



Komputerowe wspomaganie prac inżynierskich

Bazy danych, modele danych

Zakres zajęć

- Pojęcie bazy danych.
- Modele danych:
 - pojęcie modelu danych,
 - generacje modeli danych.
- Relacyjny model danych:
 - pojęcie relacji,
 - encje,
 - związki encji.
- Projektowanie baz danych.
- Hurtownie danych.



Baza danych

System zarządzania BD

Baza danych i jej własności

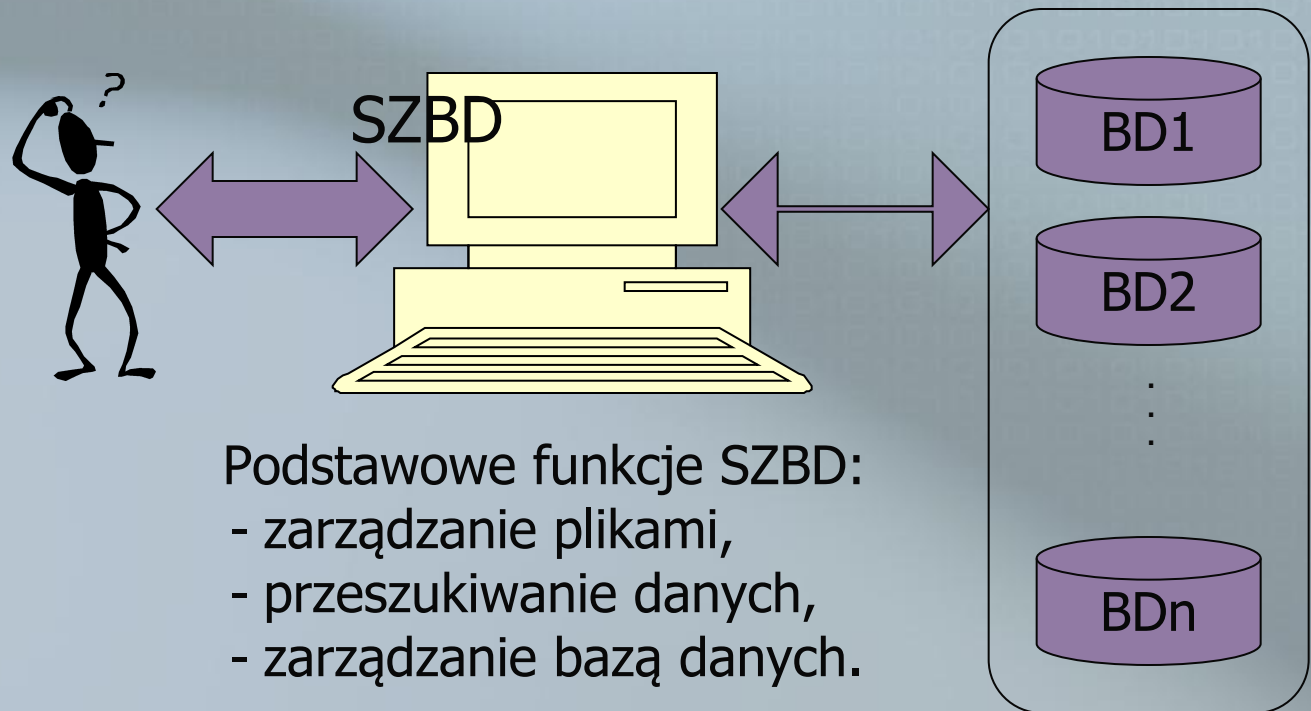
„**Baza danych** jest modelem pewnego aspektu rzeczywistości danej organizacji. Rzeczywistość tę nazywamy obszarem analizy (**OA**). (...) **Baza danych** jest magazynem danych z nałożoną na niego **wewnętrzną strukturą**. Ogólnym celem takiego magazynu jest **przechowywanie danych** związanych z pewnym zbiorem **zadań organizacyjnych**.*”

Własności bazy danych:

- współdzielenie danych – dane są dostępne dla więcej niż jednego użytkownika
- integracja danych – bez zbędnych i powtarzających się niepotrzebnie danych
- integralność danych – właściwe odzwierciedlenie obszaru analizy przez model
- bezpieczeństwo danych – ograniczenia dostępu do danych
- abstrakcja danych – odwzorowywanie istotnych szczegółów obszaru analizy
- niezależność danych – oddzielenie danych od procesów, które ich używają

System zarządzania bazami danych

System zarządzania bazami danych (SZBD, ang. DBMS) jest zbiorem narzędzi stanowiących warstwę pośredniczącą pomiędzy bazą danych a użytkownikiem i umożliwiającą dostęp do danych oraz zarządzanie bazami danych.



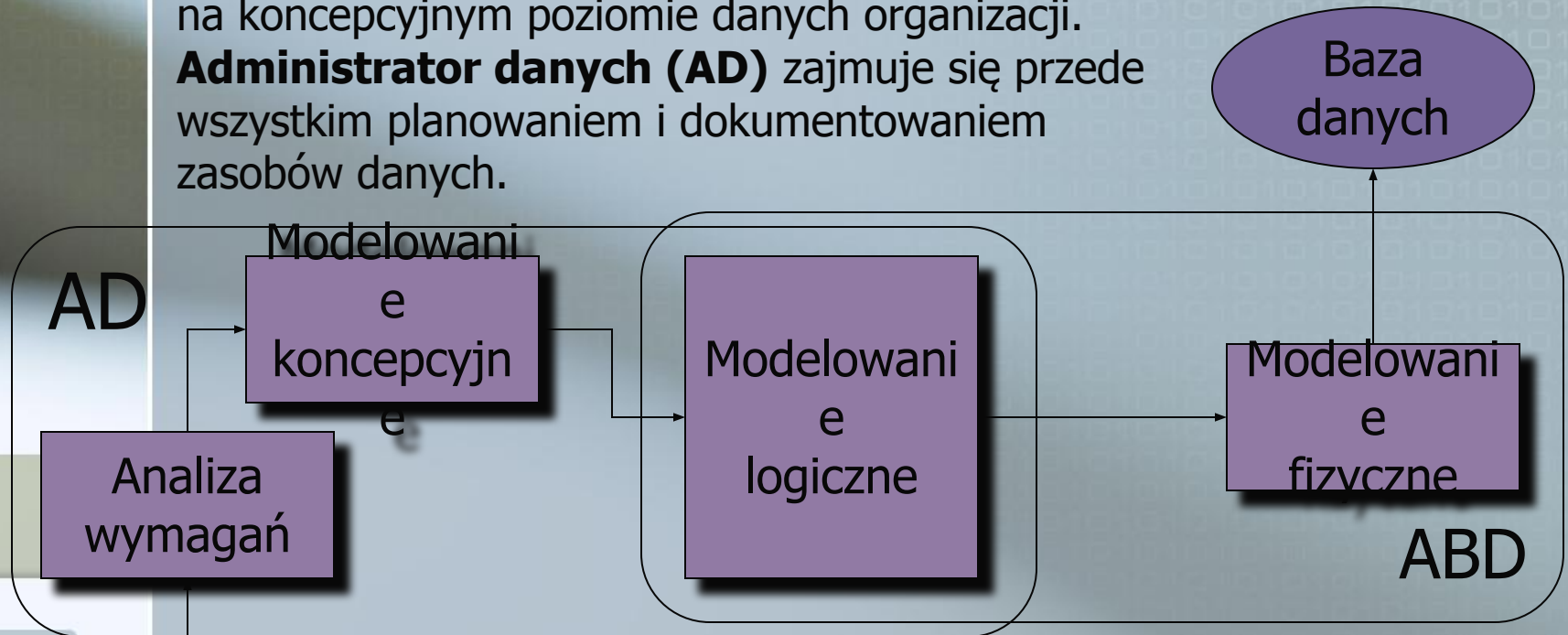
Podstawowe funkcje SZBD:

- zarządzanie plikami,
- przeszukiwanie danych,
- zarządzanie bazą danych.

