

Computer Communications - CSC1 551

Copyright © William C. Cheng

2

### Sender Reliable Error Detection

- Wait for ACKs from all receivers, re-send on timeout or selective ACK
- (+) easy resource management
- (-) wait for ACK
- (-) receiver state in sender, not scalable
- (-) ACK implosion

Computer Communications - CSC1 551

Copyright © William C. Cheng

4

### Implosion

Computer Communications - CSC1 551

Copyright © William C. Cheng

6

### Implosion (Cont...)

Computer Communications - CSC1 551

Copyright © William C. Cheng

1

# CS551

## Reliable Multicast

Bill Cheng

<http://merlot.usc.edu/cs551-f12>

Computer Communications - CSC1 551

Copyright © William C. Cheng

3

### Receiver Reliable Error Detection

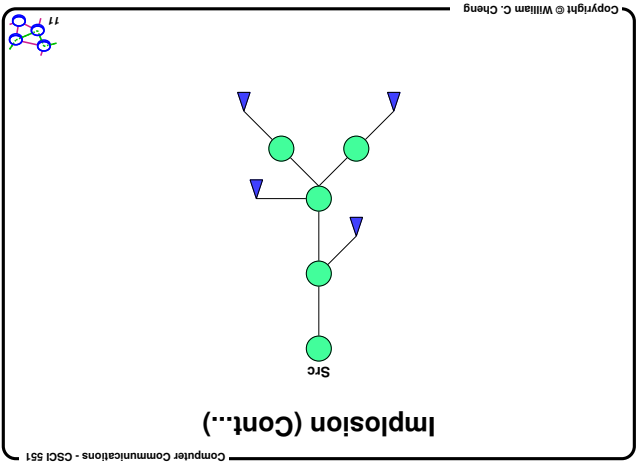
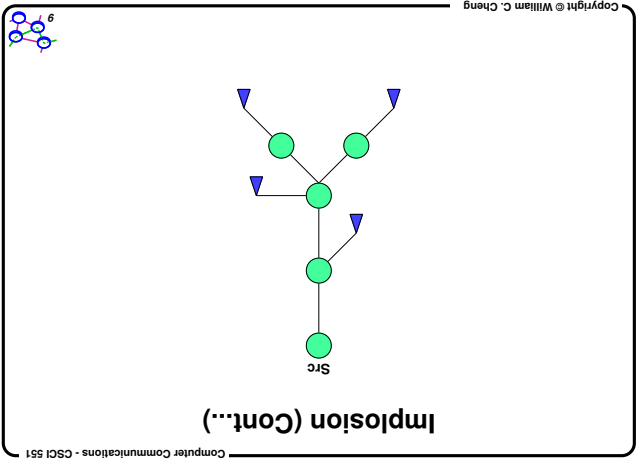
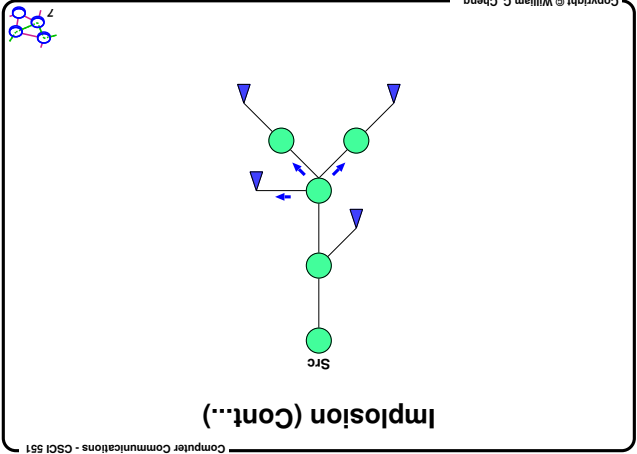
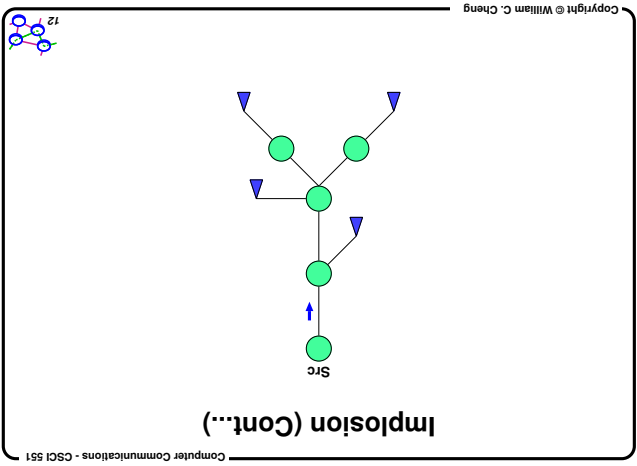
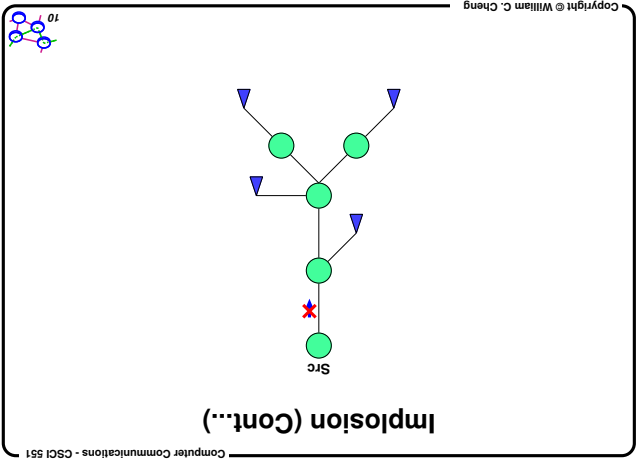
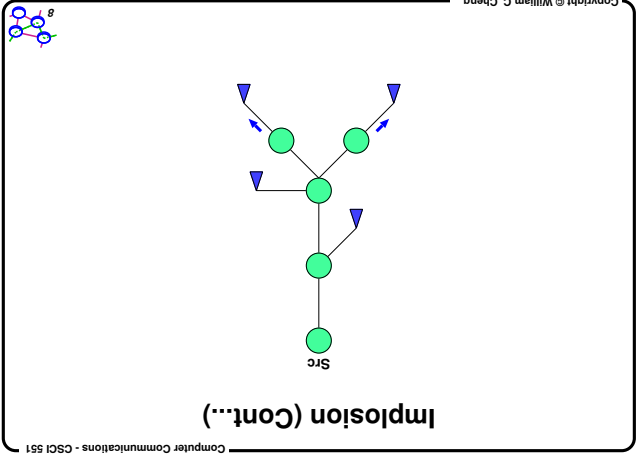
- Receivers NACKs lost packets
- (+) no state at sender - good for multicast
- (-) does not provide 100% reliability
- (-) NACK implosion

