

Bazy danych

Andrzej Grzybowski

Instytut Fizyki, Uniwersytet Śląski

Wykład 5

Normalizacja relacji bazy danych jako podstawa relacyjnego modelowania danych

(wykład przygotowany z wykorzystaniem materiałów z seminarium prof. Zbyszko Królikowskiego, wygłoszonego w ramach II seminarium PLOUG (Polish Oracle Users Group, tj. Stowarzyszenia Polskiej Grupy Użytkowników Systemu Oracle):

<http://www.ploug.org.pl/wp-content/uploads/seminarium-02-normalizacja.pdf> ,

<http://www.ploug.org.pl/wp-content/uploads/seminarium-02-2.pdf> ,

które zostały uzupełnione o inne nowsze źródła, w tym skrypt Zbigniewa Jurkiewicza

<http://mst.mimuw.edu.pl/lecture.php?lecture=bad&part=Ch6>)

Powody normalizacji relacji bazy danych: zmniejszenie redundancji składowanych danych i ograniczenie anomalii manipulowania danymi

Przykład dla relacji **ZATRUDNIENI** o nagłówku:

$N(\text{ZATRUDNIENI}) = \{\text{ID_PRACOWNIKA}, \text{IMIE}, \text{NAZWISKO}, \text{ID_DZIAŁU}, \text{NAZWA_DZIAŁU}\}$

i kluczu $PK(\text{ZATRUDNIENI}) = \{\text{ID_PRACOWNIKA}\}$

- **Anomalia wstawiania:**

(i) wstawienie danych dla nowego pracownika z działu o $\text{ID_DZIAŁU} = 20$ wymaga zadbania o spójność wartości atrybutu NAZWY_DZIAŁU z innymi krotkami, w których zapisane są dane pracowników z działu 20;

(ii) nie można wstawić danych nowego działu, w którym nie ma jeszcze pracowników;

- **Anomalia usuwania:**

usuwając ostatnią krotkę z danymi pracownika z działu o $\text{ID_DZIAŁU} = 20$, tracimy informację, że taki dział istnieje;

- **Anomalia modyfikacji (aktualizacji):**

zmiana (wartości) NAZWY_DZIAŁU o $\text{ID_DZIAŁU} = 20$ powoduje konieczność aktualizacji jego nazwy w innych krotkach, zawierających dane pracowników zatrudnionych w tym dziale, w celu zachowania spójności przechowywanych danych.

Ogólne cechy procedury normalizacji, dotyczące postaci normalnych rozważanych w tym wykładzie

Zasady dekompozycji relacji R o nagłówku $N(R)=\{A_1,A_2,\dots,A_n\}$

na relacje S i T o nagłówkach $N(S)=\{B_1,B_2,\dots,B_m\}$ i $N(T)=\{C_1,C_2,\dots,C_k\}$

- $N(R) = N(S) \cup N(T)$
- Krotki relacji S i T powstają odpowiednio przez projekcje

$$S = \pi_{B_1,B_2,\dots,B_m}(R) \text{ i } T = \pi_{C_1,C_2,\dots,C_k}(R)$$

Wymagane własności procesu normalizacji w przypadku większości rozważanych w tym wykładzie postaci normalnych relacji bazy danych:

- Żaden atrybut dekomponowanej relacji R nie może zostać stracony w procesie normalizacji – **bezstratność atrybutów**
- Dane dekomponowanej relacji R nie mogą zostać zgubione w procesie normalizacji – **bezstratność danych**
- Zależności funkcyjne pomiędzy atrybutami w dekomponowanej relacji R muszą znaleźć swoją reprezentację w relacjach powstałych w procesie normalizacji – **bezstratność zależności funkcyjnych**

Dekompozycja relacji w takim procesie normalizacji jest odwracalna!

Zależność funkcyjna atrybutów relacji bazy danych

Definicja zależności funkcyjnej podzbiorów atrybutów relacji

Niech X i Y będą podzbiorami atrybutów relacji R .

X wyznacza funkcyjnie Y (lub Y jest funkcyjnie zależne od X), wtedy i tylko wtedy, gdy dla dwóch dowolnych krotek t_1 i t_2 takich, że $t_1[X] = t_2[X]$ zachodzi zawsze $t_1[Y] = t_2[Y]$.

Innymi słowy dla każdej podkrotki $t[X]$ znajdziemy wśród krotek t relacji R nie więcej niż jedną odpowiadającą podkrotkę $t[Y]$, która może jednak wystąpić w różnych krotkach tej relacji,

tzn. każda podkrotka $t[X]$ identyfikuje pewną podkrotkę $t[Y]$ w relacji R .

Taką zależność funkcyjną oznaczamy $X \rightarrow Y$.

Ważne pytanie: Czy klucz PK relacji R ma związek z zależnościami funkcyjnymi atrybutów relacji, a jeśli tak, to jaki?

Odpowiedź: Niech klucz $PK(R) = X = \{X_1, X_2, \dots, X_n\}$, natomiast podzbiór $Y = \{Y_1, Y_2, \dots, Y_m\}$ zawiera pozostałe atrybuty relacji R .

Ponieważ klucz identyfikuje każdą krotkę t w relacji R , tzn. $PK(R) \rightarrow Y$, a zatem $PK(R) \rightarrow Y_1, PK(R) \rightarrow Y_2, \dots, PK(R) \rightarrow Y_m$.

