

Родственные паттерны

Компоновщик: посетители могут использоваться для выполнения операции над всеми объектами структуры, определенной с помощью паттерна компоновщик.

Интерпретатор: посетитель может использоваться для выполнения интерпретации.

Обсуждение паттернов поведения

Инкапсуляция вариаций

Инкапсуляция вариаций – элемент многих паттернов поведения. Если определенная часть программы подвержена периодическим изменениям, эти паттерны позволяют определить объект для инкапсуляции такого аспекта. Другие части программы, зависящие от данного аспекта, могут кооперироваться с ним. Обычно паттерны поведения определяют абстрактный класс, с помощью которого описывается инкапсулирующий объект. Своим названием паттерн как раз и обязан этому объекту.¹ Например:

- объект стратегия инкапсулирует алгоритм;
- объект состояние инкапсулирует поведение, зависящее от состояния;
- объект посредник инкапсулирует протокол общения между объектами;
- объект итератор инкапсулирует способ доступа и обхода компонентов составного объекта.

Перечисленные паттерны описывают подверженные изменениям аспекты программы. В большинстве паттернов фигурируют два вида объектов: новый объект (или объекты), который инкапсулирует аспект, и существующий объект (или объекты), который пользуется новыми. Если бы не паттерн, то функциональность новых объектов пришлось бы делать неотъемлемой частью существующих. Например, код объекта-стратегии, вероятно, был бы «зашит» в контекст стратегии, а код объекта-состояния был бы реализован непосредственно в контексте состояния.

Но не все паттерны поведения разбивают функциональность таким образом. Например, паттерн цепочка обязанностей связан с произвольным числом объектов (то есть цепочкой), причем все они могут уже существовать в системе.

Цепочка обязанностей иллюстрирует еще одно различие между паттернами поведения: не все они определяют статические отношения взаимосвязи между классами. В частности, цепочка обязанностей показывает, как организовать обмен информацией между заранее неизвестным числом объектов. В других паттернах участвуют объекты, передаваемые в качестве аргументов.

Объекты как аргументы

В нескольких паттернах участвует объект, *всегда* используемый только как аргумент. Одним из них является посетитель. Объект-посетитель – это аргумент

¹ Эта тема красной нитью проходит и через другие паттерны. Абстрактная фабрика, строитель и прототип инкапсулируют знание о том, как создаются объекты. Декоратор инкапсулирует обязанности, которые могут быть добавлены к объекту. Мост отделяет абстракцию от ее реализации, позволяя изменять их независимо друг от друга.