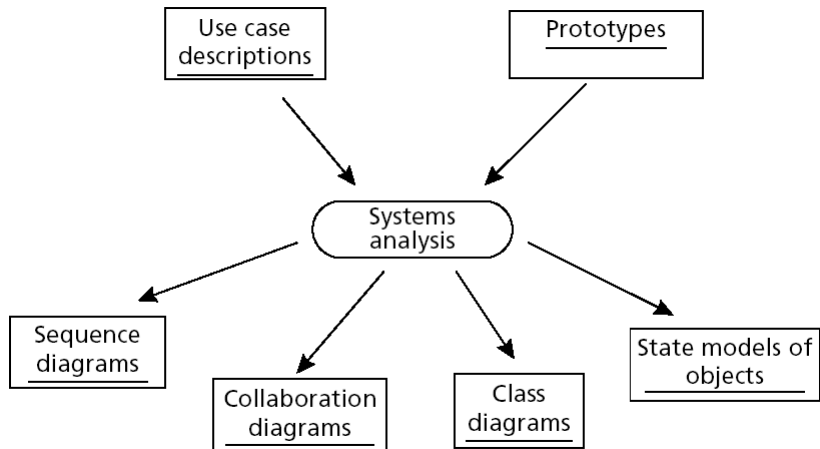
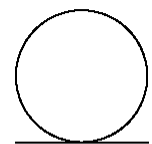


**Figure 12.1** Inputs and outputs of systems analysis

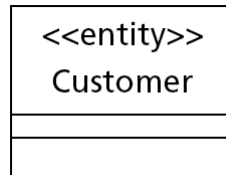
Software Development with UML – Copyright Ken Lunn 2003

**Figure 12.2** UML notation for an entity class

Customer

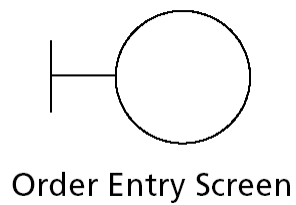
Software Development with UML – Copyright Ken Lunn 2003

**Figure 12.3** The UML notation for an entity class using a label to indicate the stereotype



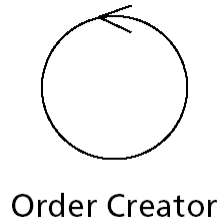
Software Development with UML – Copyright Ken Lunn 2003

**Figure 12.4** UML notation for a boundary class



Software Development with UML – Copyright Ken Lunn 2003

**Figure 12.5** UML notation for a control object



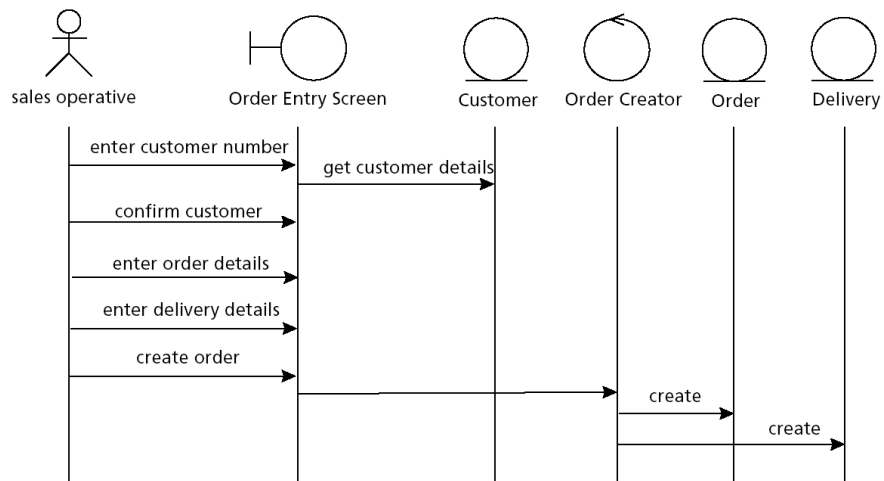
Software Development with UML – Copyright Ken Lunn 2003

