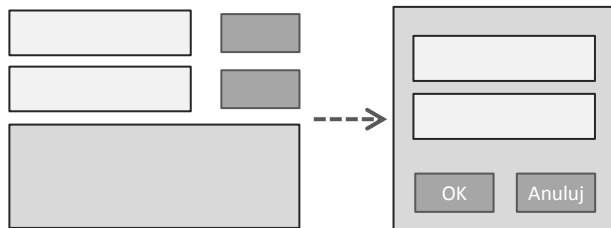
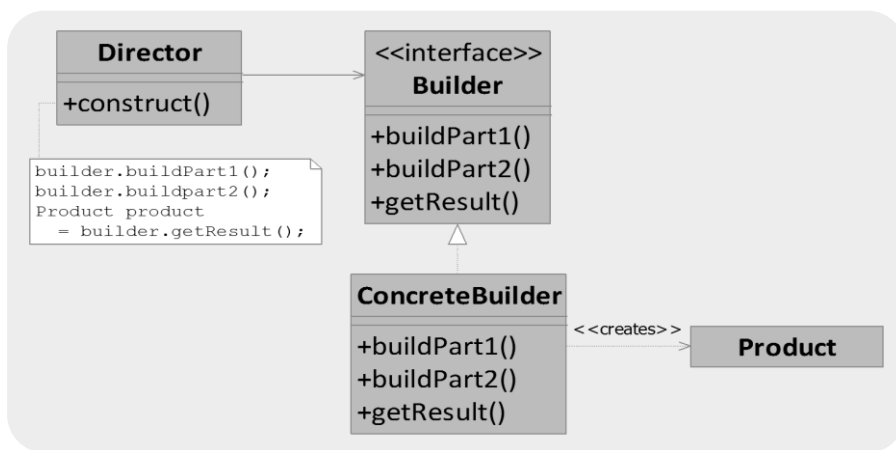


## WZORCE PROJEKTOWE (CHEAT SHEET)

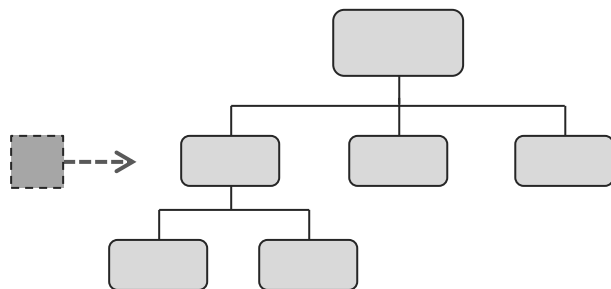
### BUILDER (BUDOWNICZY)



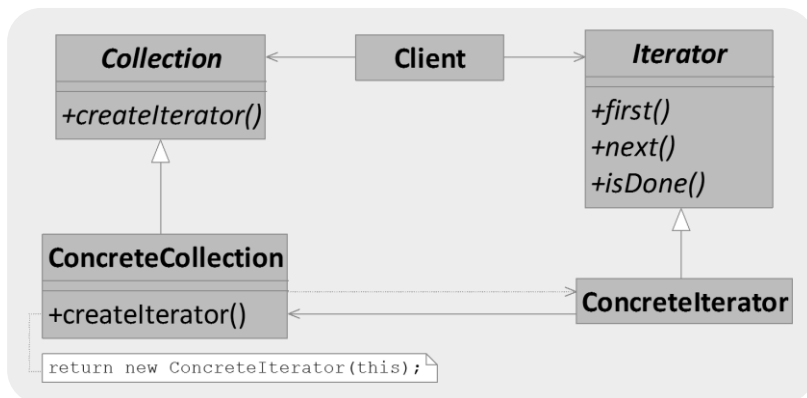
Wieloetapowe  
tworzenie złożonych  
obiektów



