

Dialog Maps Acceptors are also commonly used to model user interfaces. A **dialog map** is a state diagram whose nodes represent user interface states. Dialog maps are thus used to model user interface behavior. For example, the dialog map in Figure 13-3-5 models the behavior of a simple stopwatch program.

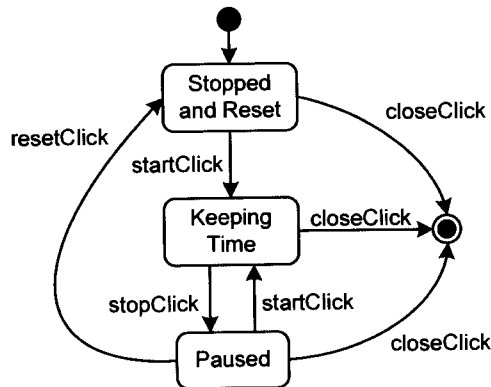


Figure 13-3-5 Stopwatch Program Dialog Map

This dialog map shows that the program first displays itself in the state **Stopped and Reset**. If the **startClick** event occurs, the user interface changes to the **Keeping Time** state. The **stopClick** event transitions to the **Paused** state. The **startClick** event again returns to the **Keeping Time** state, and the **resetClick** event returns to the **Stopped and Reset** state. If the **closeClick** event occurs in any state, the program halts.

User Interface Diagrams A useful adjunct to a dialog map is a user interface diagram. A **user interface diagram** is a drawing of (part of) a product's visual display when it is in a particular state. User interface diagrams are static models of the visual display.

There are no particular rules governing user interface diagrams: They are simply drawings of the user interface in some state. However, one graphic device—the **callout**—is especially helpful with user interface diagrams because it is often not completely clear from a drawing alone what a particular element represents. Also, constraints on data values, enabled state, visibility, and so forth are often not clear from a diagram. Consequently, text may be needed to add missing information to user interface diagrams. A **callout** is a note attached to a line or arrow picking out part of a diagram. Figure 13-3-6 shows a simulated gas station user interface diagram with callouts.