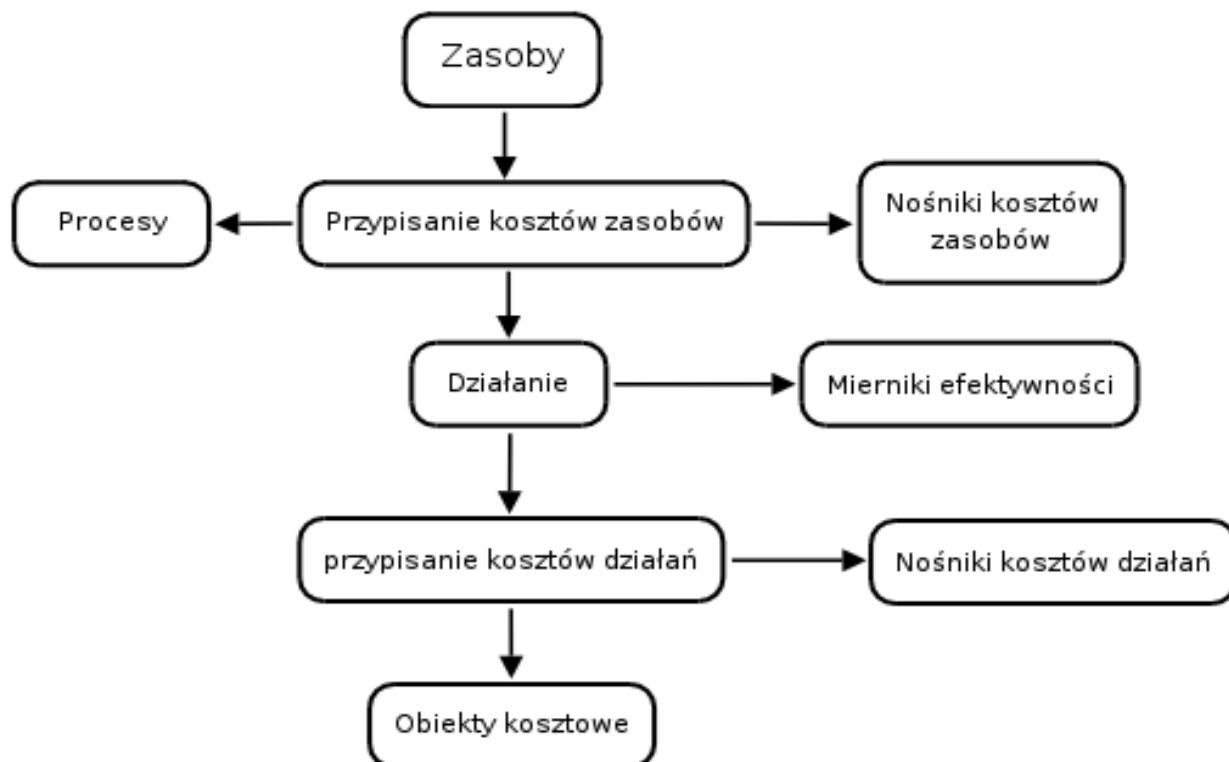


Projektowanie obiektowe

20.12.2005

Punktem wyjścia w obiektowym tworzeniu SI jest model biznesowy.

np.:



Większość SI ma strukturę warstwową, zawiera:

- składniki wewnętrzne (jądro)
- składniki zewnętrzne

Dzięki temu są one elastyczne (można do nich w każdej chwili dołączyć nowe komponenty).

Jednak przyłączenie nowego urządzenia wymaga szeregu zabiegów w różnych warstwach, co jest dość trudne.

