CS2105 Lecture 3

Reliable Protocol

28 January, 2013

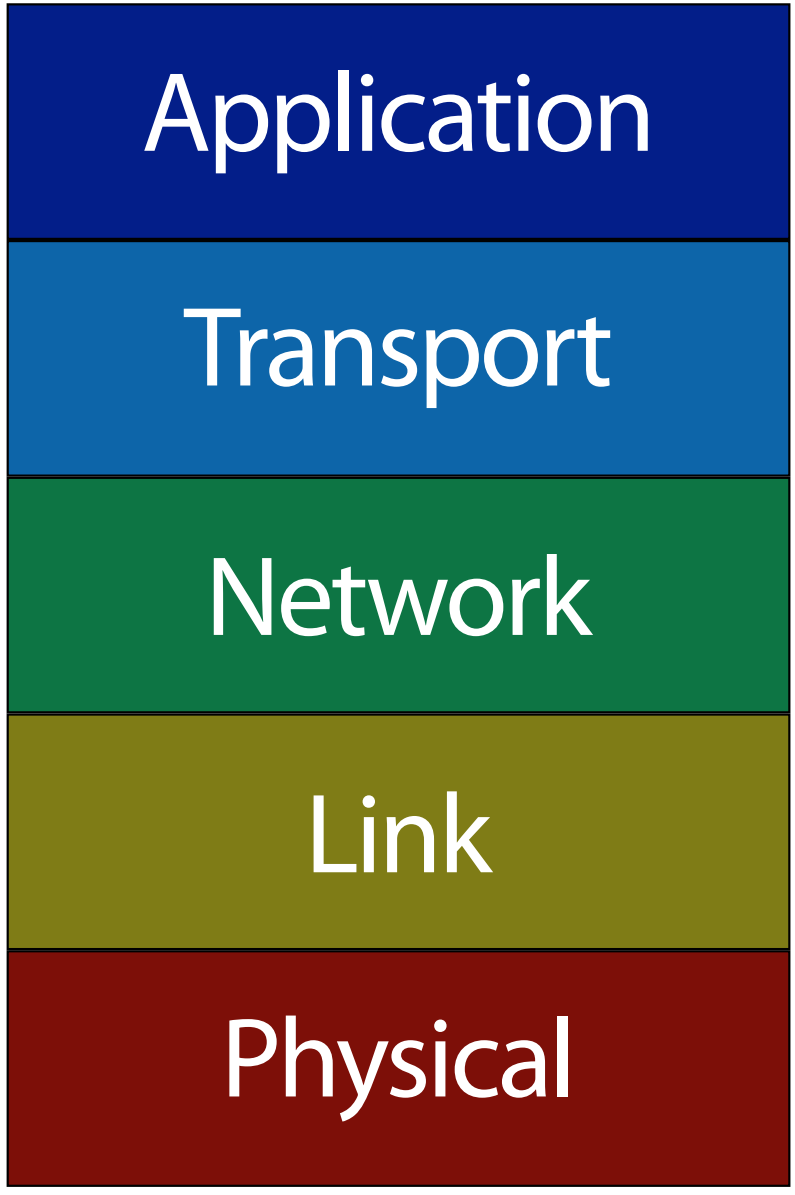
CS2105 Lecture 3

Reliable Protocol

28 January, 2013

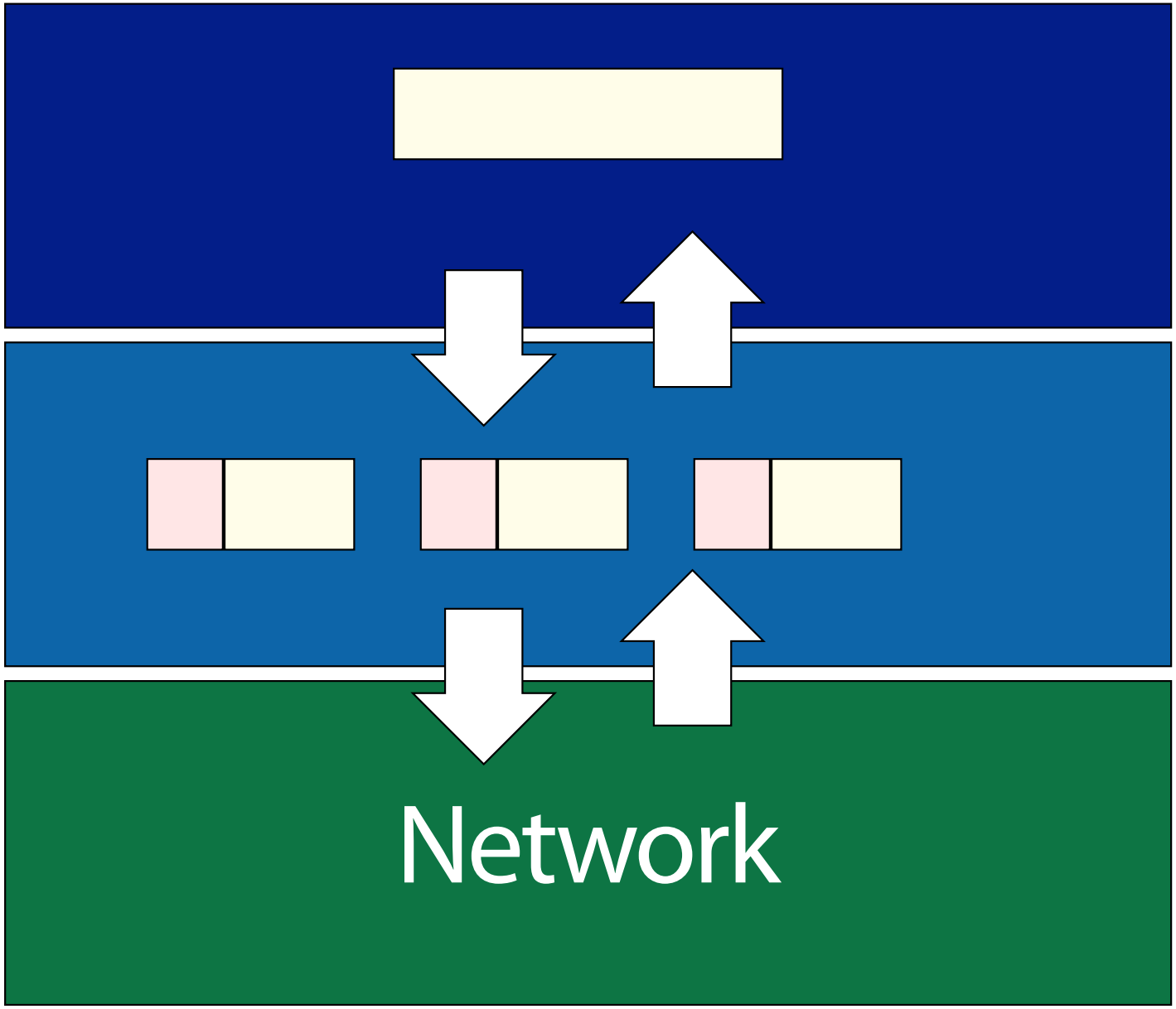
After this class, you are expected to:

- understand the interface between the transport layer and the network layer
- be able to design your own reliable protocols with ACK, NAK, sequence numbers, timeout, and retransmission.
- know how to calculate the utilization of a channel.