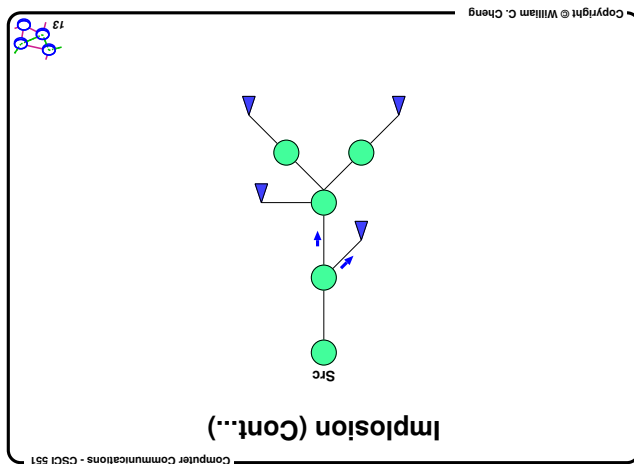
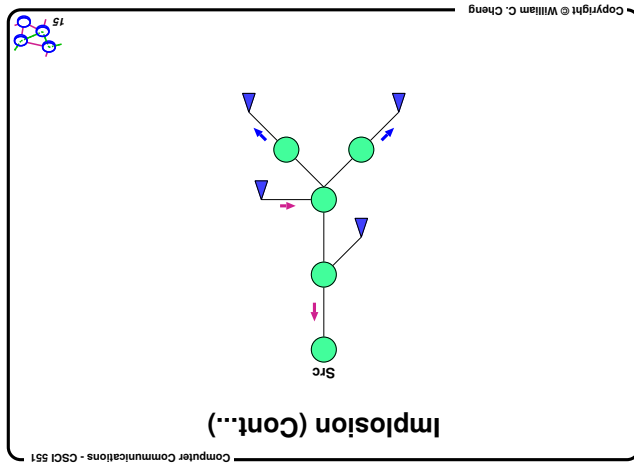
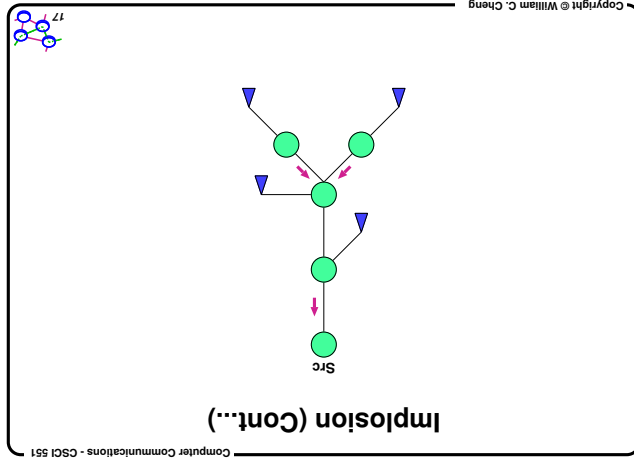
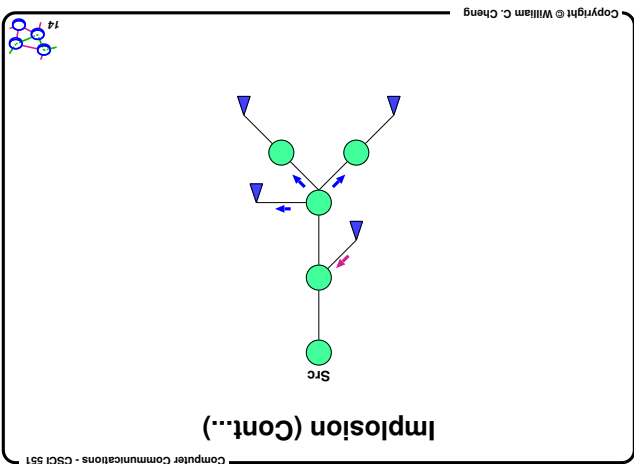
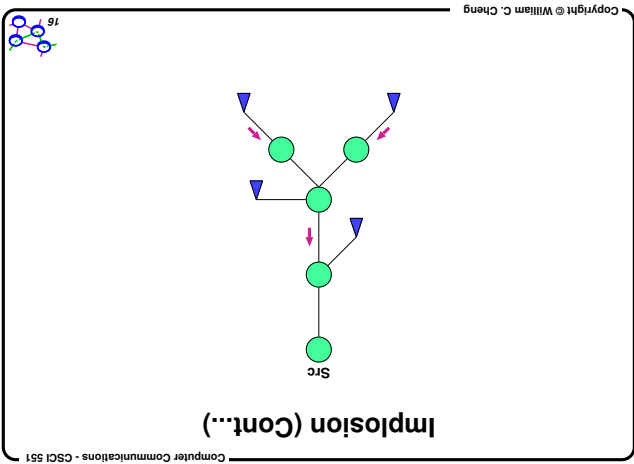
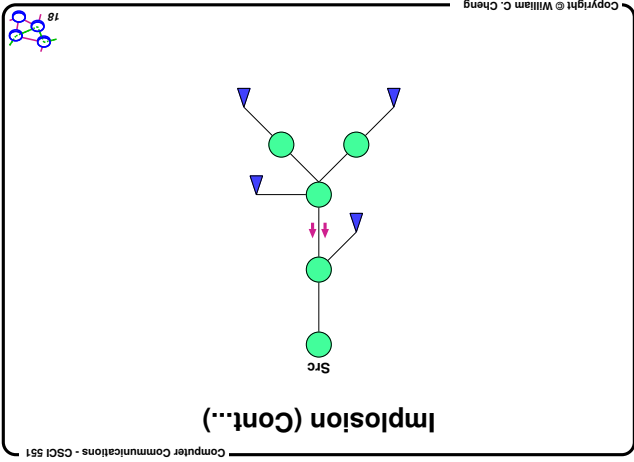
Computer Communications - CSC1 551