Zależność funkcyjna atrybutów relacji bazy danych

Przykłady:

1) $Id_części \rightarrow Nazwa_części$

<i>Id_części</i>	<i>Nazwa_części</i>
53	Pompa
57	Filtr
54	Pompa
32	Koło
88	Silnik
59	Błotnik

Dodatkowa krotka (53,'Filtr') złamałaby zależność funkcyjną $Id_części \rightarrow Nazwa_części$

2) $Id_dostawcy \rightarrow Nazwa_dostawcy$

Określmy klucz relacji DOSTAWCY:

$K(DOSTAWCY) = \{Id_dostawcy\}$

<i>Id_dostawcy</i>	<i>Nazwa_dostawcy</i>	<i>Adres_dostawcy</i>
300	NISSAN	Liege
400	HONDA	Glasgow
500	TOYOTA	Tokyo
600	HONDA	London

Łatwo zauważyć, że nie tylko $K(DOSTAWCY) \rightarrow Nazwa_dostawcy$, ale także $K(DOSTAWCY) \rightarrow Adres_dostawcy$

Pierwsza postać normalna relacji bazy danych (1NF – *First Normal Form*)

Definicja:

Relacja R jest w pierwszej postaci, jeśli relacja spełnia postulat atomowości Codda (każda wartość każdego atrybutu w każdej krotce jest pojedynczą (elementarną, nierozkładalną, atomową) wartością z dziedziny tego atrybutu).

Przykład:

Rozważmy relację ZAMÓWIENIA o nagłówku $N(\text{ZAMÓWIENIA}) = \{\text{Nr_zamówienia}, \text{Id_dostawcy}, \text{Nazwa_dostawcy}, \text{Adres_dostawcy}, \text{Id_części}, \text{Nazwa_części}, \text{Ilość}, \text{Magazny}, \text{Adres_magazynu}\}$, która jest **relacją nieznormalizowaną** (*unnormalized relation*) inaczej **relacją zagnieżdżoną** (*nested relation*) o wielowartościowych elementach krotek, dla kilku następujących atrybutów: $\{\text{Id_części}, \text{Nazwa_części}, \text{Ilość}, \text{Magazny}, \text{Adres_magazynu}\}$.

Pierwsza postać normalna relacji bazy danych (1NF – First Normal Form)

ZAMÓWIENIA (relacja nieznormalizowana - 0NF)

<i>Nr_zamówienia</i>	<i>Id_dostawcy</i>	<i>Nazwa_dostawcy</i>	<i>Adres_dostawcy</i>	<i>Id_części</i>	<i>Nazwa_części</i>	<i>Ilość</i>	<i>Magazyn</i>	<i>Adres_magazynu</i>
001	300	Nissan	Liege	53	Pompa	100	5	Warszawa
				57	Filtr	50	5	Warszawa
				59	Błotnik	500	6	Szczecin
002	400	HONDA	Glasgow	54	Pompa	500	5	Warszawa
				32	Koło	100	6	Szczecin
003	500	TOYOTA	Tokyo	88	Silnik	15	7	Poznań
004	600	HONDA	London	59	Błotnik	400	6	Szczecin
				21	Lampa	50	7	Poznań

ZAMÓWIENIA (relacja w 1NF)

<i>Nr_zamówienia</i>	<i>Id_dostawcy</i>	<i>Nazwa_dostawcy</i>	<i>Adres_dostawcy</i>	<i>Id_części</i>	<i>Nazwa_części</i>	<i>Ilość</i>	<i>Magazyn</i>	<i>Adres_magazynu</i>
001	300	NISSAN	Liege	53	Pompa	100	5	Warszawa
001	300	NISSAN	Liege	57	Filtr	50	5	Warszawa
001	300	NISSAN	Liege	59	Błotnik	500	6	Szczecin
002	400	HONDA	Glasgow	54	Pompa	500	5	Warszawa
002	400	HONDA	Glasgow	32	Koło	100	6	Szczecin
003	500	TOYOTA	Tokyo	88	Silnik	15	7	Poznań
004	600	HONDA	London	59	Błotnik	400	6	Szczecin
004	600	HONDA	London	21	Lampa	50	7	Poznań

Klucz relacji w bazie danych - przypomnienie

Kluczem relacji R o nagłówku $N(R) = \{A_1, A_2, \dots, A_n\}$

jest podzbiór atrybutów $K = \{A_{i_1}, A_{i_2}, \dots, A_{i_k}\}$ tej relacji (tzn. $K \subset N(R)$) o własnościach:

- 1) jeśli x i y są różnymi krotkami relacji R , to $x(K) \neq y(K)$ – innymi słowy nie ma różnych wartości dla tej samej krotki – i zawsze istnieje taki atrybut A_{i_m} w K , że $x(A_{i_m}) \neq y(A_{i_m})$
- 2) nie istnieje podzbiór H zbioru K , różny od K , mający własność (1).

Atrybuty, które spełniają własność (1), a niekoniecznie własność (2), tworzą **superklucz**.

Ważny wniosek: Klucz jest najmniejszym podzbiorem superklucza, jednoznacznie określającym krotki relacji.

Jeżeli więcej niż jeden zbiór atrybutów spełnia warunki klucza, to takie zbiory atrybutów nazywa się **kluczami kandydującymi**.

