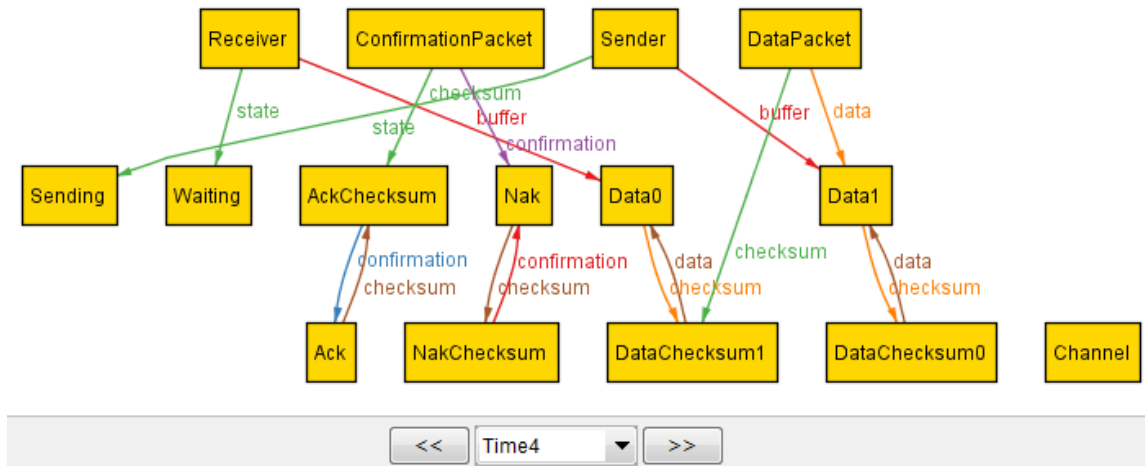
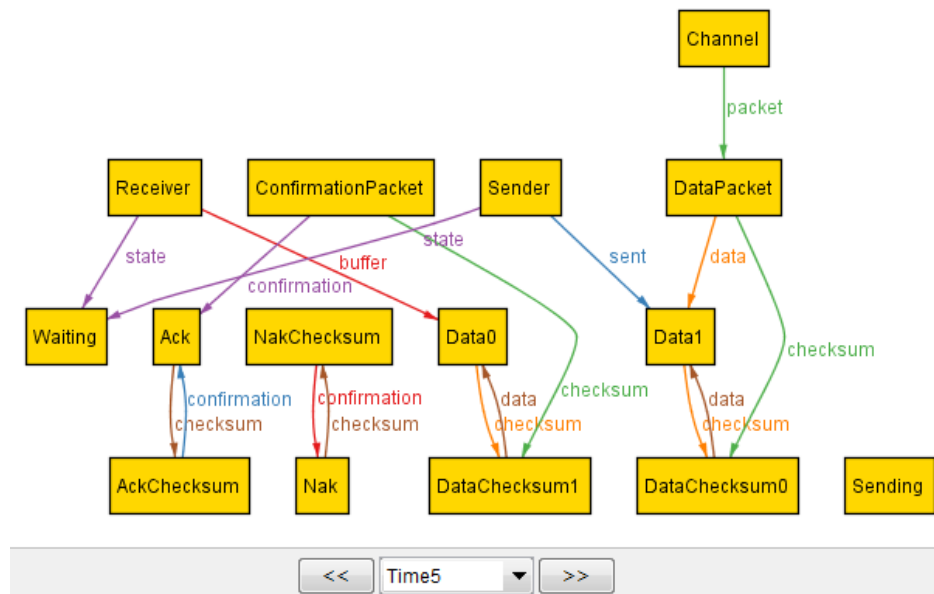


