SENG5802: Software Engineering II - Software Design
Isolating Variability and Simplifying Responsibilities

Outline

Outline

Outline

State pattern

Sometimes the behavior of an object depends on its state.

- Each incoming event may cause some action, and possibly a change in state.
- Think of a network connection. States might be Listening, Established, Closed.

Sometimes you find yourself writing switch or case statements, or if-then-else chains, in multiple methods, checking the same "conditions".

- Are you seeing multiple states? Are you keeping track of which state you are in? Do you know what causes state to change?

The pattern

Support state-dependent behavior.

ConcreteState object may need access to internals of Context.

Often implemented as inner classes.

State machines

A finite state machine is easily implemented as a Context with a set of State objects.

- Each state type inspects the event, takes action, sets the next state.
- Some or all of the behavior can be specified by metadata.

Outline

Template Method

Sometimes what varies are the details of an algorithm. The sequence of steps is static, but the individual steps need to vary.

How it works

Define the “skeleton” of an algorithm, defer individual steps to subclasses.

Outline

Outline

Command Pattern

Sometimes you need to encapsulate “operations” in some domain.

- To support “undo” functionality in a user interface.
- To communicate user intent between a remote UI and a backend service.
- To support user scripting.

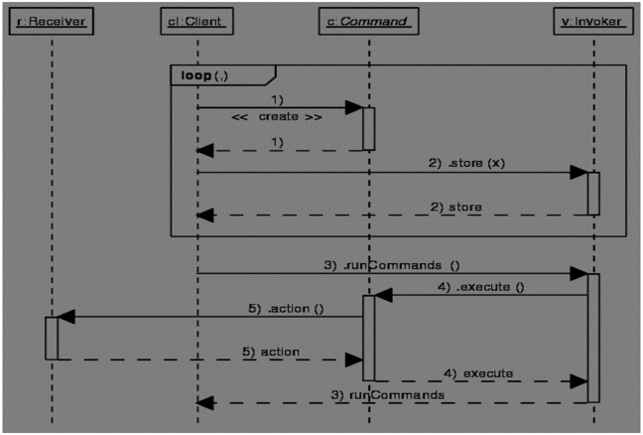
Command explained

Client builds up a list of Commands with respect to some Receiver.

Later, an Invoker actually runs them.

An **UndoableCommand** has an **undo()** method.

Interaction



Timers

Common example of the use of Command.

Set up operations that take place later, when some event is detected or after some delay.

Example: Delayed action

Outline

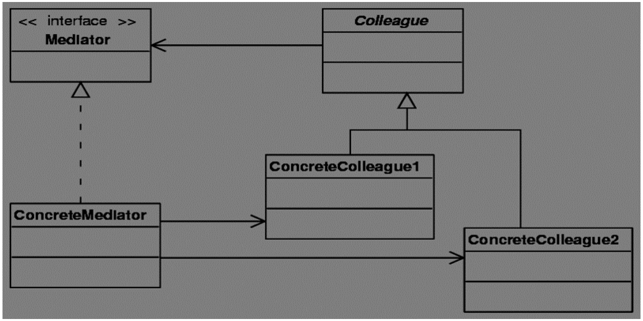
Mediator Pattern

A common problem with frameworks (GUI frameworks, for example) is the need to interconnect various elements with application-specific behaviors.

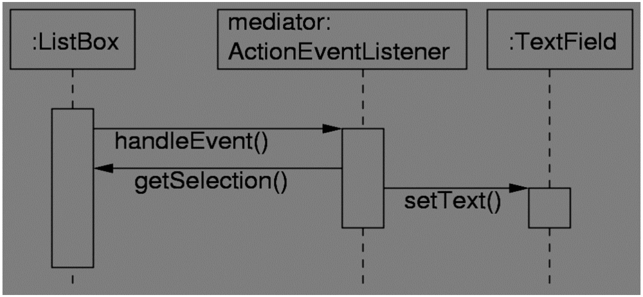
Think about the need to have the text in some field change when a button is poked.

The framework classes do not contain the necessary interconnections.

Mediator Pattern



Interaction



Example

Events are Commands.

Entity types commonly use the State pattern.

If Event type is supplied by framework, the act() method can be handled by a Mediator.

Outline

Singleton Pattern

A design often calls for just one of a something.

You have two obvious choices:

1. Class with static attributes and methods
2. Single instance of some class

Singleton Details

Usually created with “lazy evaluation” when getInstance() is first called

Constructor should be private or protected

In some languages (Java, Smalltalk), the only true globals are named classes