Engineering Design for Embedded Systems: Assignment 11: Identifying and understanding UML diagrams

Prepared by Devin Binnie

Suggested Completion Date: April 8, 2013

Diagram Identification. We've included 2 UML diagrams. Your task is to identify the type of diagram and understand its meaning. Dissect each part of the diagram and identify its purpose and functionality. Explain the relationships between the parts of each diagram.
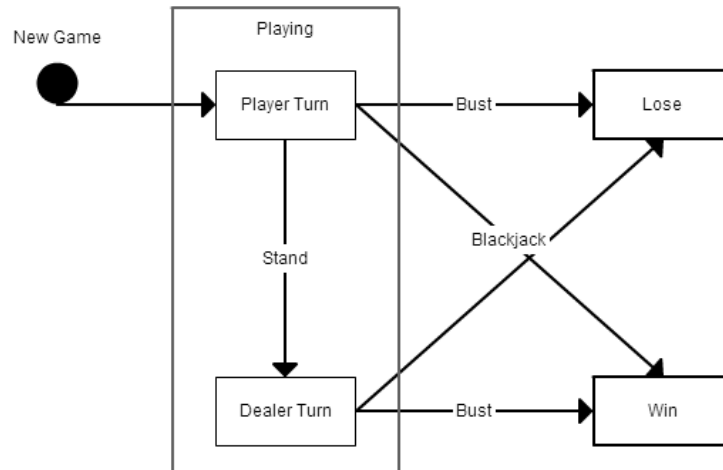

Figure 1: UML Diagram 1.

Diagram Creation. Next, create UML diagrams, including any objects you need, as follows:

1. A class diagram showing objects used in the communication between a client computer (your PC or laptop) and a server computer, when downloading a packet of data.

2. A state diagram including the different states of your computer: sleeping, hibernating, shut down, rebooting, running Operating System. Also, include an 'On' superstate.

3. A state diagram illustrating the operation of a step detection algorithm (as in Lab 2).

4. A state diagram for the control unit (CU) of a windshield wiper of a car. The input to the CU comes from a 3-position rotary switch. The switch positions are `Off`, `LowSpeed` and

`HighSpeed`. A correspondingly named event is sent to the CU when the switch is put on one of these positions. The Control Unit sets the power applied to the windshield wiper motor by invoking method `WPowSet` with a parameter taking values `NoV`, `LowV` and `HighV`.

5. A state diagram for the Control Unit of a window air conditioner with a fan and a compressor unit. The air conditioner has the following front panel controls: a button labelled `On`, a button labelled `Off`, and a rotary dial that sets the desired temperature setting. When the A/C is on, the fan runs continuously and the compressor is turned on and off depending on whether the room temperature is above or below the temperature set on the rotary dial.

A button press is reported as events `OnButtonPressed` and `OffButtonPressed`. The temperature dial setting is available via a memory-mapped variable `SetTemp`, and the actual room temperature sensor readings are in a memory-mapped variable `CurTemp`. The Control Unit controls the A/C compressor unit by invoking methods `CompOn()` and `CompOff()` and the fan by invoking methods `FanOn()` and `FanOff()`.

To reduce the wear on the mechanical parts of the air conditioner, we use some hysteresis—the compressor should only be turned off when the room temperature falls below `SetTemp`-delta, and turned on only when the room temperature rises above `SetTemp`+delta, where delta is a small number, say one degree C.
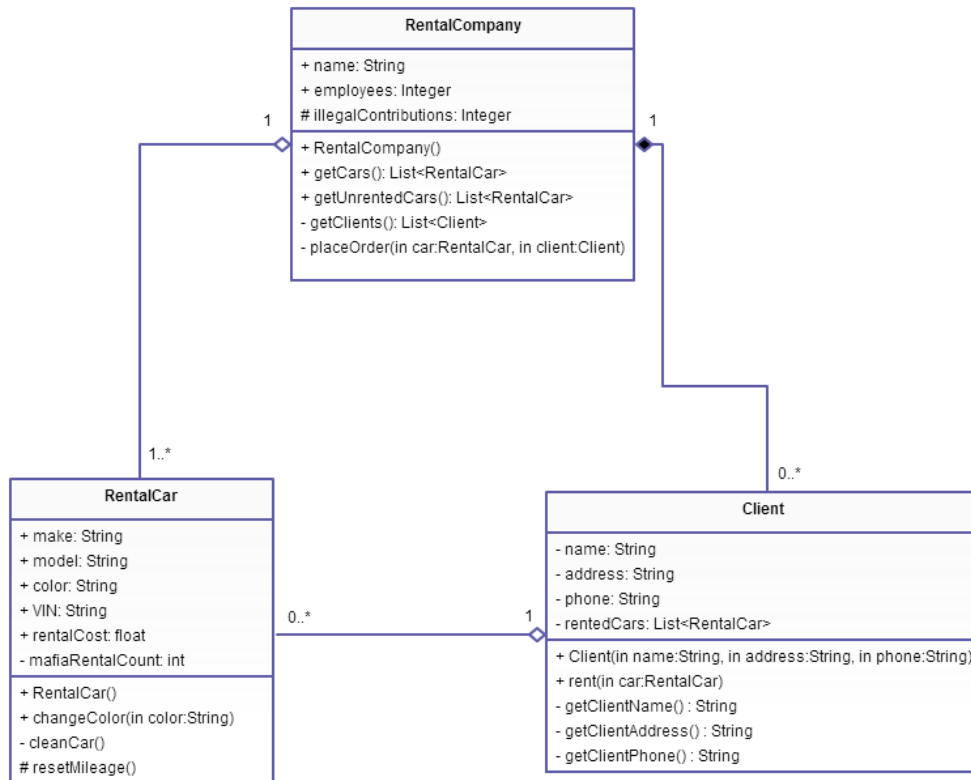

Figure 2: UML Diagram 2.