Administrowanie danymi, a administrowanie bazą danych

Administrowanie danymi jest funkcją działającą na koncepcyjnym poziomie danych organizacji.

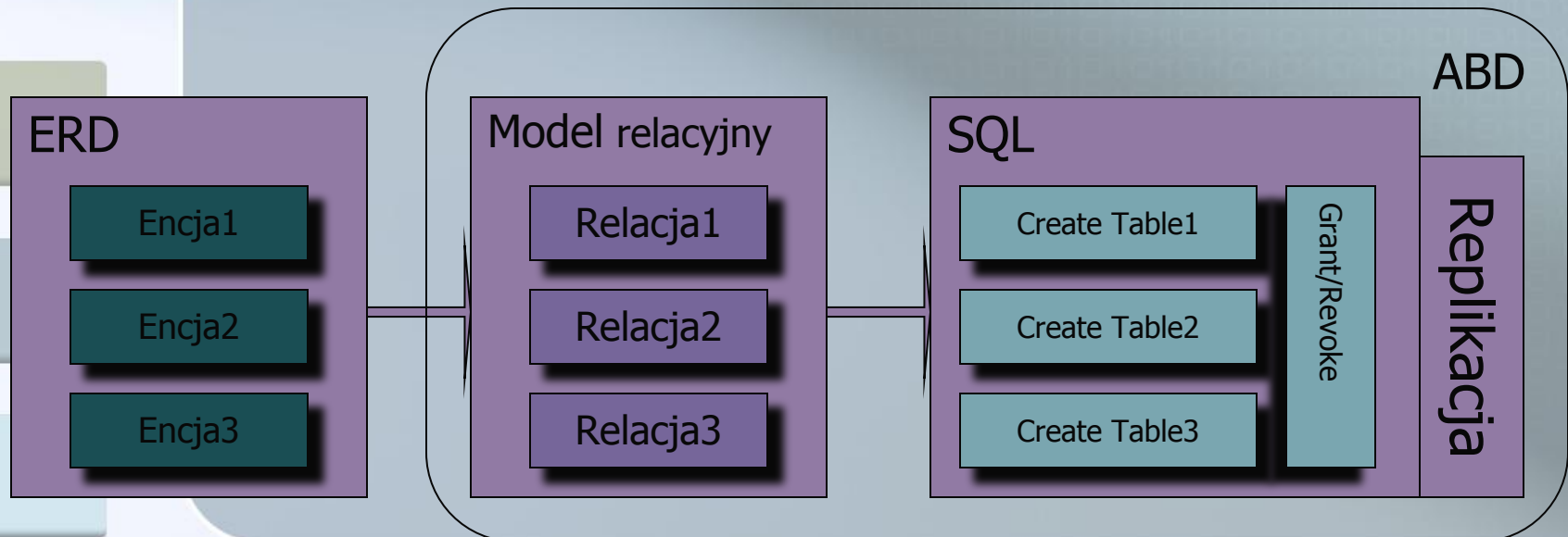
Administrator danych (AD) zajmuje się przede wszystkim planowaniem i dokumentowaniem zasobów danych.



Administrowanie bazą danych jest funkcją działającą na poziomie implementacyjnym bazy danych. **Administrator bazy danych (ABD)** odpowiedzialny jest za techniczną implementację bazy danych, zarządzanie nią i sprawowanie kontroli nad zasadami jej użycia.

Funkcje administratora bazy danych

- Podstawowe funkcje Administratora bazy danych:
- projektowanie fizyczne i implementacja bazy danych,
 - kontrola dostępu do danych,
 - monitorowanie użycia danych,
 - archiwizowanie danych,
 - tworzenie kopii zapasowych i odtwarzanie danych,
 - szkolenie użytkowników bazy danych.





Pojęcie modelu danych

Generacje modeli danych

Modele danych

Model danych w kontekście architektury danych to „... zbiór zasad dotyczących struktur danych, ich używania oraz określania warunków wspomagających utrzymywanie zgodności danych z rzeczywistością.*”

Modele danych obejmują trzy główne kategorie reguł:

- **definicje danych** – zbiór reguł określających logiczną strukturę danych
- **zasady operowania danymi** – reguły dotyczące procesu dostępu do danych i ich modyfikacji;
- **zasady integralności danych** – reguły określające jakie operacje na danych są dopuszczalne aby dane pozostały nadal poprawne.



Modele danych

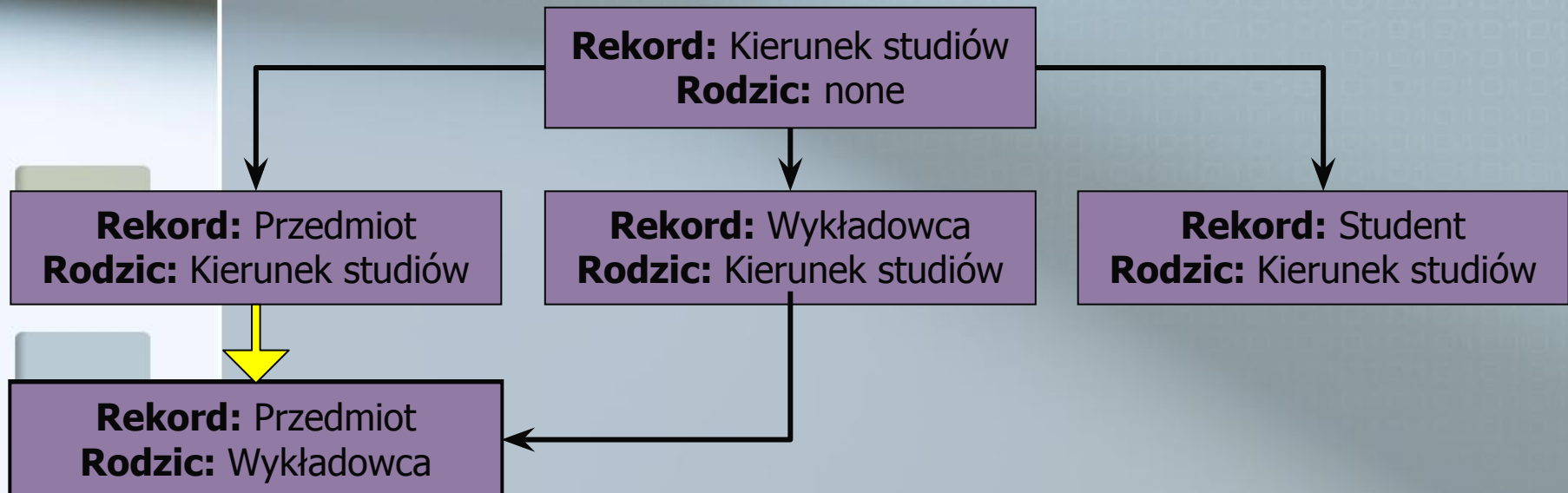
Generacje architektonicznych modeli danych:

- **modele proste**, w których obiekty reprezentowane są za pomocą struktury rekordów zgrupowanych w strukturach plików (jeden plik stanowi zbiór rekordów);
- **modele klasyczne**, w których następuje nadbudowanie odpowiedniej struktury nad zbiorami rekordów, do modeli klasycznych należą model hierarchiczny, model sieciowy oraz model relacyjny;
- **modele semantyczne**, w których wykorzystywane są narzędzia umożliwiające określenie znaczenia danych w schemacie bazy danych, do tej generacji zaliczany jest niekiedy obiektowy model danych.