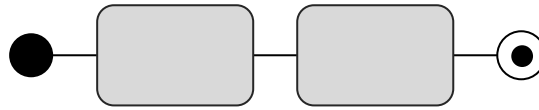
### ITERATOR



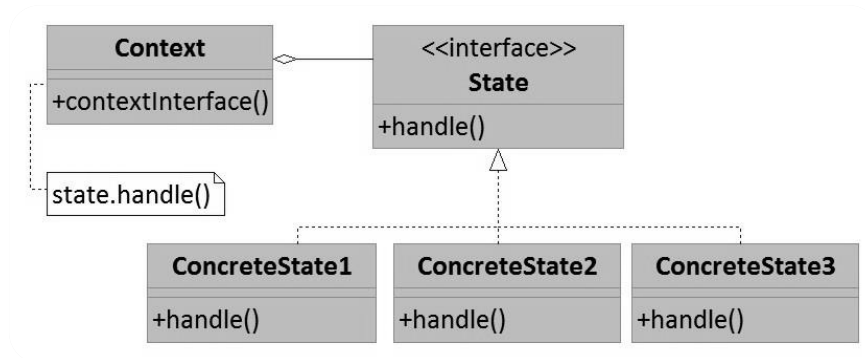
Poruszanie się po  
złożonej strukturze  
w prosty sposób



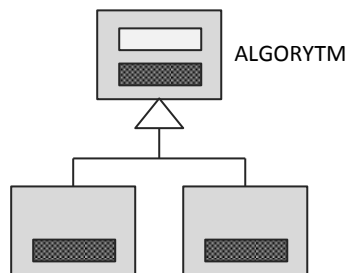
**STATE**  
(STAN)



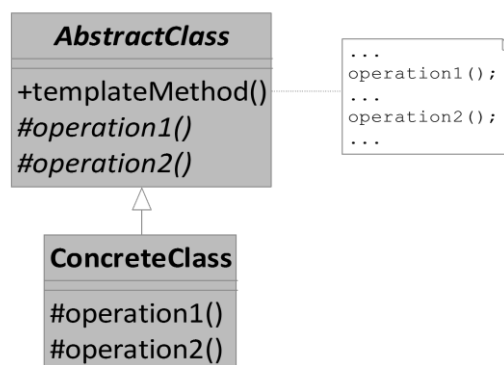
Obiektowa  
implementacja  
prostej maszyny  
stanowej



**TEMPLATE METHOD**  
(METODA SZABLONU)



Niektóre kroki  
algorytmu się różnią  
(mechanizm:  
dziedziczenie)



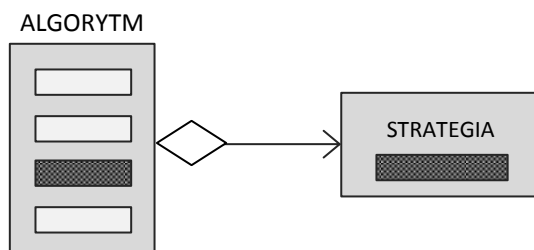
**FLUENT  
INTERFACE**

```

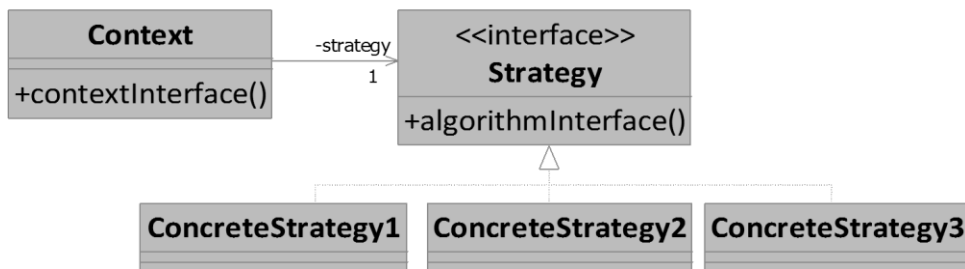
Person.new()
  .age(15) .address
    ("Piotrkowska")
  
```

Uproszczenie  
konstrukcji służących  
do pracy z obiektami  
dziedzinowymi

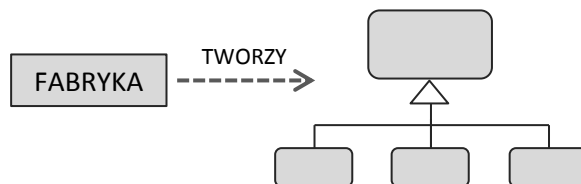
**STRATEGY**  
(STRATEGIA)



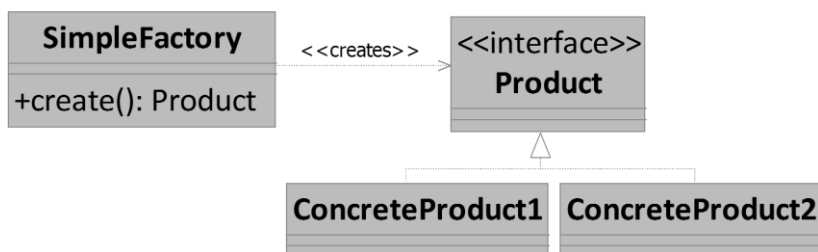
Różne wersje  
algorytmu  
(mechanizm:  
kompozycja)