### Simple scenario

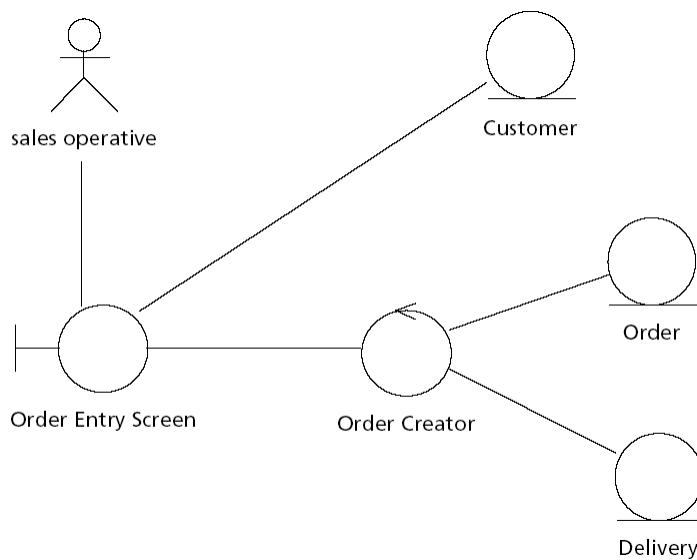
Suppose that we have a simple scenario from an order-taking use case:

1. The sales operative takes the customer number and enters it on the screen.
2. The customer details are retrieved and displayed on the screen.
3. The sales operative checks that the customer details match those given by the customer, and ticks a confirm box.
4. The sales operative enters the order details.
5. The sales operative enters the delivery details.
6. The sales operative requests that the order is created.

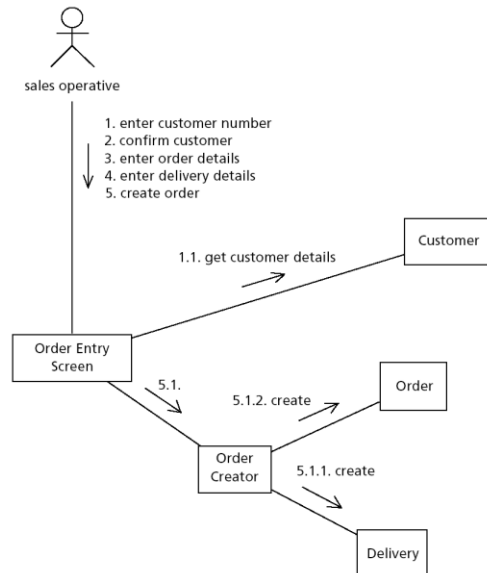
Software Development with UML – Copyright Ken Lunn 2003

**Figure 12.6** A simple UML sequence diagram

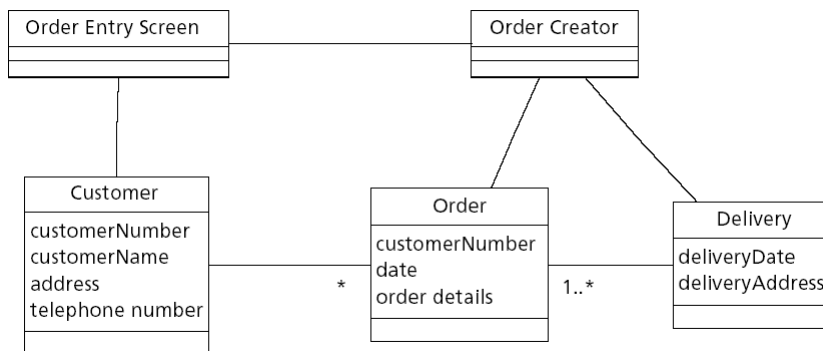
Software Development with UML – Copyright Ken Lunn 2003