Application
Transport
Network
Link
Physical

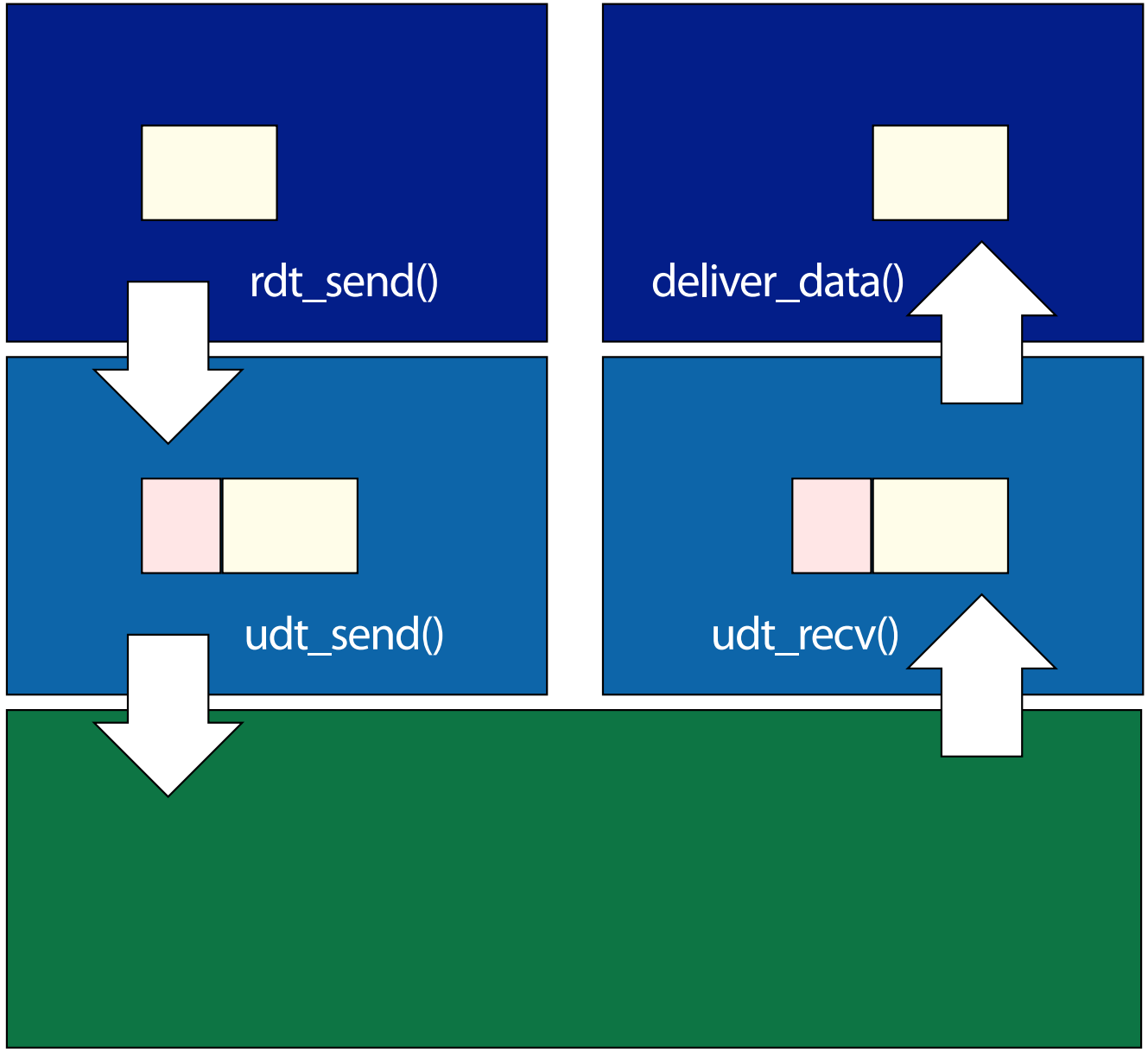
Transport layer resides **on end hosts** and provides **process-to-process** communication.

Network layer provides
host-to-host,
best-effort,
unreliable
communication.

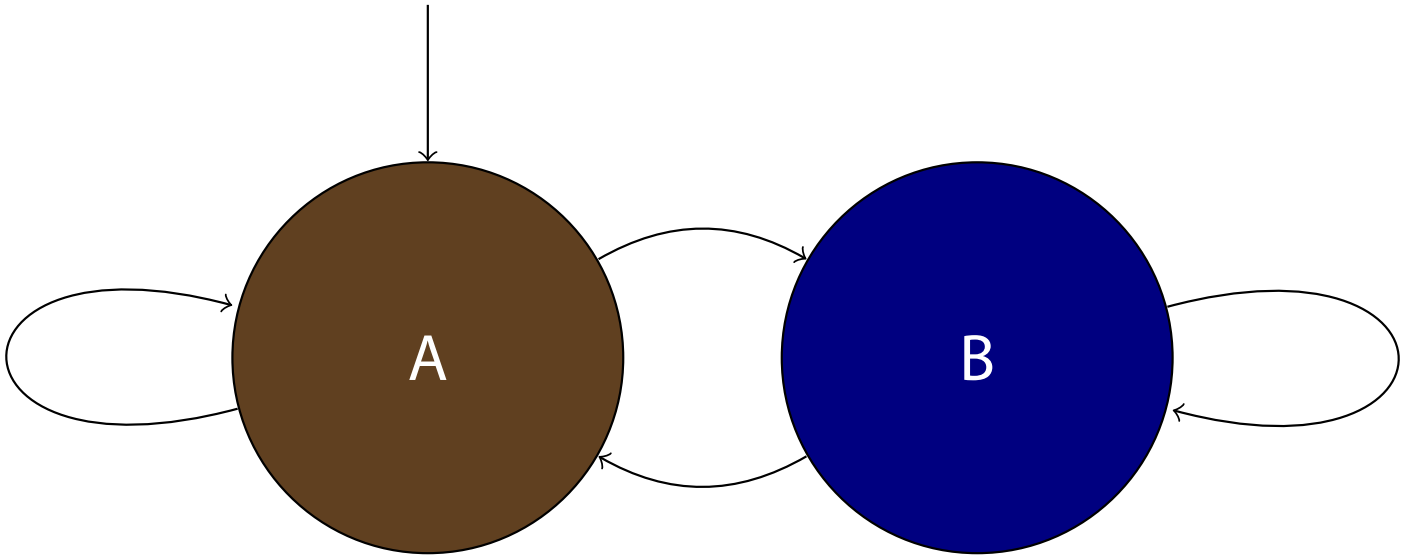

Network

How to build a **reliable transfer protocol** on top of unreliable communication?