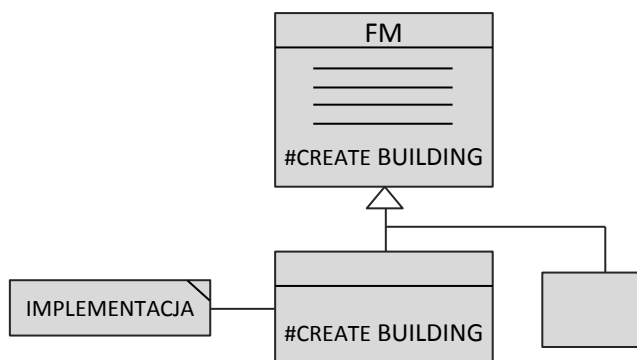
**SIMPLE FACTORY**  
(FABRYKA PROSTA)



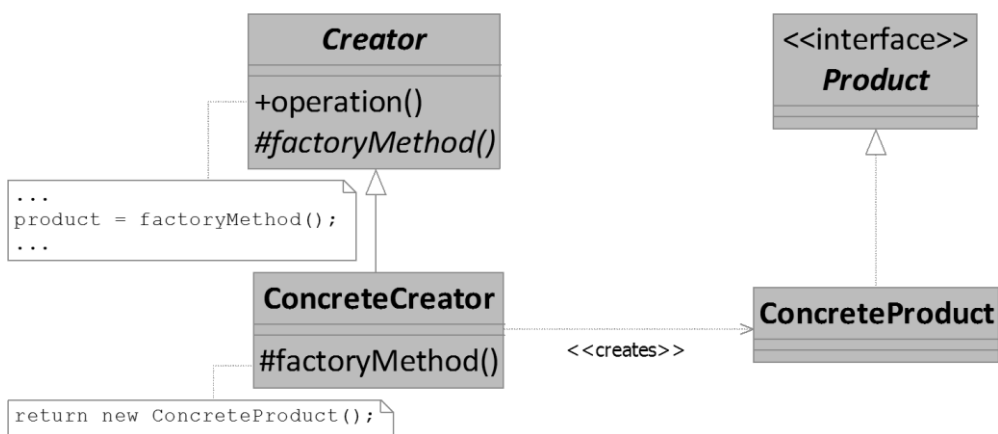
Stworzenie  
konkretnej  
implementacji na  
podstawie  
parametrów



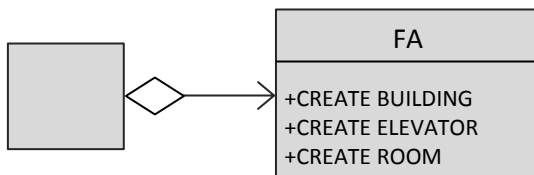
## FACTORY METHOD (METODA FABRYKI)



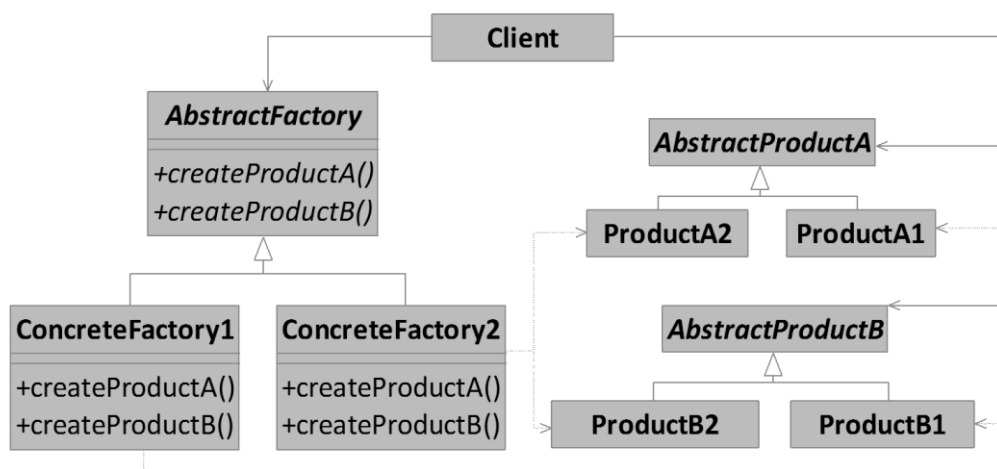
Fabryka w  
kontekście metody  
szablonu