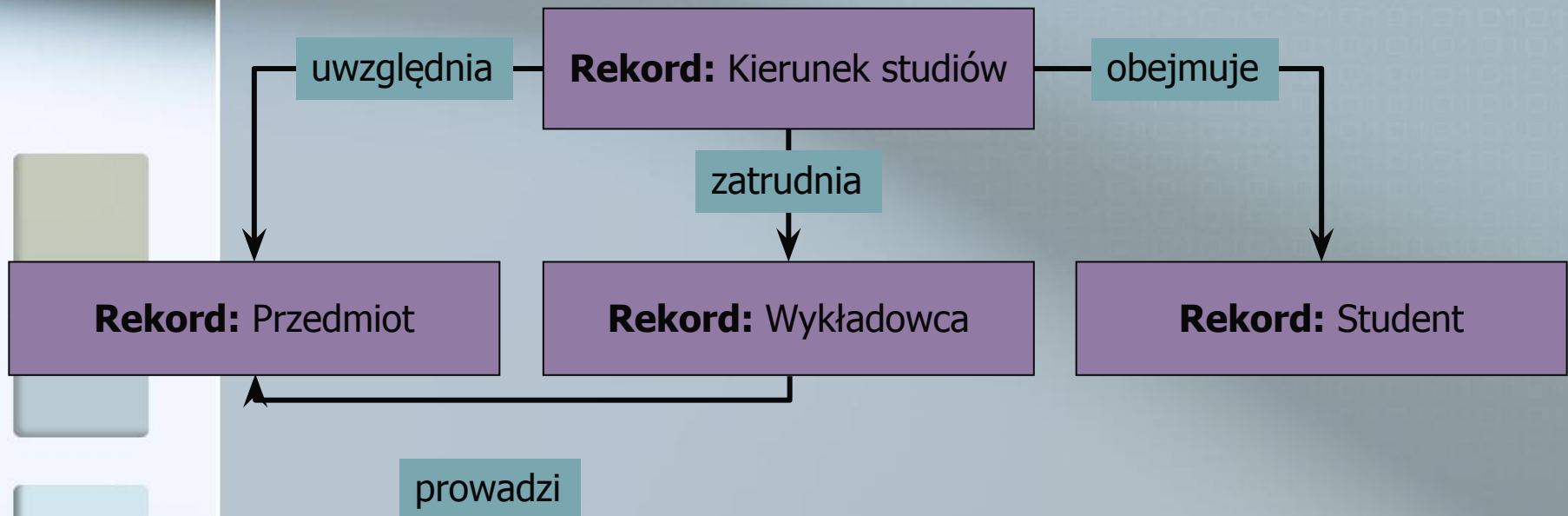
Hierarchiczny model danych

Hierarchiczny model danych wykorzystuje dwie podstawowe struktury danych: typy rekordów oraz hierarchiczne związki jeden-do-wiele (typu rodzic/dziecko). Model ten ma znaczenie historyczne.



Sieciowy model danych

Sieciowy model danych wykorzystuje również dwie podstawowe struktury danych: typy rekordów oraz typy kolekcji, stanowiące związki jeden-do-wiele. Istotne jest, że jeden typ rekordu podrzędnego może być podporządkowany kilku typom rekordów nadrzędnych. Model ten stanowi rozwinięcie modelu hierarchicznego.





Relacyjny model danych

Relacyjny model danych

Relacyjny model danych wykorzystuje tylko jedną strukturę danych – relację. **Relacja jest dwuwymiarową tabelą** spełniającą następujące zasady:

- każda relacja ma jednoznaczną nazwę;
- każda kolumna w relacji ma jednoznaczną nazwę w ramach relacji;
- wszystkie wartości w kolumnie muszą być tego samego typu;
- porządek kolumn w relacji jest nieistotny;
- każdy wiersz relacji musi być różny (powtarzanie wierszy jest niedozwolone);
- porządek wierszy nie jest istotny;
- każde pole leżące na przecięciu kolumny/wiersza w relacji powinno zawierać wartość atomową.

Twórcą założeń relacyjnego modelu danych jest dr E. F. Codd (z wykształcenia matematyk), który opublikował je w roku 1970.

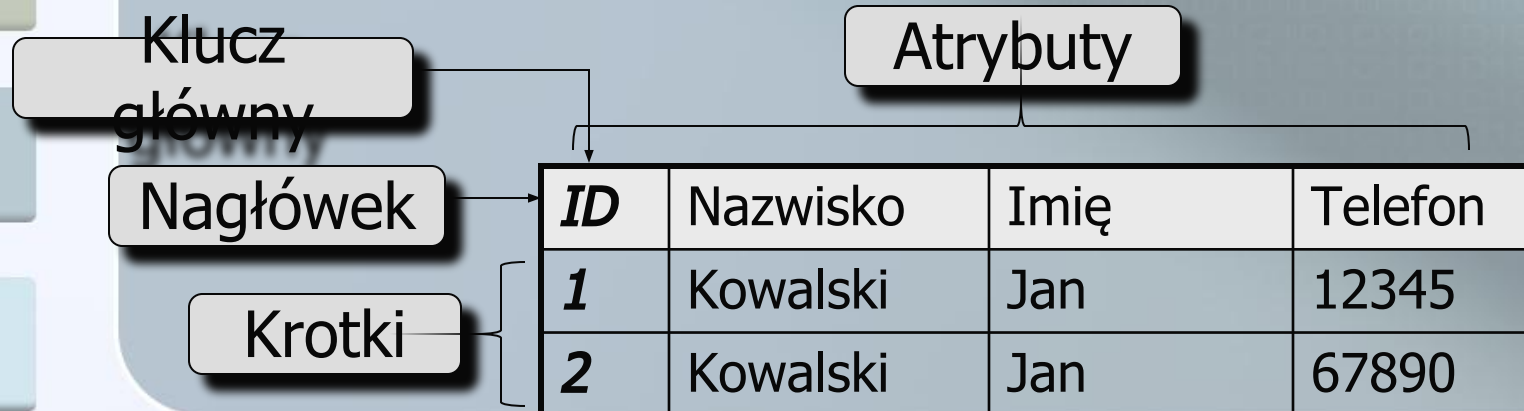
Relacyjny model danych c.d.

Każda relacja posiada **nagłówek**, określający jej **atrybuty**. Nazwa relacji wraz ze zbiorem jej atrybutów określana jest mianem **schematu relacji**, przedstawianego w sposób następujący:

NazwaRelacji(atrybut-1, atrybut-2,...,atrybut-n).

Kolejne wiersze relacji, poza wierszem nagłówkowym nazywane są **krotkami**.

Dla każdej relacji musi być określony jednoznaczny identyfikator określany mianem **klucz główny**.



Pojęcie ENCJI

Encja to pewien wyodrębniony logicznie i jednoznacznie określony byt (obiekt), rozpoznawalny w badanej rzeczywistości i pełniący w niej określoną rolę. Encja może być zarówno obiektem fizycznym (takim jak np. samochód, drzewo, książka itp.) jak również zdarzeniem (np. sprzedaż samochodu, zasadzenie drzewa, zakup książki itp.). Każda encja jest jednoznacznie identyfikowana na podstawie swojej nazwy. Przyjęło się, że nazwy encji są *rzeczownikami w liczbie pojedynczej*. Graficznie każda encja jest reprezentowana przez prostokąt.

Badana rzeczywistość:
**organizacja zajęć
dydaktycznych**

Kierunek

Przedmiot

Wykładowc
a

Student

Sala

Grupa

Atrybuty ENCJI

Atrybuty encji to cechy (własności) charakteryzujące daną encję w badanej rzeczywistości, którym przypisywane są określone wartości. Wartości poszczególnych atrybutów pozwalają odróżniać encje od siebie.

Encje posiadające te same własności tworzą **typy (zbiory) encji**. W praktyce, dla uproszczenia przyjęto używać określenia *encja* zarówno w odniesieniu do typu encji, jak również do określonego wystąpienia encji (określonej instancji encji).

