

Chapter 17

Timing Diagrams

After leaving secondary school, I started out in electronic engineering before I switched into computing. So I feel a certain anguished familiarity when I see the UML define timing diagrams as one of its standard diagrams. Timing diagrams have been around in electronic engineering for a long time and never seemed to need the UML's help to define their meaning. But since they are in the UML, they deserve a brief mention.

Timing diagrams are another form of interaction diagram, where the focus is on timing constraints: either for a single object or, more usefully, for a bunch of objects. Let's take a simple scenario based on the pump and hotplate for a coffee pot. Let's imagine a rule that says that at least 10 seconds must pass between the pump coming on and the hotplate coming on. When the water reservoir becomes empty, the pump switches off, and the hotplate cannot stay on for more than 15 minutes more.

Figures 17.1 and 17.2 are alternative ways of showing these timing constraints. Both diagrams show the same basic information. The main difference is that Figure 17.1 shows the state changes by moving from one horizontal line to another, while Figure 17.2 retains the same horizontal position but shows state changes with a cross. The style of Figure 17.1 works better when there are just a few states, as in this case, and Figure 17.2 is better when there are many states to deal with.

The dashed lines that I've used on the $\{>10s\}$ constraints are optional. Use them if you think they help clarify exactly what events the timing constrains.

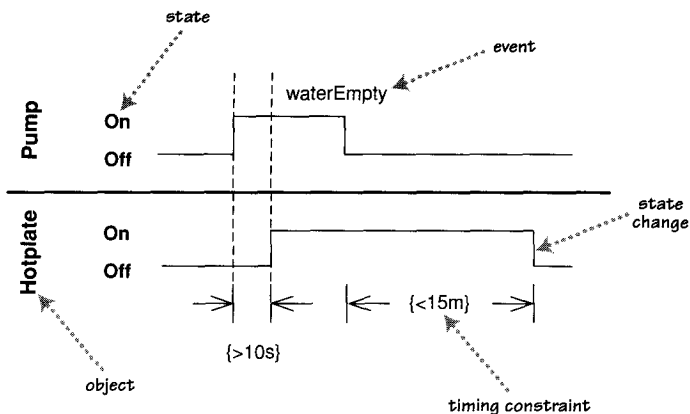


Figure 17.1 Timing diagram showing states as lines

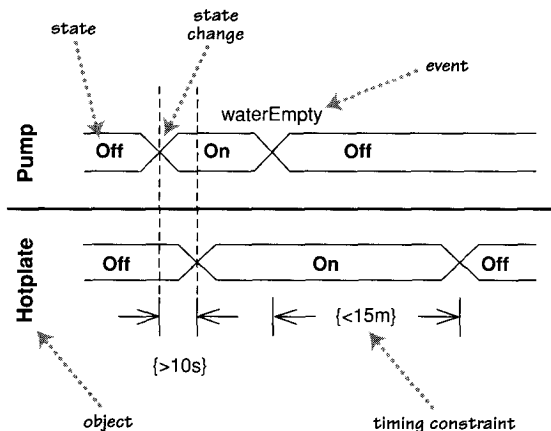


Figure 17.2 Timing diagram showing states as areas

When to Use Timing Diagrams

Timing diagrams are useful for showing timing constraints between state changes on different objects. The diagrams are particularly familiar to hardware engineers.