changes. This is achieved through the use of components – subsystems that each have a single, well-defined function. RUP describes how to construct an architecture combining both new and previously existing components, thus encouraging the reuse of software as part of the development process.

4 *Visually model software*

RUP is based around the Unified Modelling Language (UML) as a vehicle for development. UML has become an industry standard, and incorporates a wide range of techniques and tools to support developers. The techniques offered by UML bring with them all the advantages of visual modelling. For example, UML diagrams facilitate communication between developers and users and between members of the development team, they offer a number of different views of the system which combine to give a complete picture, they help developers to decompose the problem into smaller, more manageable chunks, and they provide a means of abstraction, concentrating on important information while hiding details that are currently irrelevant.

5 *Verify software quality*

RUP provides the techniques to support quality assessment of functionality, reliability and performance throughout the development process. The RUP approach to quality is based on objective measures and criteria for success; it involves all members of the development team and applies to all the activities that are carried out as part of the system development.

6 *Control changes to software*

Changes are the norm in a software development project, so an effective development process must be able to monitor and control them. RUP provides tools to do this, and also supports the work of developers by offering protection in one area of development from changes that occur in another.

RUP is an increasingly popular approach to developing software systems, and is already laying claim to be the industry standard. However, it would be overkill to work through all the details of RUP in this book, since the book is based around the development of a small, simple information system. We therefore describe the development of the Wheels bike hire system within a simplified object-oriented framework.