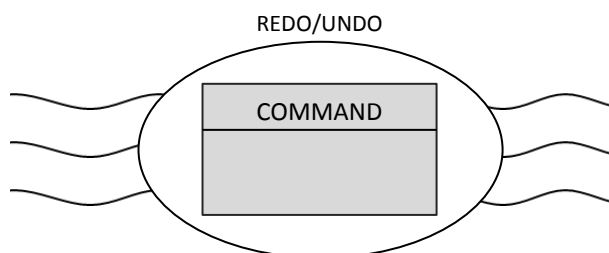
## ABSTRACT FACTORY (FABRYKA AKBSTRAKCYJNA)



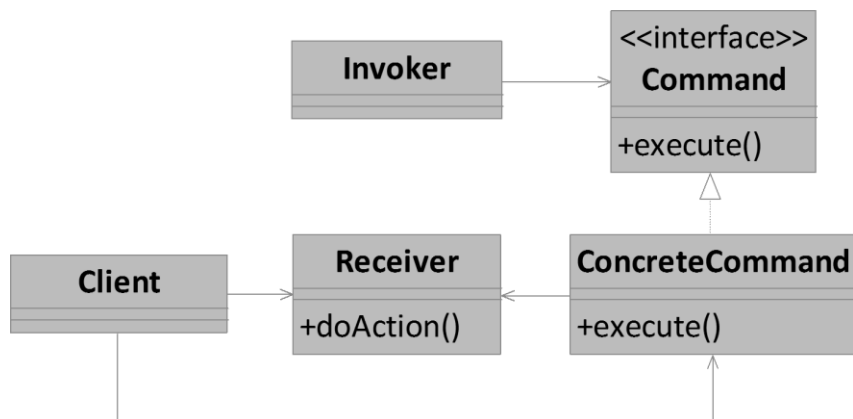
Tworzenie rodziny  
obiektów na  
zasadzie kompozycji



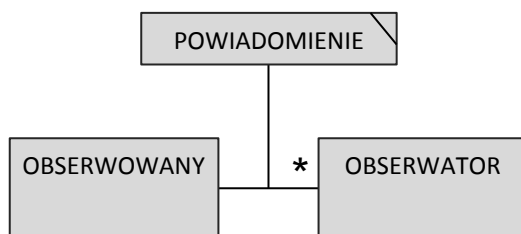
**COMMAND**  
(KOMENDA)



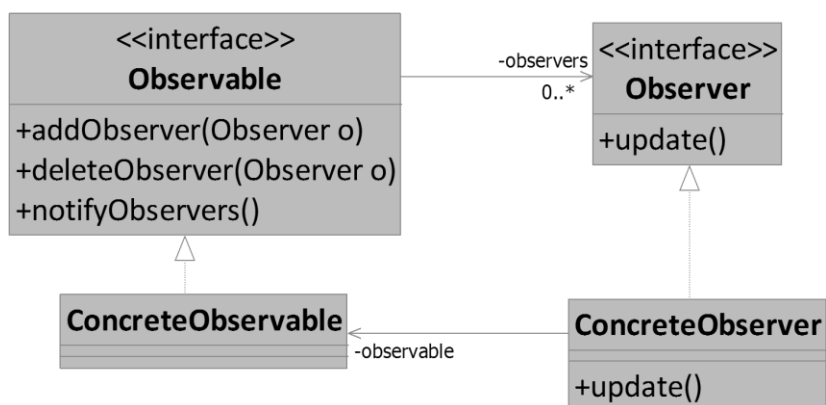
Wspólny interfejs dla niezależnych elementów reprezentujących pewne operacje w systemie



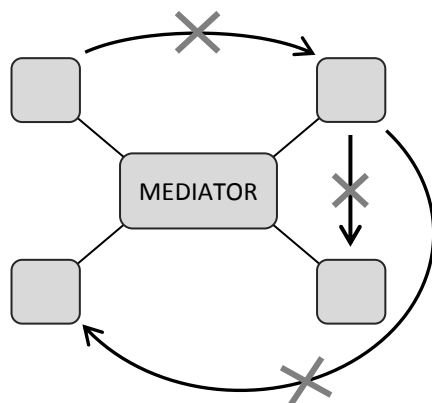
**OBSERVER**  
(OBSERWATOR)