Ważna uwaga: Klucze wybierane są spośród kluczy kandydujących.

Druga postać normalna relacji bazy danych (2NF – Second Normal Form)

Definicja pełnej zależności funkcyjnej:

Zbiór atrybutów Y jest **w pełni** funkcyjnie zależny od zbioru atrybutów X w relacji R , jeżeli $X \rightarrow Y$ i **nie istnieje** podzbiór $X' \subset X$ taki, że $X' \rightarrow Y$.

Przykład:

$\{X1, X2, Y1\} \subset N(R)$ i $\{X1, X2\} \rightarrow Y1$,

ale nie jest prawdą, że $X1 \rightarrow Y1$ ani nie jest prawdą, że $X2 \rightarrow Y1$.

Definicja częściowej zależności funkcyjnej:

Zbiór atrybutów Y jest **częściowo** funkcyjnie zależny od zbioru atrybutów X w relacji R , jeżeli $X \rightarrow Y$ i **istnieje** podzbiór $X' \subset X$ taki, że $X' \rightarrow Y$.

Przykład:

$\{X1, X2, Y1\} \subset N(R)$ i $\{X1, X2\} \rightarrow Y1$,

inaczej niż poprzednio $X2 \rightarrow Y1$, ale nadal nie jest prawdą, że $X1 \rightarrow Y1$.

Dlatego wystarczy jeden atrybut $X2$ zamiast dwóch $\{X1, X2\}$ do wyznaczenia atrybutu $Y1$.

Druga postać normalna relacji bazy danych (2NF – *Second Normal Form*)

Definicja drugiej postaci normalnej (wersja 1):

Relacja R jest w drugiej postaci normalnej, jeżeli jest w pierwszej postaci normalnej i **żaden atrybut wtórny** (spoza klucza/kluczy tej relacji) **nie jest częściowo funkcyjnie zależny od żadnego z kluczy relacji R.**

Definicja drugiej postaci normalnej (wersja 2):

Relacja R jest w drugiej postaci normalnej, jeżeli jest w pierwszej postaci normalnej i **każdy atrybut wtórny** (spoza klucza/kluczy tej relacji) **jest w pełni funkcyjnie zależny od klucza podstawowego relacji R.**

Druga postać normalna relacji bazy danych (2NF – Second Normal Form)

Przykłady związane z 2NF:

Założmy, że relacje R i S o nagłówkach $N(R) = \{A,B,C\}$ i $N(S) = \{E,F,G\}$ są w 1NF, a ich kluczami (podstawowymi) są odpowiednio $PK(R) = \{A,B\}$ i $PK(S) = \{E,F\}$.

Założmy, że na przedstawionych diagramach zależności funkcyjnych zostały podane wszystkie zależności funkcyjne w relacjach R i S.



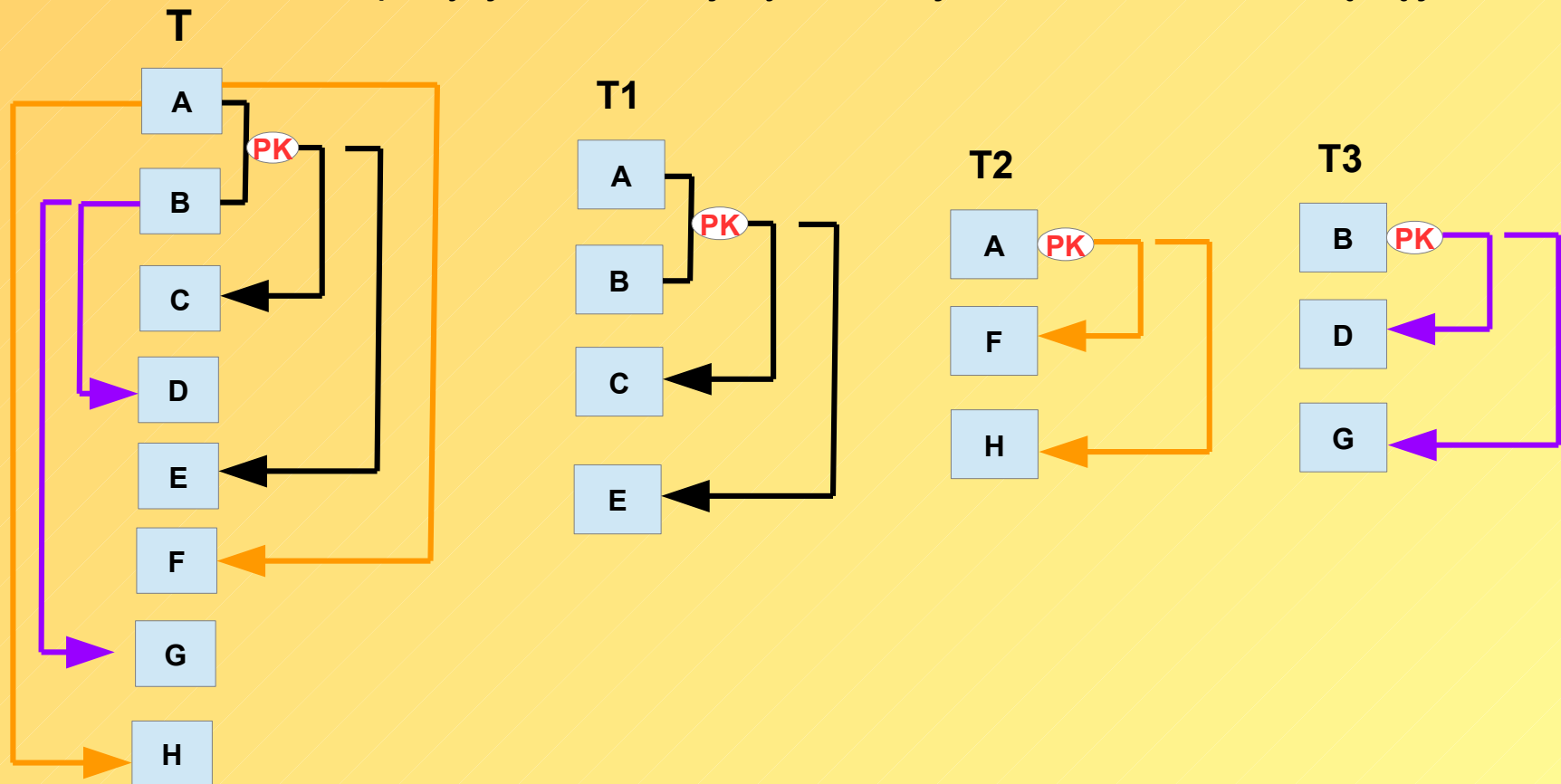
Relacja **R** jest w 2NF, ponieważ jest w 1NF i jej jedyny atrybut wtórny C w pełni zależy funkcyjnie od klucza {A,B} tej relacji.

