Announcements

- Program #2 regrades requests due by 11:00 today
- HW #2 was collected at the start of class
- Reading
  - Today: 6.3-6.4
  - Thursday: 3.1
Sliding Window Protocol

- **Need to**
  - have multiple outstanding packets
  - limit total number of outstanding packets
  - permit re-transmissions to occur

- **Sliding Window**
  - permit at most N outstanding packets
  - when packet is ACK’d advance window to first non-ACK’d pkt

- **Retransmission**
  - Go-back N
    - when a packet is lost, restart from that packet
    - provides in-order delivery, but wastes bandwidth
  - Selective Retransmission
    - use timeout to re-sent lost packet
    - use NACK as a **hint** that something was lost
Sliding Window Example

From: *Computer Networks*, 3rd Ed. by Andrew S. Tanenbaum, (c)1996 Prentice Hall.
Buffer Management

● **Unreliable Network**
  – sender must buffer all un-acked packets
  – receiver can buffer if space is available
    • if not, drop packet and wait to re-transmission

● **Buffer Size**
  – does one size fit all?
    • are TPDUs of uniform size?
  – might use a fixed size buffer smaller than max TPDU
    • requires support for multiple buffers per TPDU

● **Possible to decouple buffer allocation from window**
  – ACKs contain both buffer credits and ACKSs

● **Buffer Copies**
  – possible for each layer to copy the buffer, but this is slow
  – handoff pointers to data, but requires coordination between layers
Multiplexing in the Transport Layer

- **Upward multiplexing**
  - putting multiple transport connections onto one network connection
  - used to accommodate pricing strategies that charge for connections

- **Downward multiplexing**
  - using several network connections per transport connection
  - permits use of multiple copies of network resources
    - if the network layer uses sliding windows
      - a high latency network may under utilize the link
      - multiple connections each get a window
    - per connection buffer allocation
      - get more buffers
    - round-robin scheduling
      - get a larger share of link bandwidth
Crash Recovery

- **Router or Link Crashes**
  - Data in transit can be lost.
  - End nodes have sufficient state to recover lost data.
  - Transport protocol can hide network failures from the application.

- **Host Crashes**
  - Transport level state will be lost at one end.
  - Does the transport layer have sufficient info to recover?, **No!**
    - Information must flow down to network and up to transport user
      - ACKs go down, and data goes up.
      - It is not possible to make these two operations atomic.
    - lack of stable storage causes this problem
  - Result, higher up layer must deal with host crashes
Protocol State Machines

<LISTEN, P1>
<CONNECT, P1>
<LISTEN, ~P2>, A2

Idle

<Connect, ~P1>, A3
<timeout, *>, A4
<clear_req, *>, A5
<call_req, ~P3>, A6
<call_req, P3>, A7
<LISTEN, P2>, A8

Waiting

<Clear_req, *>, A9
<LISTEN, ~P2>, A10
<Call_acc, *>, A11

Established

<SEND, P5>, A12
<Call_acc, *>, A13
<Clear_req, *>, A14
<LISTEN, *>
<clear Req, *>, A15
<data, *>, A16
<clear Conf, *>

Sending

<DISCON, ~P4>, A17
<Clear_req, *>, A18
<credit, *>, A19

Receiving

<DISCON, P4>, A20
<Clear_req, *>, A21
<credit, *>, A22

Disconnecting

<SEND, P5>, A23
<Clear_req, *>, A24
<credit, *>, A25

Queued

<timeout, *>, A26
<Clear_req, *>, A27
<credit, *>, A28
## Predicates And State Transitions

<table>
<thead>
<tr>
<th>Pred</th>
<th>Meaning</th>
<th>Act</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>P1</td>
<td>Connection table full</td>
<td>A1</td>
<td>Send Call_acc</td>
</tr>
<tr>
<td>P2</td>
<td>Call_req pending</td>
<td>A2</td>
<td>Wait for Call_req</td>
</tr>
<tr>
<td>P3</td>
<td>LISTEN Pending</td>
<td>A3</td>
<td>Send Call_req</td>
</tr>
<tr>
<td>P4</td>
<td>Clear_req Pending</td>
<td>A4</td>
<td>Start Timer</td>
</tr>
<tr>
<td>P5</td>
<td>Credit Available</td>
<td>A5</td>
<td>Send Clear_conf</td>
</tr>
<tr>
<td></td>
<td></td>
<td>A6</td>
<td>Send Clear_req</td>
</tr>
<tr>
<td></td>
<td></td>
<td>A7</td>
<td>Send message</td>
</tr>
<tr>
<td></td>
<td></td>
<td>A8</td>
<td>Wait for credit</td>
</tr>
<tr>
<td></td>
<td></td>
<td>A9</td>
<td>Send Credit</td>
</tr>
<tr>
<td></td>
<td></td>
<td>A10</td>
<td>Set Clr_req_recv_flag</td>
</tr>
<tr>
<td></td>
<td></td>
<td>A11</td>
<td>Record credit</td>
</tr>
<tr>
<td></td>
<td></td>
<td>A12</td>
<td>Accept message</td>
</tr>
</tbody>
</table>