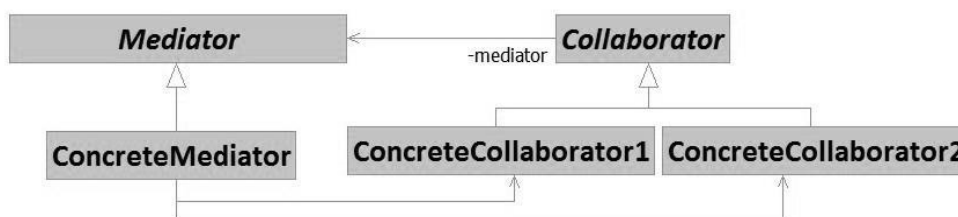
Podstawowy model komunikacji asynchronicznej



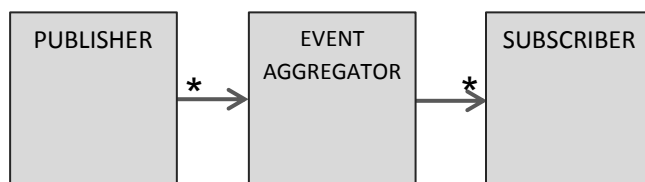
## MEDIATOR



Obiekt pośredniczący  
w komunikacji  
pomiędzy różnymi  
obiektami

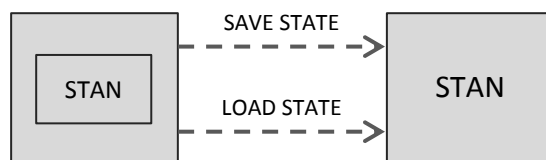


## PUBLISHER - SUBSCRIBER

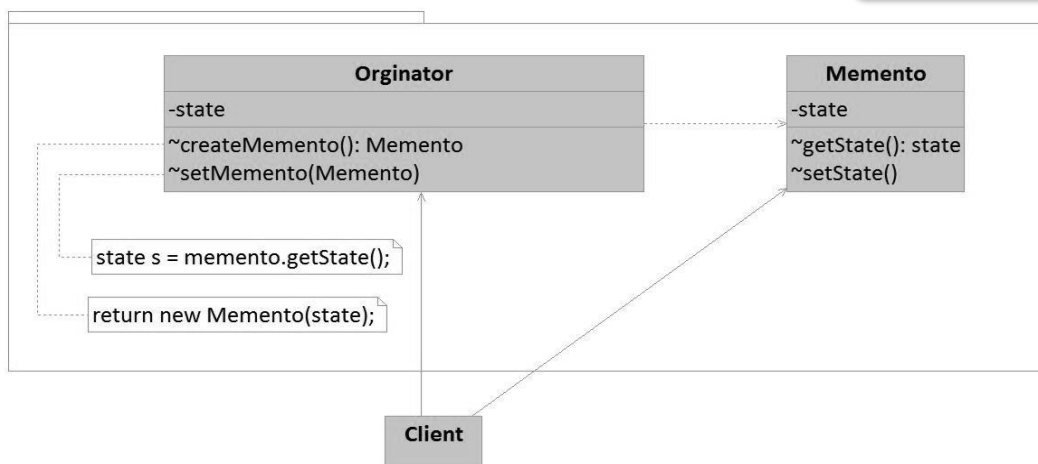


Złożony model  
zdarzeniowy  
(umożliwia  
komunikację \*-  
wiele do wielu)

