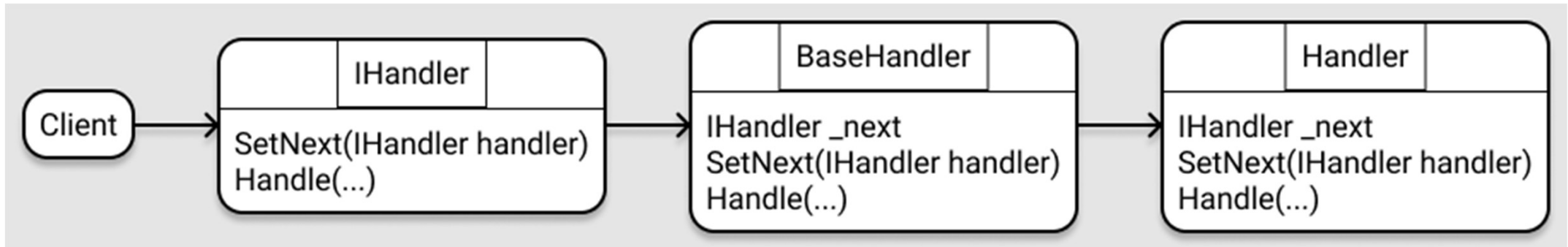


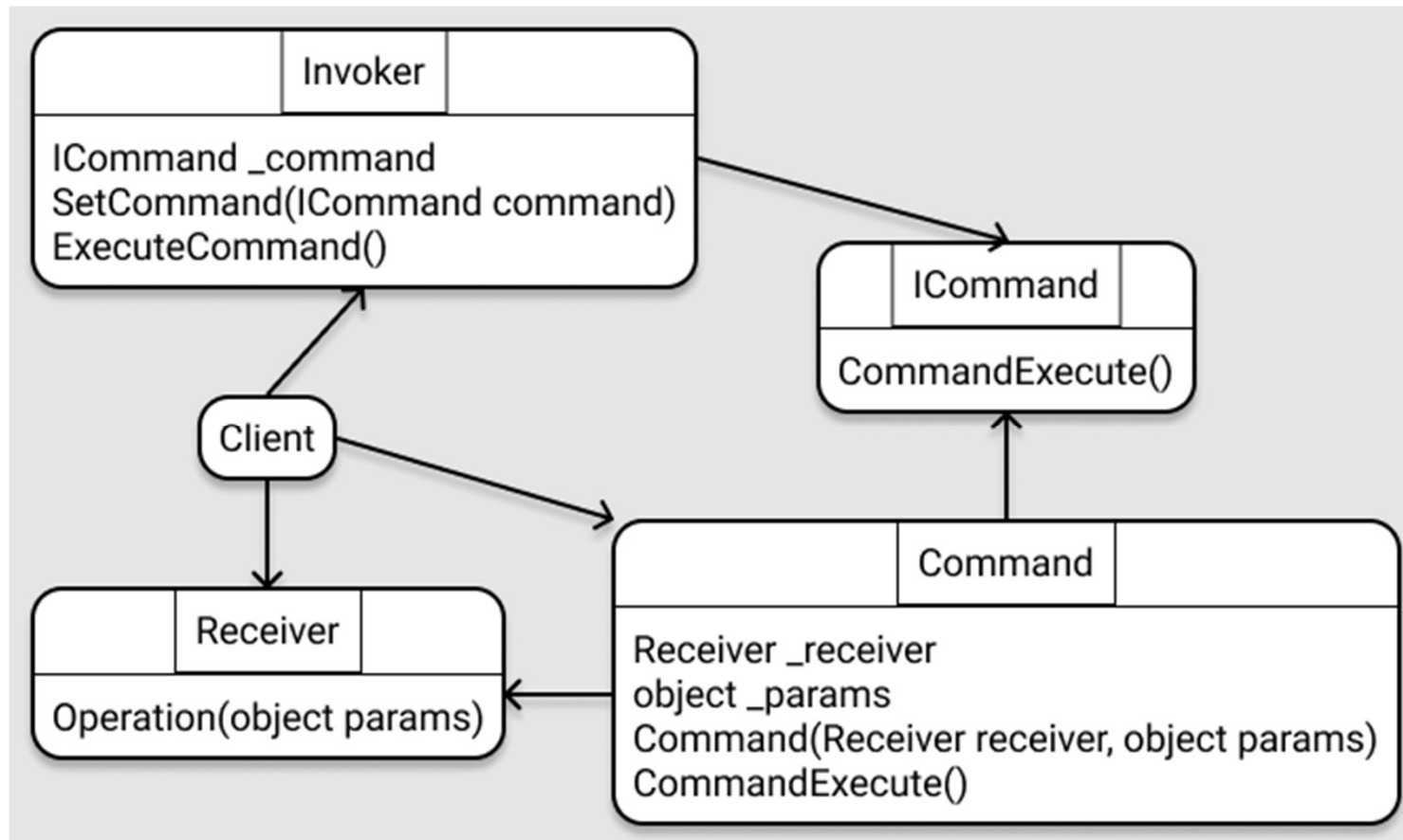
Łańcuch odpowiedzialności



Łańcuch odpowiedzialności

- Użyteczny, gdy program ma przetwarzać różne rodzaje żądań na różne sposoby, ale dokładne typy żądań i ich sekwencje są nieznane wcześniej
