Java Design Patterns

Flyweight

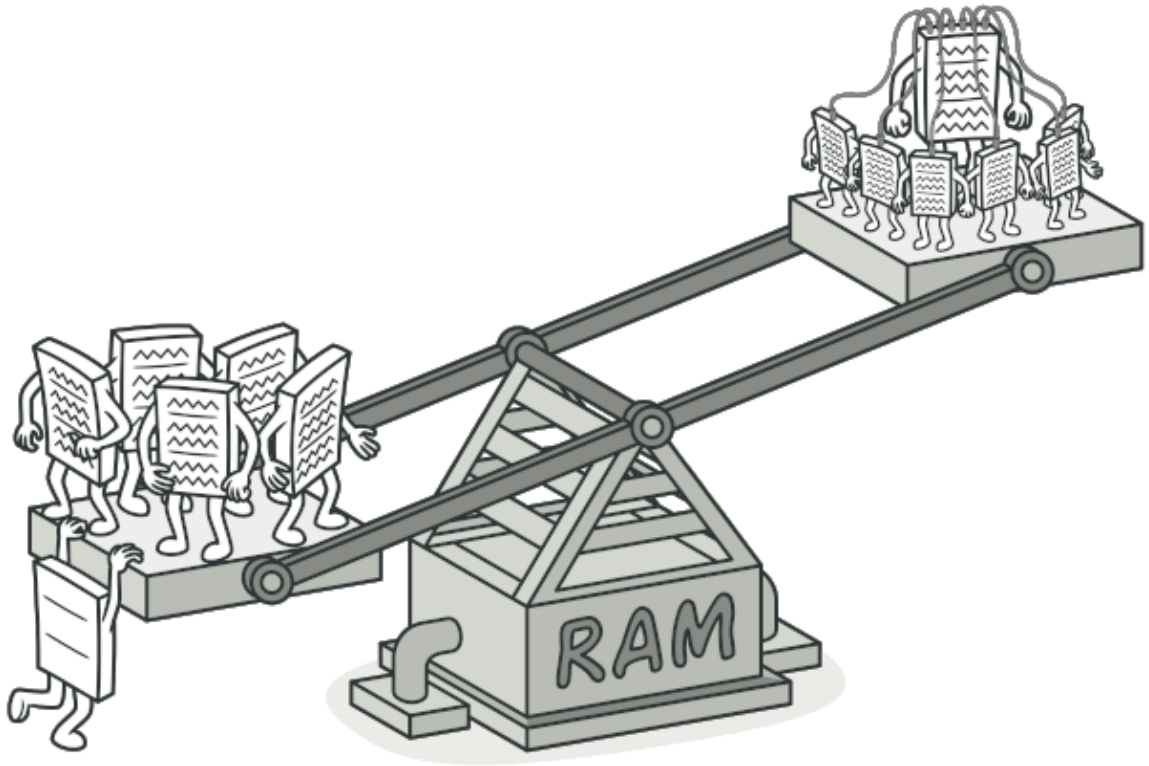
Тема

Flyweight

Суть паттерна

Легковес

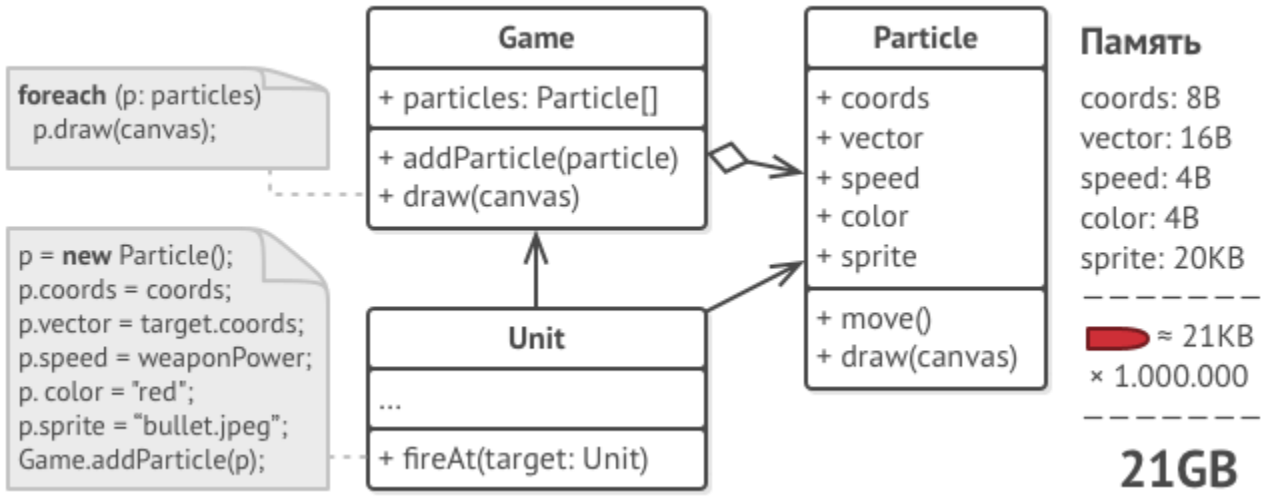
Легковес — это структурный паттерн проектирования, который позволяет вместить большое количество объектов в отведённую оперативной память за счёт экономного разделения общего состояния объектов между собой, вместо хранения одинаковых данных в каждом объекте.