Student	Sala
<i>Nazwisko</i>	<i>Numer</i>
<i>Imię</i>	<i>Rodzaj</i>
<i>Rok_Studiów</i>	<i>Ilość_miejsc</i>
<i>Nr_indeksu</i>	<i>Sprzęt</i>

Typ Encji: Studenci

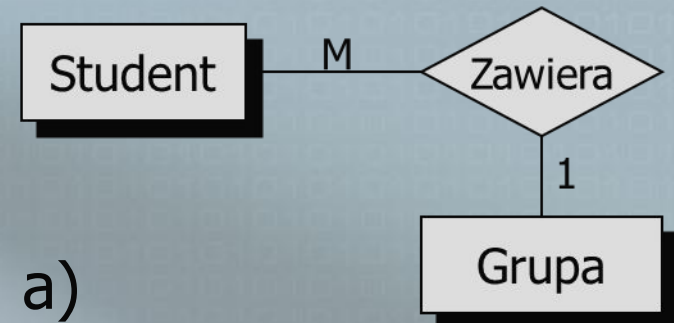
Instancja encji:
Nazwisko: Kowalski
Imię: Jan
Rok_Studiów: I
Nr_indeksu: R-10/03

Instancja encji:
Nazwisko: Nowak
Imię: Anna
Rok_Studiów: IV
Nr_indeksu: R-24/99

Związki ENCJI

Związki określają powiązania pomiędzy poszczególnymi encjami. W najprostszymi modelach uwzględniane są związki występujące pomiędzy dwoma encjami. Pomiedzy dwoma różnymi encjami może zachodzić wiele związków, ale pomiędzy dwoma tymi samymi encjami może zachodzić tylko jeden związek.

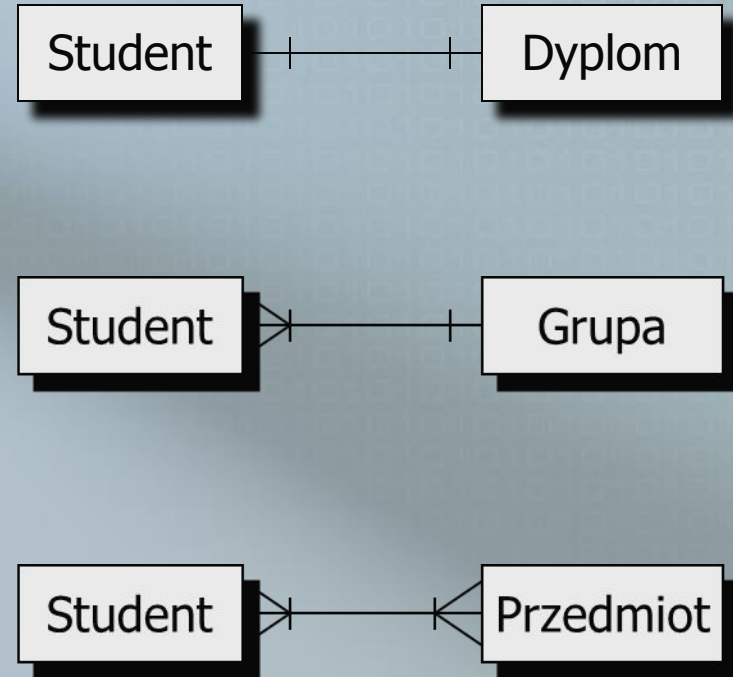
Każdy związek posiada swoją nazwę. Przyjęło się, że nazwy związków są czasownikami. Graficznie związek jest zwykle reprezentowany przez romb połączony liniami z encjami, pomiędzy którymi zachodzi wraz z oznaczeniem jego liczebności (a) lub też przez samą linię zakończoną symbolami określającymi jego liczebność (b). W niektórych notacjach nazwy związków się pomija.



Liczebność związku ENCJI

Liczebność (stopień) związku encji określa liczbę instancji biorących udział w danym związku. Rozróżnia się związki:

- **jednojednoznaczne (jeden-do-jeden, 1:1)** – każdej instancji pierwszej encji odpowiada dokładnie jedna instancja drugiej encji i odwrotnie;
- **jednoznaczne (jeden-do-wiele, 1:M)** – każdej instancji pierwszej encji odpowiada M instancji drugiej encji, ale każdej instancji drugiej encji odpowiada tylko jedna instancja pierwszej encji;
- **wieloznaczne (wiele-do-wiele, M:N)** – każdej instancji pierwszej encji odpowiada M instancji drugiej encji, a każdej instancji drugiej encji odpowiada N instancji pierwszej encji.





Projektowanie baz danych

Diagram związków ENCJI (ERD)

Diagram związków encji (ERD) jest graficznym przedstawieniem encji występujących w modelu wraz ze związkami między nimi. Stanowi podstawowe narzędzie koncepcyjnego etapu projektowania bazy danych.

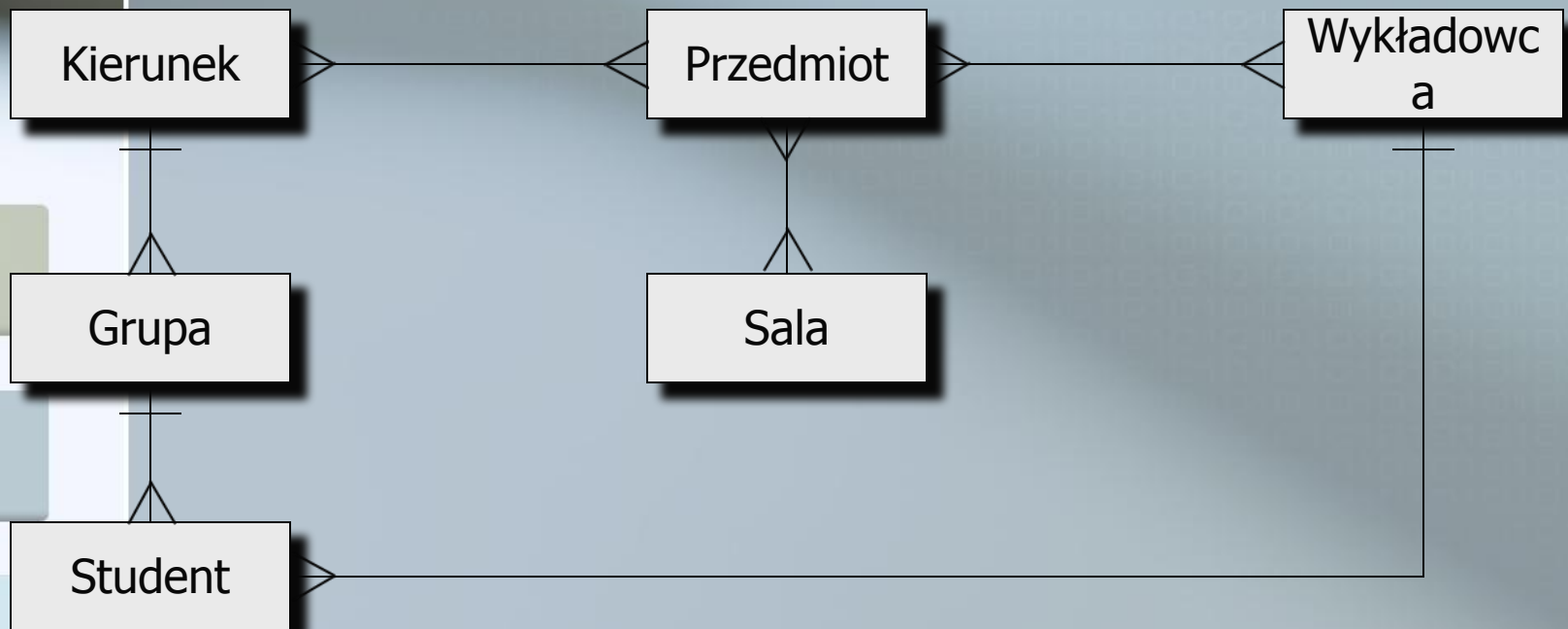


Diagram związków ENCJI (ERD)