Relacja **S** nie jest w 2NF, ponieważ jej atrybut wtórny **G** zależy częściowo od klucza {E,F} tej relacji z powodu zależności $E \rightarrow G$.

Druga postać normalna relacji bazy danych (2NF – Second Normal Form)

Algorytm dekompozycji relacji będącej w 1NF do 2NF:

Założmy, że relacja T o nagłówku $N(T) = \{A, B, C, D, E, F, G, H\}$ jest w 1NF, a jej kluczem jest $PK(T) = \{A, B\}$. Założmy, że na przedstawionym diagramie zależności funkcyjnych zostały podane wszystkie istotne zależności funkcyjne w relacji T. Wynika z nich konieczność dekompozycji do 3 mniejszych relacji T1, T2 i T3, które będą już w 2NF.



Trzecia postać normalna relacji bazy danych (3NF – Third Normal Form)

Definicja trzeciej postaci normalnej:

Relacja R jest w trzeciej postaci normalnej, jeżeli jest w drugiej postaci normalnej i **żaden atrybut wtórny** tej relacji (spoza klucza/kluczy relacji R) **nie jest przechodnio zależny funkcyjnie od klucza podstawowego tej relacji**.

Przykład:

$N(R) = \{X1, X2, Z1, Z2, Y1, Y2, Y3\}$,

Niech $KK1 = \{X1\}$, $KK2 = \{X2\}$, $\{Z1, Z2\} \notin KK1(R)$, $\{Z1, Z2\} \notin KK2(R)$.

Stwierdzamy, że $\{X1, X2\} \rightarrow \{Y1, Y2\}$. Następnie znajdujemy, że $\{X1, X2\} \rightarrow \{Z1, Z2\}$ i $\{Z1, Z2\} \rightarrow \{Y1, Y2\}$. Stąd **$\{Y1, Y2\}$ zależy przechodnio funkcyjnie od $\{X1, X2\}$** , a także od $\{X1\}$ i $\{X2\}$ jako kluczy kandydujących.

Dodatkowo stwierdzamy, że $\{X1, X2\} \rightarrow Y3$. Wiemy, że $\{X1, X2\} \rightarrow \{Z1, Z2\}$, ale znajdujemy, że $Y3$ nie zależy funkcyjnie od $\{Z1, Z2\}$.

Dlatego **$Y3$ nie zależy przechodnio funkcyjnie od $\{X1, X2\}$** , a także od $\{X1\}$ i $\{X2\}$ jako kluczy kandydujących.

Trzecia postać normalna relacji bazy danych (3NF – Third Normal Form)

Przykłady związane z 3NF:

Założmy, że relacje R i S o nagłówkach $N(R) = \{A, B, C, D\}$ i $N(S) = \{E, F, G, H\}$ są w 2NF, a ich kluczami (podstawowymi) są odpowiednio $PK(R) = \{A\}$ i $PK(S) = \{E\}$.

Założmy, że na przedstawionych diagramach zależności funkcyjnych zostały podane wszystkie zależności funkcyjne w relacjach R i S.



