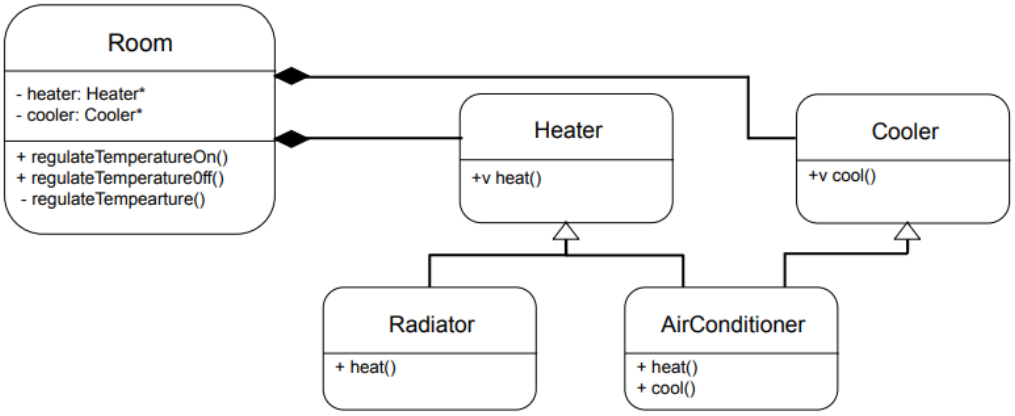
As explained in Note2, we want to decouple the heating and cooling functions from the “Room” class. To do so, we can create abstract classes “Heater” and “Cooler”. Since the implementation of these functions may vary across different classes, it is an ideal scenario to apply the state pattern. Following this approach, the updated solution is shown in the figure below.



The new design now has fewer hierarchies. It is more maintainable, expandable, and reusable.

I am fully aware that the actual temperature change in a room depends on various factors, such as its size, insulation, and the power of the heating/cooling unit. However, for the sake of simplicity, I will assume that the heat() and cool() functions directly affect the room temperature.