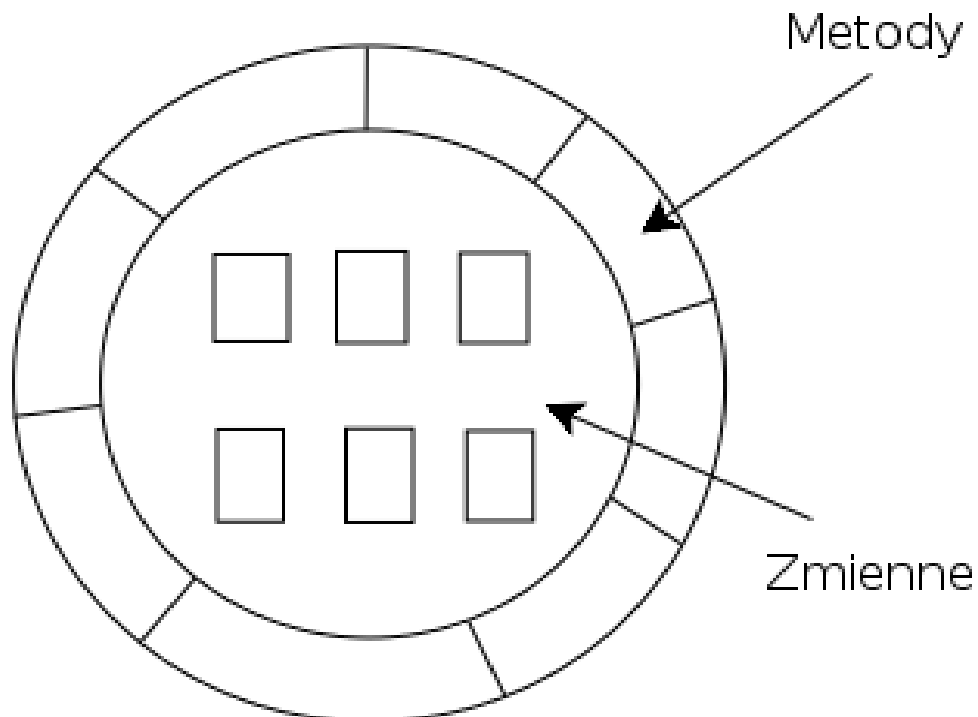
Podstawowe koncepcje projektowania obiektowego:

- koncepcje obiektu:
 - stan wewnętrzny
 - sposób zachowania
 - tożsamość unikatowa
- zasada abstrakcji:
 - kompozycyjnej
 - uogólniającej
- zasada hierarchizacji:
 - kompozycyjnej
 - dziedziczenia

- zasada modularności
- zasada enkapsulacji

Pojęciem kluczowym w projektowaniu jest KLASA.

Cechą charakterystyczną koncepcji klasy jest ukrycie danych przed obiektami:

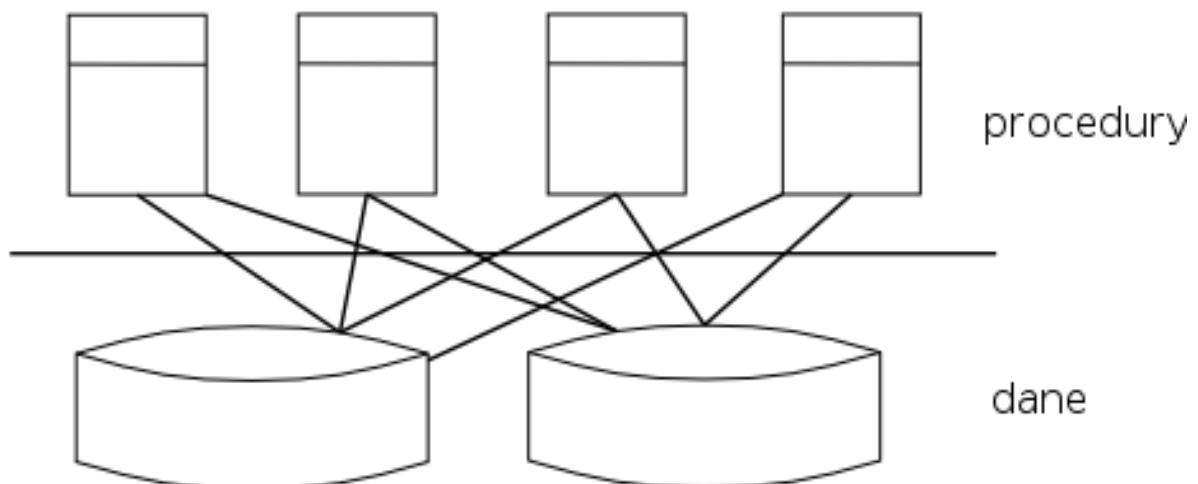


Obiekty zewnętrzne mają dostęp do funkcji (metod) za pomocą których mogą zlecać wykonanie określonych operacji na danych.

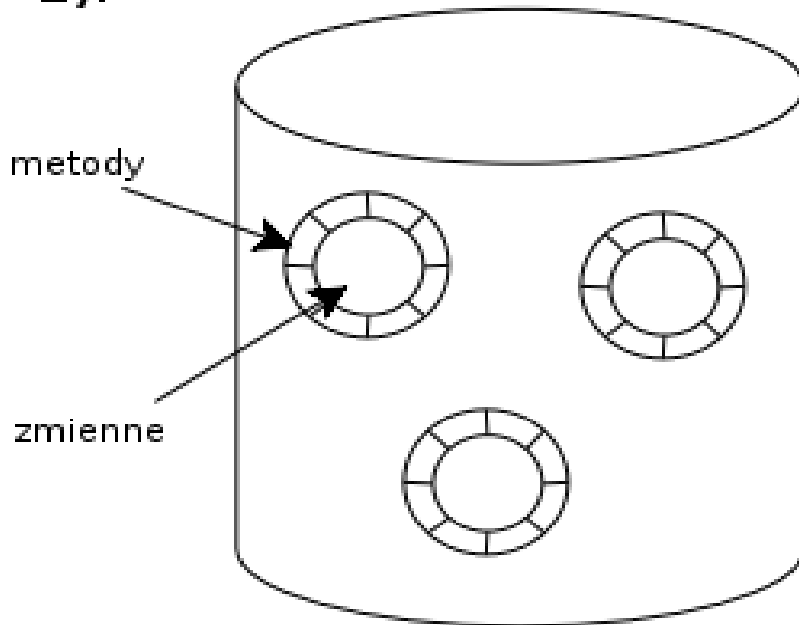
ENKAPSULACJA – zamknięcie danych w „pancerzu” skojarzonych z nimi metod