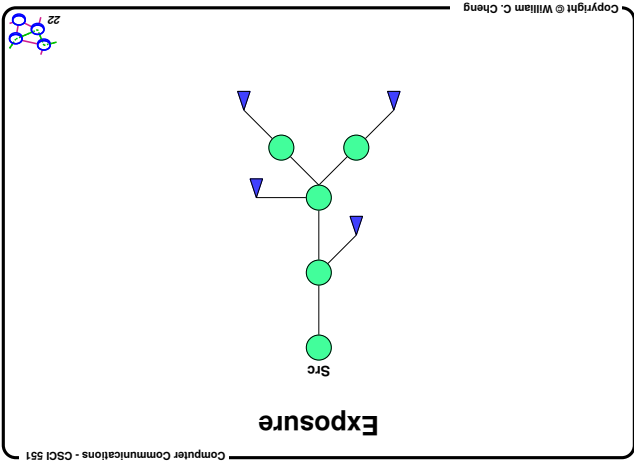
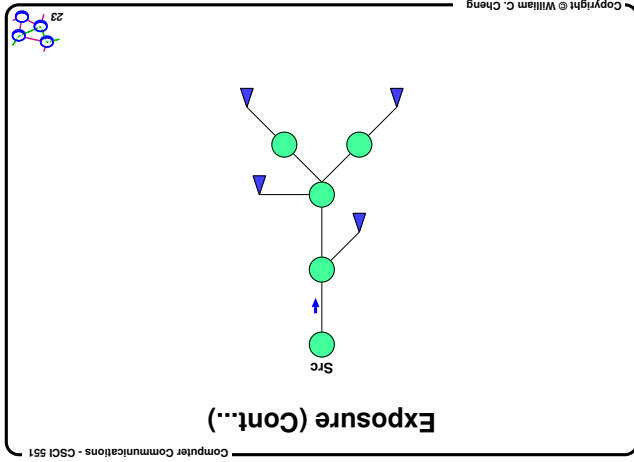
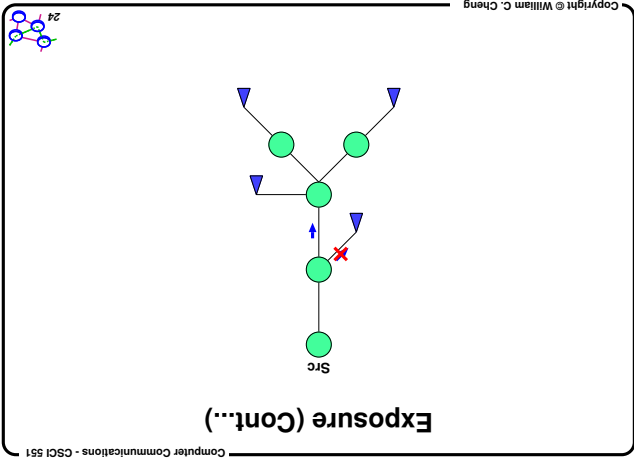
Copyright © William C. Cheng