Unreliable: may not deliver at all or deliver with error


rdt_send()
deliver_data()
udt_send()
udt_recv()

Finite State Machine

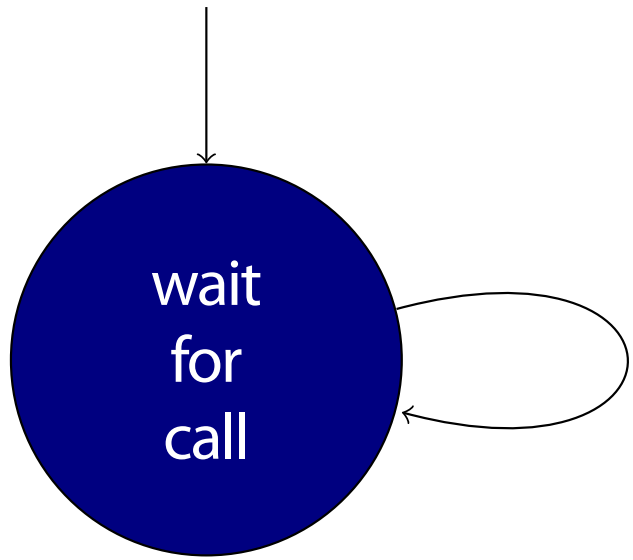


Event
Action

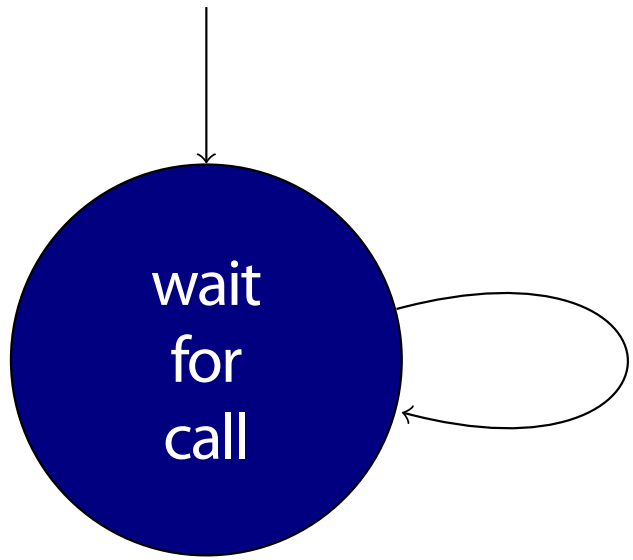
rdt 1.0

Assume underlying channel is reliable

rdt 1.0 sender



rdt 1.0 receiver



rdt 2.0

Underlying channel may introduce bit errors

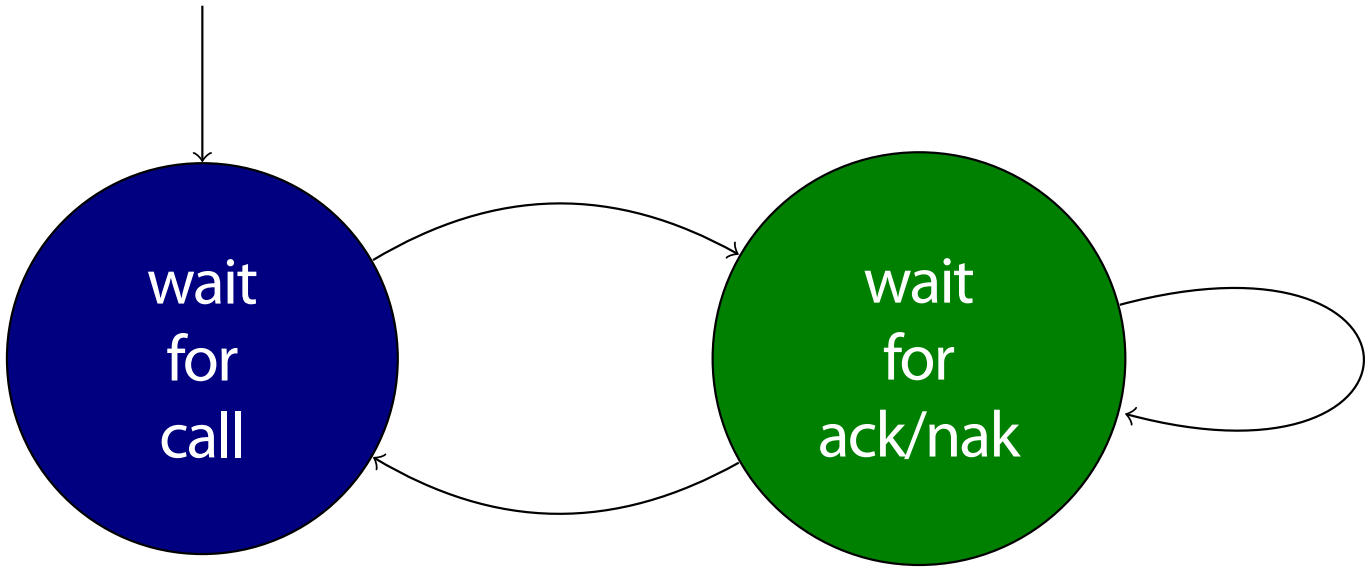
ARQ
Automated Repeat reQuest

Receiver detects errors
Receiver feeds back to sender
Sender retransmits

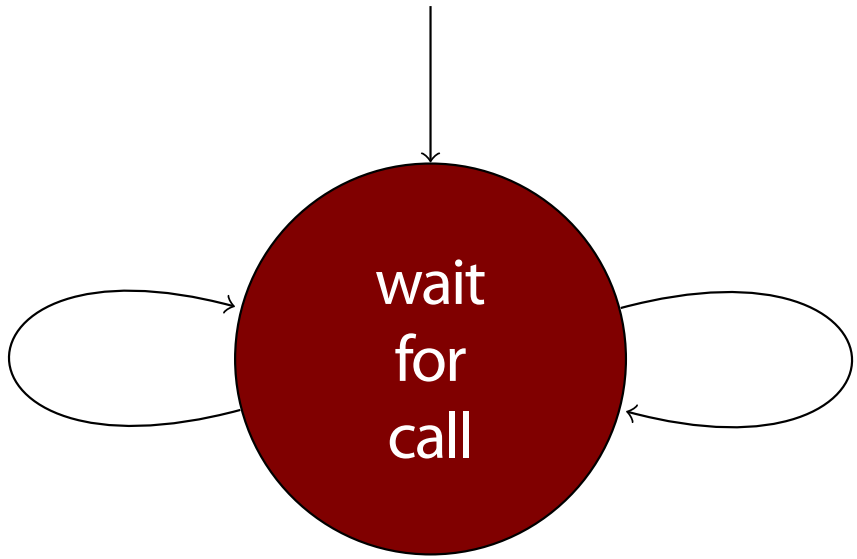
Receiver detects errors
Receiver feeds back to sender
Sender retransmits

We will talk about checksum, the technique to detect errors, in more detail in the subsequent lectures.