Проблема

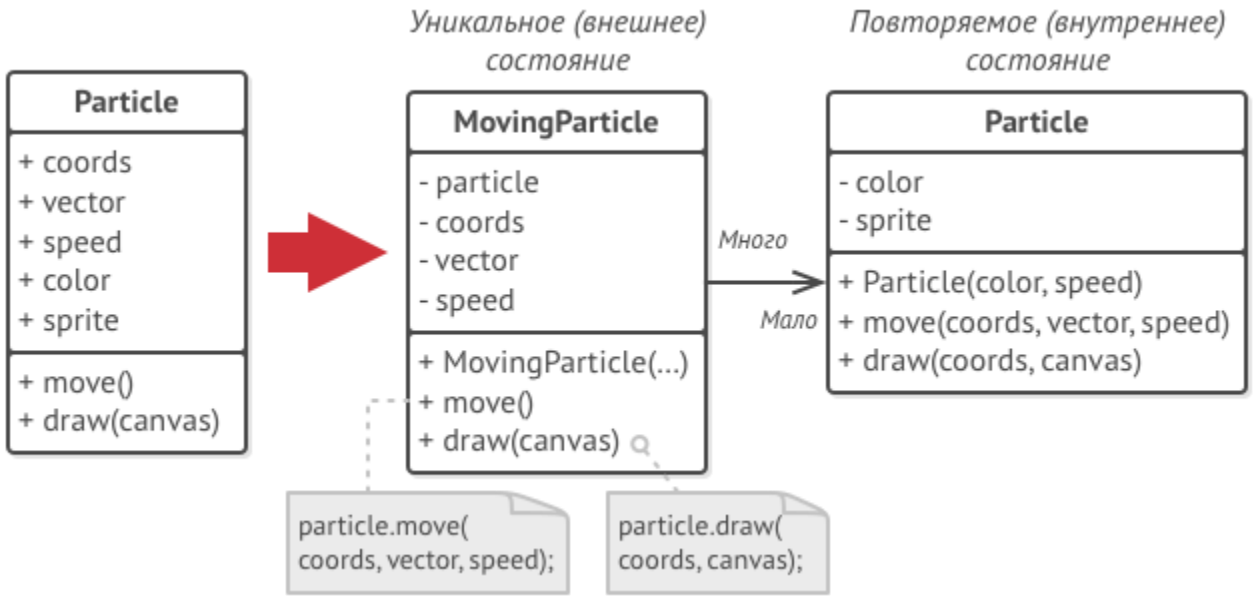
Постановка задачи

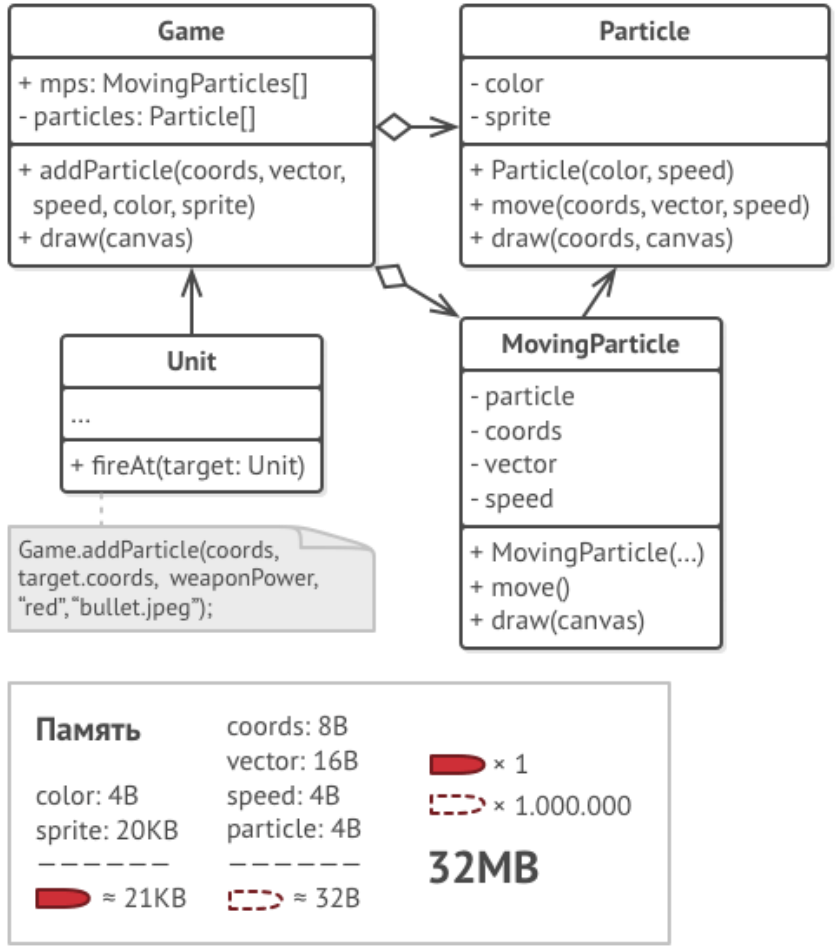
На досуге вы решили написать небольшую игру-стрелялку, в которой игроки перемещаются по карте и стреляют друг в друга. Фишкой игры должна была стать реалистичная система частиц. Пули, снаряды, осколки от взрывов — всё это должно красиво летать и радовать взгляд.



Решение задачи

Если внимательно посмотреть на класс частиц, то можно заметить, что цвет и спрайт занимают больше всего памяти. Более того, они хранятся в каждом объекте, хотя фактически их значения одинаковые для большинства частиц.

