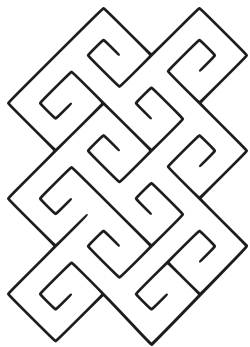


OBJETOS

Patrones de diseño



Agenda

- Patrones de comportamiento
 - Chain of responsibility
 - Command
 - Template method
 - Strategy
 - State

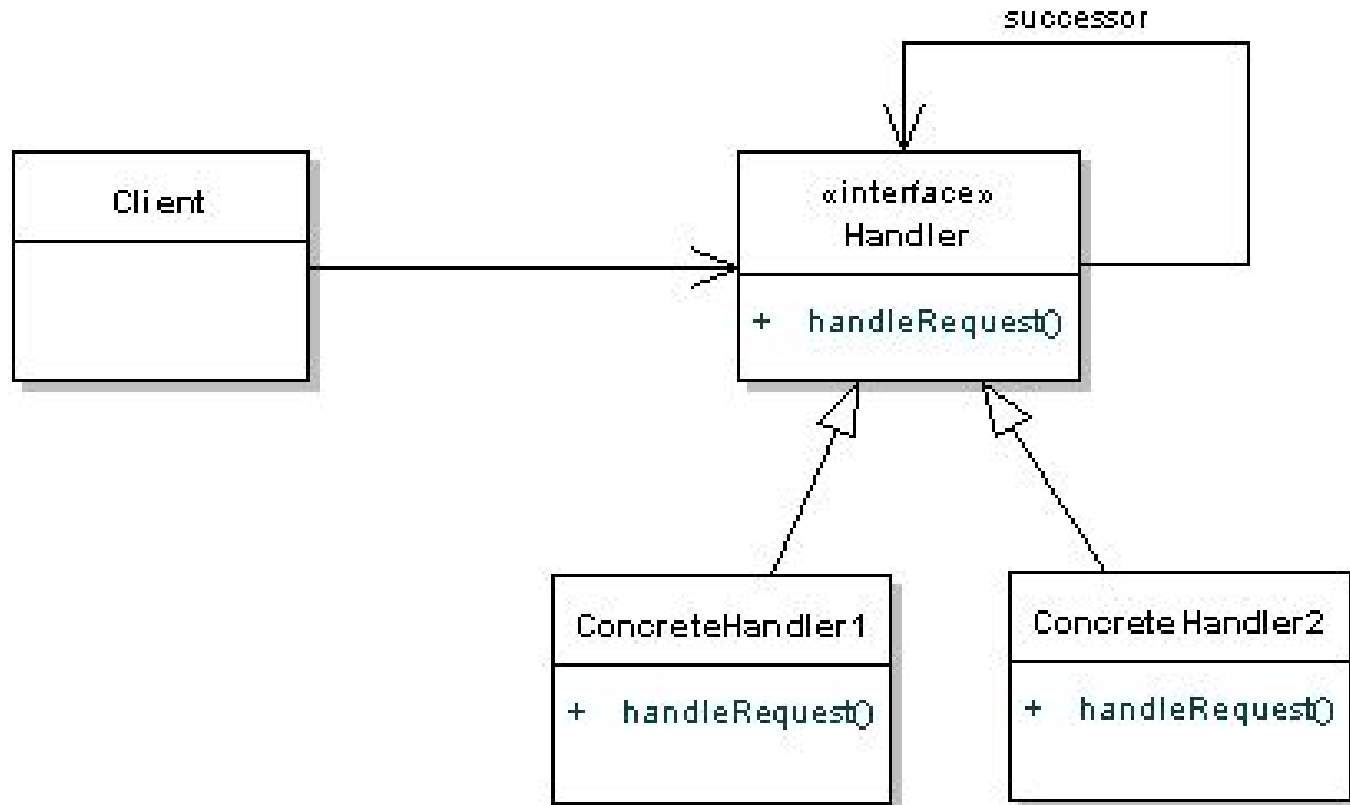


Chain of Responsibility

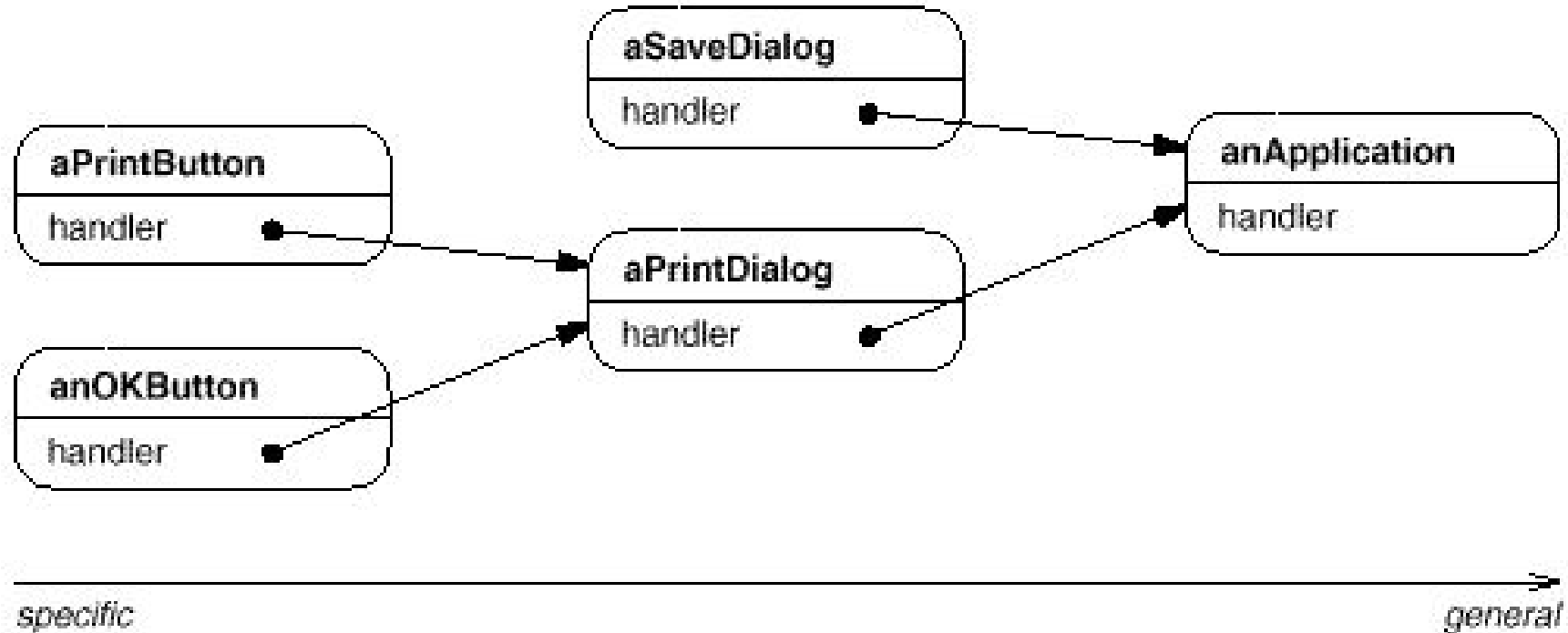
Desacopla al que emite un mensaje del que lo maneja. El mensaje se transmite a lo largo de la cadena hasta que es procesado por algún eslabón. Cada eslabón responde según un contexto, pero el mensaje es el mismo.

- Organiza una respuesta a una acción según una jerarquía
- Permite la ampliación mediante la implementación de nuevas respuestas
- Simplifica el problema a resolver dividiéndolo

