as possible. Suffice to say that with more than one stream, there is the possibility that the intrinsic bursty nature of TCP streams can fill the tiny buffers installed in low-end, cheap, workgroup switches, thus resulting in erratic packet loss. The only solution is to use a more expensive switch (yes, fast memory is expensive...) and replace the cheap one.

– Main.TobyRodwell - 31 Jan 2007