rdt 2.0 sender



rdt 2.0 receiver

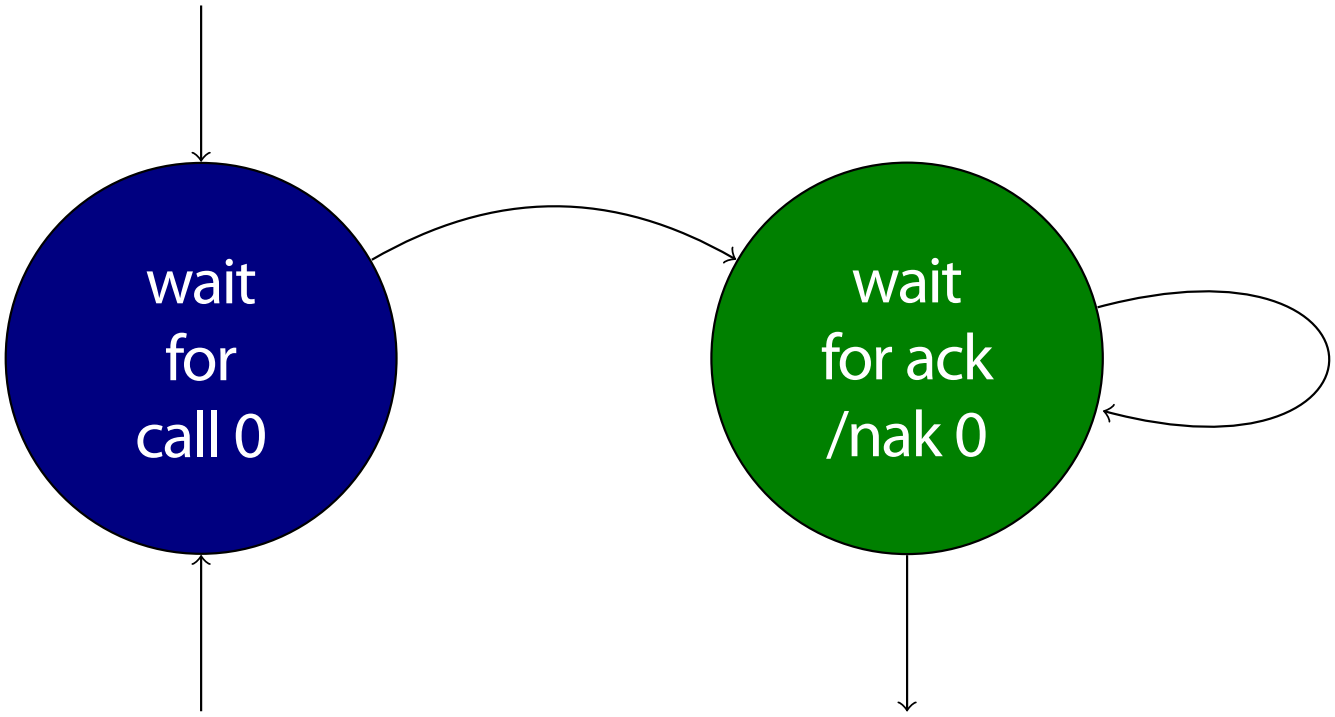