5

### Implosion (Cont...)







Copyright © William C. Cheng

Re-transmitter = sender

= receiver

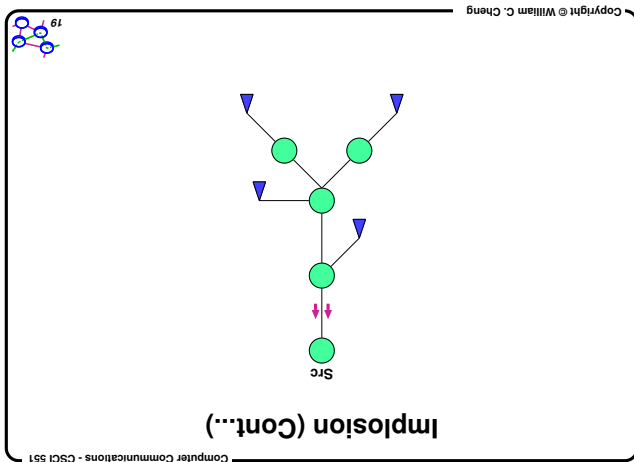
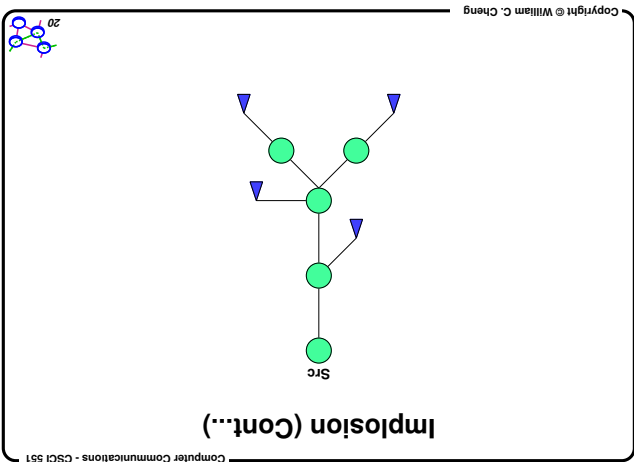
How to retransmit?  
= unicast, multicast, scoped multicast, retransmission group, etc.

