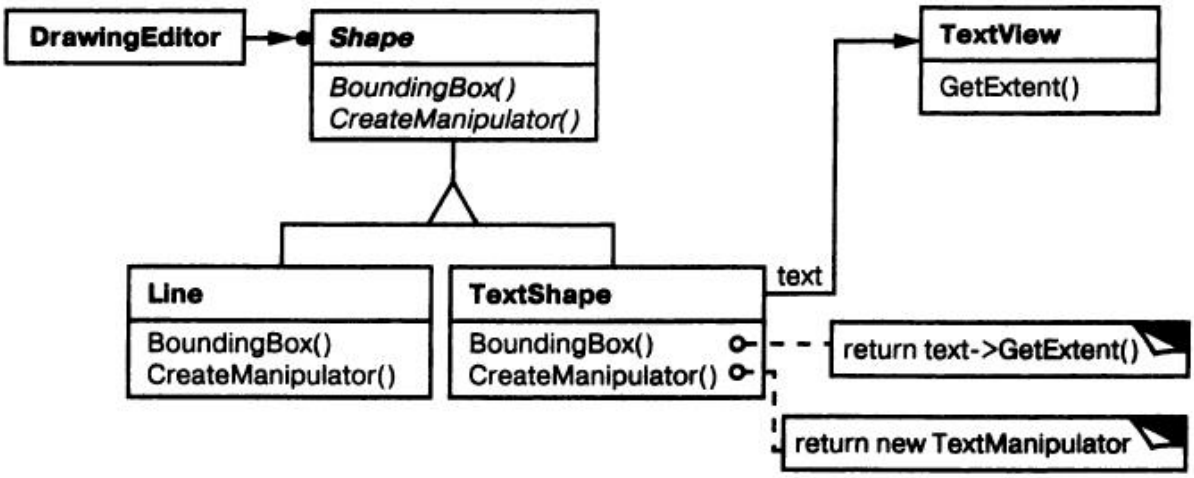
Структурные паттерны

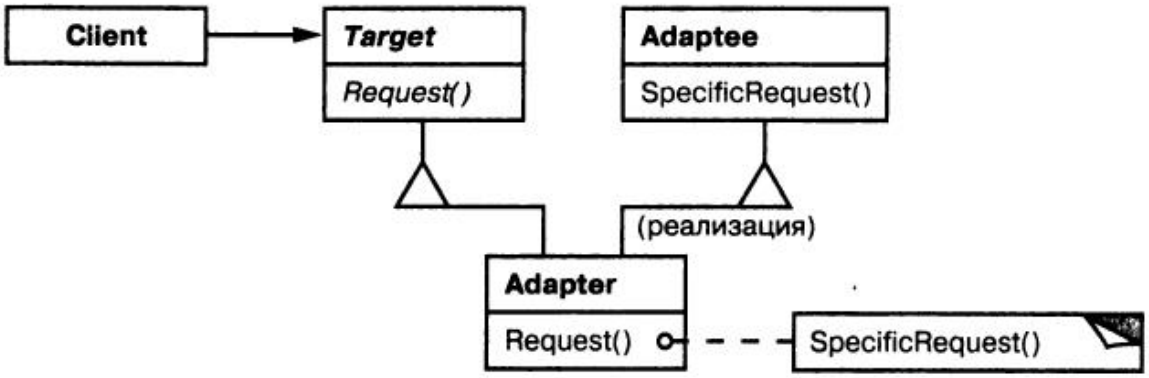
Adapter

Преобразует интерфейс одного класса в интерфейс другого, который ожидают клиенты. Обеспечивает совместную работу классов с несовместимыми интерфейсами, которая без него была бы невозможна.

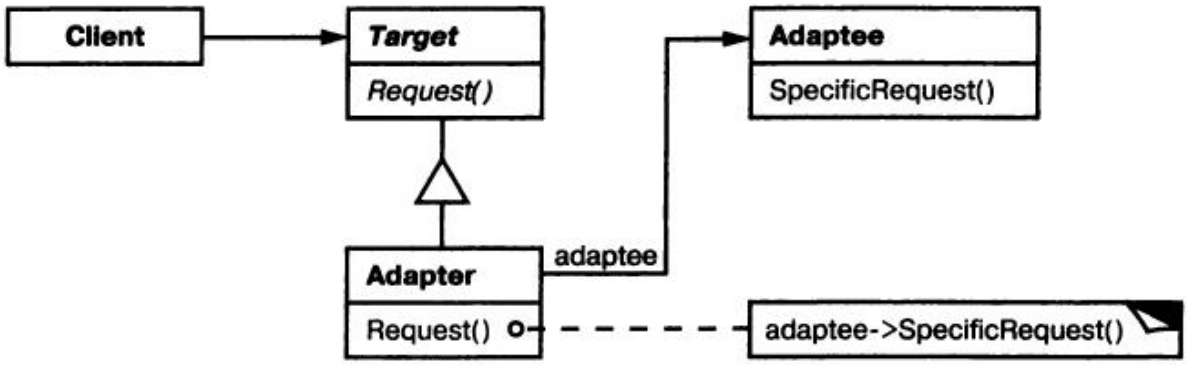
Пример: использование сторонней библиотеки в проекте.