Dzięki temu można zmienić tradycyjny układ systemu w którym wiele programów korzysta z wielu danych 1) na układ w którym programy i dane są zblokowane razem 2).

1).

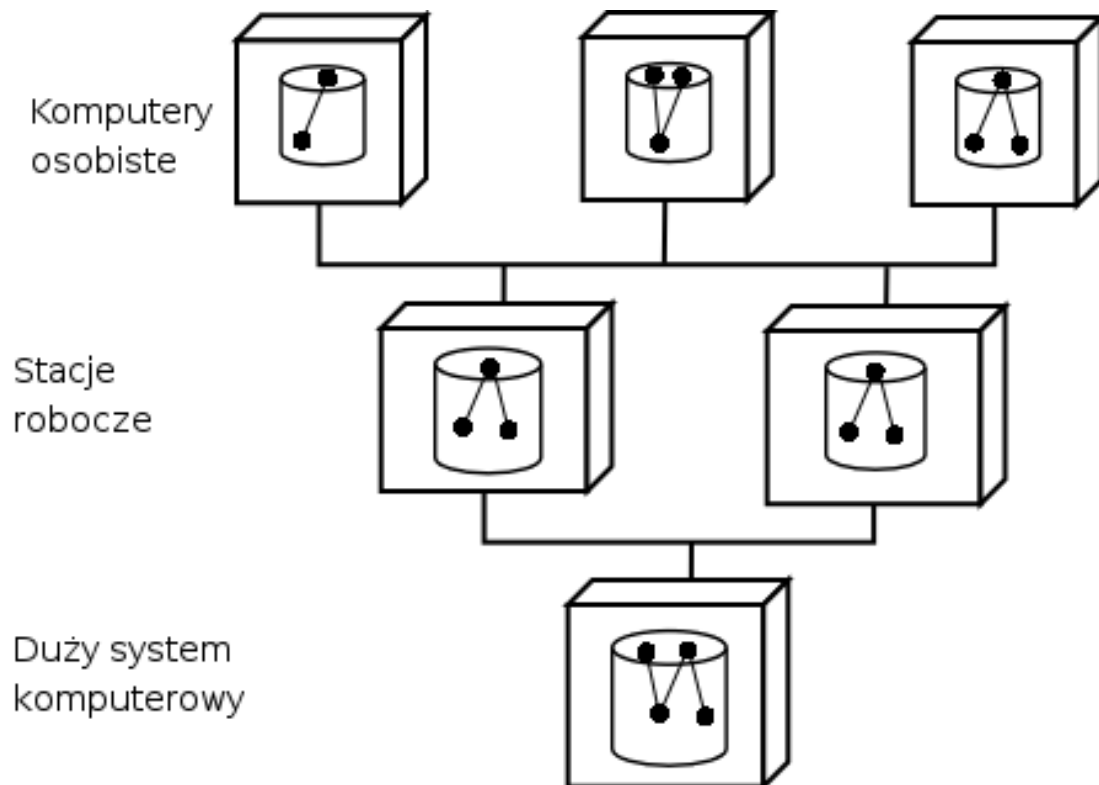


2).