Chain of Responsibility

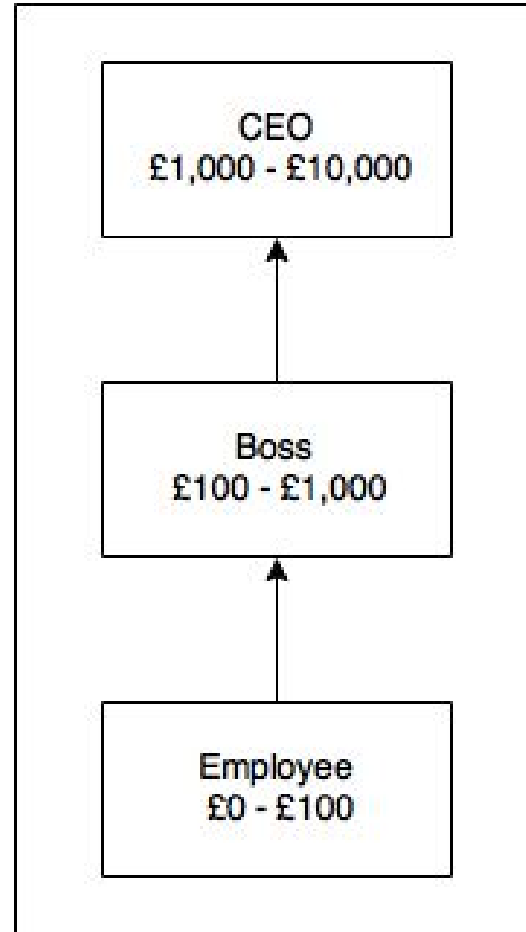


Chain of Responsibility



Chain of Responsibility

- Contratación de un servicio
- Aprobación de un gasto

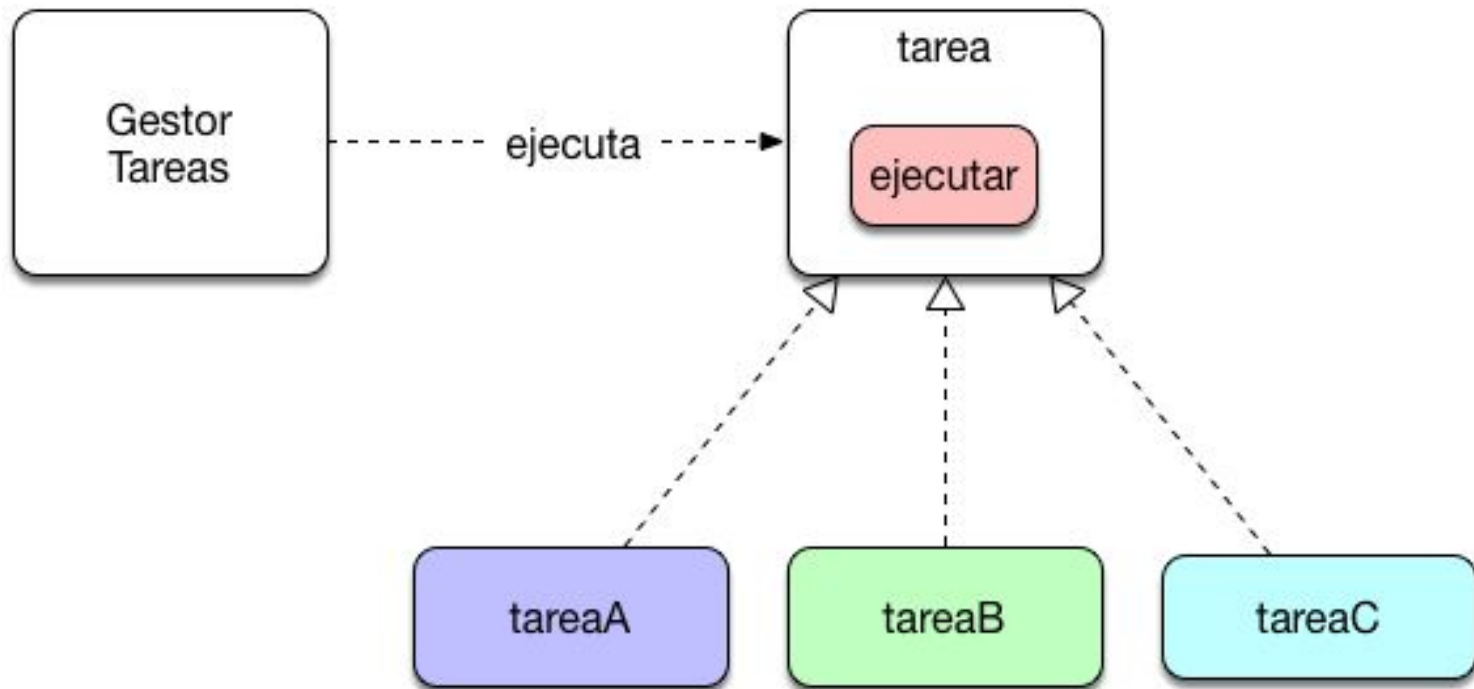


Command

Encapsula una orden o un pedido en forma de un objeto, permitiendo enviar callbacks para hacer uso de herramientas de propósito general en problemas puntuales.