Структура (адаптер класса; множественное наследование):



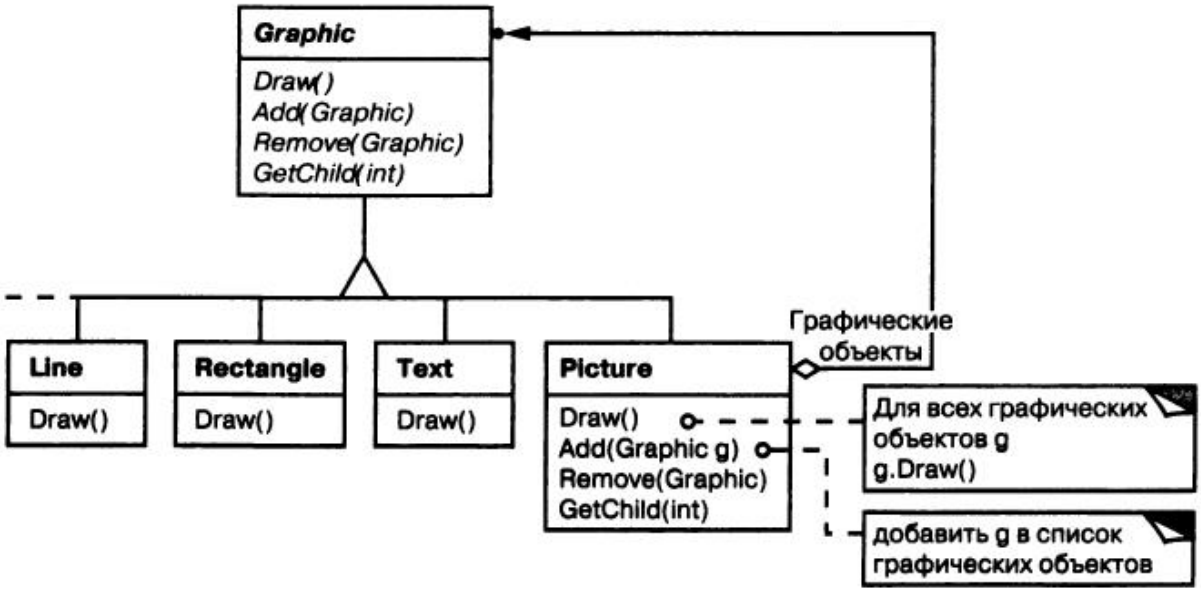
Структура (адаптер объекта; композиция):