- Pozwala na wykonanie sekwencji działań w określonej kolejności lub zmieniać ją w czasie wykonywania programu

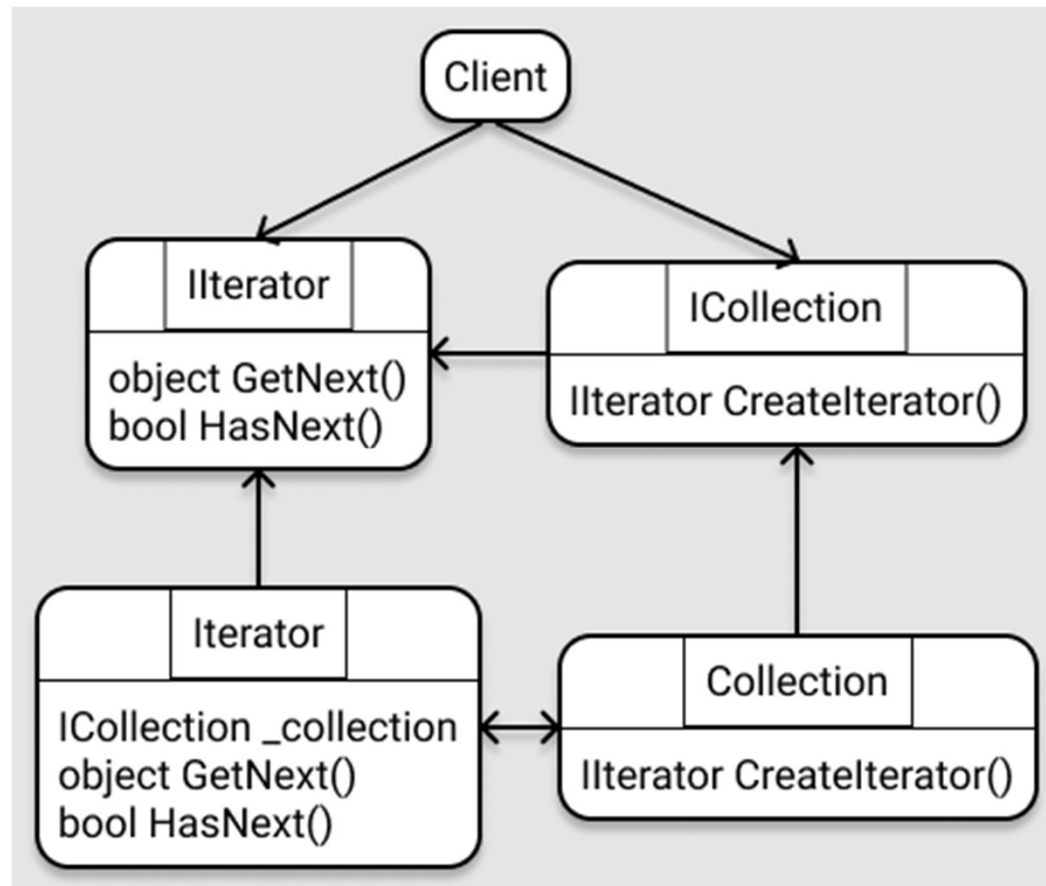
Komenda



Komenda

- Pozwala na parametryzowanie obiektów za pomocą operacji
- Pozwala na kolejkowanie operacji, planowanie ich, a nawet zdalne wykonywanie
- Stosowany do wykonywania operacji, które można cofnąć

