- Identify the events that affect an object of the class
- Identify the different states that objects of the class can be in, including the start state and (possibly) multiple stop states
- Check whether any events that are listed separately should be represented as the same event with different conditions (guards)
- Check whether there are any actions that the system must perform in response to an event or whilst in a given state; these should be represented as actions in the transition or state labels
- Begin to construct the diagram from the start state, the event that creates an object of the class, and the state that the object moves into
- Build up the diagram, working through the events and states on the list and adding them to the diagram
- Check that all guards and actions have been included on the relevant transition labels
- Check whether a superstate should be included to cater for events that may occur at any time during the life of an object
- Check the completed diagram against the information that has been gathered about the behaviour of the class.

Bibliography

Bennett, S., McRobb, S. and Farmer, R. (2002) *Object-Oriented Systems Analysis and Design Using UML* (2nd edition), McGraw-Hill, London.

Britton, C. and Doake, J. (2002) *Software System Development: A Gentle Introduction* (3rd edition), McGraw-Hill, London.

Fowler, M. (2000) *UML Distilled: A Brief Guide to the Standard Object Modeling Language* (2nd edition), Addison-Wesley, Reading, MA.

Quick check questions

You can find the answers to these in the chapter.

a What aspect of a system is modelled by a state diagram?

b What is meant by 'state' in this context?

c What is meant by 'event' in this context?