**Figure 12.8** A UML collaboration diagram

Software Development with UML – Copyright Ken Lunn 2003

**Figure 12.9** A collaboration diagram without stereotype displays but with messages

Software Development with UML – Copyright Ken Lunn 2003

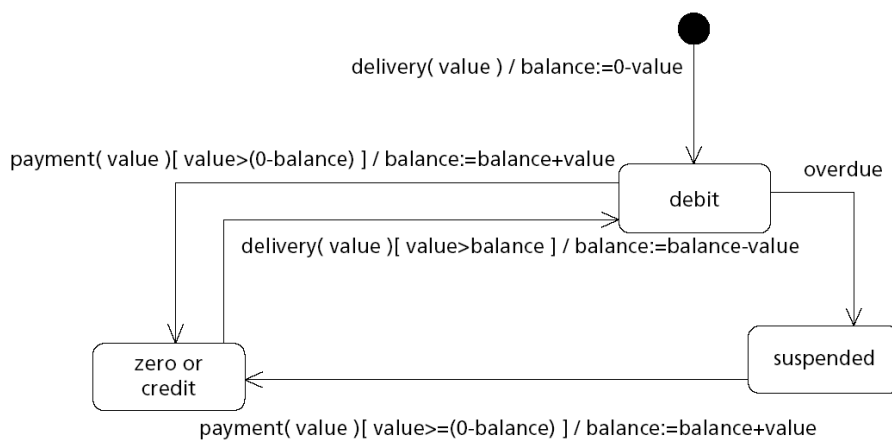
**Figure 12.10** A UML class diagram

Software Development with UML – Copyright Ken Lunn 2003

Figure 12.11 A prototype screen

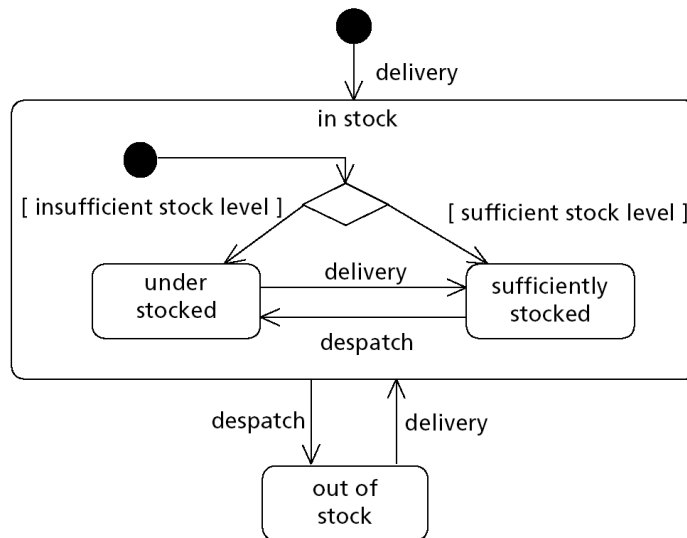
Software Development with UML – Copyright Ken Lunn 2003

Figure 12.13 A statechart diagram for a customer account with events, guards and actions



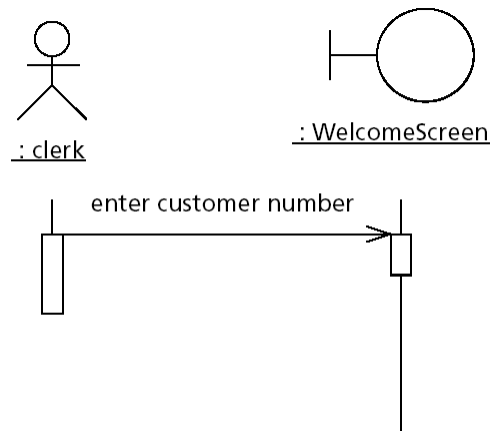
Software Development with UML – Copyright Ken Lunn 2003