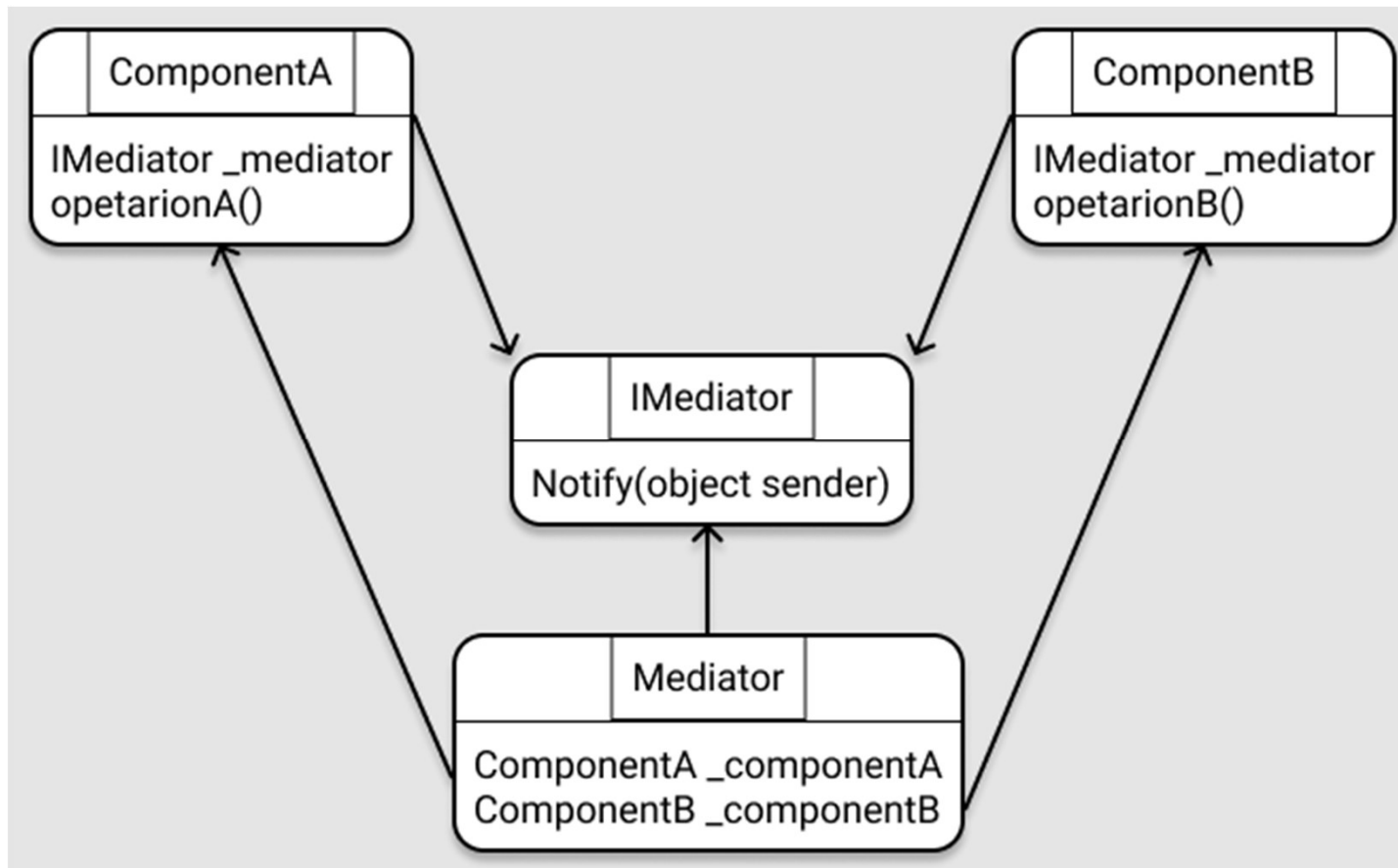
Iterator



Iterator

- Może ukryć złożoność struktury danych przed klientem
- Redukuje powtarzający się kod przechodzenia po elementach kolekcji