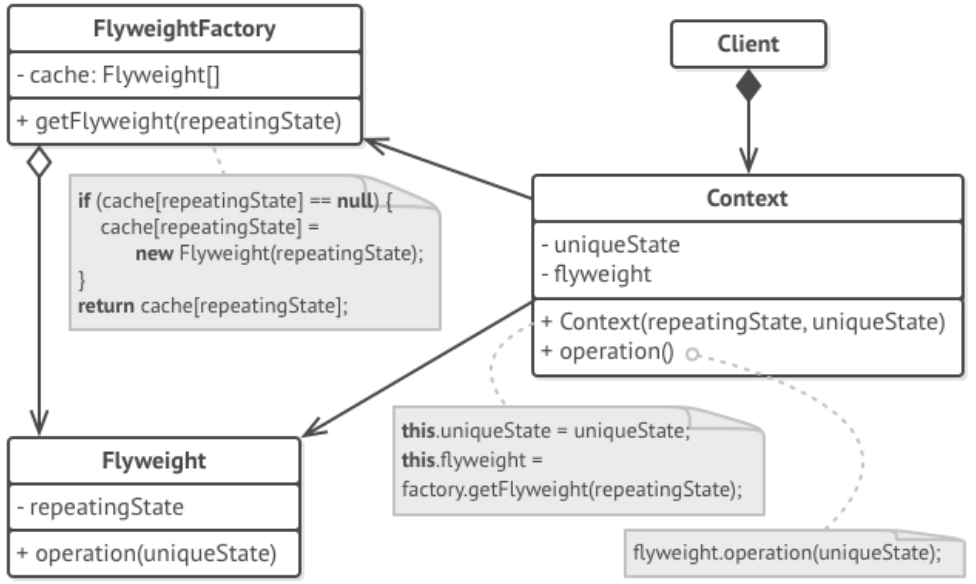




Структура

Структура паттерна

1. Вы всегда должны помнить о том, что Легковес применяется в программе, имеющей громадное количество одинаковых объектов. Паттерн разделил данные этих объектов на две части — контексты и легковесы.
2. Легковес содержит состояние, которое повторялось во множестве первоначальных объектов. Один и тот же легковес можно использовать в связке с множеством контекстов. Состояние, которое хранится здесь, называется *внутренним*, а то, которое он получает извне — *внешним*.
3. Контекст содержит «внешнюю» часть состояния, уникальную для каждого объекта. Контекст связан с одним из объектов-легковесов, хранящих оставшееся состояние.
4. Поведение оригинального объекта чаще всего оставляют в Легковесе, передавая значения контекста через параметры методов.
5. Клиент вычисляет или хранит контекст, то есть внешнее состояние легковесов. Для клиента легковесы выглядят как шаблонные объекты, которые можно настроить во время использования, передав контекст через параметры.
6. Фабрика легковесов управляет созданием и повторным использованием легковесов.



Применимость

Применение паттерна

1. Когда не хватает оперативной памяти для поддержки всех нужных объектов.

Шаги реализации

Алгоритм реализации паттерна

1. Разделите поля класса, который станет легковесом, на две части:
   - внутреннее состояние: значения этих полей одинаковы для большого числа объектов.
   - внешнее состояние (контекст): значения полей уникальны для каждого объекта.
2. Оставьте поля внутреннего состояние в классе, но убедитесь, что их значения неизменяемы. Эти поля должны инициализироваться только через конструктор.
3. Превратите поля внешнего состояния в аргументы методов, где эти поля использовались. Затем, удалите поля из класса.
4. Создайте фабрику, которая будет кешировать и повторно отдавать уже созданные объекты. Клиент должен запрашивать легковеса с определённым внутренним состоянием из этой фабрики, а не создавать его напрямую.
5. Клиент должен хранить или вычислять значения внешнего состояния (контекст) и передавать его в методы объекта легковеса.

Плюсы и недостатки

Плюсы:

- Экономит оперативную память.

Минусы:

- Расходует процессорное время на поиск/вычисление контекста.
- Усложняет код программы за счёт множества дополнительных классов.

Отношение с другими паттернами

1. Компоновщик часто совмещают с Легковесом, чтобы реализовать общие ветки дерева и сэкономить при этом память.
2. Легковес показывает, как создавать много мелких объектов, а Фасад показывает, как создать один объект, который отображает целую подсистему.
3. Паттерн Легковес может напоминать Одиночку, если для конкретной задачи у вас получилось уменьшить количество объектов к одному. Но помните, что между паттернами есть два кардинальных отличия:
   - В отличие от *Одиночки*, вы можете иметь множество объектов-легковесов.
   - Объекты-легковесов должны быть неизменяемыми, тогда как объект-одиночки допускает изменение своего состояния.

Информационный видеосервис для разработчиков программного обеспечения

ITVDN

IT VIDEO DEVELOPERS NETWORK

TestProvider PROMETRIC TEST CENTER CyberBionic systematics Microsoft Partner Silver Learning Windows Azure Cloud Partner EBA European Business Association