# Reading 3 -- Sequence x State diagrams

Goal: To verify that every state transition for an object can be achieved by the messages sent and received by that object.

Inputs to Process:

1. Sequence diagrams that describe the classes, objects, and possibly actors of a system and how they collaborate to capture services of the system.
2. State diagrams that describe the internal states in which an object may exist, and the possible transitions between states.

For **each state diagram,** perform the following steps:

1) **Read the state diagram to understand the possible states of the object and the actions that trigger transitions between them.**

INPUTS: State diagram (SD).
OUTPUTS: Transition Actions (marked and labeled in green on SD);
Discrepancy reports.

☞ Determine which class is being modeled by this state diagram.

**Could you identify the type of the object that this state machine is modeling? If you can't find this information, you have found a defect of omission. Fill in a discrepancy report for this.**

☞ Trace the sequence of states and the *transition actions* (system changes during the lifetime of the object, which trigger a transition from one state to another) through the state diagram. Begin at the start state and follow the transitions until you reach the end state. Make sure you have covered all transitions.

☞ Highlight transition actions (represented by arrows) as you come to them using a green pen. For example, the state diagram provided in Example 5 contains seven transition actions. The arrow leading from the state labeled "authorizing" back to itself represents an action that does not actually change the state of the object. Give each action a unique label [A1, A2, …].

☞ Think about the states and actions you have just identified, and how they fit together.

**Is it possible to understand and describe what is going on with the object just by reading this state machine? If not, you may have discovered a defect of ambiguity; the behavior of this class over its lifetime is not well described.**

2) **Read the sequence diagrams to understand how the transition actions are achieved by messages that are sent and received by the relevant object.**

INPUTS: State diagram (SD);<br>
Transition Actions (marked and labeled in green on SD);<br>
Sequence diagrams (SqD).

OUTPUTS: Object messages (marked and labeled in green on SqD);<br>
Discrepancy reports.

☞ Take the sequence diagrams and choose the ones that use the object modeled by the state diagram; use only this subset of the sequence diagrams in the remainder of this step.

**Could you find sequence diagrams in which the object participates? If not, you have found a defect of omission (necessary descriptions of how this object contributes to system services are missing) or extraneous information (this object does not contribute to system services and is not needed). Fill out a discrepancy report describing this defect.**

☞ For each sequence diagram identified in the previous step:

☞ Read the diagram to identify the system service being described and the messages that this object receives.

☞ Think about which object states on the state diagram are *semantically* related to the system service. Highlight the state transitions leading to and from these states, and use this subset for the remainder of this step.

☞ Map the object messages on the sequence diagram to the state transitions on the state diagram. Each transition action may map to one message, or a sequence of messages. To do this, you will need to think about the *semantics* behind the system messages. Are they contributing to achieving some larger system service or functionality? Do they have something to do with the types of states this object should be in? When you have made a mapping, mark the related messages and transition actions with a star (*). Label the messages with the same label given to their associated action on the state diagram.

**Semantically, could you do this mapping? Were there additional messages needed to achieve the state transition? If so, you have discovered a defect of omission. Important messages are missing from the sequence diagrams for this object. Fill out a discrepancy report describing this defect.**

☞ Look for constraints and conditions on the messages you just mapped to state transitions. An example constraint might be “t>0”, that is, whether or not a message is sent depends on the value of some attribute t. Look to see that any constraints/conditions found are captured somehow on the state diagram. This information might be captured by: 1) state information (i.e. the fact that t>0 corresponds to a particular state of the system; 2) transition information (i.e. some state transition occurs when t>0 becomes true or false; 3) nothing (i.e. this information is not relevant or important for the state diagram).

**Could you find some correspondence between the constraint/condition information on the state and sequence diagrams? Does the information included on both diagrams have the same meaning? If not, you have detected an inconsistency between the diagrams. Fill out a discrepancy report describing this defect. Is there some constraint/condition information on the sequence diagram that was not captured on the state diagram? If so, is it important enough to the idea of object states that it should have been somehow represented? If yes, you have detected an omission. Necessary information has been left out of the state diagram. Fill out a discrepancy report describing this defect.**

3) **Review the marked-up diagrams to make sure that all transition actions are accounted for.**

INPUTS: Transition Actions (marked and labeled in green on SD);
Object messages (marked and labeled in green on SqD);.

OUTPUTS: Discrepancy reports.

☞ Review the state diagram looking for unstarred transition actions that could not be associated with object messages.

**If the transition action was labeled with a constraint, could you find a message or sequence of messages capable of satisfying the constraint? If not, you have found a defect of inconsistency. The state diagram requires system services that are not described on any sequence diagram. Fill out a discrepancy report describing this defect. If the transition action was labeled with an event, could you find a message, a sequence of messages, or some event performed by an actor that achieves the transition action? If not, you have found a defect of inconsistency. The state diagram requires system services that are not described on any sequence diagram. Fill out a discrepancy report describing this defect.**

☞ If the starred messages and transition actions identified in the previous step appear on the same sequence diagram, make sure they appear in a logical order. That is, suppose the messages that achieve action A1 appear before the messages that achieve action A2 on one sequence diagram. This means that A1 must take place chronologically before A2. Then you should make sure that A1 could be reached before A2 on the state diagram as well.

**Does the order of events not match between the two diagrams? If so, there is an inconsistency. The system services and functionality are described in different ways on the two diagrams. Fill out a discrepancy report describing this defect.**