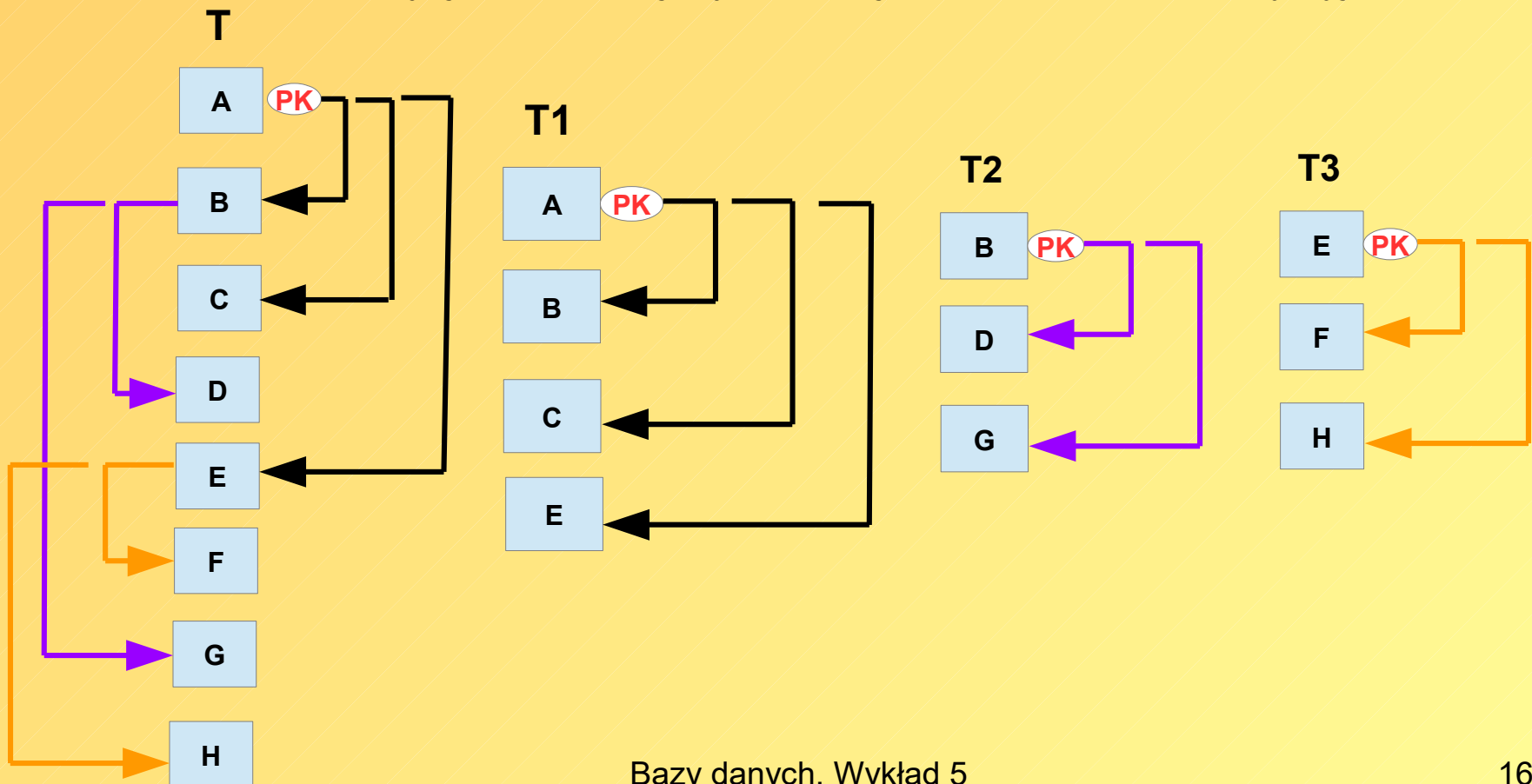
Relacja **R** jest w 3NF, ponieważ jest w 2FN i żaden jej atrybut wtórny B, C, D nie jest przechodnio zależny funkcyjnie od klucza {A} tej relacji.

Relacja **S** nie jest w 3NF, ponieważ jej atrybut wtórny **G** zależy przechodnio funkcyjnie od klucza {E} tej relacji z powodu przechodniej zależności funkcyjnej $E \rightarrow G$, której przechodniość wynika z występowania zależności funkcyjnych $E \rightarrow F$ i $F \rightarrow G$.

Trzecia postać normalna relacji bazy danych (3NF – Third Normal Form)

Algorytm dekompozycji relacji będącej w 2NF do 3NF:

Założmy, że relacja T o nagłówku $N(T) = \{A, B, C, D, E, F, G, H\}$ jest w 2NF, a jej kluczem jest $PK(T) = \{A\}$. Założmy, że na przedstawionym diagramie zależności funkcyjnych zostały podane wszystkie istotne zależności funkcyjne w relacji T. Wynika z nich konieczność dekompozycji do 3 mniejszych relacji T1, T2 i T3, które będą już w 3NF.



Postać normalna Boyce'a-Codda relacji bazy danych (*BCNF – Boyce-Codd Normal Form*)

Definicja trywialnej zależności funkcyjnej:

Zależność funkcyjna jest trywialna wtedy i tylko wtedy, gdy **prawa strona zależności jest podzbiorem lewej** ($\{X, Y\} \rightarrow Y$).

Definicja nietrywialnej zależności funkcyjnej:

Zależność funkcyjna jest nietrywialna wtedy i tylko wtedy, gdy **prawa strona zależności nie jest podzbiorem lewej** ($X \rightarrow Y$ i $Y \not\subset X$).

Definicja postaci normalnej Boyce'a-Codda (wersja uproszczona):

Relacja R będąca w pierwszej postaci normalnej jest w postaci normalnej Boyce'a-Codda wtedy i tylko wtedy, gdy elementem determinującym każdej nietrywialnej, lewostronnie nieredukowalnej zależności funkcyjnej tej relacji jest klucz kandydujący (potencjalny).