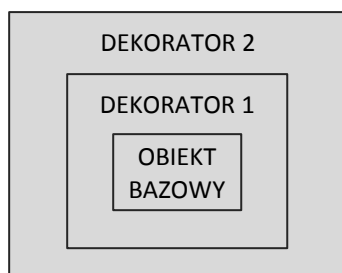
## MEMENTO



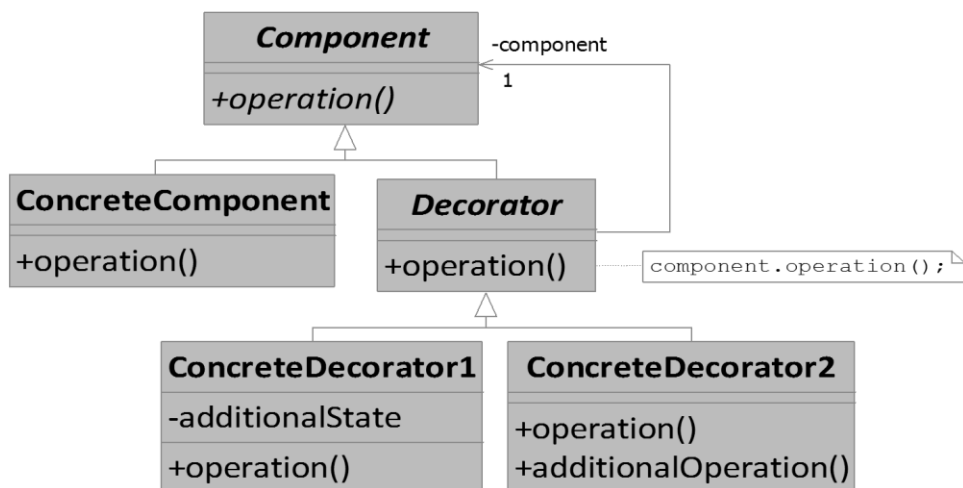
Zapamiętywanie  
i odtwarzanie stanu  
fragmentu systemu



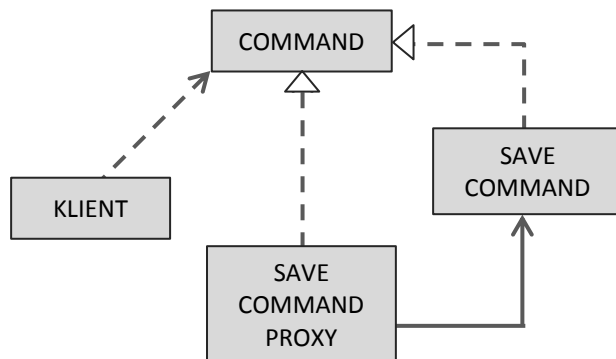
## DEKORATOR



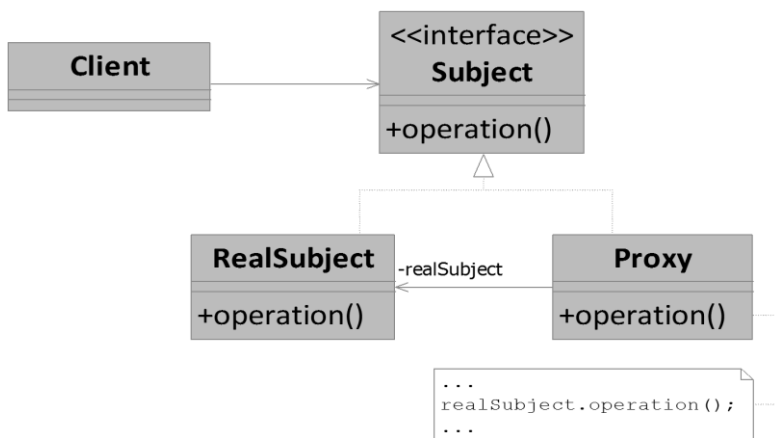
Dynamiczne  
wzbogacanie  
możliwości obiektu  
(rozszerzenie  
odpowiedzialności)



