Matthew Mahan

Dr. Bowman

CSC 403 001

13 October 2023

Object Oriented Design Patterns: Memento

The Memento pattern is a behavioral design pattern for cases in which a class must return to a previous state. For example, an art program that allows the user to undo or redo a brush stroke implements the Memento pattern to undo that brush stroke. Another example may be a video game that allows the player to save the game and exit. Then at a later time, the player may boot the game again and load the previous state of the game to resume their progress.

There are two formal aspects to the problem a Memento solves:

1. The internal state of an object at a specific time needs to be saved externally so that it may be accessed at a later time regardless of the internal state of the object at that later time.
2. The internal state of the object should not be exposed. This means the object’s encapsulation should not be violated.

The solution described by the pattern involves three classes. These classes are the Originator, the Memento, and the Caretaker. The Originator class is the class whose state has some significance and needs to be recorded. The Memento is a class which is able to hold a copy of the Originators internal state. The Originator is responsible for saving its state to a Memento as well as loading state from a Memento. Finally, the Caretaker class is responsible for the handling of Mementos when the Originator is not using them. The Caretaker may request the Originator to create a new Memento or the Caretaker may request the Originator to set its state to a passed in Memento.

In total, the effect of this pattern is that copies of the Originator’s state may be created and loaded arbitrarily without decapsulating the data it contains. Data is only ever passed between the interfaces of the objects and is never located outside. This is advantageous for keeping the state safe. Also, using techniques of serialization, a generic Memento can be used for many Originators. However, one disadvantage is that a very large Originator with produce an equally large Memento. This has the potential to consume large amounts of memory or storage if excessive Mementos are created.