

Wzorce behawioralne

- opisujące zachowanie i odpowiedzialność współpracujących ze sobą obiektów
- umożliwiają organizację, zarządzanie i łączenie zachowań

Wzorce behawioralne

- Chain of responsibility
- Command
- Interpreter
- Iterator
- Mediator
- Memento
- Observer
- State
- Strategy
- Template
- Visitor

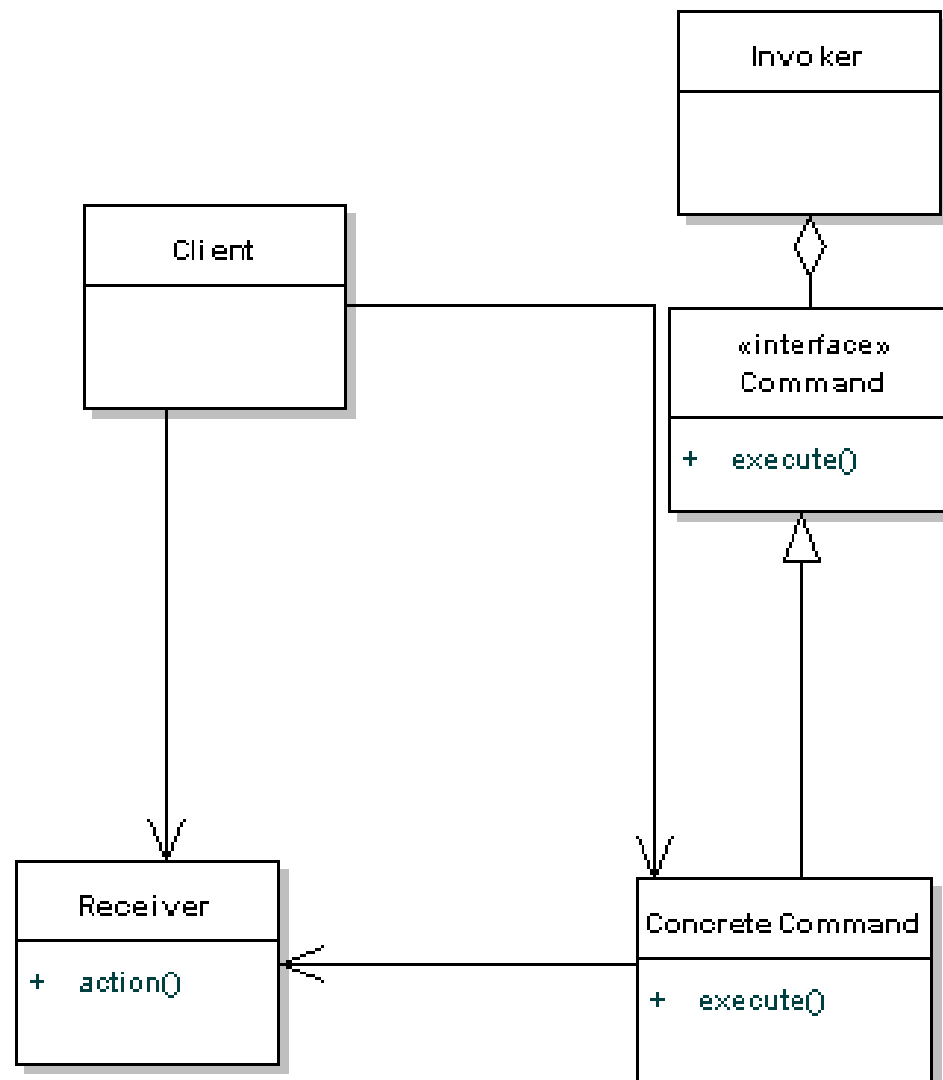
Command



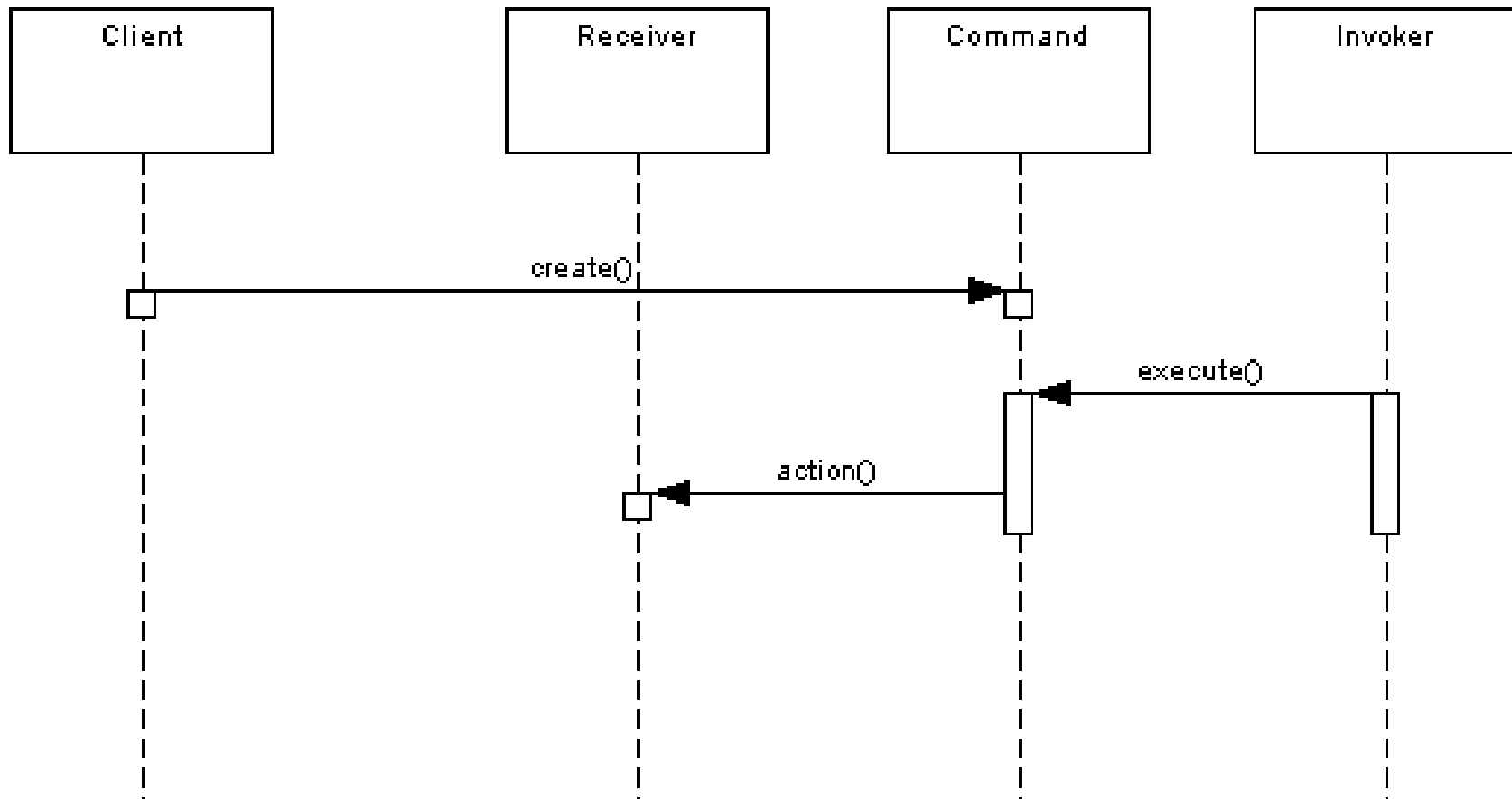
Wzorzec – Command (Polecenie)

- Ukrycie logiki obsługi poleceń w obiektach poleceń.
- Interfejs takich obiektów jest bardzo prosty, występuje najczęściej tylko jedna metoda, typowo `void execute()` lub `void run()`.
- Wykorzystywane w celu eliminacji długich instrukcji warunkowych w tzw. dispatcherach. Przykładem wzorca Command jest interfejs `Runnable`. Idealny do implementacji operacji cofnij-ponów.