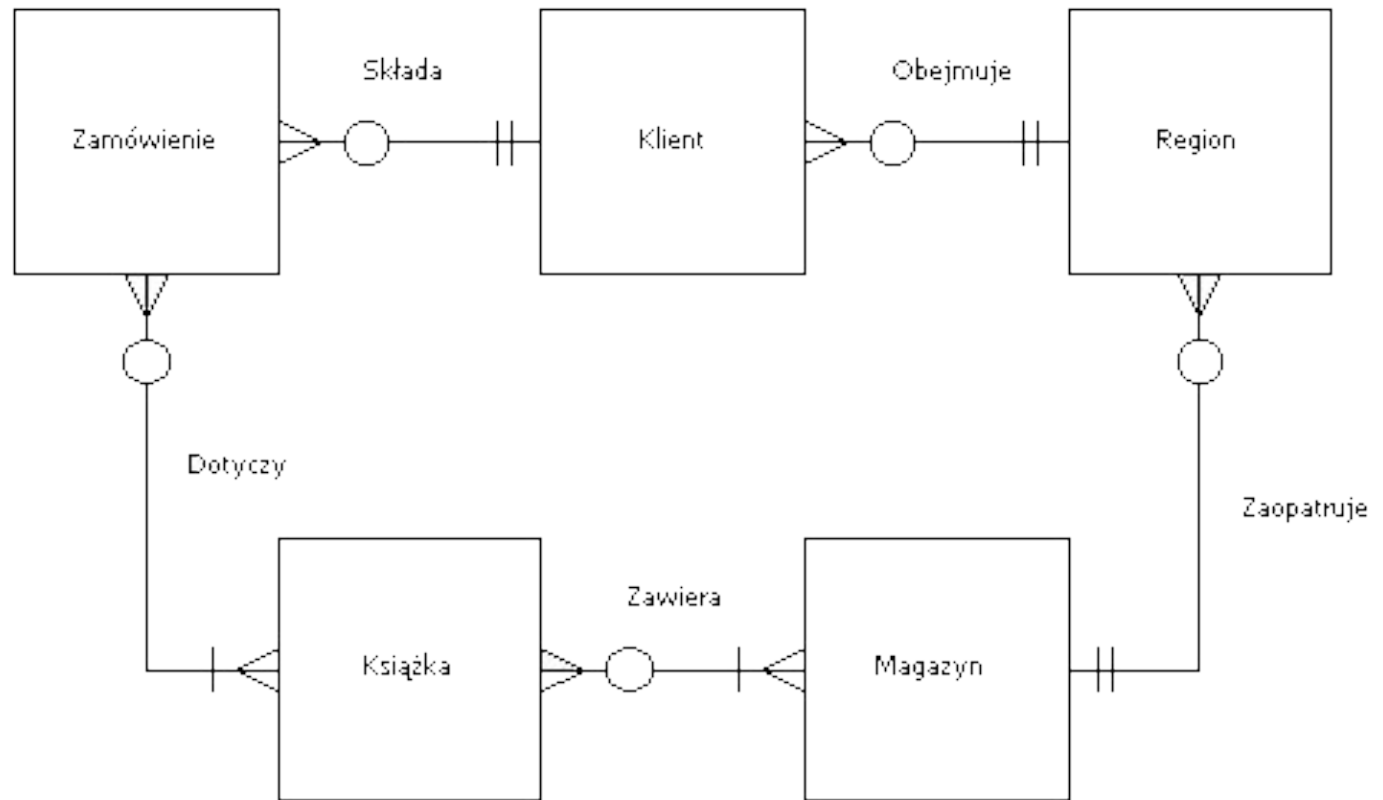


Diagram związków ENCJI (ERD)

EGZAMINATOR
Id-egz CHARACTER (4)
Nazwisko CHARACTER (25)
Imię CHARACTER (15)
Kod CHARACTER (5)
Miasto CHARACTER (15)
Ulica CHARACTER (30)
Numer CHARACTER (8)
Tel CHARACTER (12)
Fax CHARACTER (12)
E-mail CHARACTER (30)

EGZAMIN
nr-egz INTEGER (7)
nr-EKUK <FK1> CHARACTER (7)
nr-modulu <FK2> INTEGER (1)
Id-egz <FK3> CHARACTER (4)
Data DATE (8)
kod-o <FK4> INTEGER (5)
zdał LOGICAL (1)

EGZAMINOWANY
nr-EKUK CHARACTER (7)
Nazwisko CHARACTER (25)
Imię CHARACTER (15)
Data-ur DATE (8)
Miejsce CHARACTER (15)
PESEL CHARACTER (11)
Kod CHARACTER (5)
Miasto CHARACTER (15)
Ulica CHARACTER (30)
Numer CHARACTER (8)
Tel CHARACTER (12)
Fax CHARACTER (12)
E-mail CHARACTER (30)
nr-ECDL CHARACTER (9)
data-ECDL DATE (8)

OSRODEK
kod-o INTEGER (5)
nazwa-o CHARACTER (30)
Kod CHARACTER (5)
Miasto CHARACTER (15)
Ulica CHARACTER (30)
Numer CHARACTER (8)

MÓDUL
nr-modulu INTEGER (1)
nazwa-m CHARACTER (40)
opis CHARACTER (200)

przeprowadza

zdaje

jest miejscem

jest tematem



Diagram związków ENCJI (ERD)

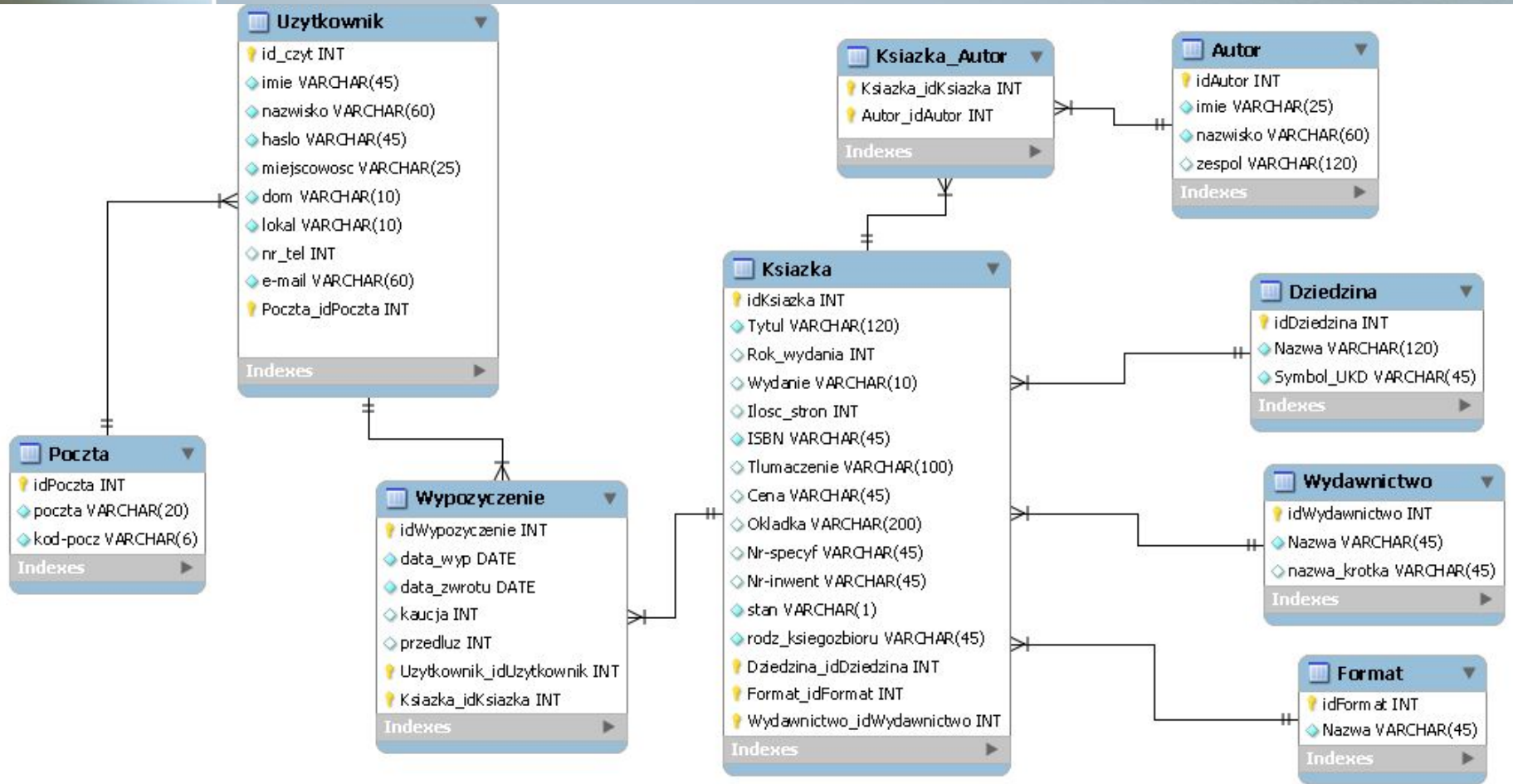
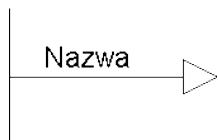
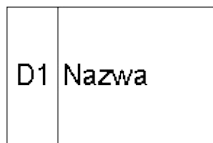
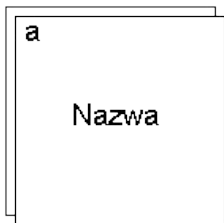
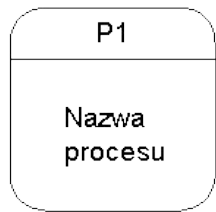