Revisiting Milestone 2

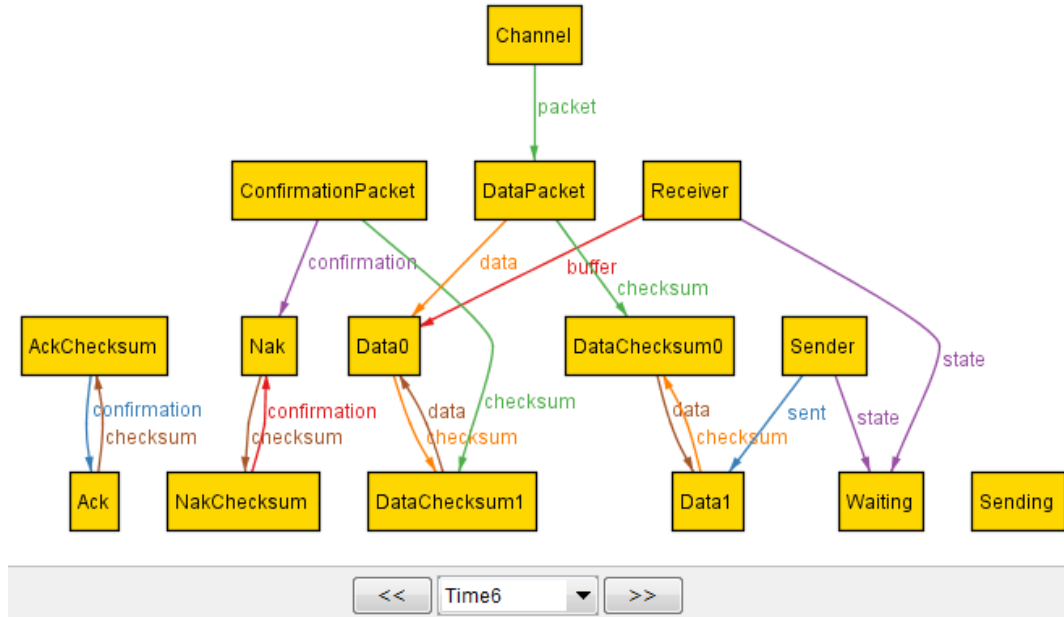
We did not have the exact specification that you were looking for in Milestone 2 (ability to actually lose packets), so we had to fix that part first. Here is a quick rundown of what it looks like losing a data packet in our model:



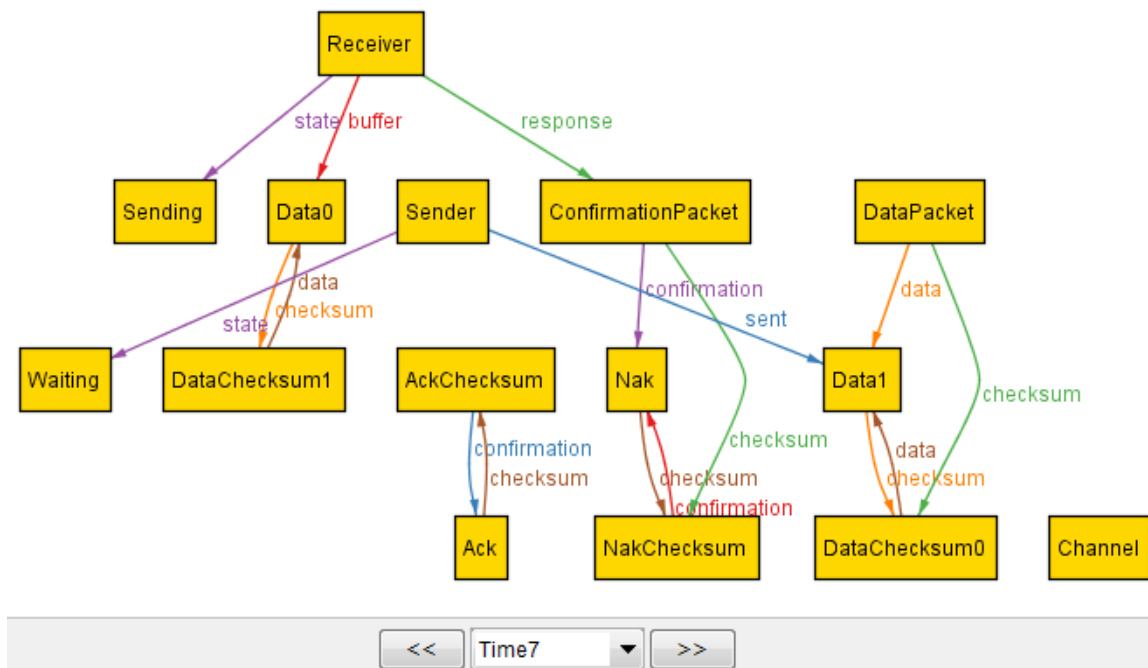
I have skipped ahead to the state where we have successfully sent Data0 from the Sender's buffer to the receiver's. Now begins the interesting part!