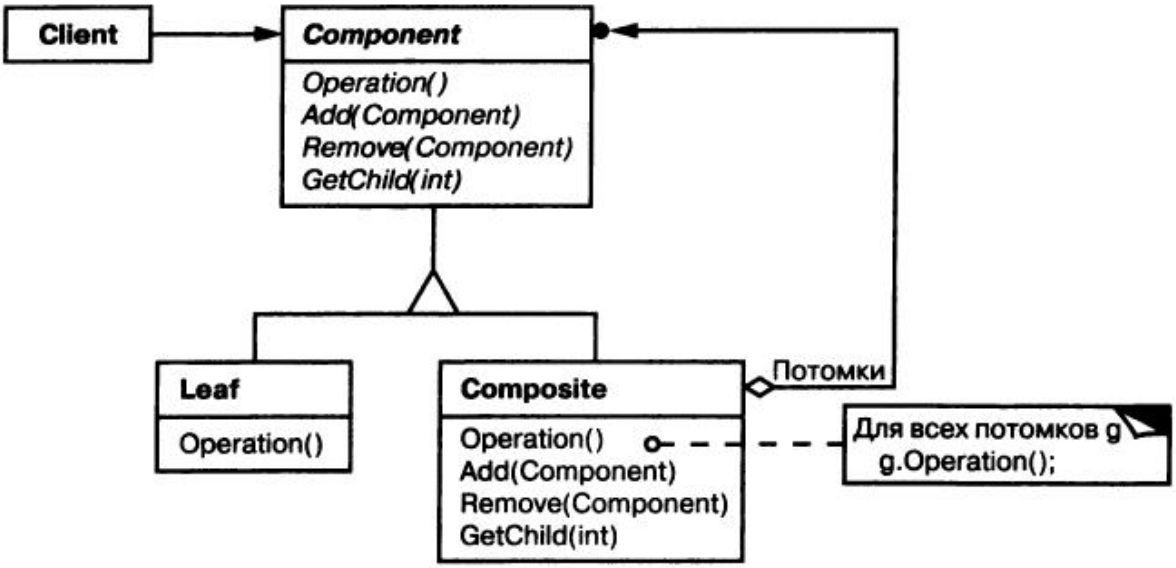
Composite

Компонует объекты в древовидные структуры для представления иерархий часть-целое. Позволяет клиентам единообразно трактовать индивидуальные и составные объекты.

Пример: флэппи-бёрд (птица, труба, бэкграунд — лэйаут).



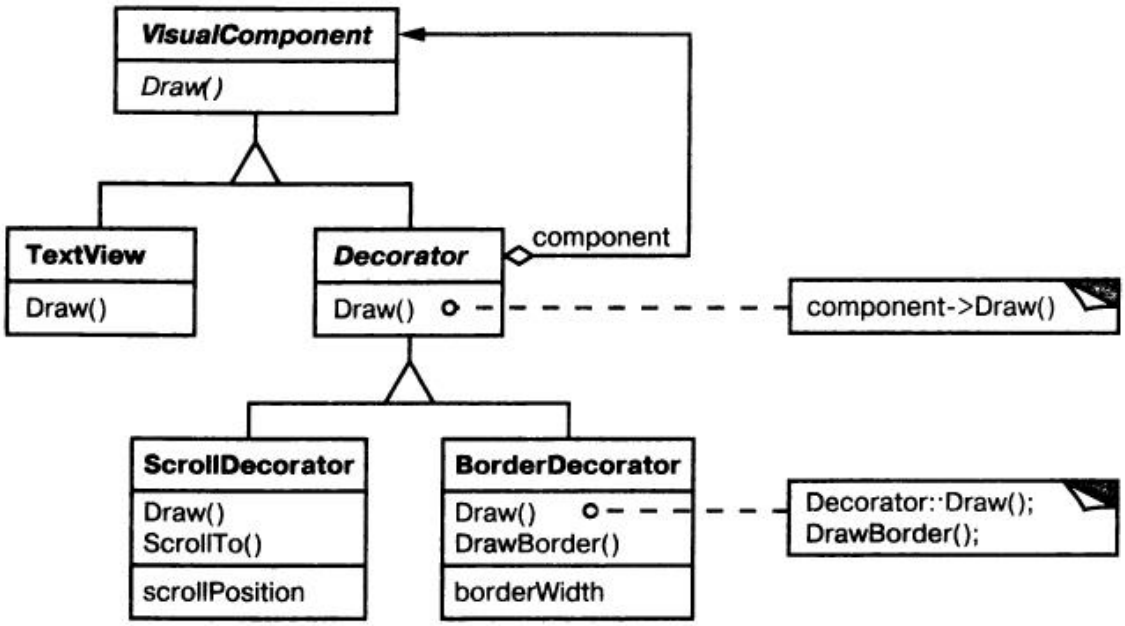
Структура:



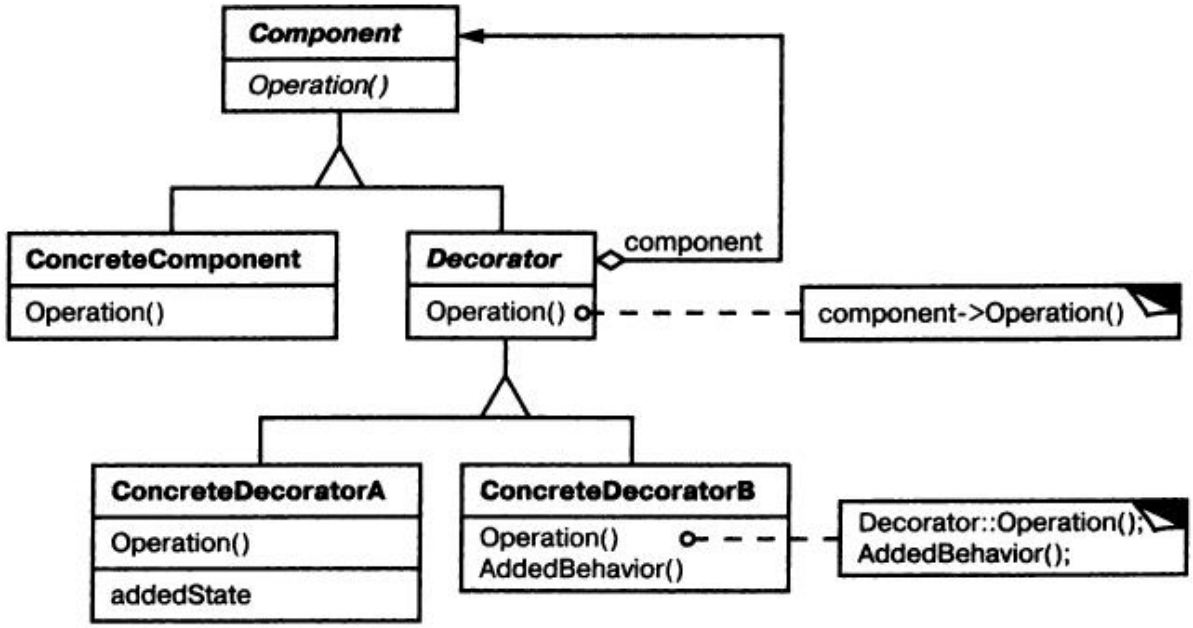
Decorator

Динамически добавляет объекту новые обязанности. Является гибкой альтернативой порождению классов с целью расширения функциональности.

Пример: текствью с бордером и скроллом.



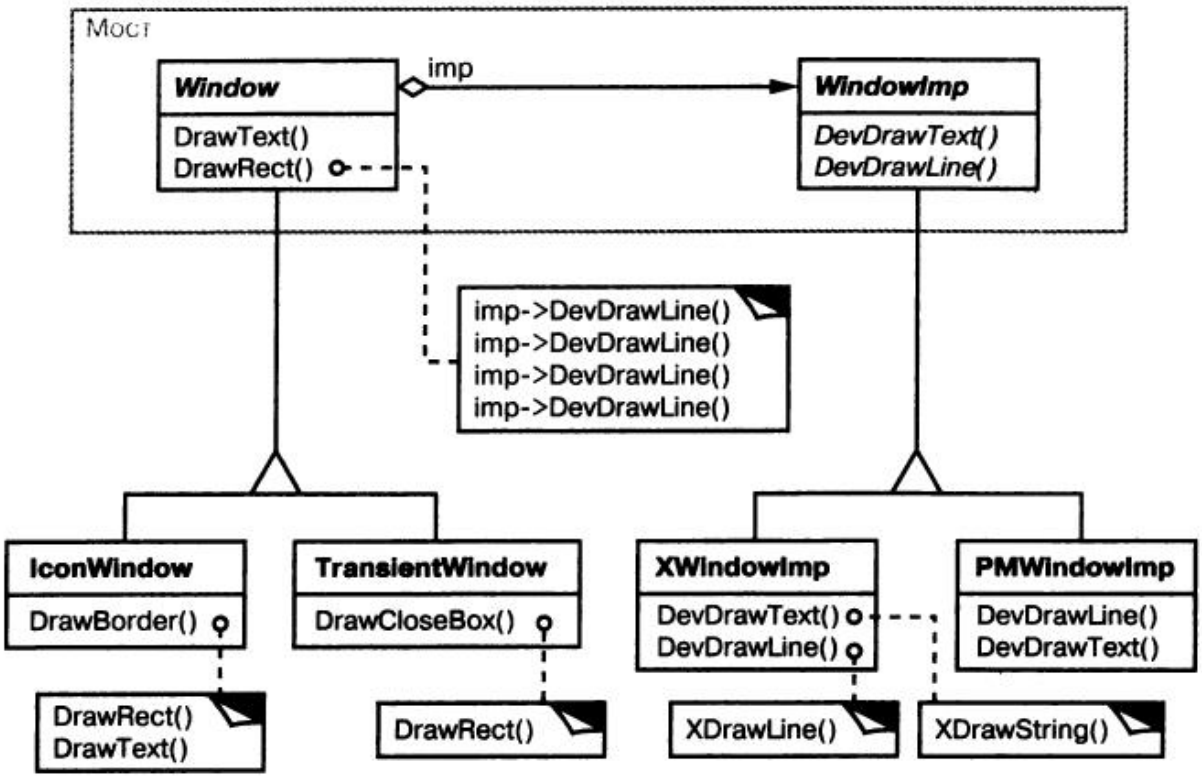
Структура:



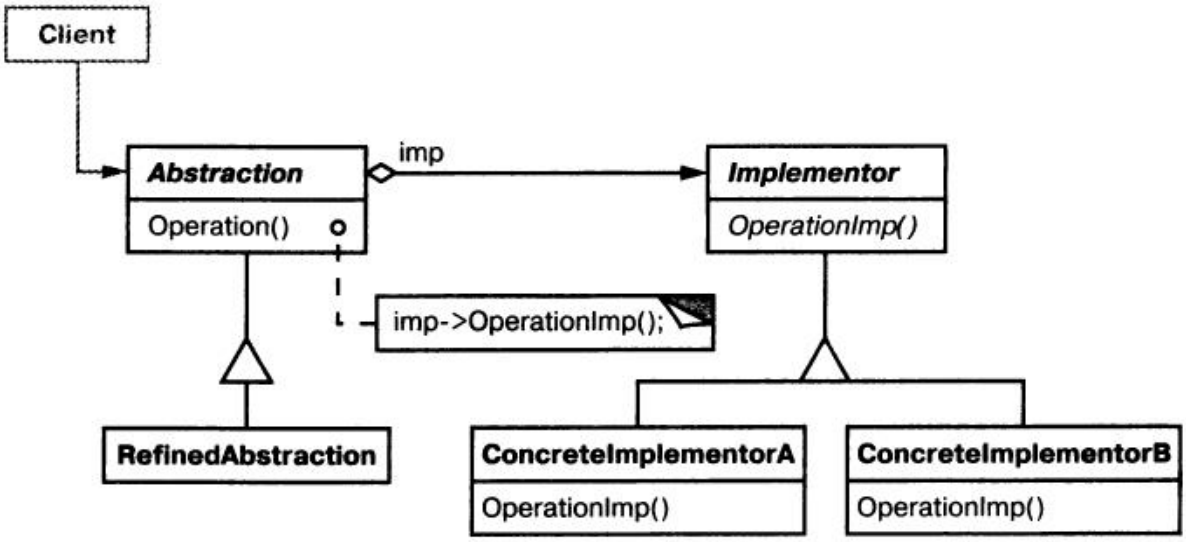
Bridge

Отделяет абстракцию от её реализации так, чтобы то и другое можно было изменять независимо.

Пример: окна в системах; сервисы и использование нескольких протоколов.



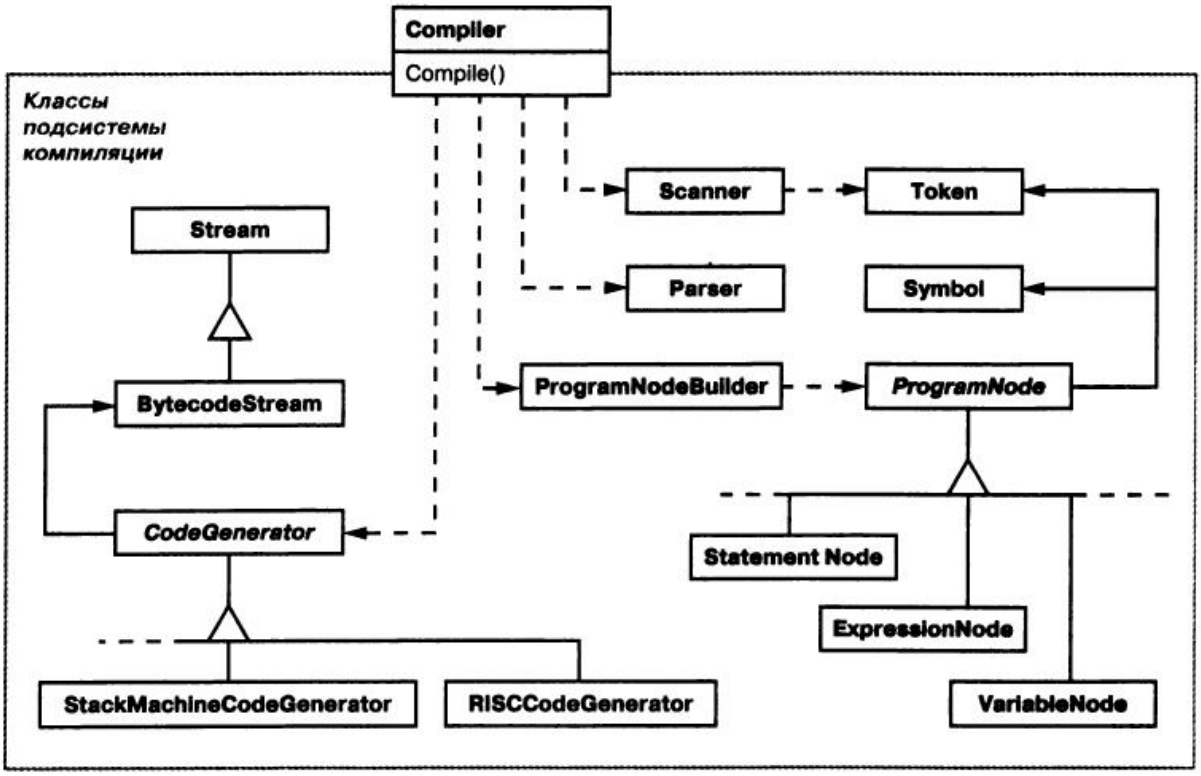
Структура:



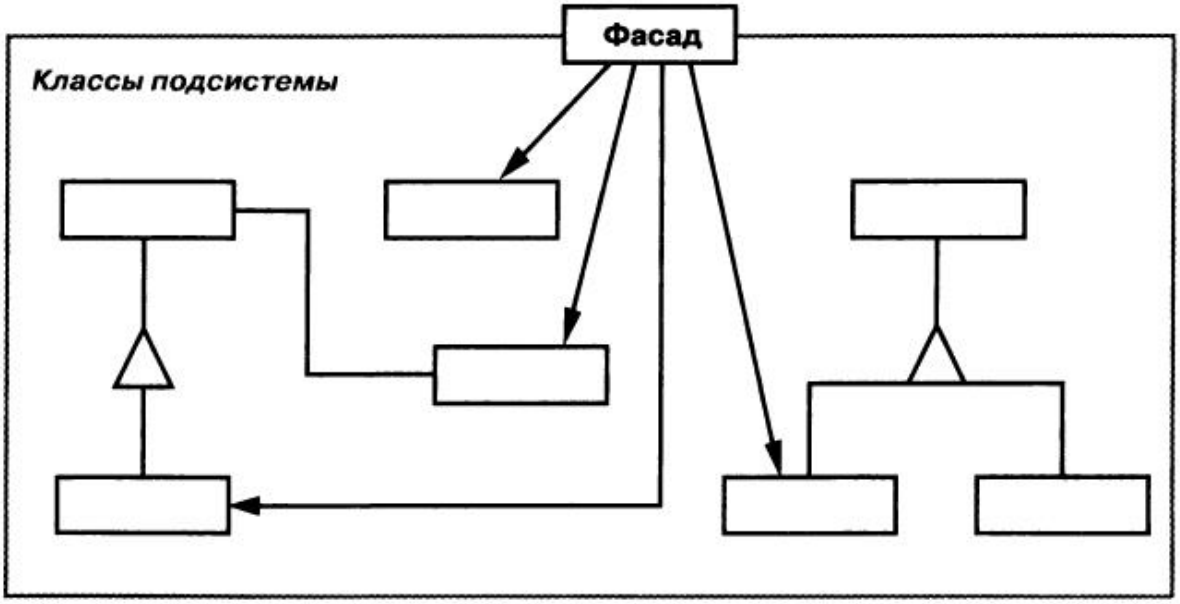
Facade

Предоставляет унифицированный интерфейс вместо набора интерфейсов некоторой подсистемы. Определяет интерфейс более высокого уровня, который упрощает использование подсистемы.

Пример: много библиотек, какие-то нужно объединить.