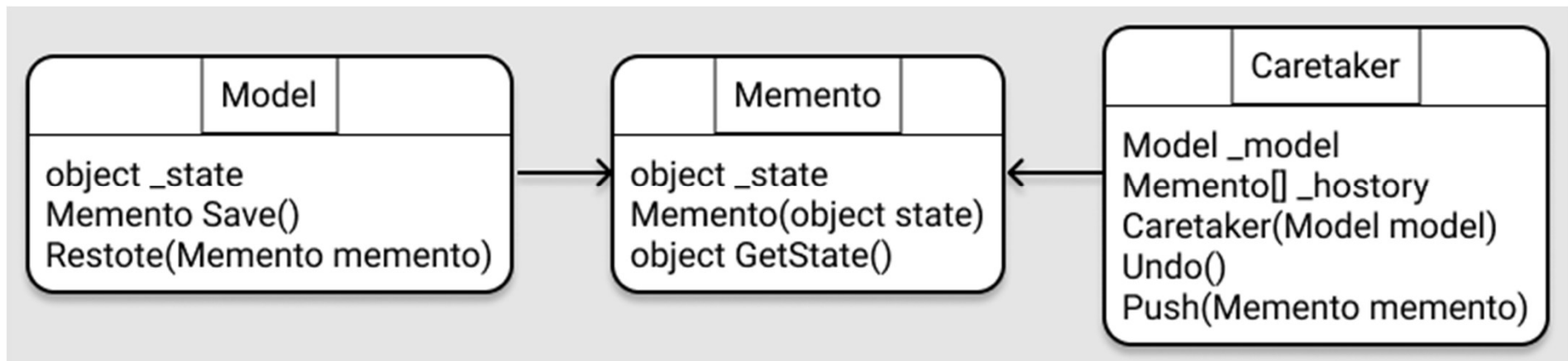
Mediator



Mediator

- Wyodrębnia relacje między klasami do osobnej klasy
- Pozwala na ponowne użycie komponentu w innym programie mimo, że jest on mocno powiązany z innymi składnikami
- Pozwala używać pojedynczą klasę w różnych kontekstach