Command



Command



Przykłady

- `java.lang.Runnable` w JDK
- `javax.swing.Action` w JDK
- `pl.anicos.patterns.exercises.command`

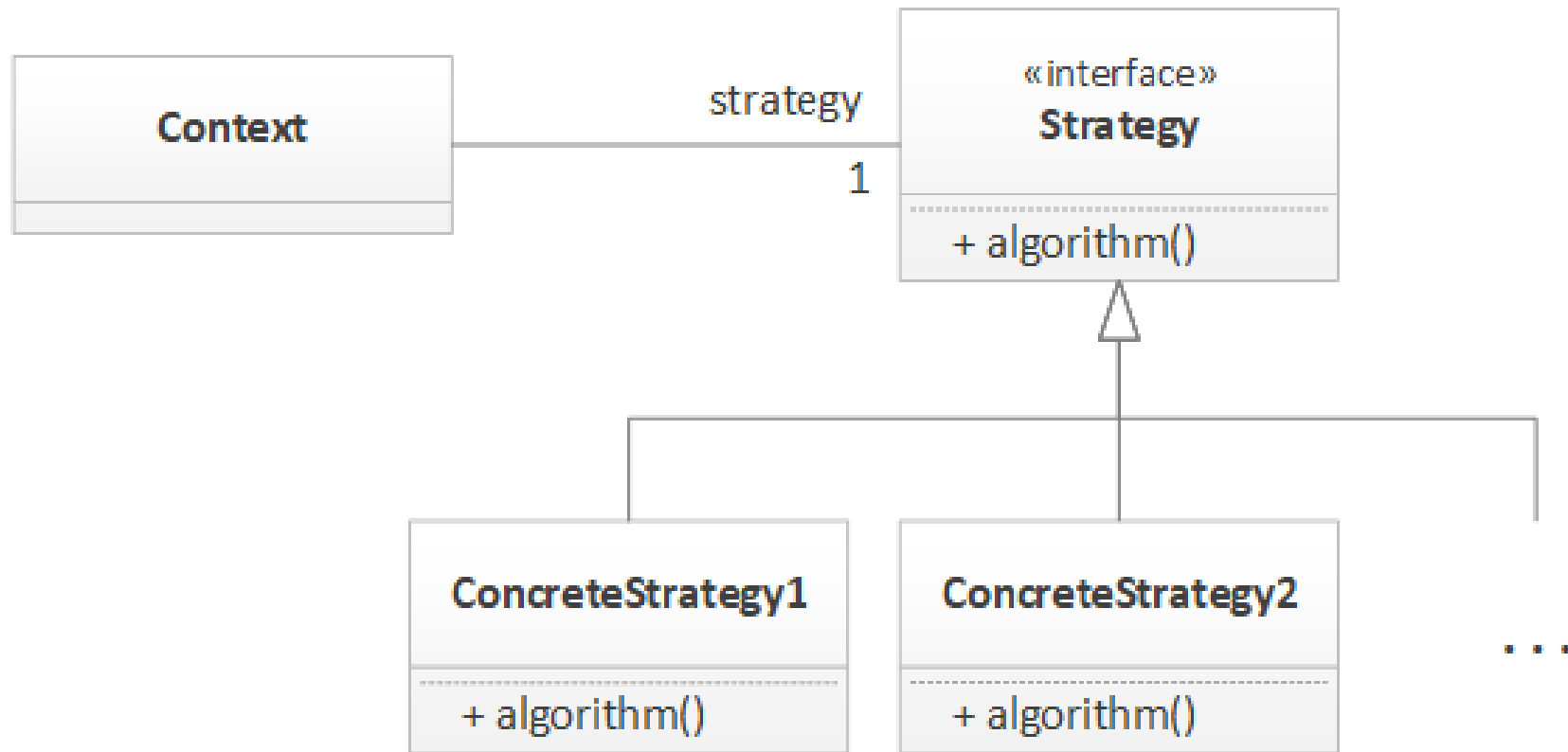
Strategy

- Polega na hermetyzowaniu operacji umożliwiając stworzenie zamiennych implementacji
- Zamiast używać instrukcji **if** czy **switch** użyj **strategii**
- Zawsze przed zastosowaniem dziedziczenia warto pomyśleć, czy wzorzec strategii nie będzie lepszym rozwiązaniem.