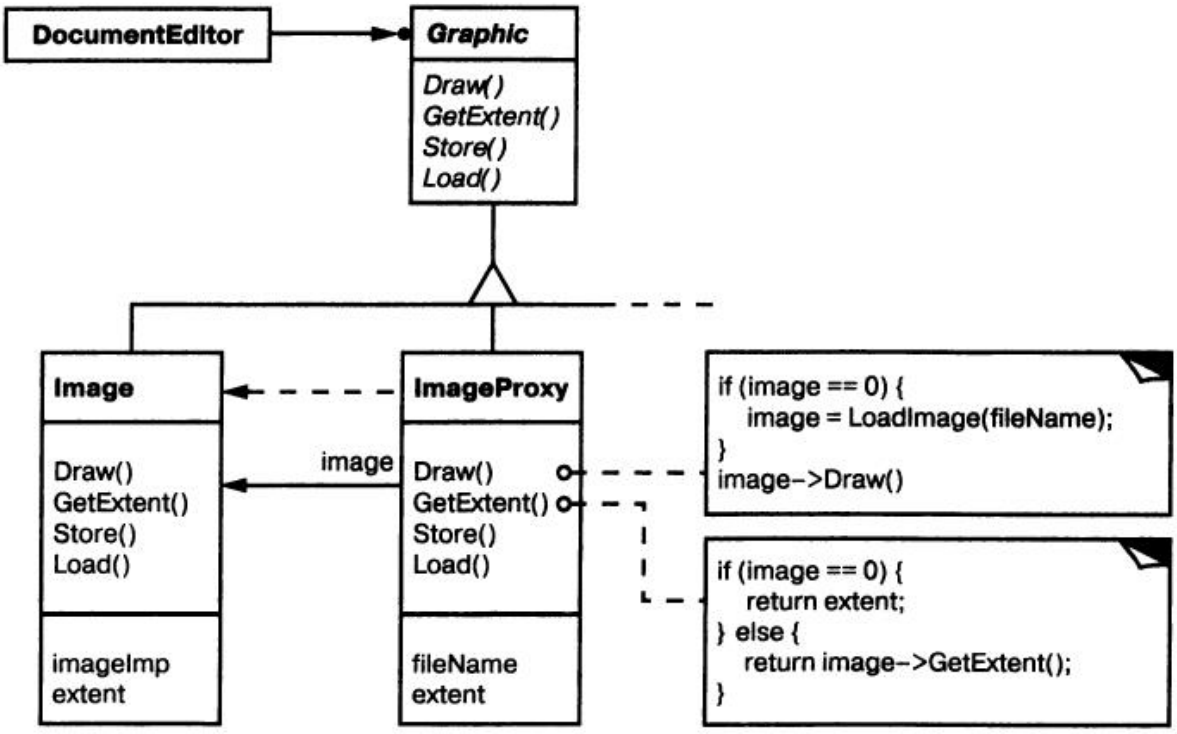
Структура:



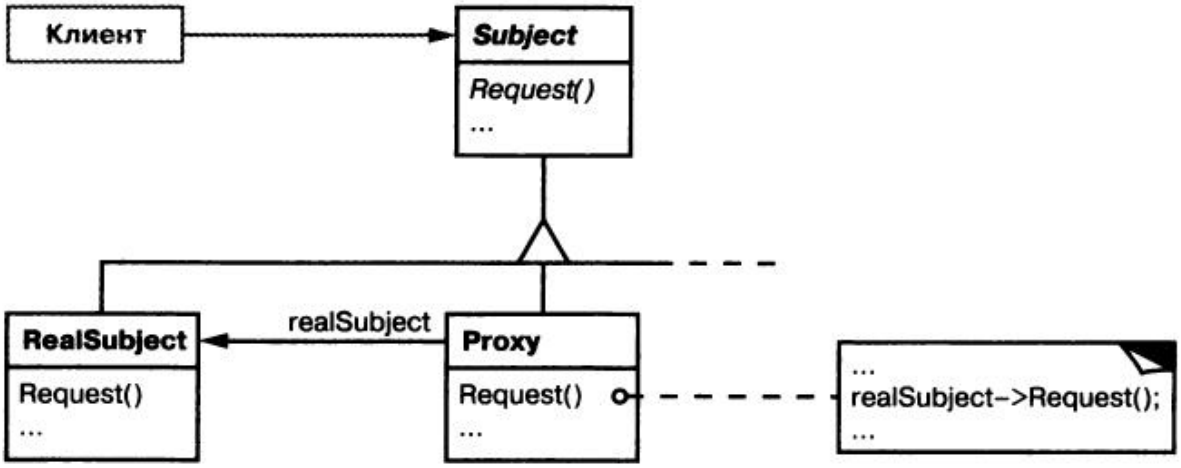
Proxy

Является суррогатом другого объекта и контролирует доступ к нему.

Пример: ограничение доступа к новостям через прокси.



