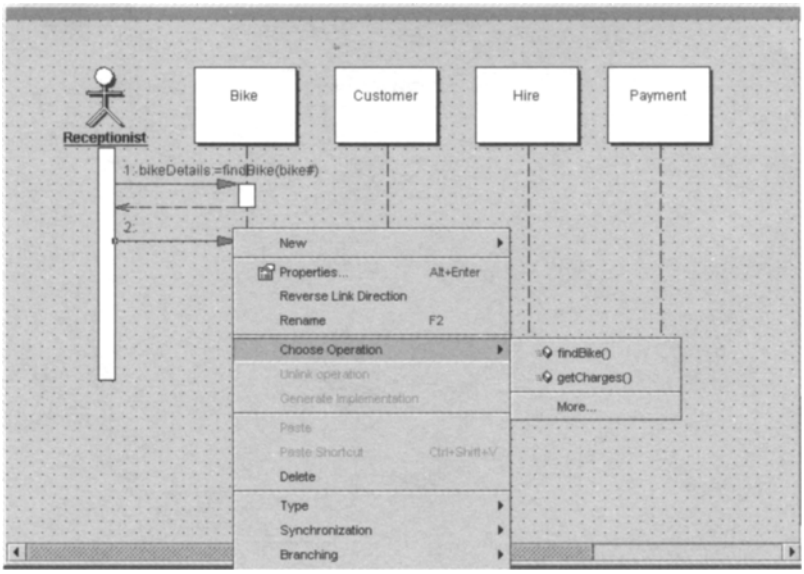

Figure 6.15 *Screen offering developer a choice of known operations for a message label*

diagram, the objects involved in each scenario and the operations on classes. A good CASE tool (see Chapter 1) will support model consistency by allowing the developer to link the objects on the interaction diagram to a list of classes it knows about from the class diagram. Similarly, it allows the developer to choose a label for the message arrow from a list of operations defined on the target object's class. Figure 6.15 shows a CASE tool offering a choice of operations (findBike() and getCharges()) for message number 2. The operations findBike() and getCharges() are defined on the class Bike in the class diagram.

Sequence diagrams are also useful for checking existing models; we may find, when doing the sequence diagrams, that we need an extra operation, or that we never use one that we did specify. A good CASE tool will allow us to add or delete operations and will update models (such as the class diagram) that are affected by our decision.

Using packages in interaction diagrams

On complicated interaction diagrams it is sometimes useful to suppress some of the details. The layout of collaboration diagrams makes it easy to identify groups of tightly coupled objects which can conveniently be regarded as a unit while we concentrate on what is happening in the rest of the diagram. In Figure 6.16a, objects e, f and g show complicated inter-object messaging. They can be grouped into a package and can be treated as a single entity while we concentrate on the interactions between objects a, b, c and d as in Figure 6.16b.