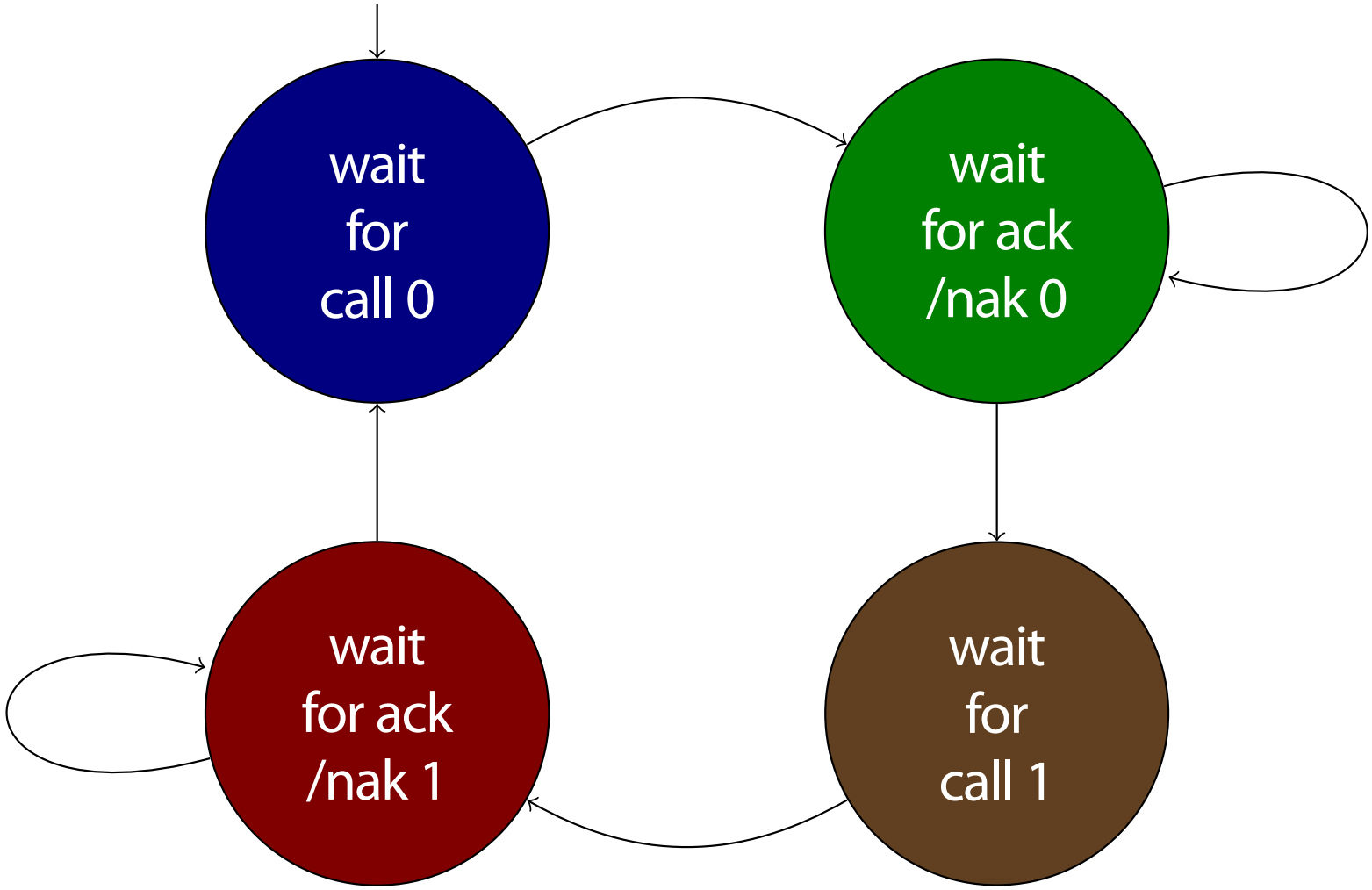
Stop-and-Wait Protocol

Bug:
What if ACK/NAK is corrupted?