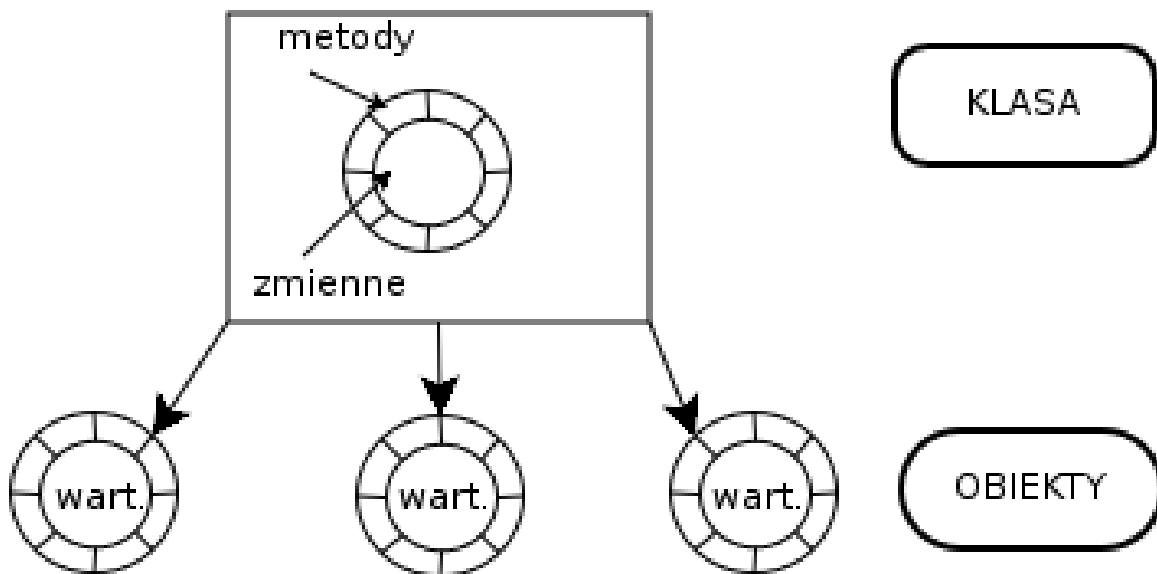


W takim systemie każdy z obiektów może być programowany przez oddzielnego programistę.

Cały system można wtedy budować zestawiając obiekty i klasy w stosownych konfiguracjach na poszczególnych poziomach systemu.

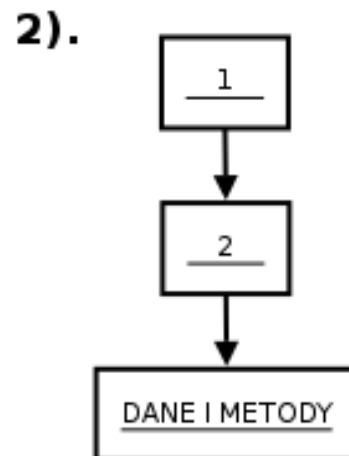
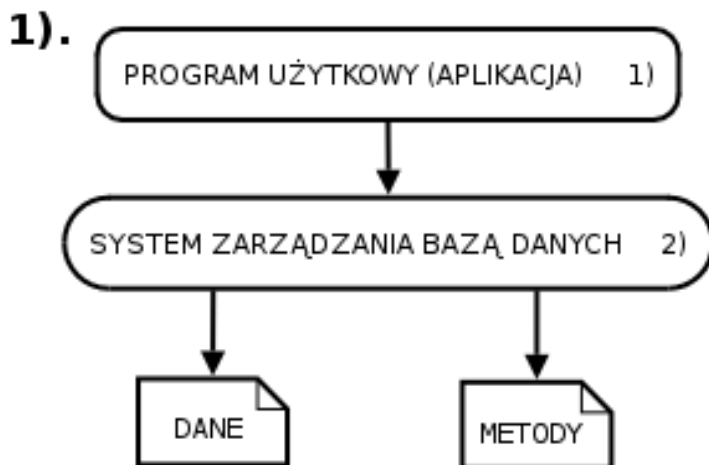


Na podstawie abstrakcyjnej definicji klasy można wygenerować konkretne obiekty.



Z klasy można wytworzyć dowolnie dużo obiektów.

- 1) W tradycyjnym podejściu głównym „aktorem” działającym w systemie jest program zarządzający który aktywizuje poszczególne programy i dane
- 2) W podejściu obiektowym wszystkie obiekty mogą być aktywne równocześnie a każdy z nich wykonuje swoją część zadań



Obiekty komunikują się ze sobą przesyłając komunikaty.
Z jednej klasy można wygenerować dalsze klasy.



Klasy potomne dziedziczą część właściwości nadklasy i tak powstaje hierarchia klas i podklas, ułatwiająca operowania złożonymi obiektami. W każdej chwili można także dodać nową podklasę. Atrybuty dotyczące obiektów wszystkich podklas wygodnie jest umieszczać w definicji klasy nadrzędnej.

Budowa systemu polega na zestawieniu go z klas, z których część jest stworzona specjalnie dla danego systemu, ale większość pochodzi z zasobów przeznaczonych do wielokrotnego użytku (np.: biblioteka klas).

Typowe źródła klas dla projektu:

- klasy unikalne <- opracowane wewnętrznie
- klasy wyspecjalizowane <- dostawca klas
- klasy ogólne <- dostawca języka