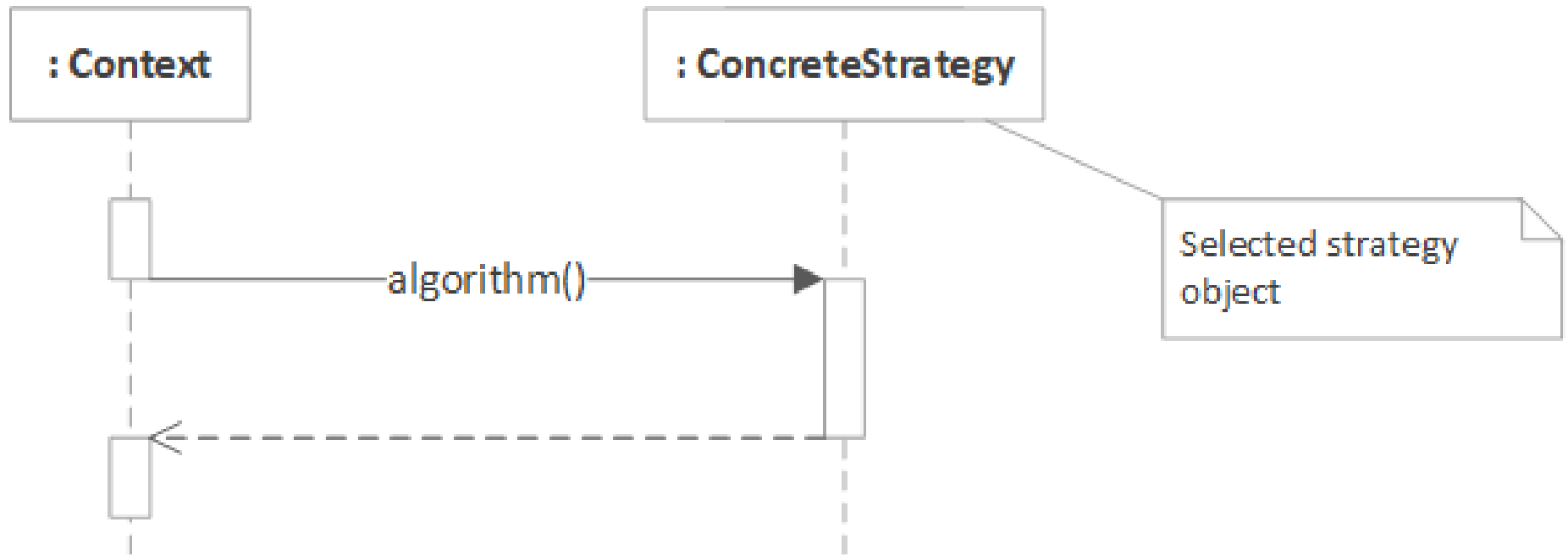
Przykład z życia wzięty

- Tworzymy sklep internetowy oferujący swoje usługi w kilku państwach. Jak wiadomo prawo podatkowe znacząco różni się w poszczególnych krajach.
- Powstaje problem naliczenia odpowiedniego podatku dla klientów pochodzących z odmiennych państw.
- Jak to rozwiązać?
- Wybrać odpowiednią stawkę za pomocą licznych instrukcji warunkowych? Nie, do tego świetnie nadaje się **wzorzec strategii**.

Strategy – diagram klas



Strategy – diagram aktywności



Użycie Strategii

```
public class StrategyTest {  
    @Test  
    public void runStrategy() {  
        Cart cart = new Cart();  
        cart.addLineItem(new Item("Milk", 3.0));  
        cart.addLineItem(new Item("Water", 2.0));  
        double polandPrice = cart.calculateFinalPrice(TaxPolicyFactory.getTaxPolicy("Poland"));  
        System.out.println("Final price in poland id " + polandPrice);  
        double germanPrice = cart.calculateFinalPrice(TaxPolicyFactory.getTaxPolicy("Germany"));  
        System.out.println("Final price in Germany id " + germanPrice);  
        double andoreaPrice = cart.calculateFinalPrice(TaxPolicyFactory.getTaxPolicy("Andora"));  
        System.out.println("Final price in Andora id " + andoreaPrice);  
    }  
}
```

Przykłady Strategy

- `java.util.Comparator#compare()` podczas wywołania `Collections#sort()`
- `pl.anicos.patterns.exercises.strategy.StrategyTest`