Internet Engineering

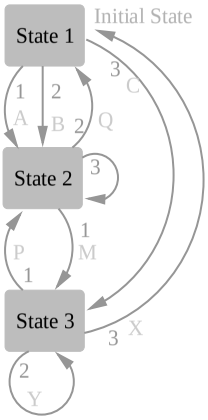
241-461

Robert Elz

kre@munnari.OZ.AU
kre@coe.psu.ac.th

http://fivedots.coe.psu.ac.th/~kre

FSM (example)



Three States 1 2 and 3

Three events that occur 1 2 and 3

Several output actions

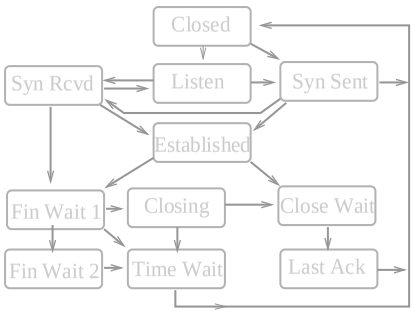
State 1 is the initial state

For input events

```
2 1 3 1 3 2 3 2 1 1
                    3
```

```
What states does FSM pass through?
1 2 3 1 2 2 1 3 3 2 3
What output actions are performed?
B M X A - Q C Y P M
```

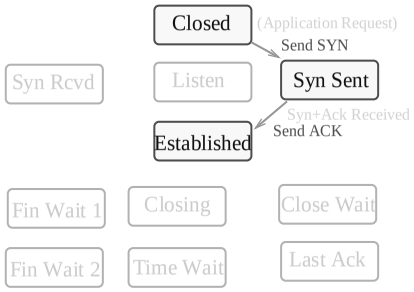
TCP Connections (States)



FSM of TCP connection machinery

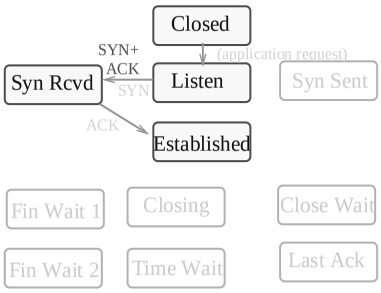
- Inputs & Outputs to come later

TCP Client Open



- 3-way Handshake
- SYN SYN+ACK ACK

TCP Server Open



- Same 3-way handshake
  - SYN SYN+ACK ACK

TCP Opens

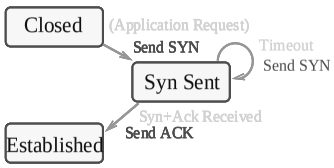
- Start in CLOSED state
  - Neither system knows anything about other
- Send SYNchronise
  - Inform other end we want to communicate
  - Tell it what sequence number we use
- SYN is a kind of data
  - Uses one sequence number itself
  - Always the first sequence number of this connection
- Sender picks sequence number
  - Not just Start at zero
  - To meet some objectives
    - No old packets from previous connection
    - Not easy for anyone to guess

TCP Opens (2)

- SYN is data
  - So must be acknowledged
- Receiver of SYN server
  - Must send ACKnowledge
    - So sender knows it was received
- TCP is full duplex
    - data goes both directions
  - Need sequence numbers both directions
  - So must SYNchronise in reverse
    - Send SYN back to client
- Data and ACK can travel in same packet
  - Data here is SYN
    - Therefore SYN+ACK

TCP Opens (3)

- Data in SYN+ACK packet must be acknowledged
  - That is the SYN
- Must ACKnowledge that SYN
  - But just ACK this time
  - SYN already sent
    - Other than retransmit
    - Just one SYN per connection
      - In each direction



- Investigate
  - 2 systems talking to each other
    - Both acting as clients