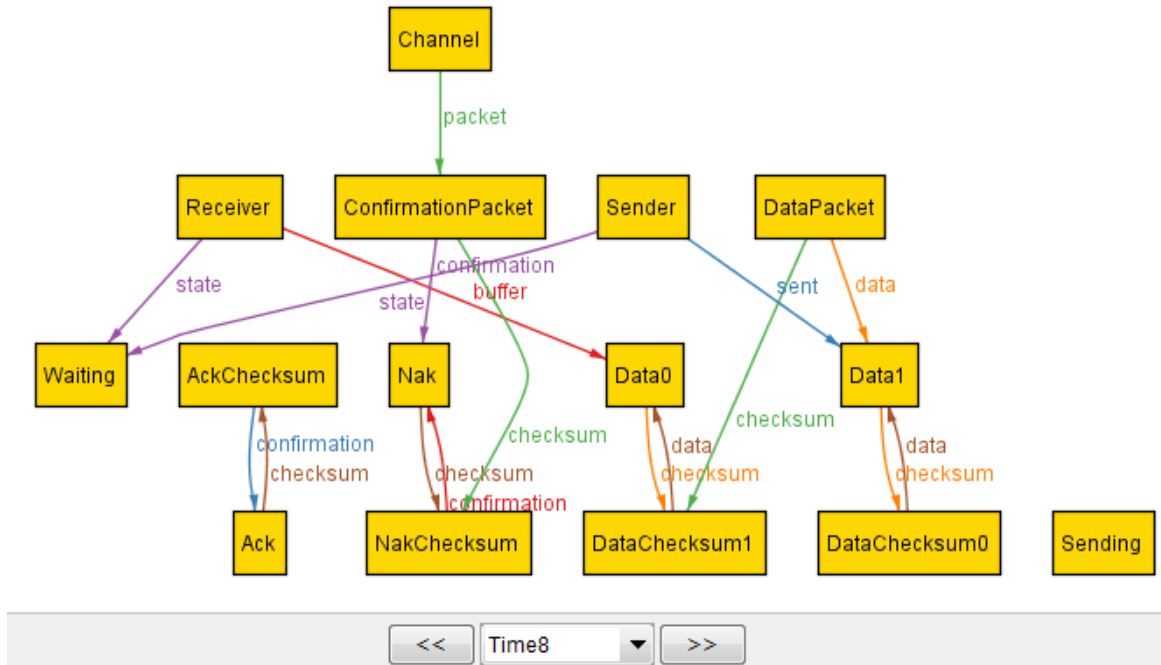
Here, the Sender has created a new packet with the last piece of data in its buffer, Data1, and dropped it into the Channel to send over to the Receiver.



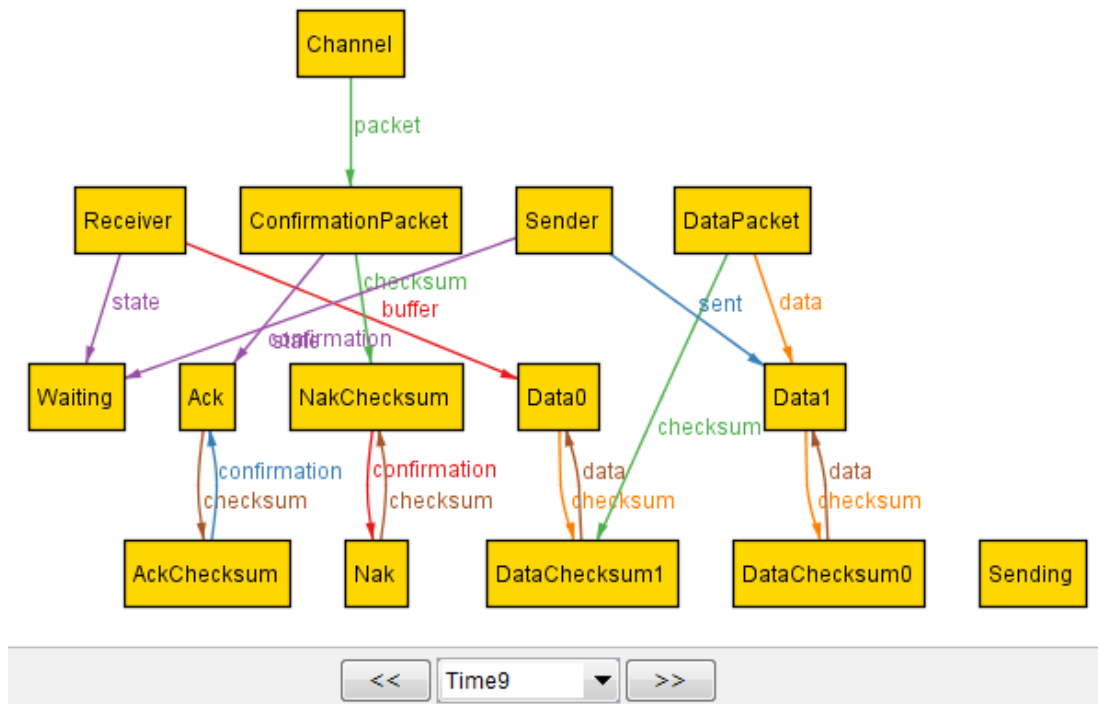
Now, the data is malformed inside of the Channel (modeled by the packet's switch from Data1 to Data0).