Diagram przepływu danych (DFD)

Diagram przepływu danych (*ang. Data Flow Diagram - DFD*) jest graficzną prezentacją przepływu danych w obrębie obszaru analizy. Uwzględnia:

- **Funkcje** — (procesy) realizujące określone cele; jeśli funkcji nie można rozbić na podfunkcje, wówczas nosi ona nazwę elementarnej.
- **Magazyny danych** — trwałe lub tymczasowe składnice danych, które są argumentami dla funkcji.
- **Terminatory** — obiekty, które nie są częścią systemu, ale stanowią odbiorców bądź źródła danych lub argumentów funkcji.
- **Przepływy** — elementy pokazujące kierunek przesyłu danych (np. bajtów, znaków, pakietów..).

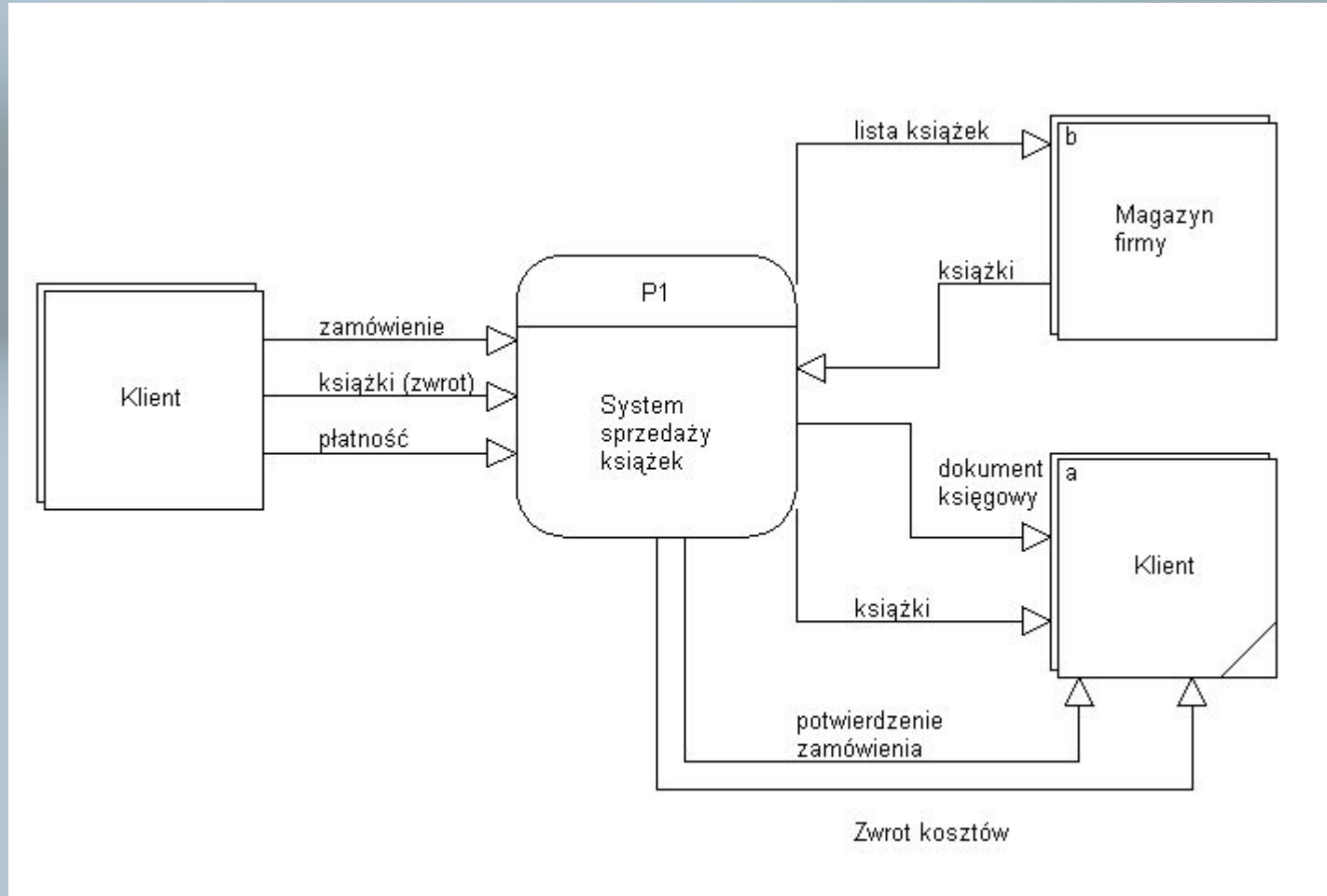


DFD – poziomy szczegółowości

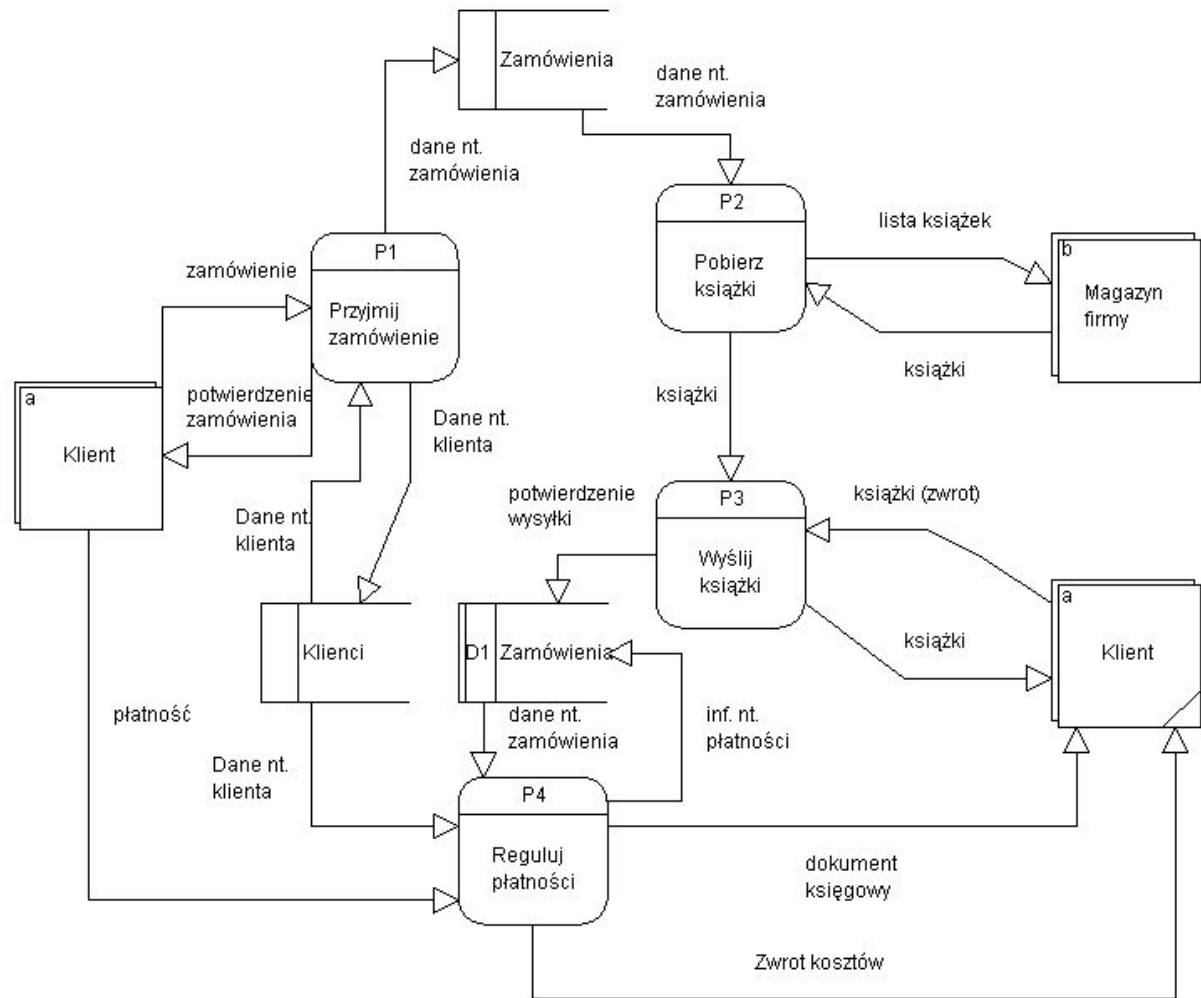
DFD obrazuje za pomocą przepływów kierunek przesyłania danych pomiędzy funkcjami, magazynami i obiektami zewnętrznymi. **DFD** są zwykle prezentowane na różnych stopniach szczegółowości. Rozróżnia się:

- **diagramy kontekstowe**, które pokazują granice systemu, źródła i odbiorców danych oraz główne wejścia i wyjścia systemu;
- **diagramy systemowe** (inaczej zerowe), które przedstawiają główne funkcje systemu;
- **diagramy szczegółowe** (procesów elementarnych), które pokazują szczegółowe realizacje funkcji systemowych

DFD – przykład diagramu kontekstowego



DFD – przykład diagramu systemowego



Normalizacja

Normalizacja jest procesem pozwalającym *metodą dekompozycji odwracalnej* wyeliminować anomalie, pojawiające się w trakcie korzystania z pliku danych. Do anomalii tych należą:

- uboczny efekt usunięcia (np. usunięcie studenta nr 15),
- uboczny efekt modyfikacji (np. zmiana wykładowcy),
- uboczny efekt wstawienia (np. dopisanie nowego studenta).

Nazwa przedmiotu	Nr wyk.	Nazwisko wykładowca	Nr stud.	Nazwisko studenta	Ocena	Typ oceny
Administrowanie BD	1	Iwan St.	12	Kowalski J.	5	Zal
Administrowanie BD	1	Iwan St.	12	Kowalski J.	4	Egz
Administrowanie BD	1	Iwan St.	34	Nowak A.	4	Zal
Administrowanie BD	1	Iwan St.	44	Nowak W.	5	Zal
Administrowanie BD	1	Iwan St.	44	Nowak W.	5	Egz
Systemy operacyjne	1	Iwan St.	12	Kowalski J.	4	Zal
Systemy operacyjne	1	Iwan St.	12	Kowalski J.	4	Egz
J. Angielski	5	Brown J.	15	Wiśniak K.	3	Zal

Etapy normalizacji

Normalizacja przeprowadzana jest etapowo. Zwykle doprowadzenie tabel do trzeciej postaci normalnej jest wystarczające i kończy proces normalizacji. W niektórych wypadkach konieczne jest przeprowadzenie dalszych etapów.

Etapy przeprowadzania normalizacji:

- przygotowanie zbioru danych w postaci tabeli;
- przekształcenie nieznormalizowanej tabeli do pierwszej postaci normalnej (1NF);
- przekształcenie tabeli w pierwszej postaci normalnej do drugiej postaci normalnej (2NF);
- przekształcenie tabeli w drugiej postaci normalnej do trzeciej postaci normalnej (3NF);
- przekształcenie tabeli do postaci normalnej Boyce'a-Codda (BCNF);
- przekształcenie tabeli do czwartej i piątej postaci normalnej (4NF oraz 5NF).

Pierwsza postać normalna (1NF)

Relacja jest w **pierwszej postaci normalnej (1NF)** wtedy i tylko wtedy, gdy każdy atrybut niekluczowy jest funkcyjnie zależny od klucza głównego.*

<u>Nazwa przedmiotu</u>	Nr wyk.	Nazwisko wykładowca
Administrowanie BD	1	Iwan St.
Systemy operacyjne	1	Iwan St.
J. Angielski	5	Brown J.

<u>Nazwa przedmiotu</u>	<u>Nr stud.</u>	<u>Typ oceny</u>	Nazwisko studenta	Ocena
Administrowanie BD	12	Zal	Kowalski J.	5
Administrowanie BD	12	Egz	Kowalski J.	4
Administrowanie BD	34	Zal	Nowak A.	4
Administrowanie BD	44	Zal	Nowak W.	5
Administrowanie BD	44	Egz	Nowak W.	5
Systemy operacyjne	12	Zal	Kowalski J.	4
Systemy operacyjne	12	Egz	Kowalski J.	4
J. Angielski	15	Zal	Wiśniak K.	3

*Beynon-Davies P., *Systemy baz danych*, WNT, Warszawa 2000

Druga postać normalna (2NF)

Relacja jest w **drugiej postaci normalnej (2NF)** wtedy i tylko wtedy, gdy jest w 1NF i każdy atrybut niekluczowy jest w pełni funkcyjnie zależny od klucza głównego.*

<u>Nazwa przedmiotu</u>	Nr wyk.	Nazwisko wykładowca
Administrowanie BD	1	Iwan St.
Systemy operacyjne	1	Iwan St.
J. Angielski	5	Brown J.

<u>Nr stud.</u>	Nazwisko studenta
12	Kowalski J.
34	Nowak A.
44	Nowak W.
15	Wiśniak K.

<u>Nazwa przedmiotu</u>	<u>Nr stud.</u>	<u>Typ oceny</u>	Ocena
Administrowanie BD	12	Zal	5
Administrowanie BD	12	Egz	4
Administrowanie BD	34	Zal	4
Administrowanie BD	44	Zal	5
Administrowanie BD	44	Egz	5
Systemy operacyjne	12	Zal	4
Systemy operacyjne	12	Egz	4
J. Angielski	15	Zal	3

*Beynon-Davies P., *Systemy baz danych*, WNT, Warszawa 2000

Trzecia postać normalna (3NF)

Relacja jest w **trzeciej postaci normalnej (3NF)** wtedy i tylko wtedy, gdy jest w 2NF i każdy niekluczowy atrybut jest bezpośrednio zależny (a nie przechodnio zależny) od klucza głównego.*

<u>Nazwa przedmiotu</u>	Nr wyk.
Administrowanie BD	1
Systemy operacyjne	1
J. Angielski	5

<u>Nr wyk.</u>	Nazwisko wykładowca
1	Iwan St.
5	Brown J.

<u>Nr stud.</u>	Nazwisko studenta
12	Kowalski J.
34	Nowak A.
44	Nowak W.
15	Wiśniak K.

<u>Nazwa przedmiotu</u>	<u>Nr stud.</u>	<u>Typ oceny</u>	Ocena
Administrowanie BD	12	Zal	5
Administrowanie BD	12	Egz	4
Administrowanie BD	34	Zal	4
Administrowanie BD	44	Zal	5
Administrowanie BD	44	Egz	5
Systemy operacyjne	12	Zal	4
Systemy operacyjne	12	Egz	4
J. Angielski	15	Zal	3

*Beynon-Davies P., *Systemy baz danych*, WNT, Warszawa 2000

Słownik danych

Słownik danych jest narzędziem pozwalającym na zapisywanie metadanych (danych na temat danych) przedsiębiorstwa. Termin metadane odnosi się zarówno do zasobów danych, jak również wymagań dotyczących danych. W słowniku danych każda encja wraz ze swoimi atrybutami zostaje szczegółowo opisana (z uwzględnieniem typów i zakresów danych oraz kluczy). Słownik danych powinien zawierać również opisy poszczególnych związków wraz z ich charakterystyką.