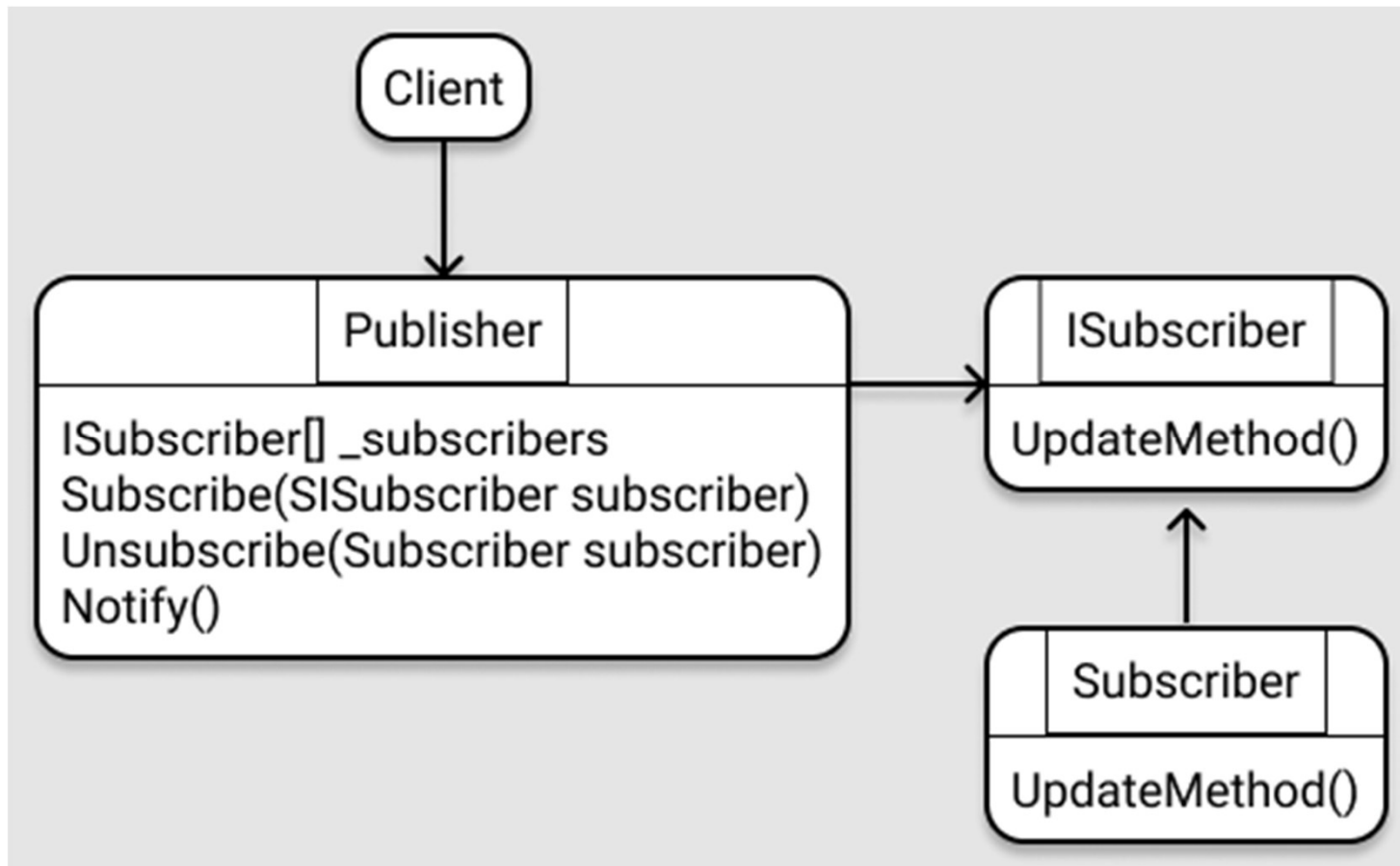
Memento



Memento

- Tworzy migawki i pozwala przywrócić stan historyczny obiektu

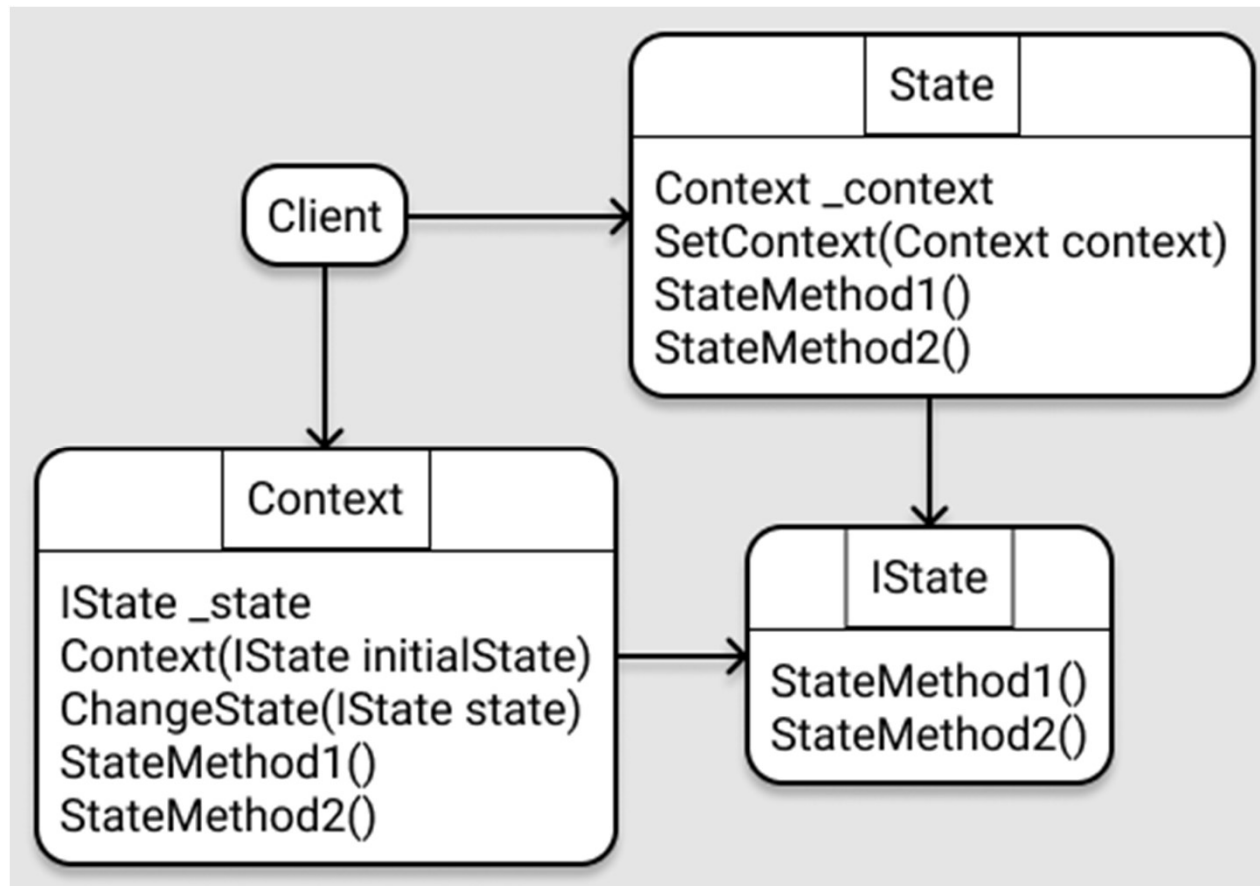
Observer



Obserwator

- Pozwala reagować na zmiany stanu obiektu przez zestaw obiektów:
 - stały
 - nieokreślony
 - dynamicznie zmieniający się