- Convierte comportamiento en un objeto
- Requiere de una interfaz que determine un contrato a cumplir
- Separa incumbencias - El que genera un mensaje del que lo resuelve
- Permite configurar para uso específico herramientas de propósito general

Command



Command

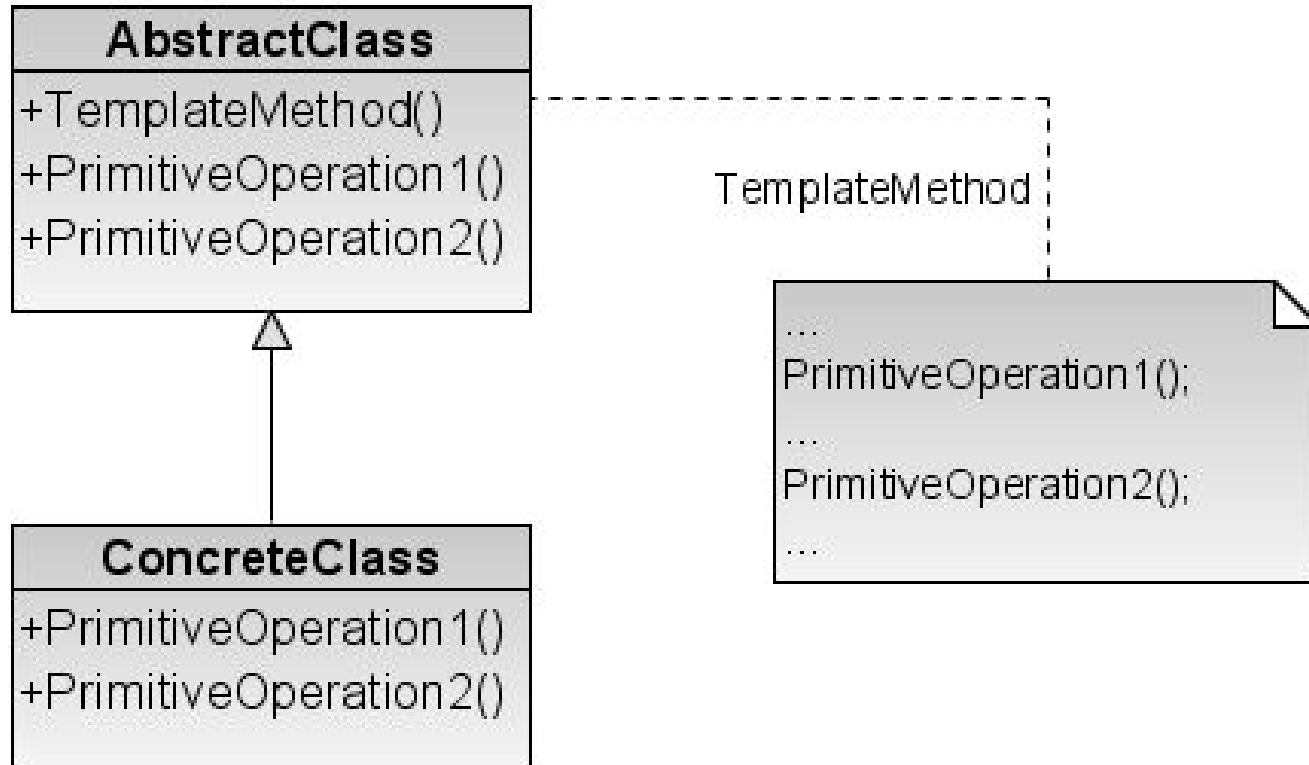


Template method

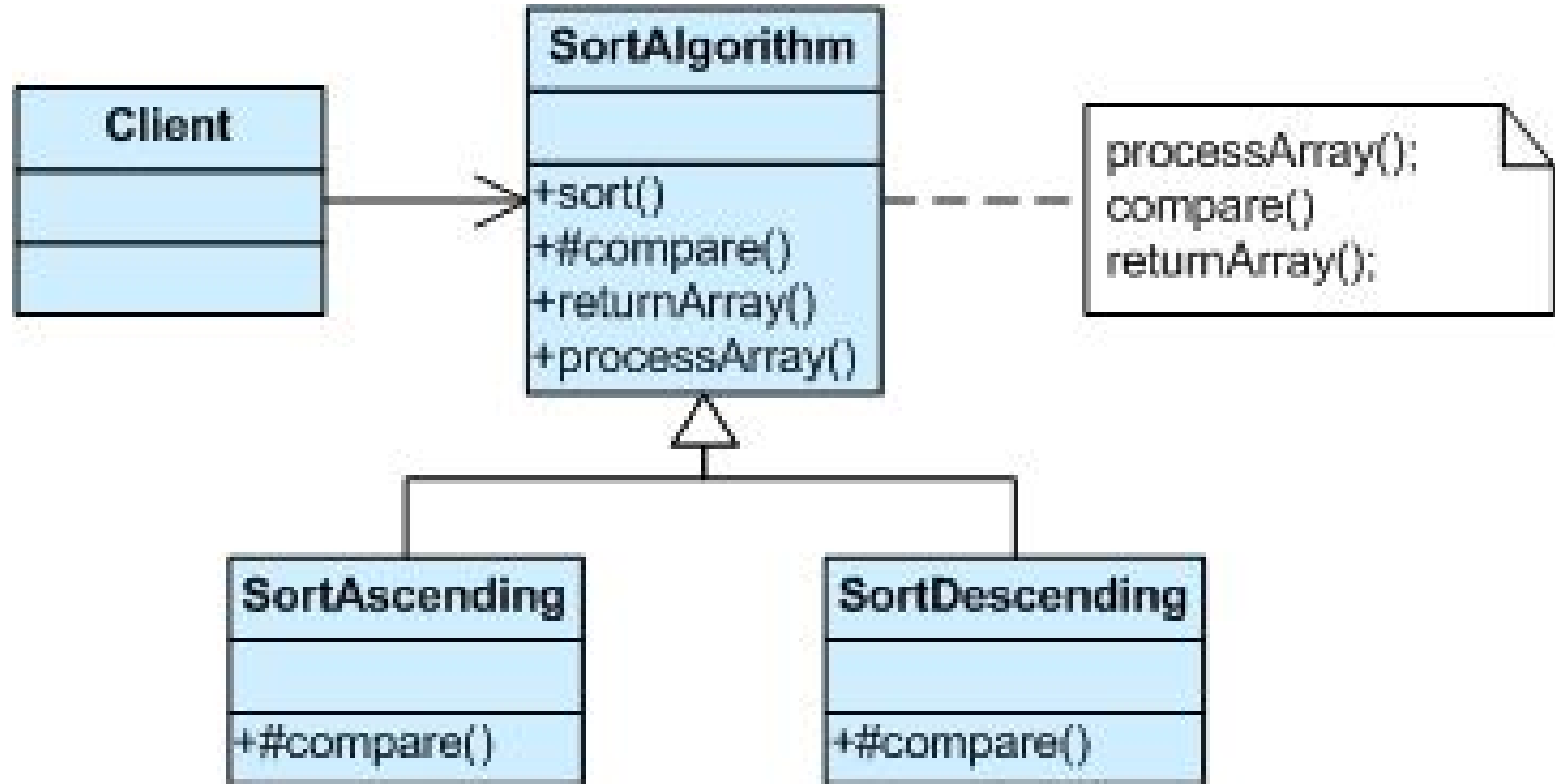
Define el esqueleto de un algoritmo, postergando o delegando ciertos pasos a las subclases concretas que lo implementen, admitiendo variaciones en el comportamiento general de la operatoria.

- Define un algoritmo en términos abstractos
- Evita la repetición de código
- Suele aparecer cuando revisamos código y hacemos un refactor
- Permite limitar y controlar el nivel de extensión en hooks

