

Podsumowanie

Główny Obszar Działania	Model	Diagramy	Podstawowe Pojęcia
Struktura	Model Obiektowy	Diagram Klas	Klasa, Obiekt, Asocjacja, Generalizacja, Zależność, Realizacja, Interface
		Diagram Obiektów	
	Model Przypadków Użycia	Diagram Przypadków Użycia	Aktor, Przypadek Użycia, «include», «Extend», generalizacja
	Model Implementacji	Diagramy Komponentów	Komponent, Interface, Zależność, Realizacja
		Diagramy Wdrożeniowe	Węzeł, Komponent, Zależność, Lokacja
Dynamika	Model Dynamiczny	Diagramy Stanu	Stan, Zdarzenie, Przejście, Akcja, Aktywność
		Diagramy Aktywności	Stan, Aktywność, Fork, Join, Romb Decyzyjny
		Diagramy Interakcji	Interakcja, Współpraca, Komunikat, Aktywacja
Zarządzanie	Model Zarządzania	Diagramy Pakietów	Pakiet, Podsystem
Rozszerzalność	Wszystkie Modele	Wszystkie Diagramy	Stereotyp, Własność Etykietowana, Ograniczenie

Podstawowe zasady obiektowości

Obiekt - struktura danych, występująca łącznie z operacjami dozwolonymi do wykonywania na niej, odpowiadająca bytowi wyróżnialnemu w analizowanej rzeczywistości.

Tożsamość obiektu - wewnętrzny identyfikator obiektu, który pozwala na odróżnienie go od innych obiektów.

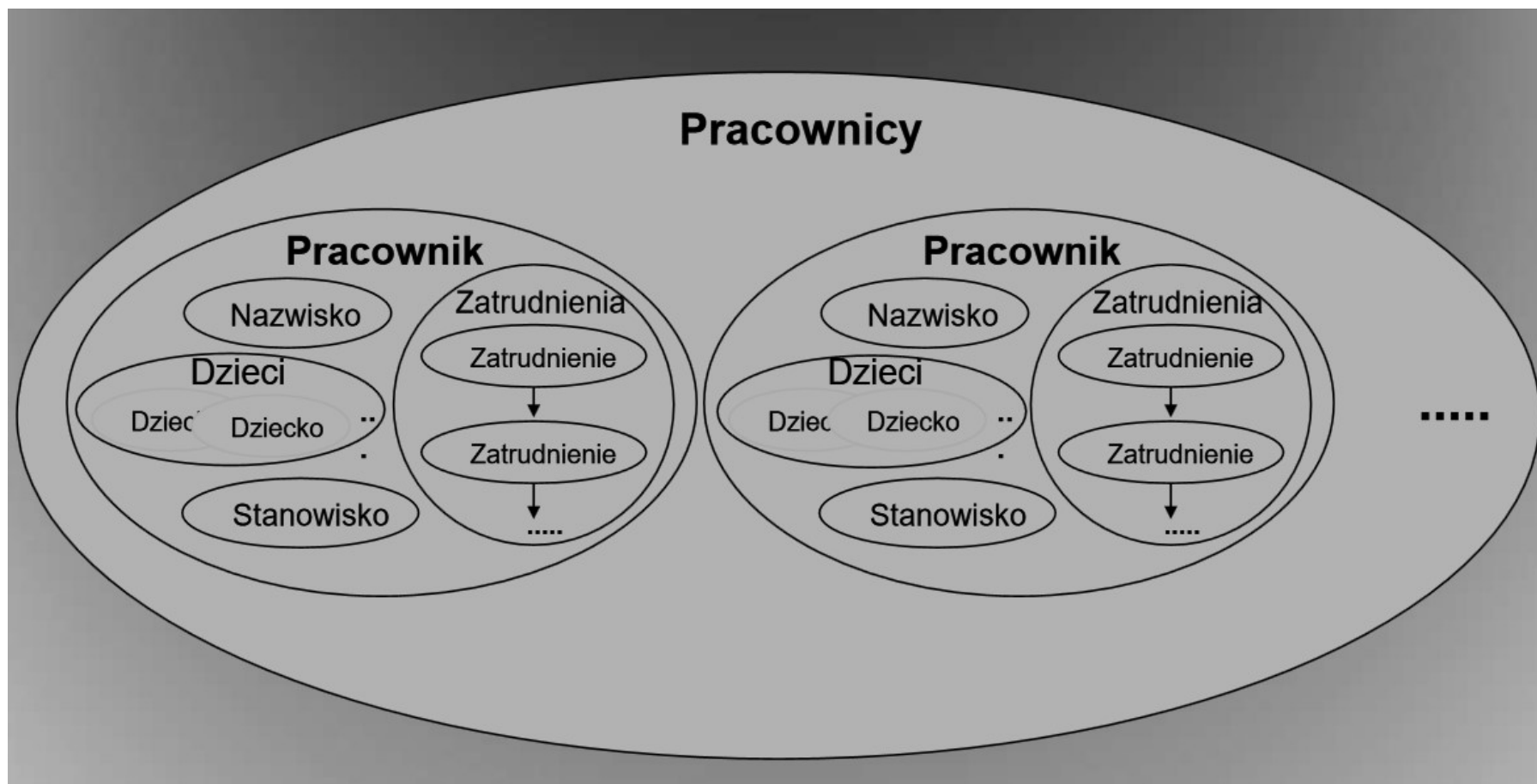
Hermetyzacja - rozróżnienie pomiędzy interfejsem do obiektu opisującym co obiekt robi, a implementacją definiującą, jak jest zbudowany i jak robi, to co ma zrobić.

Klasa - zbiór własności grupy obiektów o tych samych charakterystykach.

Dziedziczenie - wielokrotne użycie tego, co wcześniej zostało zrobione: definiowanie klas, które mają wszystkie cechy zdefiniowane wcześniej (z nadklasy) plus cechy nowe.

Polimorfizm - wybór nazwy dla operacji jest określony wyłącznie semantyką operacji. Decyzja o tym, która z metod implementujących daną operację zostanie wybrana, zależy od przynależności obiektu do odpowiedniej klasy.

Przykład obiektu złożonego



Źródła obiektowości

```
graph TD; A([Źródła obiektowości]) --> B[Języki programowania operujące na złożonych strukturach danych, wprowadzające klasy, metody, dziedziczenie i hermetyzację (Simula 67, Smalltalk).]; A --> C[Skierowanie uwagi na czynniki ludzkie w tworzeniu oprogramowania.]; A --> D[Metodyki projektowania oprogramowania, od początku bazujące na wyróżnianiu obiektów i ich klas w otaczającej nas rzeczywistości.]; A --> E[Bazy danych, od początku bazujące na obiektach (IMS, CODASYL).];
```

Języki programowania operujące na złożonych strukturach danych, wprowadzające klasy, metody, dziedziczenie i hermetyzację (Simula 67, Smalltalk).

Skierowanie uwagi na czynniki ludzkie w tworzeniu oprogramowania.

Metodyki projektowania oprogramowania, od początku bazujące na wyróżnianiu obiektów i ich klas w otaczającej nas rzeczywistości.

Bazy danych, od początku bazujące na obiektach (IMS, CODASYL).