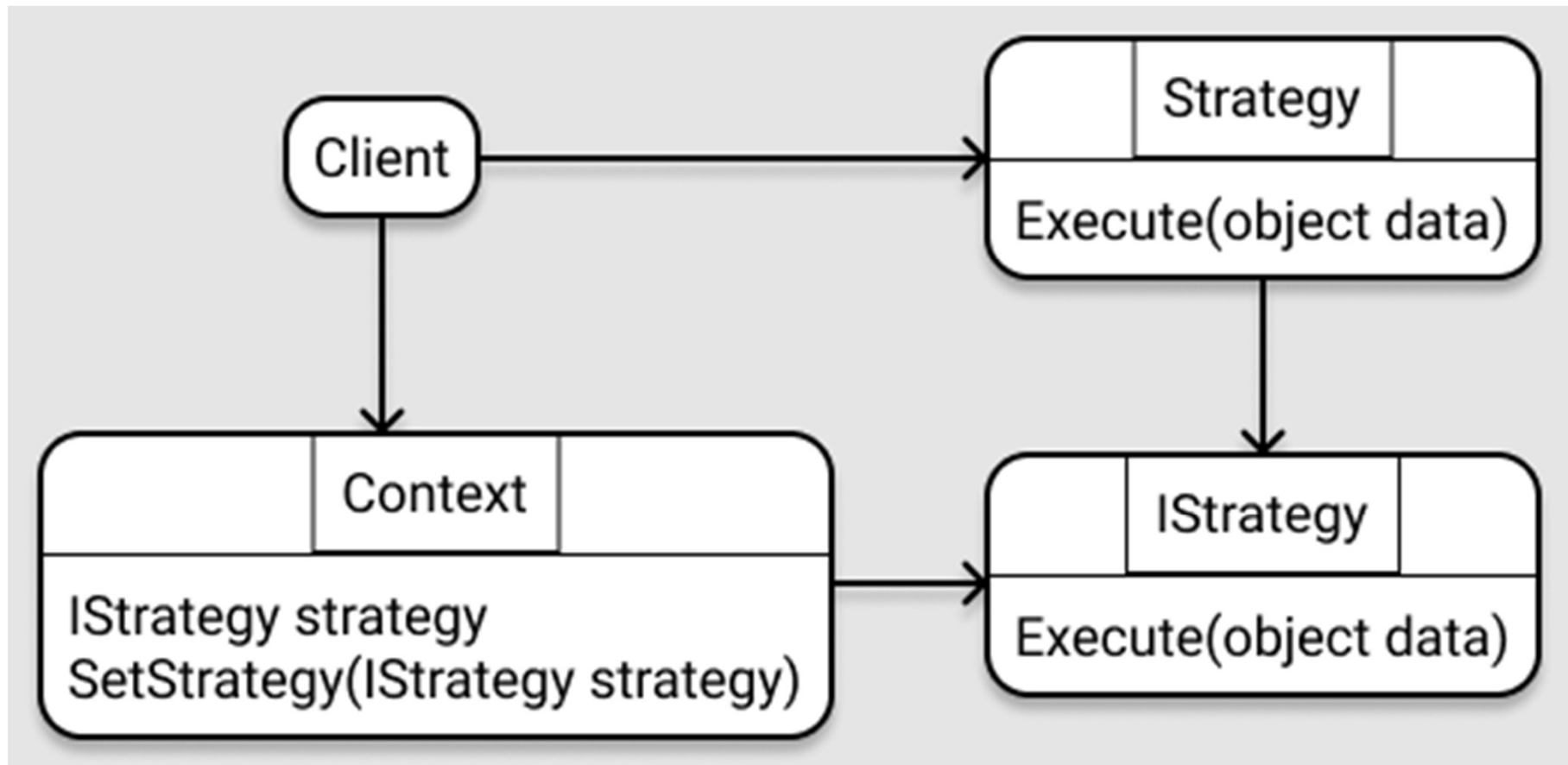
Stan



Stan

- Przydatny, gdy istnieje potrzeba zmiany zachowania obiektu w zależności od jego stanu
- Oczyszcza klasę z dużą ilością warunków, które zmieniają sposób w jaki klasa zachowuje się z bieżącymi wartościami pól

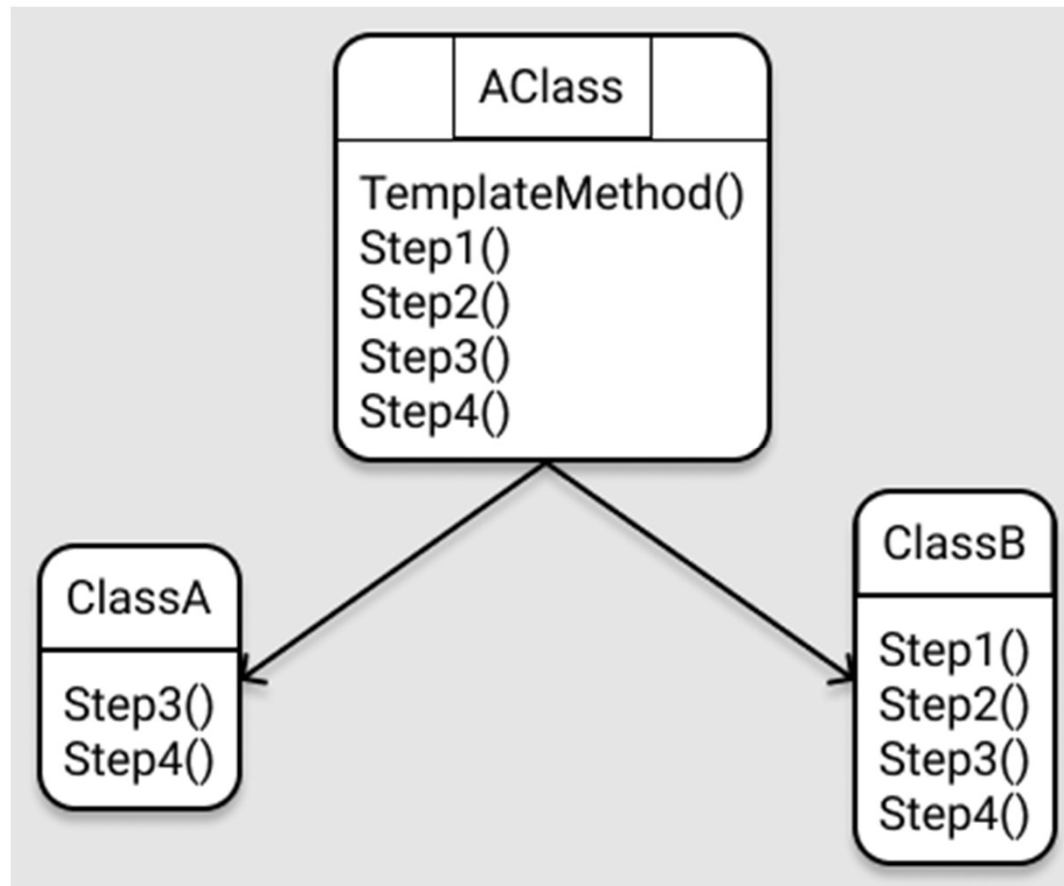
Strategia



Strategia

- Wzorzec strategii umożliwia pośrednio zmieniać zachowanie obiektu w czasie wykonywania
- Pozwala wyeliminować wiele podobnych klas, które różnią się tylko sposobem wykonywania akcji
- Izoluje logikę biznesową od szczegółów implementacji algorytmu
- Zastępuje operator warunkowy przełączający warianty tego samego algorytmu

