

Introducción a los patrones GOF

Diseño de Sistemas Software

Curso 2021/2022

Carlos Pérez Sancho



Universitat d'Alacant
Universidad de Alicante

Departament de Llenguatges i Sistemes Informàtics
Departamento de Lenguajes y Sistemas Informáticos

Patrones GOF

Documentados por primera vez en [Gamma et al., 1994], es un catálogo de patrones de diseño detallado.

El nombre GOF (Gang Of Four) hace referencia a los 4 autores del libro.

Proporcionan soluciones a problemas cotidianos en el desarrollo de software.

Se dividen en 3 tipos:

Creacionales Abstract factory, Builder, Factory method, Prototype, Singleton.

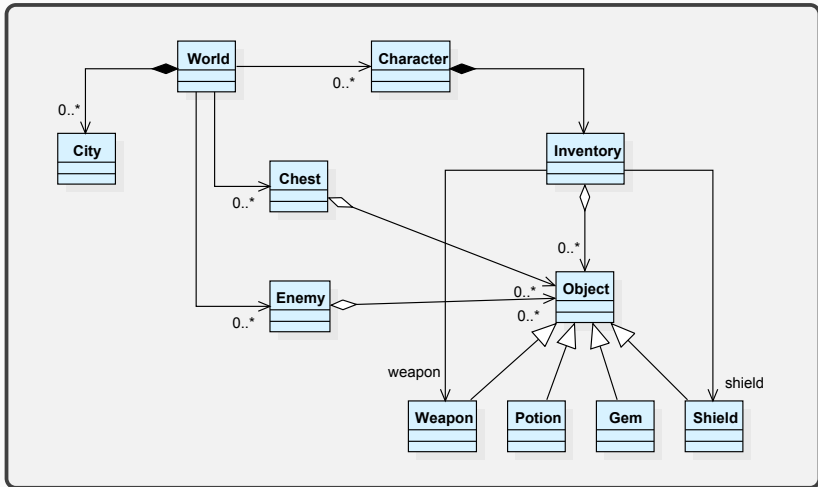
Estructurales Adapter, Bridge, Composite, Decorator, Facade, Flyweight, Proxy.

De comportamiento Chain of responsibility, Command, Interpreter, Iterator, Mediator, Memento, Observer, State, Strategy, Template method, Visitor.

Diseña un diagrama de clases para un juego de rol con las siguientes características:

- Hay numerosos personajes que se mueven por un mundo abierto, en el que pueden encontrar distintas ciudades
- Los personajes tienen un inventario en el que guardan un arma, un escudo y los objetos que van consiguiendo
- Se pueden conseguir objetos dentro de cofres o al derrotar a los pequeños enemigos que merodean por el mundo
- Los objetos pueden ser pociones, gemas, pergaminos, etc.

Solución



Aplicación de un patrón de comportamiento

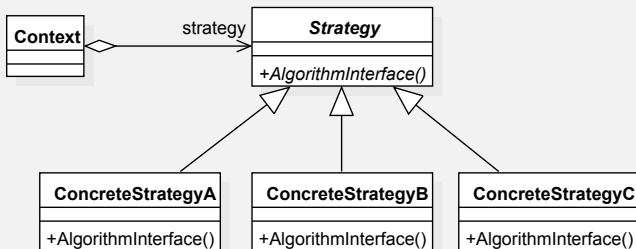
Ampliación del juego:

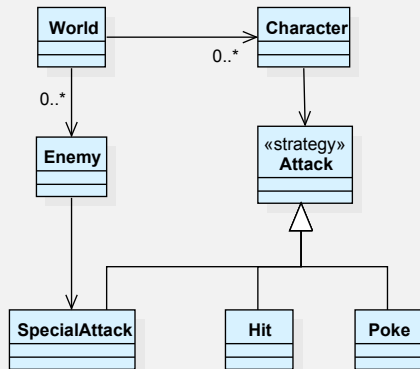
- En el juego los personajes podrán usar distintas armas, que se diferencian en los puntos de ataque (fuerza) que tienen
- En cada ataque, los personajes pueden elegir entre varios tipos de ataque: golpear, empalar o ataques especiales; ya que cada ataque afecta de forma distinta a cada enemigo
- Los personajes sólo podrán emplear los ataques especiales cuando los hayan obtenido al derrotar enemigos

Patrón Strategy (de comportamiento)

Strategy

Define una jerarquía de clases que representan algoritmos, los cuales son intercambiables. Estos algoritmos pueden ser intercambiados por la aplicación en tiempo de ejecución.