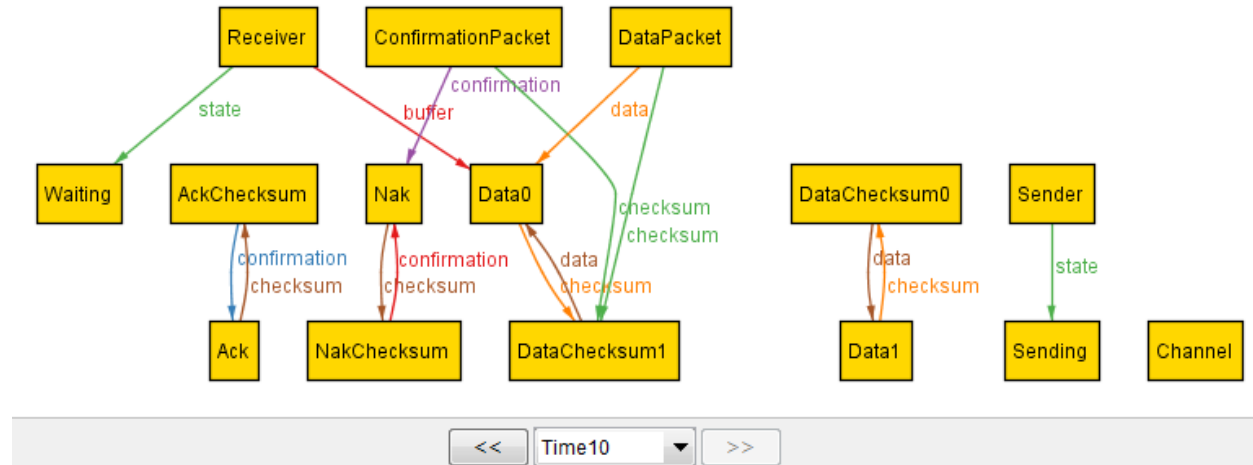
The Receiver gets this malformed data and identifies it as incorrect by its mismatching checksum. It prepares a Nak message to send back.



The Nak message is dropped into the Channel to send back to the Sender.



However, the Nak gets malformed to an Ack during the return journey over the Channel!



The Sender reads this Ack in as confirmation that the Receiver got the data (Data1), so it drops it from its own storage. Neither the Sender nor the Receiver have any knowledge of this data anymore, so it is lost! We can see that in the model as the Data1 and DataChecksum0 connected all by themselves on the outside.

Milestone 3 Stuff (RDT2.1 & RDT 2.2)

The issue outlined above no longer exists with our use of Sequence Numbers and removal of Nak response packets. We can run the same Unsuccessful predicate as we did above to show a scenario when a packet is lost with our new specification, but we can find no instance, even with relatively unbounded time (100 Time). **This shows the new addition to the Specification for Milestone 3!**

Executing "Run Unsuccessful for exactly 3 Data, 5 Checksum, 100 Time"
Solver=sat4j Bitwidth=0 MaxSeq=0 SkolemDepth=1 Symmetry=20
572719 vars. 4137 primary vars. 153681 clauses. 2496ms.
No instance found. Predicate may be inconsistent. 983ms.

However, we are still able to transfer all 3 pieces of data over in the minimum 13 Time like before.

Executing "Run Successful for exactly 3 Data, 5 Checksum, 13 Time"
Solver=sat4j Bitwidth=0 MaxSeq=0 SkolemDepth=1 Symmetry=20
16701 vars. 570 primary vars. 19429 clauses. 62ms.
Instance found. Predicate is consistent. 31ms.