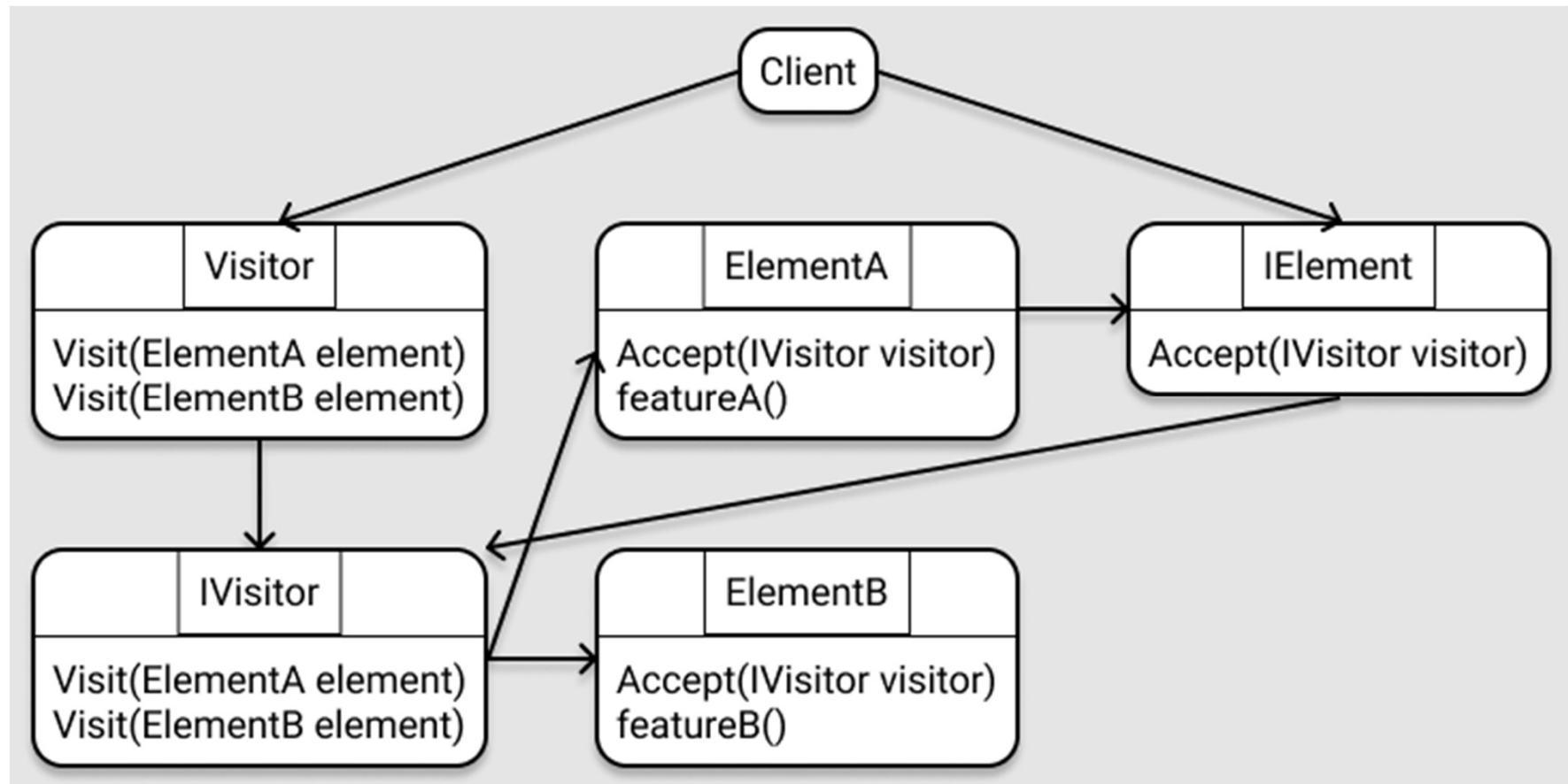
Szablon



Szablon

- Pozwala na rozszerzanie określonych kroków algorytmu
- Zmniejsza powielanie kodu, gdy wiele klas ma zaimplementowany podobny algorytm

Wizytator



Wizytator

- Umożliwia wykonywanie operacji na zestawie obiektów o różnych klasach
- Oczyszcza logikę biznesową z funkcji pomocniczych
- Użyteczny, gdy pewne zachowania mają sens tylko w niektórych klasach hierarchii
- Rozwiązuje problem „double dispatch”