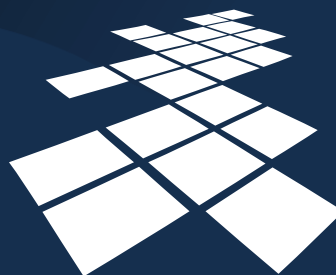


Normalizacja schematów logicznych relacji

Wykład przygotował:
Tadeusz Morzy



UCZELNIA
ONLINE

BD – wykład 5

Celem niniejszego wykładu jest przedstawienie i omówienie procesu normalizacji. Proces normalizacji traktujemy jako proces, podczas którego schematy relacji posiadające niepożądane cechy są dekomponowane na mniejsze schematy relacji o pożądanych własnościach.

Wykład rozpoczniemy od krótkiego przykładu motywacyjnego, ilustrującego problem. Następnie, wprowadzimy pojęcie zależności funkcyjnych stanowiących punkt wyjścia procesu normalizacji. Następnie przejdziemy do omówienia kolejnych postaci normalnych.



- Dana jest następująca relacja Dostawcy:

Nazwisko	Adres	Produkt	Cena
Kowalski	ul. Krucza 10	chipsy	1,50
Kowalski	ul. Krucza 10	orzeszki	3,50
...
Kowalski	ul. Krucza 10	gruszki	4,50
Nowak	ul. Malwowa 4	chipsy	2,00
Nowak	ul. Malwowa 4	orzeszki	4,00
...

Rozważmy następujący przykład. Dana jest następująca relacja Dostawcy, jak na slajdzie, składająca się z 4 atrybutów. Załóżmy, że atrybut nazwisko jest unikalny, to jest nie ma dwóch dostawców o tym samym nazwisku. Relacja Dostawcy zawiera informacje o dostawcach (o ich adresach), dostarczanych produktach i cenach dostarczanych produktów.



Motywacja (2)

- Załóżmy, że trybut **Nazwisko** jest unikalny, tj. nie ma dwóch dostawców o tym samym nazwisku.
- Cechy relacji **Dostawca**:
 - redundancja danych - problem spójności danych
 - anomalia wprowadzania danych
 - anomalia usuwania danych
 - anomalia uaktualniania danych
- Rozwiązaniem: dekompozycja relacji Dostawca na dwie relacje: Dostawca i Dostawy

Analizując relację Dostawcy zauważmy, że relacja ta charakteryzuje się następującymi cechami. Po pierwsze, obserwujemy redundancję danych – adres dostawcy jest pamiętany tyle razy ile różnych produktów dany dostawca dostarcza. Problem redundancji danych nie sprowadza się do problemu zajętości pamięci, aktualnie pamięci są bardzo tanie, lecz problemu potencjalnej niespójności danych. W momencie zmiany adresu dostawcy, zmiana ta musi być odzwierciedlona we wszystkich krotkach zawierających adres dostawcy. W przeciwnym razie pojawi się problem spójności danych. Po drugie, obserwujemy tzw. anomalię wprowadzania danych. Załóżmy, że chcemy wprowadzić informację o nowym dostawcy, tj. jego nazwisko i adres. Niestety, informacji tej nie można wprowadzić do relacji Dostawca tak długo, jak długo dostawca nie dostarcza żadnych produktów. Po trzecie, obserwujemy anomalię usuwania danych. Załóżmy, że rezygnujemy z usług dostawcy Nowak. Usuwając informację o dostawach Nowaka mimo woli usuwamy informacje o samym dostawcy Nowak. Wreszcie, obserwujemy anomalię uaktualniania danych. Aktualizując adres dostawcy, jak już wspominaliśmy, aktualizację tę musimy wprowadzić do wszystkich krotek zawierających adres dostawcy.

Reasumując, schemat relacji Dostawca posiada szereg niepożądanych własności, które w późniejszym czasie będą utrudniały przygotowanie aplikacji operującej na tej relacji. Zauważmy, że rozwiązaniem wszystkich omówionych problemów jest dekompozycja relacji Dostawca na dwie relacje: Dostawca i Dostawy.



Motywacja (3)

Dostawca

Nazwisko	Adres
Kowalski	ul. Krucza 10
...	...
...	...
...	...
Nowak	ul. Malwowa 4
...	...

Dostawy

Nazwisko	Produkt	Cena
Kowalski	chipsy	1,50
Kowalski	orzeszki	3,50
Kowalski	gruszki	4,50
Nowak	chipsy	2,00
Nowak	orzeszki	4,00
...