**Figure 12.14** A simple example of substates in a statechart diagram

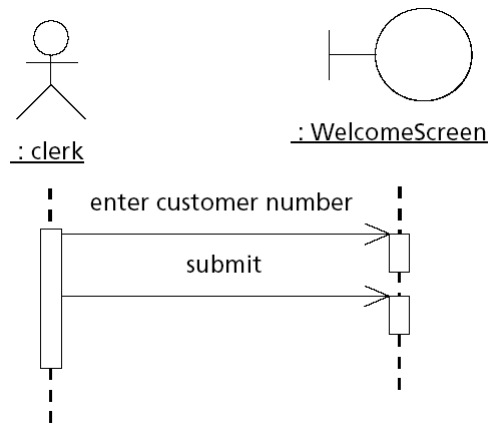


Software Development with UML – Copyright Ken Lunn 2003

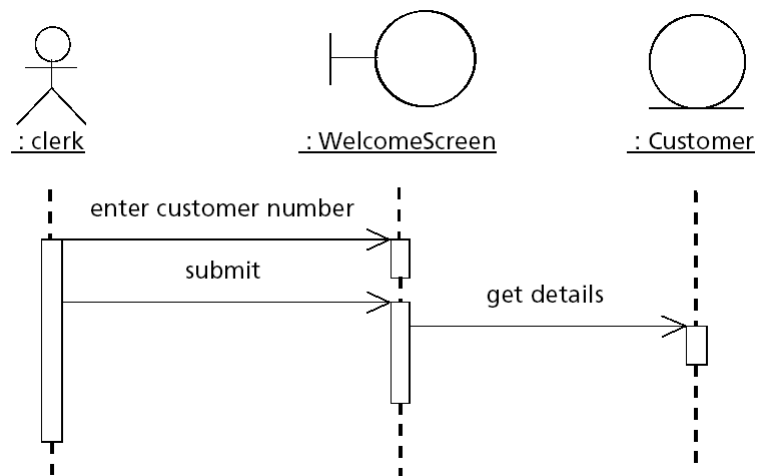
**Figure 12.15** The beginnings of a sequence diagram, built from a use case scenario



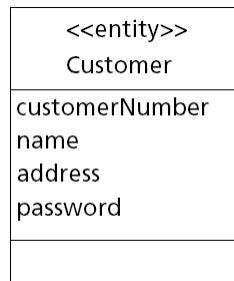
Software Development with UML – Copyright Ken Lunn 2003

**Figure 12.16** Further steps in the sequence diagram

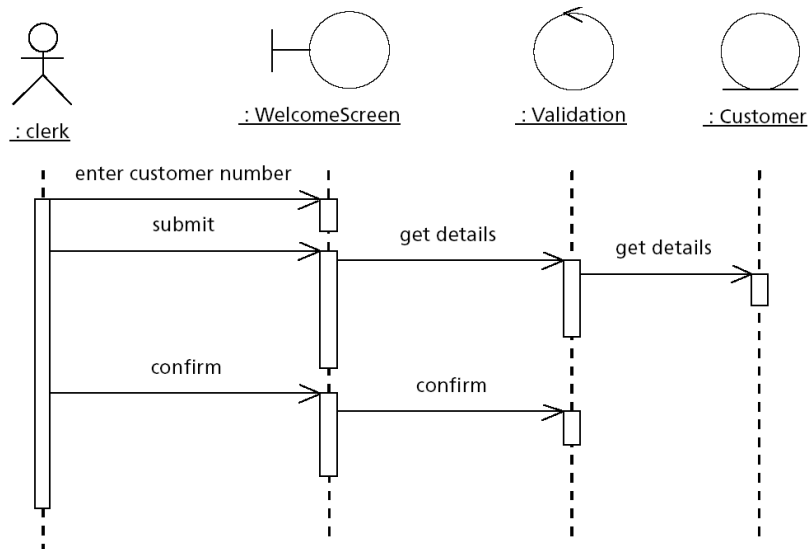
Software Development with UML – Copyright Ken Lunn 2003

**Figure 12.17** Further addition to the sequence diagram

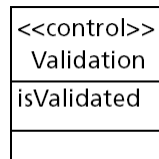
Software Development with UML – Copyright Ken Lunn 2003