Symboliczna interpretacja zależności dozwolonych w relacji R przez tę definicję BCNF:

$$X \rightarrow Y \wedge Y \not\subset X \wedge \sim \left(\bigvee_{X' \subset X \wedge X' \neq X} X' \rightarrow Y \right) \wedge X = \text{KK}(R)$$

Postać normalna Boyce'a-Codda relacji bazy danych (*BCNF – Boyce-Codd Normal Form*)

Definicja postaci normalnej Boyce'a-Codda (wersja ogólna):

Relacja R będąca w pierwszej postaci normalnej jest w postaci normalnej Boyce'a-Codda wtedy i tylko wtedy, gdy wszystkie zależności $X \rightarrow Y$ w tej relacji spełniają przynajmniej jeden z poniższych warunków:

- $X \rightarrow Y$ jest trywialną ($Y \subset X$) zależnością funkcyjną ,
- dla każdej nietrywialnej ($Y \not\subset X$) zależności $X \rightarrow Y$ wyznacznik X tej zależności jest nadkluczem

Symboliczna interpretacja nietrywialnych zależności dozwolonych przez BCNF w relacji R:

$$X \rightarrow Y \quad \wedge \quad Y \not\subset X \quad \wedge \quad X = SK(R)$$

Inna definicja trzeciej postaci normalnej, ułatwiająca rozróżnienie BCNF od 3NF:

Relacja jest w 3NF, jeśli jest w 1NF i dla każdej nietrywialnej zależności $X \rightarrow Y$:

- wyznacznik X tej zależności jest nadkluczem (warunek spełniony także w BCNF!)
lub
- Y zawiera atrybuty tylko atrybuty z kluczy (kluczy kandydujących może być kilka!)

Postać normalna Boyce'a-Codda relacji bazy danych (*BCNF – Boyce-Codd Normal Form*)

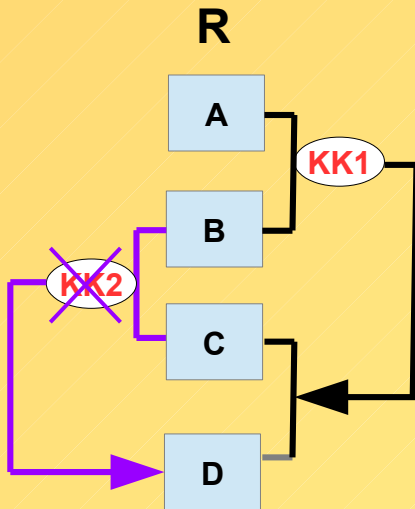
Ważna wskazówka, dotycząca 3NF i postaci normalnej Boyce'a-Codda:

Jeżeli relacja R nie ma dwóch lub więcej nierozłącznych (zwanych też przekrywającymi się) kluczy kandydujących i jest w 3NF, to również jest w postaci normalnej Boyce'a-Codda.

Przykład relacji w 3NF, która nie jest w BCNF:

Zakładamy, że na poniższym diagramie są wszystkie istotne zależności funkcyjne w relacji R, która jest w 2NF.

Na diagramie nie ma przechodnich zależności funkcyjnych atrybutów wtórnych (niekluczowych) od klucza, więc R jest w 3NF.



$\{A,B\} \rightarrow \{C,D\}$, zatem $KK1(R) = \{A,B\}$
jest kluczem kandydującym relacji R.

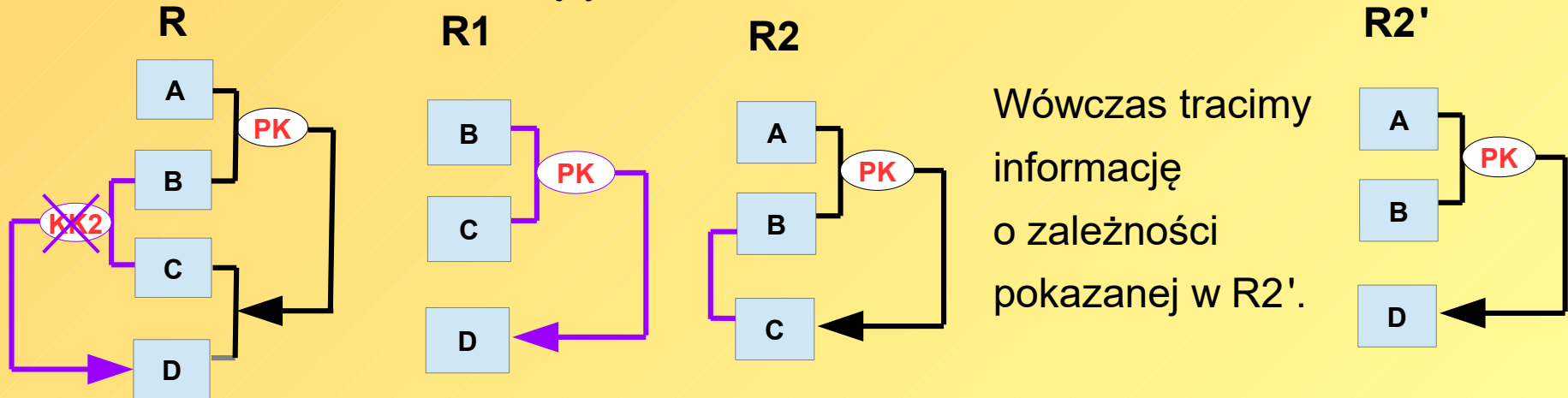
$\{B,C\} \rightarrow \{D\}$,
ale nie zachodzi zależność $\{B,C\} \rightarrow \{A\}$,
więc $\{B,C\}$ nie jest kluczem kandydującym
relacji R. Stąd relacja R nie jest w BCNF.