Dekompozycja bez utraty informacji

BD – wykład 5 (4)

Relacja Dostawca zawiera informacje o dostawcach, natomiast relacja Dostawy zawiera informacje o dostarczanych produktach i ich cenach. Zauważmy, że w przypadku relacji Dostawca adres dostawcy jest pamiętany tylko w jednej krotce – brak redundancji danych. Zauważmy również, że dekompozycja rozwiązuje problem anomalii wstawiania – informacje o nowym dostawcy możemy wstawić do relacji Dostawca, nawet jeżeli dostawca ten nie dostarcza żadnych produktów. Dekompozycja ta rozwiązuje również problem anomalii usuwania – usunięcie informacji o dostawach z relacji Dostawy nie pociąga za sobą usunięcia informacji o samych dostawcach. Dekompozycja rozwiązuje również problem anomalii aktualizacji – zmiana adresu dostawcy dotyczy wyłącznie jednej krotki. Zauważmy, że dekompozycja relacji Dostawca na relacje Dostawca i Dostawy jest dekompozycją bez utraty informacji w tym sensie, że łącząc relację Dostawca i Dostawy wg. atrybutu połączeniowego Nazwisko możemy odtworzyć oryginalną zawartość relacji Dostawca.



Zależności funkcyjne (1)

- **Zależność funkcyjna (FD)**

Dana jest relacja r o schemacie R . X, Y są podzbiorami atrybutów R . W schemacie relacji R , X wyznacza funkcyjnie Y , lub Y jest funkcyjnie zależny od X , co zapisujemy $X \rightarrow Y$, wtedy i tylko wtedy, jeżeli dla dwóch dowolnych krotek t_1, t_2 takich, że $t_1[X] = t_2[X]$ zachodzi zawsze $t_1[Y] = t_2[Y]$, gdzie $t_i[A]$ oznacza wartość atrybutu A krotki t_i

- Przykłady:

- 1. Nazwisko \rightarrow Adres
- 2. {Nazwisko, Towar} \rightarrow Cena

Jak już wspomnieliśmy we wstępie, punktem wyjścia procesu normalizacji jest informacja o zależnościach funkcyjnych występujących w relacjach. Zależność funkcyjną definiujemy następująco:

Dana jest relacja r o schemacie R . X, Y są podzbiorami atrybutów R . W schemacie relacji R , X wyznacza funkcyjnie Y , lub Y jest funkcyjnie zależny od X , co zapisujemy $X \rightarrow Y$, wtedy i tylko wtedy, jeżeli dla dwóch dowolnych krotek t_1, t_2 takich, że $t_1[X] = t_2[X]$ zachodzi zawsze $t_1[Y] = t_2[Y]$, gdzie $t_i[A]$ oznacza wartość atrybutu A krotki t_i .

Przykładowo, relacja Dostawca zawiera dwie zależności funkcyjne: Nazwisko \rightarrow Adres i {Nazwisko, Towar} \rightarrow Cena.

Z pierwszej zależności funkcyjnej wynika, że adres dostawcy jednoznacznie zależy od nazwiska dostawcy. Natomiast z drugiej zależności funkcyjnej wynika, że cena towaru zależy od kombinacji atrybutów Nazwisko i Towar.



Zależności funkcyjne (2)

- Zależność funkcyjna określa zależność pomiędzy atrybutami. Jest to własność semantyczna, która musi być spełniona dla dowolnych wartości krotek relacji.
- Relacje które spełniają nałożone zależności funkcyjne nazywamy instancjami legalnymi
- Zależność funkcyjna jest własnością schematu relacji R , a nie konkretnego wystąpienia relacji
- Z zależności funkcyjnej wynika, że jeżeli $t_1[X] = t_2[X]$ i $X \rightarrow Y$, to zachodzi zawsze $t_1[Y] = t_2[Y]$

Należy podkreślić, że zależność funkcyjna określa zależność pomiędzy atrybutami. Jest to własność semantyczna, która musi być spełniona dla dowolnych wartości krotek relacji.

Relacje które spełniają nałożone zależności funkcyjne nazywamy instancjami legalnymi. Zależność funkcyjna jest własnością schematu relacji R , a nie konkretnego wystąpienia relacji. Jeżeli zmieni się relacja, to zależność funkcyjna nadal pozostaje ważna. Zauważmy również, że z zależności funkcyjnej wynika, że jeżeli $t_1[X] = t_2[X]$ i $X \rightarrow Y$, to zachodzi zawsze $t_1[Y] = t_2[Y]$.