**Figure 12.18** Entity object to store customer information

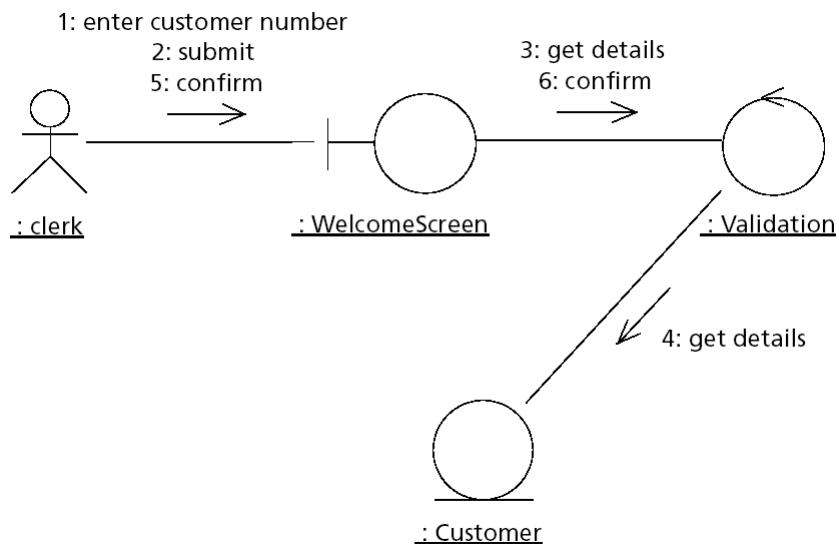
Software Development with UML – Copyright Ken Lunn 2003

**Figure 12.19** The final analysis sequence diagram for the primary path of the Validate Customer use case

Software Development with UML – Copyright Ken Lunn 2003

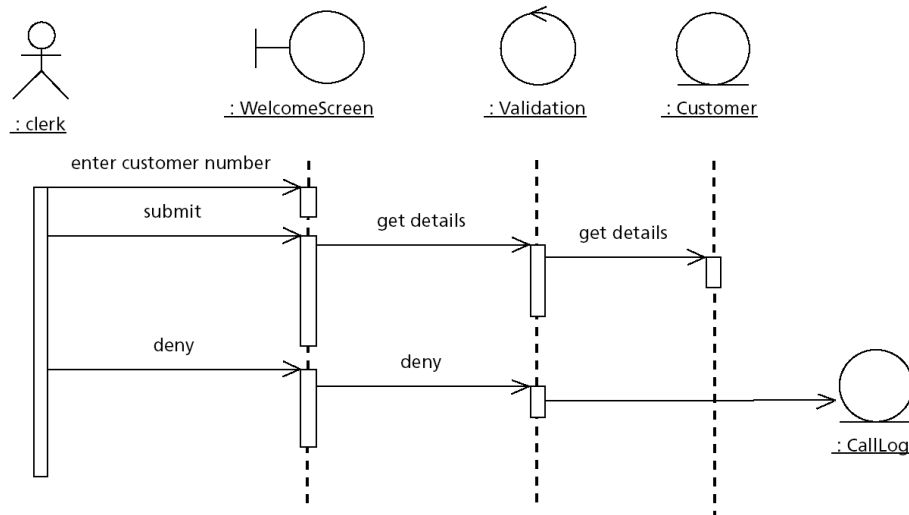
**Figure 12.20** The validation control object

Software Development with UML – Copyright Ken Lunn 2003

**Figure 12.21** A collaboration diagram to show the structure of the system that supports the validate customer use case

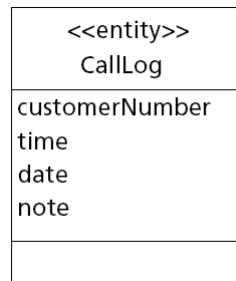
Software Development with UML – Copyright Ken Lunn 2003

**Figure 12.22** Sequence diagram to show the workings of the alternative in the Validation use case where the customer is denied



Software Development with UML – Copyright Ken Lunn 2003

**Figure 12.23** An entity object to store a call log



Software Development with UML – Copyright Ken Lunn 2003