Postać normalna Boyce'a-Codda relacji bazy danych (*BCNF – Boyce-Codd Normal Form*)

Przykład dekompozycji relacji będącej w 3NF do postaci normalnej Boyce'a-Codda:

Rozpatrzmy relację R z poprzedniego slajdu i wykorzystajmy dwustopniową regułę dekompozycji:

- Znajdźmy nietrywialne zależności funkcyjne $X \rightarrow Y$, które łamią BCNF
- Podzielmy relację w następujący sposób:
 - (i) zgromadźmy w pojedynczej relacji wszystkie atrybuty z łamiących BCNF zależności $X \rightarrow Y$ o wspólnym wyznaczniku X,
 - (ii) wówczas w innej relacji zgromadźmy atrybuty z wyznacznika (wyznaczników) X zależności łamiących BCNF razem z pozostałymi atrybutami tej relacji, które nie biorą udział w zależnościach łamiących BCNF.



Dekompozycja do BCNF nie gwarantuje zachowania zależności funkcyjnych!

Reguły wnioskowania Armstronga dla zależności funkcyjnych

- Zwrotność

$$Y \subset X \implies X \rightarrow Y$$

(często zapisuje się dokładniej: $Y \subseteq X \implies X \rightarrow Y$)

- Rozszerzanie

$$X \rightarrow Y \implies \{X,Z\} \rightarrow \{Y,Z\}$$

- Przechodniość

$$X \rightarrow Z \wedge Z \rightarrow Y \implies X \rightarrow Y$$

Czwarta postać normalna relacji bazy danych (4NF – *Fourth Normal Form*)

Definicja (funkcyjnej) zależności wielowartościowej:

Zbiór atrybutów Y jest **wielowartościowo** zależny od zbioru atrybutów X w relacji R, jeżeli dla krotek t o zgodnych podkrotkach t[X] można zmieniać podkrotkę t[Y], otrzymując krotki relacji R.

Zależność wielowartościową Y od X często oznacza się jako $X \twoheadrightarrow Y$.

Przykład:

JĘZYKI_PROGRAMOWANIA_PRACOWNIKOW

Id_pracownika	Język_programowania
1000	Python
1000	Java
1000	C++
1020	Python
1020	Java

W relacji JĘZYKI_PROGRAMOWANIA_PRACOWNIKÓW zachodzi (funkcyjna) zależność wielowartościowa $\text{Id_pracownika} \twoheadrightarrow \text{Język_programowania}$, ale nie zachodzi zależność funkcyjna $\text{Id_pracownika} \rightarrow \text{Język_programowania}$.

Czwarta postać normalna relacji bazy danych (4NF – *Fourth Normal Form*)

- Zauważmy, że pusty podzbiór atrybutów $\emptyset \subset N(R)$ jest zawsze wielowartościowo zależny od dowolnego podzbioru atrybutów $X \subset N(R)$.
- Również zawsze $X \twoheadrightarrow Y$ dla dowolnych rozłącznych ($X \cap Y = \emptyset$) podzbiorów atrybutów $X \subset N(R)$ i $Y \subset N(R)$, takich, że $X \cup Y = N(R)$.

W powyższych przypadkach mówimy o **trywialnych zależnościach wielowartościowych**.

Definicja nietrywialnej zależności wielowartościowej:

Zależność wielowartościowa $X \twoheadrightarrow Y$ w relacji R jest **nietrywialna**, gdy

- $Y \not\subseteq X$ (a dokładniej $Y \not\subseteq X$)
oraz
- $X \cup Y$ nie są wszystkimi atrybutami relacji R

Definicja czwartej postaci normalnej:

Relacja R jest w czwartej postaci normalnej, jeśli (jest w BCNF i) dla każdej nietrywialnej zależności wielowartościowej $X \twoheadrightarrow Y$ w relacji R wyznacznik X zależności wielowartościowej jest nadkluczem relacji R.

Czwarta postać normalna relacji bazy danych (4NF – *Fourth Normal Form*)

Ważny wniosek z definicji czwartej postaci normalnej:

Trywialne zależności wielowartościowe nie łamią 4NF.

Przykład:

Rozważmy relację R typu „all-key” o następującej strukturze

$R \subset D(\text{Nazwisko_pracownika}) \times D(\text{Język_programowania}) \times D(\text{Język_obcy})$,

gdzie dziedziny jej atrybutów Nazwisko_pracownika, Język_programowania, Język_obcy są następujące:

$D(\text{Nazwisko_pracownika}) = \{\text{'Kowalski'}, \text{'Malinowski'}, \text{'Nowak'}\}$

$D(\text{Język_programowania}) = \{\text{'C'}, \text{'C++'}, \text{'Java'}, \text{'Smalltalk'}, \text{'SQLPL/M'}\}$

$D(\text{Język_obcy}) = \{\text{'angielski'}, \text{'francuski'}, \text{'włoski'}, \text{'hiszpański'}, \text{'rosyjski'}\}$

W takiej relacji jedynie wszystkie atrybuty mogą stanowić klucz

$PK(R) = \{\text{Nazwisko_pracownika}, \text{Język_programowania}, \text{Język_obcy}\}$,

dlatego zaliczamy ją do relacji typu „all-key”.