Template method



Template method

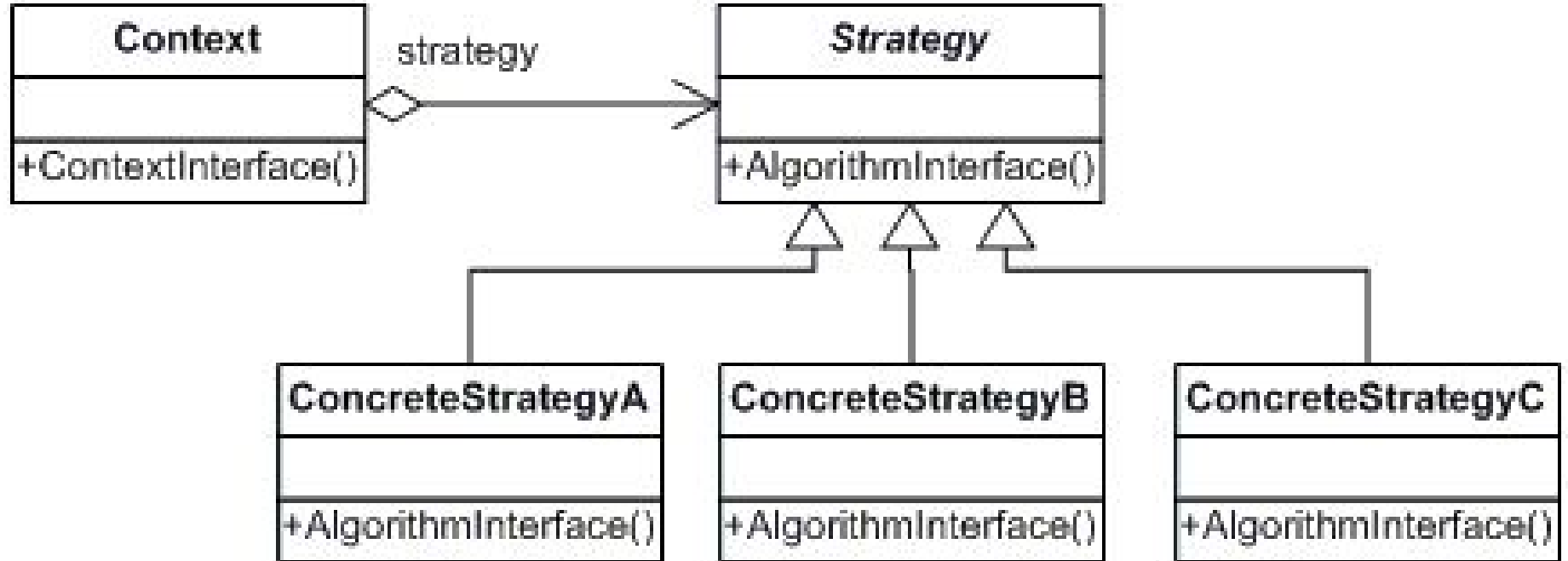


Strategy

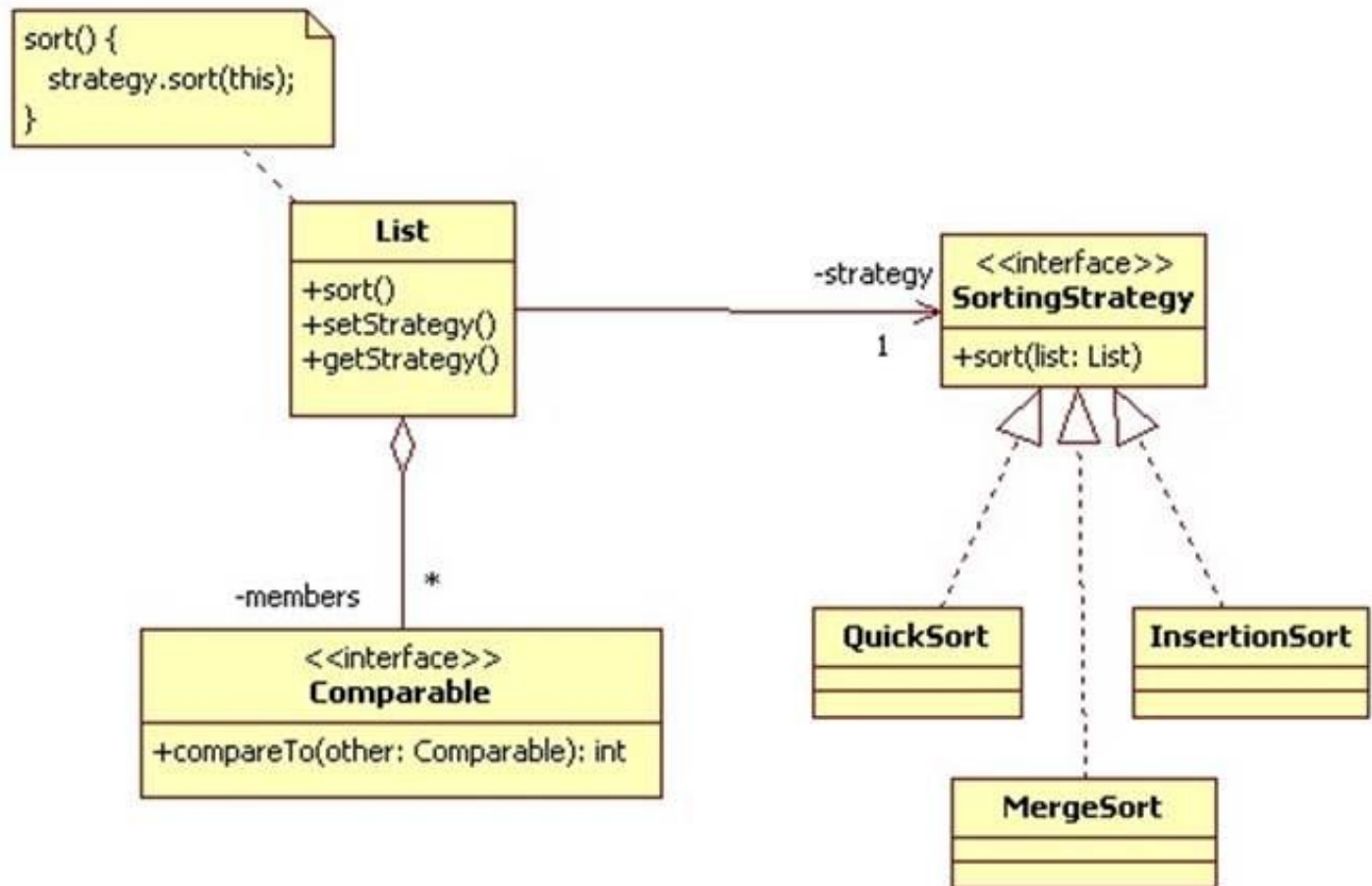
Define una familia de algoritmos, encapsulados, intercambiables, que pueden variar independientemente del código cliente que los use.