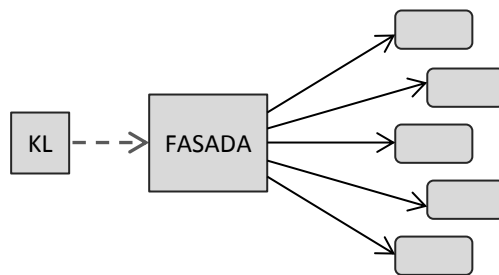
## PROXY



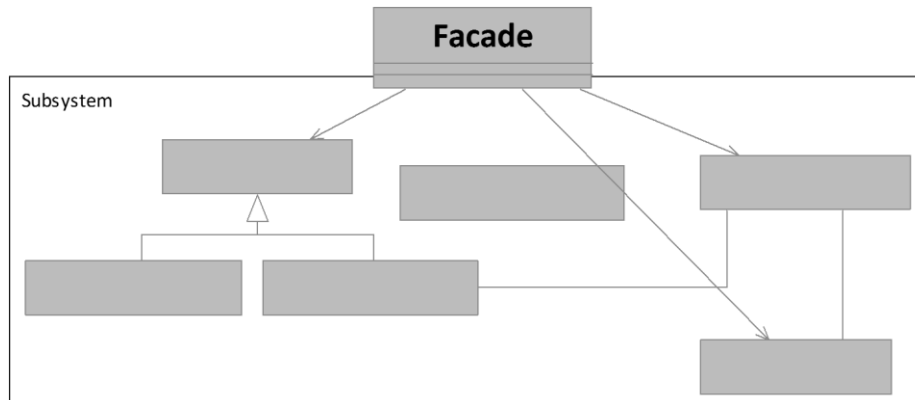
Kontrola dostępu do  
obiektu (silne  
powiązanie  
z obiektem  
docelowym)



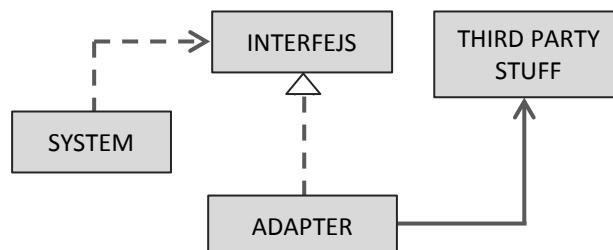
## FACADE (FASADA)



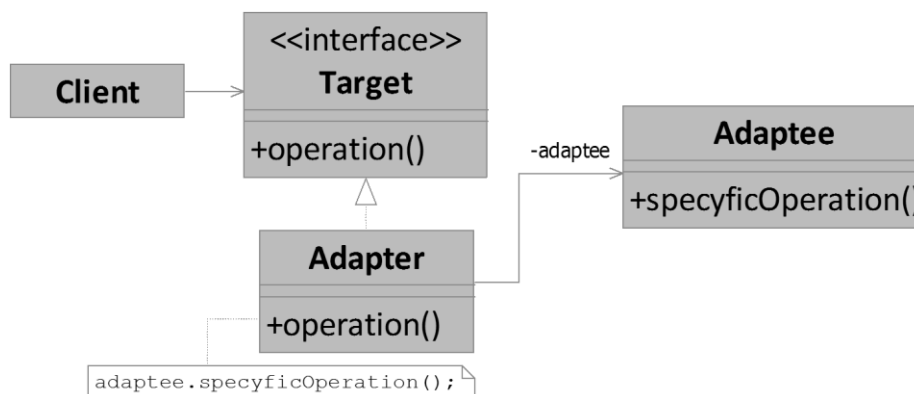
Ukrycie szczegółów  
większego  
podsystemu  
i udostępnienie  
jednego punktu  
dostępowego



## ADAPTER



Prześciółka  
dopasowująca  
zewnętrzne klasy do  
wymaganego  
interfejsu



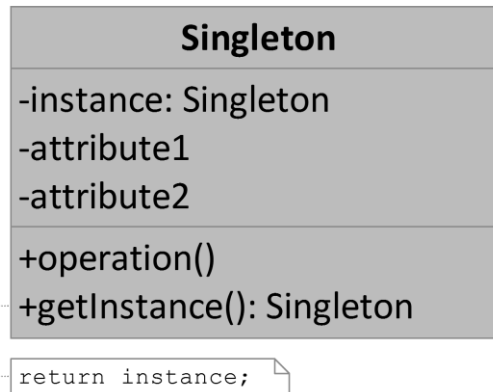


## SINGLETON

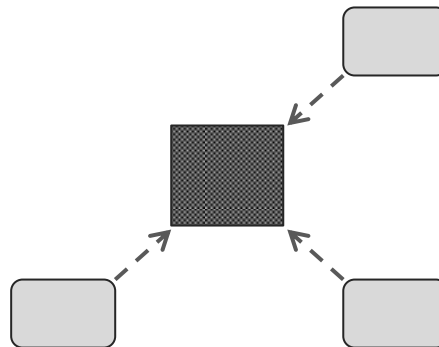
TYLKO JEDEN

1

Umożliwia  
tworzenie tylko  
jednego obiektu  
klasy



## DEPENDENCY INJECTION INVERSION OF CONTROL



Wydobycie zależności  
na zewnątrz  
(getter/setter,  
konstruktor)  
i dostarczanie ich  
z zewnątrz (poprzez  
rozbudowaną fabrykę)