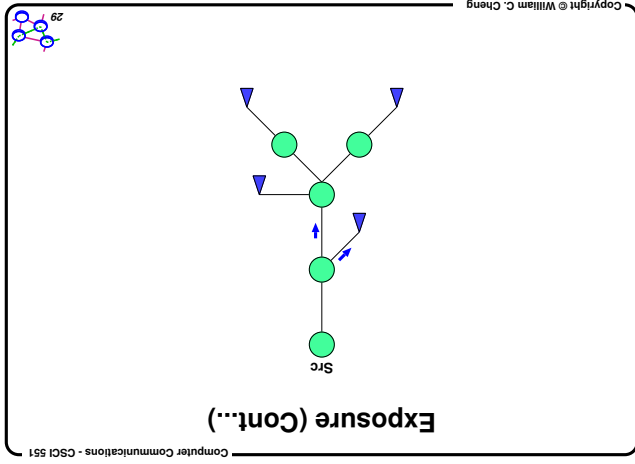
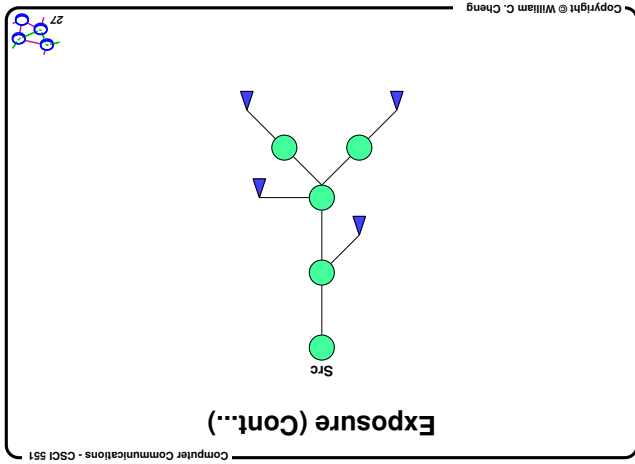
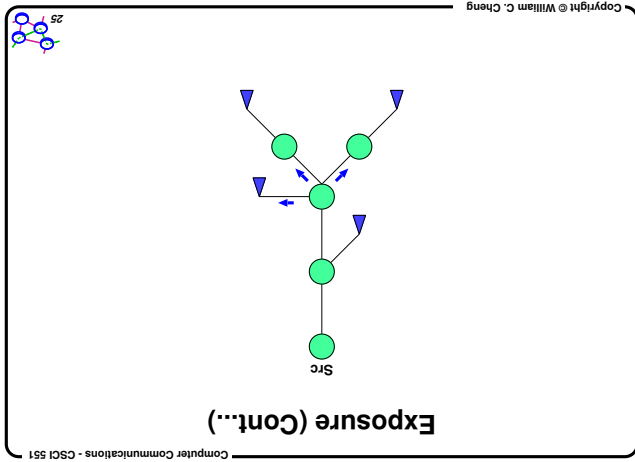
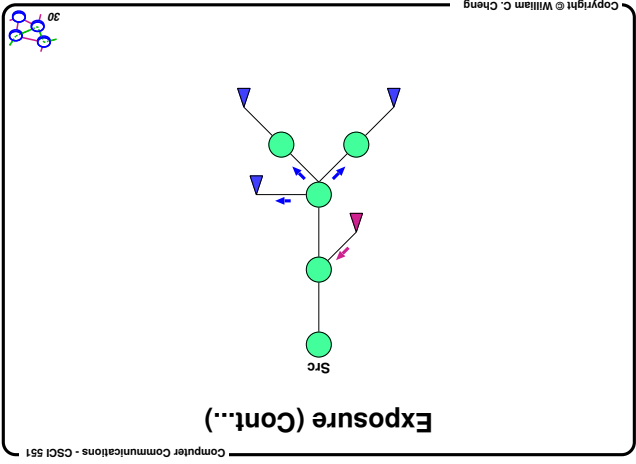
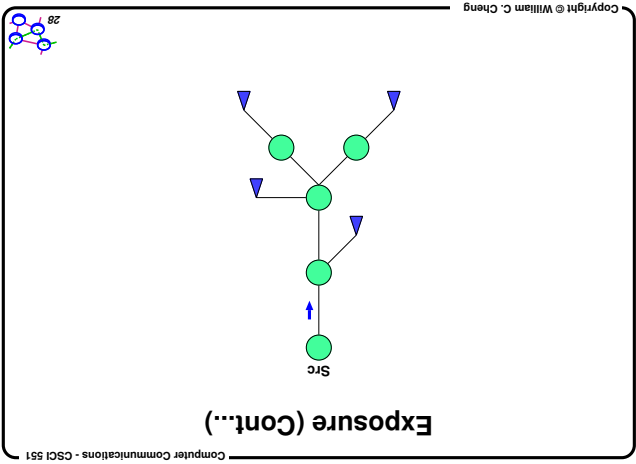
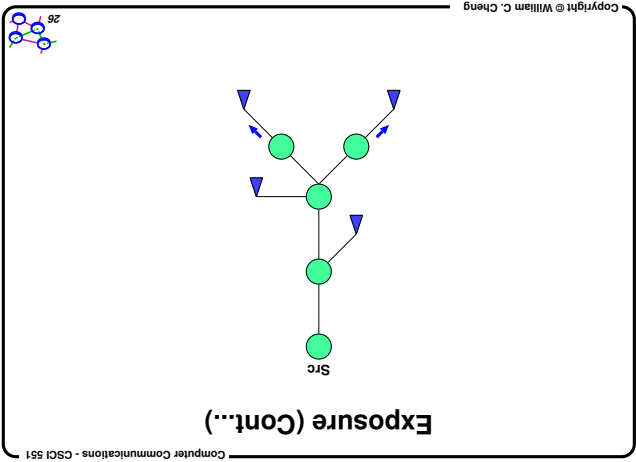
Problem with sender retransmissions = exposure