- Encapsular algoritmos que difieren, pero pueden aplicarse al mismo contexto
- Cada algoritmo será apropiado para un momento diferente
- Facilita el agregado de nuevos algoritmos y la modificación de los existentes

Strategy



Strategy

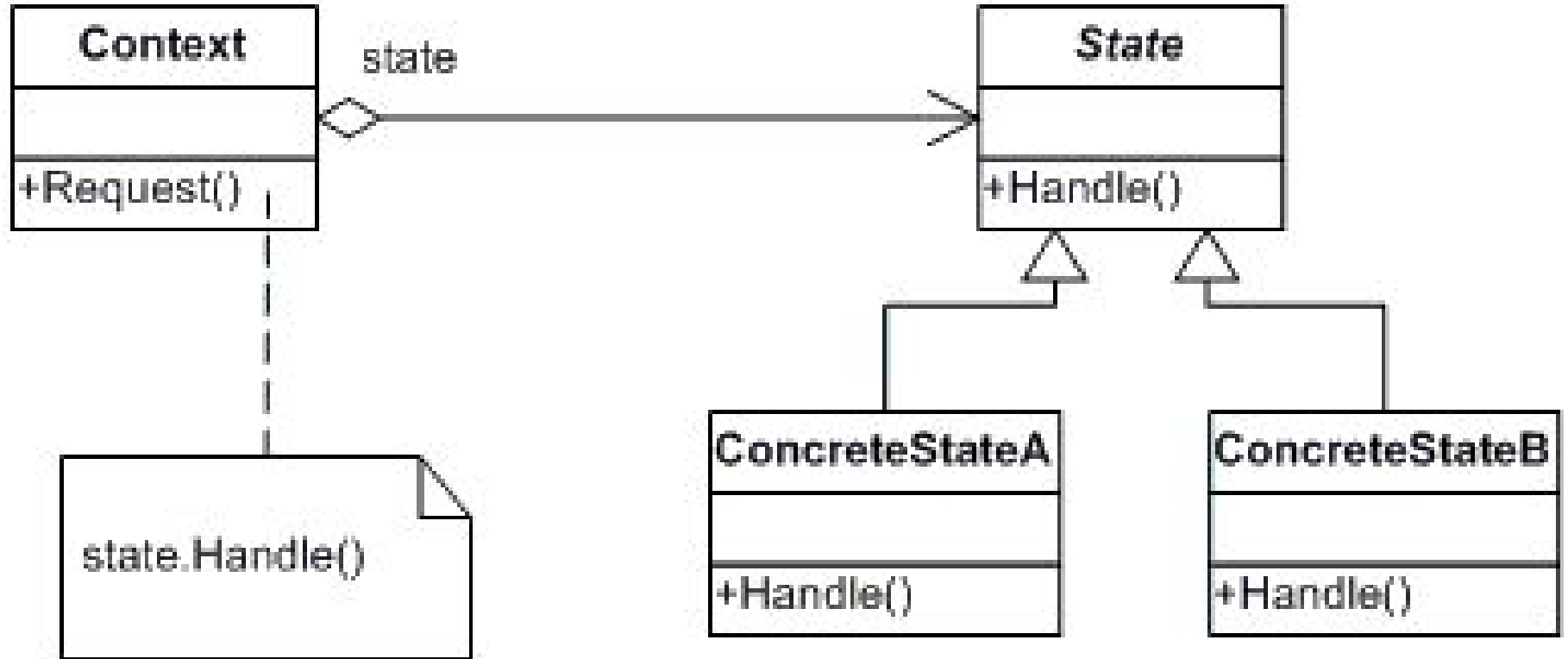


State

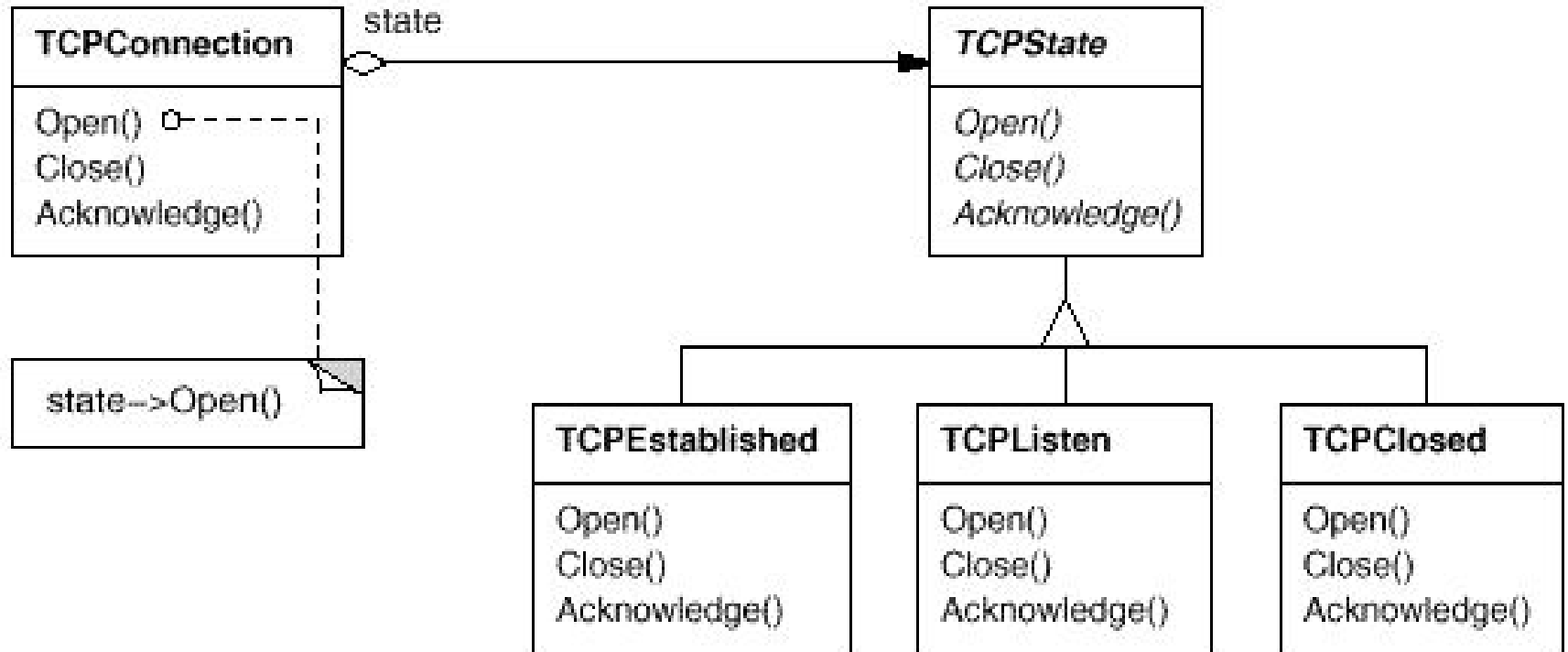
Permite a un objeto modificar su comportamiento ante un cambio en su estado interno al punto que se comporta como un objeto de otra clase.

- Lleva cuenta de su estado interno
- Delega a una jerarquía de estados, bajo una interfaz, las operaciones relacionadas al estado
- Cambia su comportamiento en runtime
- Explicita los cambios de estado
- Requiere que se defina cómo se cambiará de estado, o quién será responsable de hacerlo
- Puede devenir en la creación repetida de objetos estado

State



State



¿ ?



Bibliografía

- Design Patterns CD - Gamma, Helm, Johnson, Vlissides