Czwarta postać normalna relacji bazy danych (4NF – Fourth Normal Form)

R

<i>Nazwisko_pracownika</i>	<i>Język_programowania</i>	<i>Język_obcy</i>
Kowalski	C	angielski
Kowalski	C	francuski
Kowalski	C++	angielski
Kowalski	C++	francuski
Kowalski	Java	angielski
Kowalski	Java	francuski
Malinowski	Smalltalk	angielski
Malinowski	Smalltalk	włoski
Malinowski	Smalltalk	hiszpański
Malinowski	Java	angielski
Malinowski	Java	włoski
Malinowski	Java	hiszpański
Nowak	C	angielski
Nowak	C	rosyjski
Nowak	SQL PL/M	angielski
Nowak	SQL PL/M	rosyjski

W relacji R zauważamy dwie zależności wielowartościowe:

- (1) Nazwisko_pracownika
→→ Język_programowania,
- (2) Nazwisko_pracownika
→→ Język_obcy,

z których żadna nie jest trywialna.

Wyznacznik obydwu tych zależności wielowartościowych, którym jest **Nazwisko_pracownika** **nie jest nadkluczem relacji R**, ponieważ PK(R) jest złożony ze wszystkich atrybutów tej relacji. **Zatem R nie jest w 4NF.**

Czwarta postać normalna relacji bazy danych (4NF – *Fourth Normal Form*)

Dekompozycja zależności wielowartościowych relacji R do różnych relacji, powoduje, że w relacjach powstałych po dekompozycji mamy pojedyncze zależności wielowartościowe:

(1) Nazwisko_pracownika $\rightarrow\rightarrow$ Język_programowania,

(2) Nazwisko_pracownika $\rightarrow\rightarrow$ Język_obcy,

które są trywialne w nowych relacjach, więc obydwie relacje powstałe po takiej dekompozycji relacji R są w 4NF.

<i>Nazwisko_pracownika</i>	<i>Język_programowania</i>
Kowalski	C
Kowalski	C++
Kowalski	Java
Malinowski	Smalltalk
Malinowski	Java
Nowak	C
Nowak	SQL PL/M

<i>Nazwisko_pracownika</i>	<i>Język_obcy</i>
Kowalski	angielski
Kowalski	francuski
Malinowski	angielski
Malinowski	włoski
Malinowski	hiszpański
Nowak	angielski
Nowak	rosyjski

Reguły wnioskowania dla zależności wielowartościowych

- Promocja

Każda zależność funkcyjna jest zależnością wielowartościową:

$$X \rightarrow Y \implies X \twoheadrightarrow Y$$

Odwrotna implikacja nie zachodzi!

Z reguły promocji oraz definicji BCNF i 4NF wynika, że 4NF jest w BCNF, ale nie na odwrót!

- Uzupełnianie

W danej relacji R o nagłówku $N(R)=\{X,Y,Z\}$:

$$X \twoheadrightarrow Y \implies X \twoheadrightarrow Z$$

- Przechodniość

W danej relacji R o nagłówku $N(R)=\{X,Y,Z\}$:

$$X \twoheadrightarrow Z \wedge Z \twoheadrightarrow Y \implies X \twoheadrightarrow Y$$

Zagadnienie normalizacji do 3NF/BCNF - przykład

ZAMÓWIENIA

<i>Nr_zamowienia</i>	<i>Id_dostawcy</i>	<i>Nazwa_dostawcy</i>	<i>Adres_dostawcy</i>	<i>Id_czesci</i>	<i>Nazwa_czesci</i>	<i>Ilosc</i>	<i>Magazyn</i>	<i>Adres_magazynu</i>
001	300	NISSAN	Liege	53	Pompa	100	5	Warszawa
001	300	NISSAN	Liege	57	Filtr	50	5	Warszawa
001	300	NISSAN	Liege	59	Błotnik	500	6	Szczecin
002	400	HONDA	Glasgow	54	Pompa	500	5	Warszawa
002	400	HONDA	Glasgow	32	Koło	100	6	Szczecin
003	500	TOYOTA	Tokyo	88	Silnik	15	7	Poznań
004	600	HONDA	London	59	Błotnik	400	6	Szczecin
004	600	HONDA	London	21	Lampa	50	7	Poznań

Zawsze należy
scharakteryzować dziedzinę
rzeczywistości, która została
zamodelowana w relacji,
dla której mamy przeprowadzić
analizę postaci normalnych!
W tym celu najlepiej
sformułować założenia!

Relacja ZAMÓWIENIA zawiera informacje o zamówieniach części u dostawców i skierowaniu ich do odpowiednich magazynów. Przyjmujemy dla tej dziedziny rzeczywistości następujące założenia:

- (1) Jedno zamówienie może dotyczyć wielu części dostarczanych przez tego samego dostawcę.
- (2) Ta sama część może być dostarczana przez różnych dostawców.
- (3) Te same części są składowane w tym samym magazynie.
- (4) Różni dostawcy mogą nosić tę samą nazwę
(dlatego określa się identyfikator dostawcy i nazwę dostawcy).
- (5) Różne części mogą nosić tę samą nazwę
(dlatego określa się identyfikator części i nazwę części).