UNIVERSIDAD DE BURGOS
DEPARTAMENTO DE INGENIERÍA CIVIL
Área de Lenguajes y Sistemas Informático

Object Pool (Grand Volumen I pag 135)

Sinopsis / Propósito

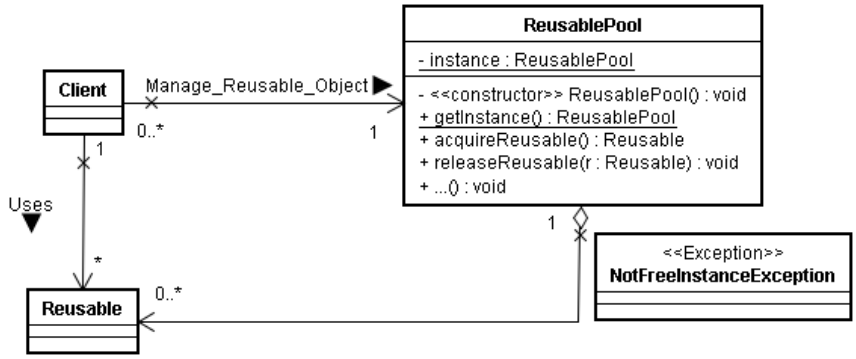
Gestiona la reutilización de objetos cuando un tipo de objetos es costoso de crear o sólo se puede crear un número limitado de objetos

Fuerzas

- Un programa sólo puede crear un número limitado de instancias para una clase en particular
- Si la creación de instancias de una clase es muy costosa, se debería evitar la nueva creación de instancias
- Un programa puede evitar crear nuevos objetos reutilizándolos cuando ellos han terminado su función en vez de tirarlos a la papelera (garbage)

Solución

Estructura



Participantes

- **`Reusable`**: Las instancias de clases en este rol colaboran con otros objetos durante un tiempo limitado. Después ellas no son necesarias para la colaboración.
- **`Client`**: Las instancias de clases en este rol usan los objetos de tipo `Reusable`.
- **`ReusablePool`**: Las instancias de clase en este rol gestionan la creación y obtención de objetos `Reusable` para ser usados por el objeto `Client`. Normalmente es deseable mantener todos los objetos `Reusable` que no se encuentran actualmente en uso en el mismo almacén para mantener una política coherente. Por ello `ReusablePool` esta diseñada como una clase `Singleton`. La política concreta utilizada en este ejemplo es mantener dos instancias de la clase `Reusable`. En el caso de recibir una petición y no existir instancias disponibles lanza una excepción **`NotFreeInstanceException`**

Colaboraciones

Un objeto `Client` invoca a `ReusablePool.acquireReusable()` cuando necesite un objeto `Reusable`. Cuando el `Client` deja de utilizar el objeto invoca al método `ReusablePool.releaseReusable(Reusable)` pasando como parámetro el objeto a liberar. La política de asignación y liberación de objetos `Reusable` esta implementada en `ReusablePool` *(*número de instancias de objetos `Reusable`, que hacer en el caso de recibir una petición y no existir instancias disponibles*...).*