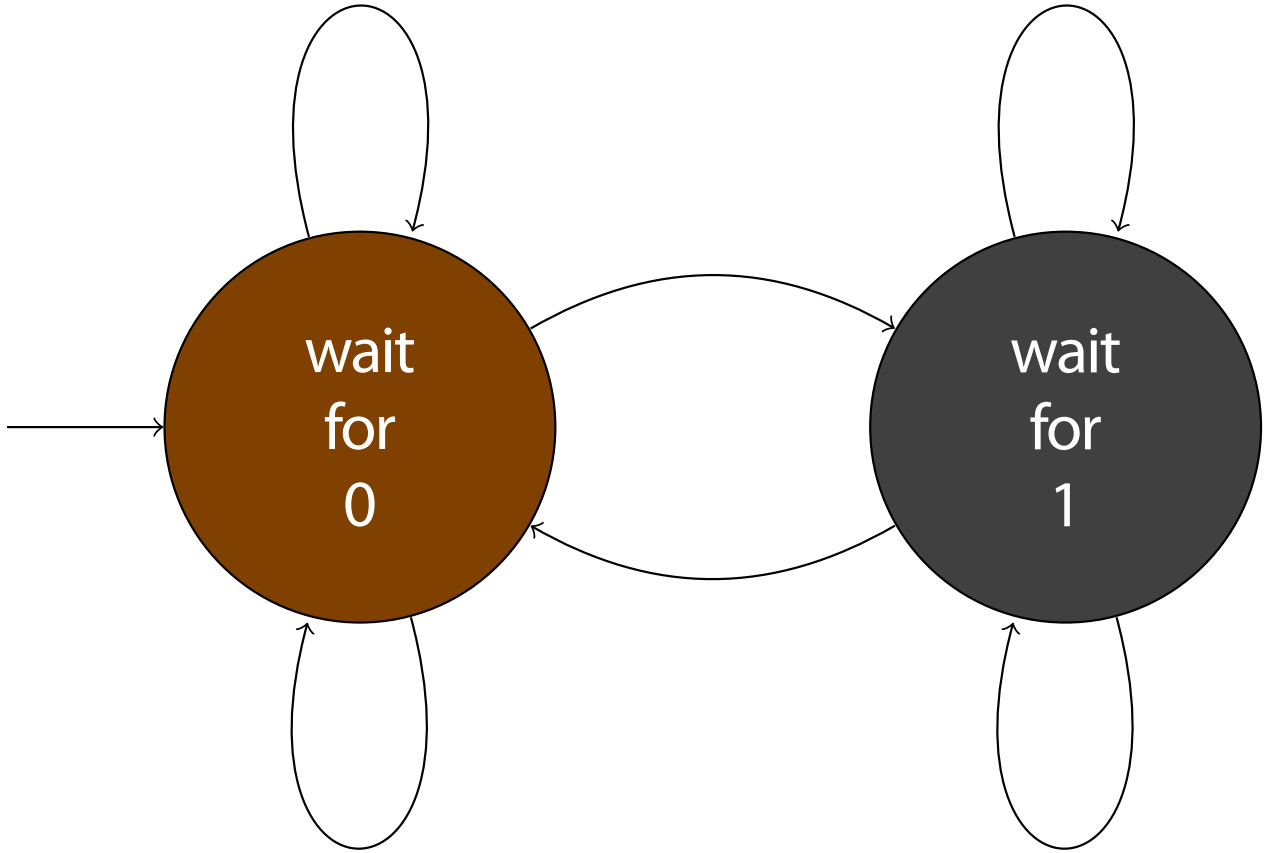
Bug Fix:
Add a sequence number

rdt 2.1 sender




wait
for
call 0
wait
for ack
/nak 0
wait
for ack
/nak 1
wait
for
call 1

rdt 2.1 receiver



rdt 2.2

Replace NAK with ACK of the last correctly received packet.