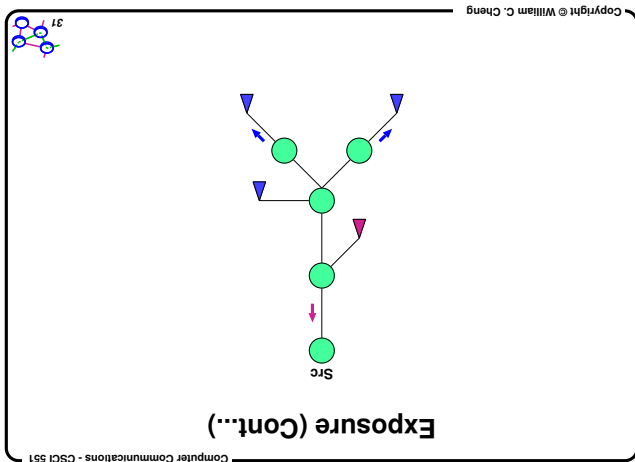
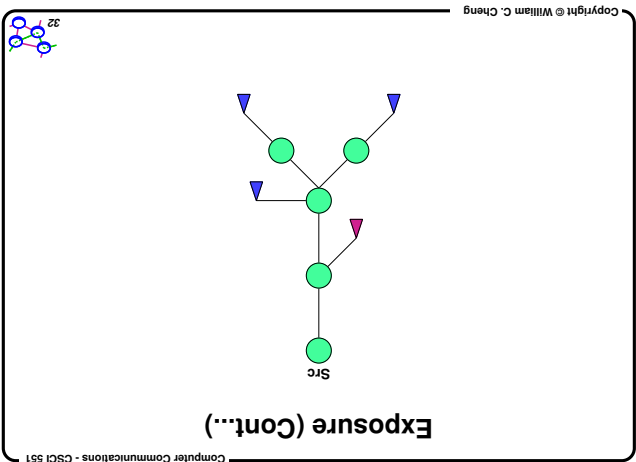
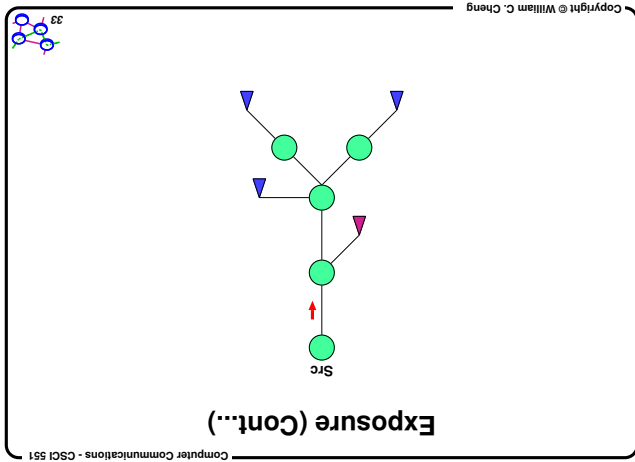
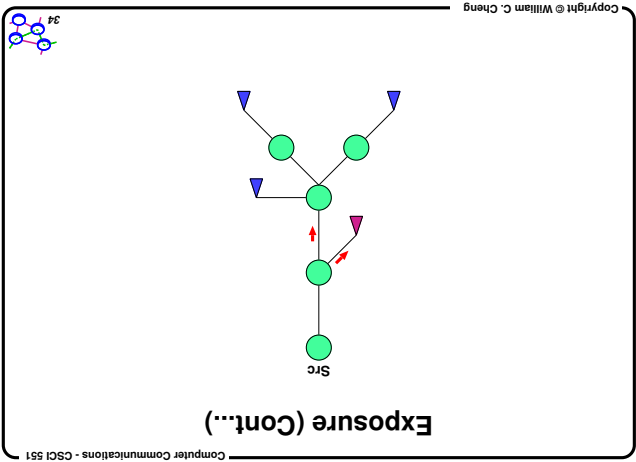
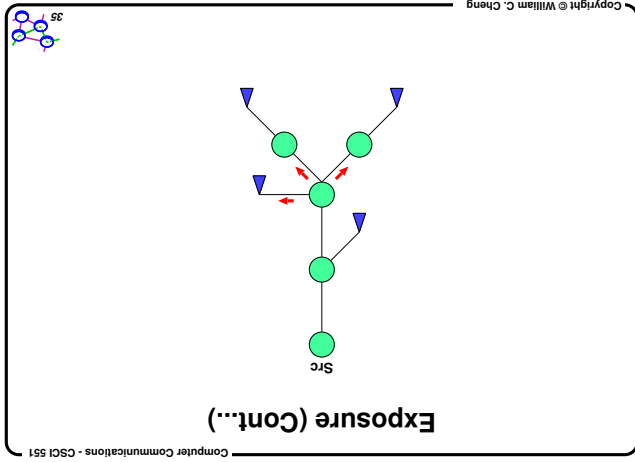
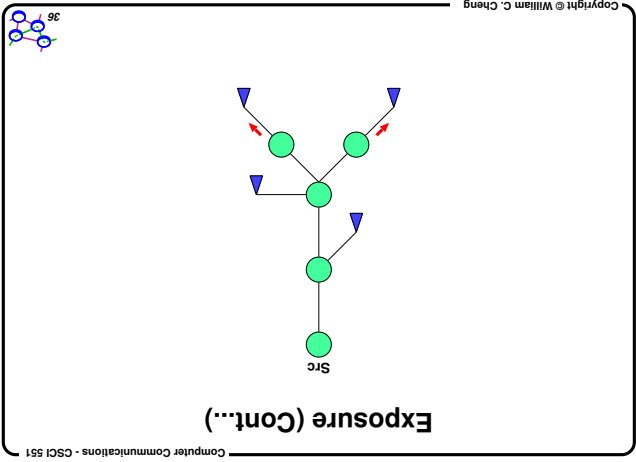
Retransmission