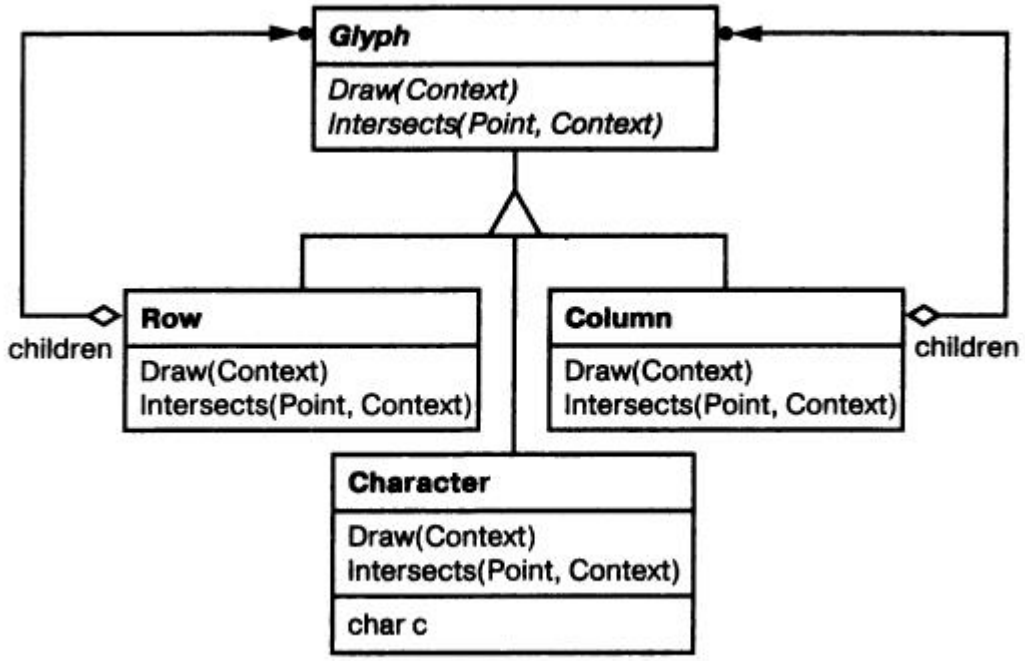
Структура:



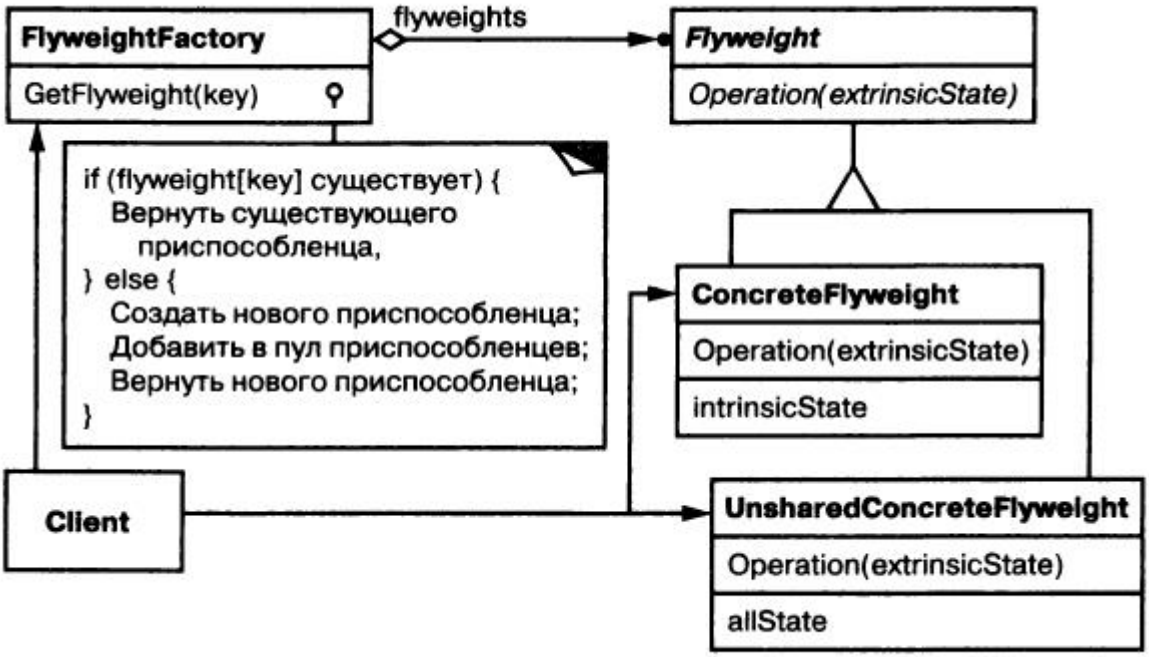
Flyweight

Использует разделение для эффективной поддержки множества мелких объектов.

Пример: отрисовка символов в тексте.



Структура:


FlyweightFactory
GetFlyweight(key)
flyweights
Flyweight
Operation(extrinsicState)
if (flyweight[key] существует) {
Вернуть существующего
приспособленца,
} else {
Создать нового приспособленца;
Добавить в пул приспособленцев;
Вернуть нового приспособленца;
}
ConcreteFlyweight
Operation(extrinsicState)
intrinsicState
Client
UnsharedConcreteFlyweight
Operation(extrinsicState)
allState