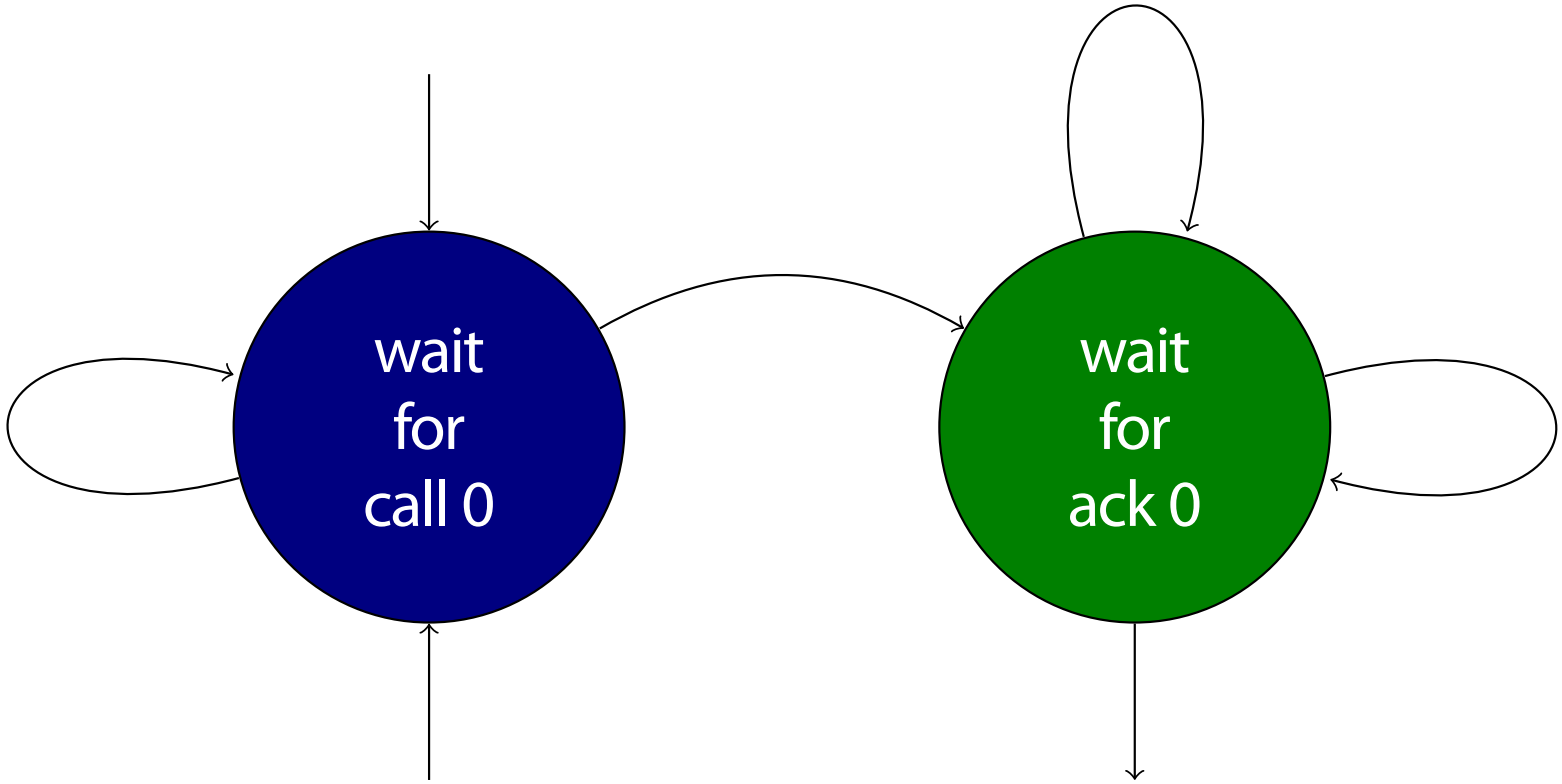
rdt 3.0

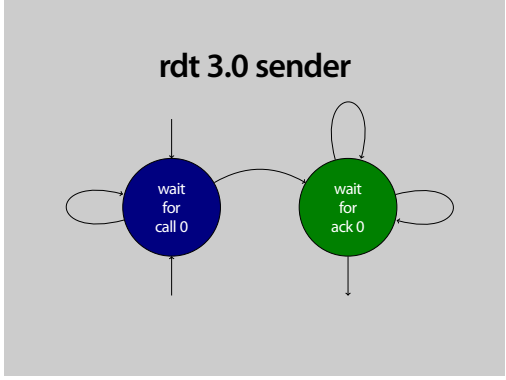
Packet can be loss or corrupted

Challenge: If ACK is lost, how to tell if the packet has been received?

Resend after timeout
(may lead to dups, but OK since we have seq numbers).

rdt 3.0 sender





The state diagram for the receiver in rdt 3.0 is given as an exercise.

Alternating Bit Protocol

RTT = 30ms

R = 1 Gbps

L = 1000 bytes