

Re: CSMA

Source: <http://unix.derkeiler.com/Newsgroups/comp.unix.programmer/2005-09/0599.html>

From: Alex Fraser (*me_at_privacy.net*)

Date: 09/25/05

Date: Sun, 25 Sep 2005 11:26:14 +0100

<rishishah@patni.com> wrote in message
news:1127596352.312029.273640@z14g2000cwz.googlegroups.com...
> *I am implementing a project in c++/unix in which I need to simulate a
> CSMA MAC layer.*
> *The project is a simulation of peer to peer network where is each peer
> is a process on some unix machine. Though each client has a separate
> TCP communication line, I have to assume that they have a common line
> and each of the process has to follow the CSMA mechanism while
> transmitting that is if there is a collision while transmitting then
> both the processes have to revert back and transmit after some random
> interval of time. I have to assume that the entire network is running
> on a 9.6kbps line. Each peer notifies every other client if it sends a
> packet on the common line.*
> *I am struggling as to how to design such a network where I have to
> detect a collision since there is no delay in the unix socket
> communication. One idea could be to send the packet to a recipient,
> wait for x msec and then send a signal to the recipient to go ahead. If
> the sender receives a packet from some other source during the xmsec
> wait time then it will not send the signal and the recipient will
> discard the packet.*

I presume you intend to handle the case where A sends to B at the same time as C sends to D.

Note that AIUI, in a half-duplex Ethernet network compliant with the specification, a collision anywhere will always be detected by the senders. Thus the situation Pascal described where the packets cross on the transmission line does not occur. The guarantee is provided by specifying that the minimum transmission time (implied by way of a minimum frame size) is longer than the maximum propagation time between any two nodes. I think simulating this simplifies matters.

Another thing that would almost certainly simplify the problem (timing issues in particular) is having all transmissions go via a "hub" process, if that is an option.

You might be better off asking about this in another group such as

comp.unix.programmer: Re: CSMA

comp.programming since the difficulties are not Unix-specific.

Alex