Nazwa encji	Nazwa atrybutu	Typ danych	Zakres danych
Student	<u>ID</u>	Integer	
	Nazwisko	Char(20)	
	Imię	Char(20)	
	Rok_studiów	Integer	Od 1 do 5
	Nr_indeksu	Char(8)	R-###/##
Sala	<u>Numer</u>	Integer	
	Rodzaj	Char(13)	Zwykła Multimedialna
	Ilość_miejsc	Integer	15 lub 30
	Sprzęt	Char(50)	

Wybór środowiska implementacji

Implementacja bazy danych obejmuje realizację projektu z zastosowaniem określonej strategii i przy użyciu wybranego środowiska implementacji (systemu zarządzania bazami danych). Dokonując wyboru środowiska implementacji należy kierować się kilkoma aspektami, determinującymi jego użycie:

- ogólnym przeznaczeniem projektowanej bazy danych;
- postacią danych;
- sposobami ich użycia i prezentacji;
- zakresem dostępu do danych (przewidywaną ilością użytkowników oraz zakresem ich uprawnień);
- wymaganiami bezpieczeństwa i możliwościami oferowanymi w tym zakresie przez system.

Istotnym etapem kończącym procedurę implementacji jest testowanie bazy danych, pozwalające na wyeliminowanie błędów.

Obiektowy model danych

Koncepcja **obiektowego modelu danych** sprowadza się do rozszerzenia modelu relacyjnego o cechy obiektowe. W modelu tym mogą występować relacje zagnieżdżone, opisujące atrybuty złożone. Podstawowymi składnikami modelu obiektowego są:

- obiekty, stanowiące pakiety danych (przechowywanych w atrybutach obiektu) i metod (umożliwiających operowanie danymi);
- klasy obiektów, stanowiące zgrupowanie podobnych obiektów, pozwalające na określanie wspólnych dla grupy obiektów atrybutów, metod i związków.

Z pojęciem klas związane jest pojęcie dziedziczenia, polegające na przejmowaniu przez podklasy atrybutów i metod z klasy nadrzędnej.

Klasa: Pracownik
Atrybuty:
Nazwisko (Char),
Staż (Int)

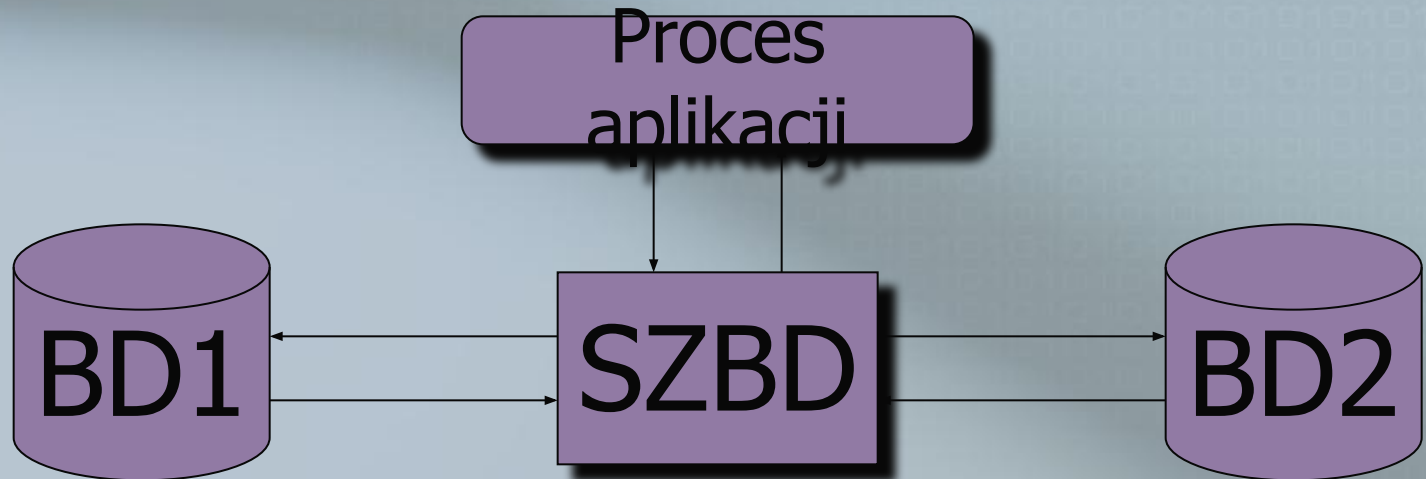
Klasa: Kierownik
Atrybuty:
Dodatek (Int)

Klasa
nadrzędna

Podklasa

Rozproszone bazy danych

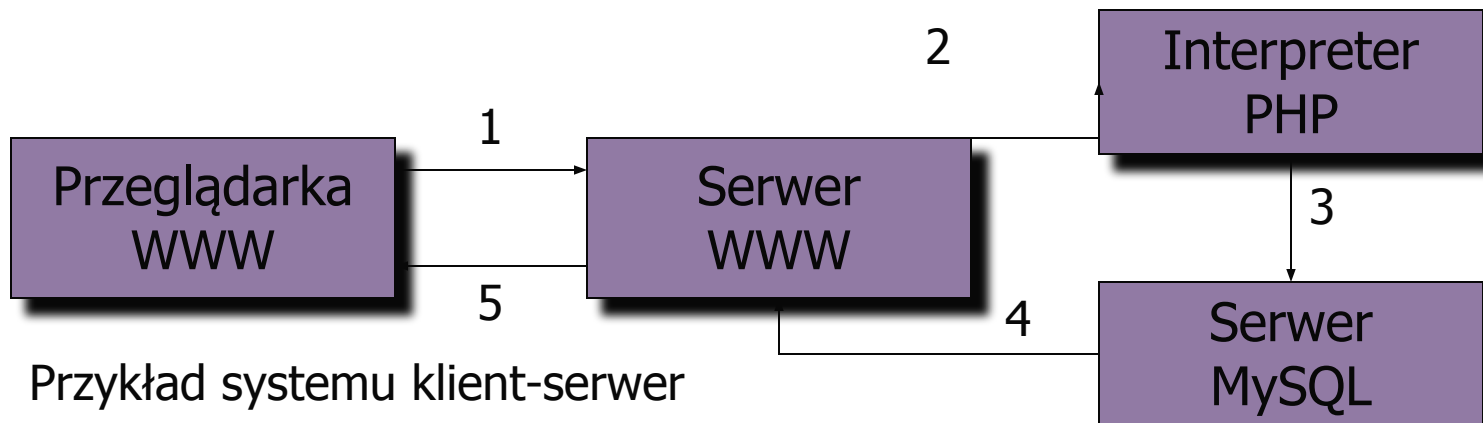
Rozproszone bazy danych opierają swoją strukturę w dużej mierze na relacyjnym modelu danych. Są to systemy, w których występuje fragmentacja danych i rozmieszczenie ich w różnych geograficznie miejscach organizacji.



Rozproszone bazy danych c.d.

Do podstawowych typów rozproszonych baz danych należą:

- systemy klient-serwer, w których baza danych jest przechowywana na jednym serwerze i może być udostępniana wielu klientom;
- systemy jednorodne, w których dane rozłożone są na wielu serwerach, opartych na tym samym SZBD;
- systemy niejednorodne, w których konfiguracje sprzętowe i oprogramowania serwerów mogą być różne;
- systemy federacyjne, składające się z pewnej liczby niezależnych baz danych, które w razie potrzeby mogą wykonywać wspólne zadania.



Do przemyślenia w domu

- Programowanie komputerów:
 - program, programowanie, język programowania
 - rozwój i generacje języków programowania
 - klasyfikacja języków programowania
 - środowiska programistyczne
- Podstawy algorytmizacji:
 - pojęcie algorytmu,
 - rodzaje algorytmów,
 - schematy blokowe.





Dziękuję za uwagę

Zapraszam w przyszłym tygodniu