

Common problems

- 1 Can I draw a state diagram for the whole system?

No – a state diagram normally only models the behaviour of the objects of a single class. One of the most common mistakes that students make when learning the technique is to try to model the behaviour of the whole system in a single state diagram.

- 2 How do I know if I need to draw a state diagram for a particular class?

In order to decide whether or not to draw a state diagram, you need to look at how objects of the class behave in response to events; you can see this by studying the lifelines of the objects in all the relevant interaction diagrams, i.e. all the ones in which objects of this class feature. When you look at the lifeline of an object in a sequence diagram, you can see all the events that happen to it (i.e. the messages sent to it) and how many there are; you can also see whether or not the object always responds in the same way. This is even more apparent in collaboration diagrams as the way they are drawn emphasizes all the messages coming to an object. In fact examining interaction diagrams is one of the main starting points when drawing state diagrams. A good (i.e. representative) set of interaction diagrams will show all the events that can happen to an object during its lifetime (all the messages that can be sent to it) and all the different ways it can respond. From this a list of events can be drawn up like the ones we compiled for the Job Application and Bike state diagrams. State diagrams and interaction diagrams look at the same events but from a different viewpoint. An interaction diagram shows how the execution of a particular scenario affects all of the objects involved. A state diagram looks at a particular class of objects and shows how all of the scenarios affect them. So when your state diagram is complete you should be able to take each scenario in turn and trace through the state diagram following the sequence of events that affect that class.

Students often try to draw a state diagram for a class that is not complex enough to need one, such as the Customer class in the Wheels system. It is only useful to draw a state diagram in cases where the way an object of that class responds to an event depends on the state it is in. This is what textbooks mean when they refer to an object having dynamic behaviour. Only classes with dynamic behaviour are worth modelling with a state diagram.