Computer Communications - CSCI 551

21







Copyright © William C. Cheng

37

**Exposure (Cont...)**

- ↳ Routers do transport level processing
- ↳ buffer packets
- ↳ fuse ACKs
- ↳ send retransmissions
- ↳ this solves implosion and exposure problems, but:
  - not scalable
  - violate end-to-end argument

Computer Communications - CSCI 551

Copyright © William C. Cheng

40

**SRM (Scalable Reliable Multicast)**

- ↳ Originally designed for wb
- ↳ Receiver reliable
- ↳ NACK-based
- ↳ Every member may multicast NACK or retransmission
- ↳ No assistance from routers

Computer Communications - CSCI 551

Copyright © William C. Cheng

42

**SRM Request Suppression**

Computer Communications - CSCI 551

Copyright © William C. Cheng

39

**CS51**

**Scalable Reliable Multicast**

**[Floyd97c]**

**Bill Cheng**

<http://merlot.usc.edu/cs51-f12>

Computer Communications - CSCI 551

Copyright © William C. Cheng

41

**SRM Retransmission**

- ↳ All transmissions go to everyone
- ↳ Receivers notice losses
- ↳ how?
  - if there is a missing sequence number from subsequence data
  - use session messages to deal with loss of last packet
- ↳ Losses result in repair requests (to everyone)
- ↳ Repair requests produce repairs (to everyone)
- ↳ Requests and repairs are spaced and avoided
- ↳ via randomization and suppression

Computer Communications - CSCI 551

Copyright © William C. Cheng

48

When noticing skipped sequence number:  
 → start a timer whose timeout is proportional to distance from Src

SRM Request Suppression (Cont...)

Computer Communications - CSCI 551

Copyright © William C. Cheng

47

When noticing skipped sequence number:  
 → start a timer whose timeout is proportional to distance from Src

SRM Request Suppression (Cont...)

Computer Communications - CSCI 551

Copyright © William C. Cheng

46

SRM Request Suppression (Cont...)

Computer Communications - CSCI 551

Copyright © William C. Cheng

45

SRM Request Suppression (Cont...)

Computer Communications - CSCI 551

Copyright © William C. Cheng

44

SRM Request Suppression (Cont...)

Computer Communications - CSCI 551

Copyright © William C. Cheng

43

SRM Request Suppression (Cont...)

Computer Communications - CSCI 551





## SRM Summary

- ↳ NACK/Retransmission suppression
  - = delay before sending
  - = delay based on RTT estimation
  - = deterministic and stochastic components
- ↳ Periodic session messages
  - = discover lost final packets
  - = used to estimate OTT from sender to receivers
- ↳ Adaptive algorithm to adjust constants

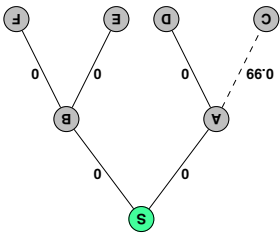
Copyright © William C. Cheng



## What's Missing?

- ↳ Loses at link (A,C)
  - causes retransmission to the whole group
- ↳ Better: only retransmit to those members who lost the packet
  - = local recovery
  - = router support
  - = A to C
  - = where have we seen this before?
- ↳ SNOOP

Copyright © William C. Cheng



Computer Communications - CSCI 551