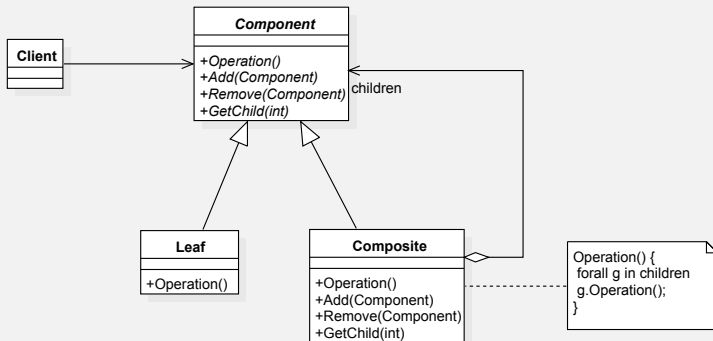
Ampliación del juego:

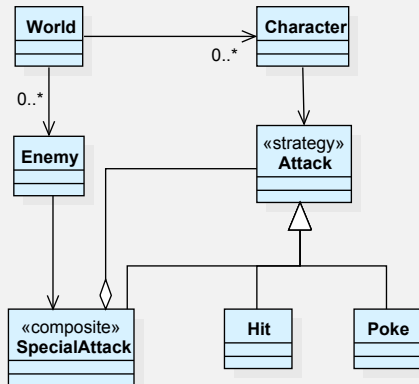
- Algunos ataques especiales consisten en combos: una combinación de otros ataques que el personaje ejecutará en secuencia

Patrón Composite (estructural)

Composite

Compone objetos en estructuras de árboles para representar jerarquías parte-todo. Permite que los clientes traten de manera uniforme a los objetos individuales y a los complejos.





Aplicación de un patrón creacional

Ampliación del juego:

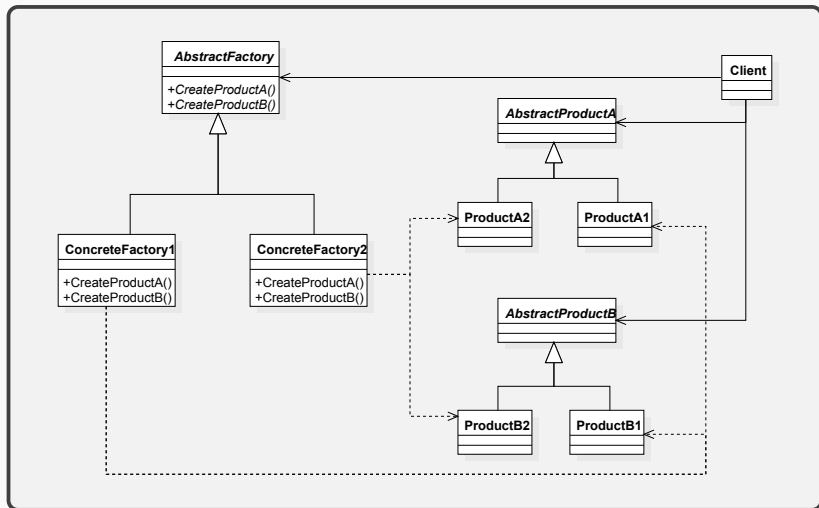
- Queremos que las ciudades se generen automáticamente a medida que los personajes recorren el mundo abierto. La clase `CityGenerator` se encargará de crearlas.
- Las ciudades están compuestas de distintos tipos de edificios: posadas, herreros, tiendas, etc.
- Los edificios son distintos en función de la raza a la que pertenecen. En el juego existen dos razas distintas, humanos y orcos, aunque se contempla que podrá haber más en el futuro.
- En ningún caso se podrán mezclar edificios de razas distintas en una misma ciudad.

Patrón Abstract factory (creacional)

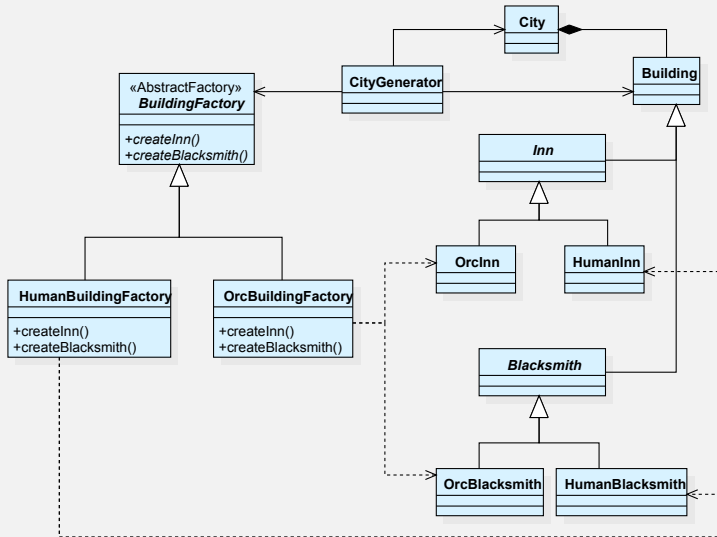
Abstract factory

Provee una interfaz para crear familias de objetos “producto” relacionados o que dependen entre sí, sin especificar sus clases concretas.

Patrón Abstract factory (creacional)



Solución



Para la próxima sesión, estudiar los patrones publicados en Moodle.



Gamma, E., Helm, R., Johnson, R., and Vlissides, J. M. (1994).

Design Patterns: Elements of Reusable Object-Oriented Software.

Addison-Wesley Professional.

Safari Books Online.