- Proces normalizacji relacji można traktować jako proces, podczas którego schematy relacji posiadające pewne niepożądane cechy są dekomponowane na mniejsze schematy relacji o pożądanych własnościach
- Proces normalizacji musi posiadać trzy dodatkowe własności:

Własność zachowania atrybutów - żaden atrybut nie zostanie zagubiony w trakcie procesu normalizacji

Własność zachowania informacji - dekompozycja relacji nie prowadzi do utraty informacji

Własność zachowania zależności - wszystkie zależności funkcyjne są reprezentowane w pojedynczych schematach relacji

Przejdziemy teraz do przedstawienia procesu normalizacji.

Proces normalizacji relacji można traktować jako proces, podczas którego schematy relacji posiadające pewne niepożądane cechy są dekomponowane na mniejsze schematy relacji o pożądanych własnościach. Proces normalizacji musi posiadać trzy dodatkowe własności:

Własność zachowania atrybutów - żaden atrybut nie zostanie zagubiony w trakcie procesu normalizacji.

Własność zachowania informacji - dekompozycja relacji nie prowadzi do utraty informacji, tj. łącząc zdekomponowane relacje możemy odtworzyć oryginalną relację.

Własność zachowania zależności - wszystkie zależności funkcyjne są reprezentowane w pojedynczych schematach relacji.

Proces normalizacji schematu relacji polega na sprawdzeniu czy dany schemat jest w odpowiedniej postaci normalnej, jeżeli nie wówczas następuje dekompozycja schematu relacji na mniejsze schematy relacji. Ponownie, weryfikowana jest postać normalna otrzymanych schematów relacji. Jeżeli nie spełniają one zadanej postaci normalnej to proces dekompozycji jest kontynuowany dopóki otrzymane schematy relacji nie będą w odpowiedniej postaci normalnej.

Zanim przejdziemy do przedstawienia postaci normalnych przypomnimy podstawowe pojęcia dotyczące schematu relacji.



Pojęcia podstawowe (1)

- **Nadkluczem (superkluczem)** schematu relacji $R = \{A_1, A_2, \dots, A_n\}$ nazywamy zbiór atrybutów $S \subseteq R$, który jednoznacznie identyfikuje wszystkie krotki relacji r o schemacie R . Innymi słowy, w żadnej relacji r o schemacie R nie istnieją dwie krotki t_1, t_2 takie, że $t_1[S] = t_2[S]$
- **Kluczem K** schematu relacji R nazywamy minimalny nadklucz, to znaczy taki nadklucz, że nie istnieje $K' \subset K$ będący nadkluczem schematu R
- Kluczem schematu Dostawca:
{Nazwisko, Produkt, Cena}

Nadkluczem (superkluczem) schematu relacji $R = \{A_1, A_2, \dots, A_n\}$ nazywamy zbiór atrybutów S będący podzbiorem zbioru R , który jednoznacznie identyfikuje wszystkie krotki relacji r o schemacie R . Innymi słowy, w żadnej relacji r o schemacie R nie istnieją dwie krotki t_1, t_2 takie, że $t_1[S] = t_2[S]$. **Kluczem K** schematu relacji R nazywamy minimalny nadklucz, to znaczy taki nadklucz, że nie istnieje żaden podzbiór zbioru K będący nadkluczem schematu R . Łatwo zauważyć, że kluczem przykładowego schematu Dostawca jest zbiór atrybutów {Nazwisko, Produkt, Cena}.



Pojęcia podstawowe (2)

- **Klucze potencjalne** (ang. *candidate keys*)

Klucz podstawowy (ang. *primary key*)

Klucze drugorzędne (ang. *secondary keys*)

- **Atrybuty:**

- atrybuty podstawowe: atrybut X jest podstawowy w schemacie R jeżeli należy do któregoś z kluczy schematu R
- atrybuty wtórne: atrybut X jest wtórny w schemacie R jeżeli nie należy do żadnego z kluczy schematu R

Schemat relacji może posiadać wiele kluczy, które nazywamy kluczami potencjalnymi. Spośród kluczy potencjalnych wybieramy jeden klucz, tzw. klucz podstawowy. Schemat relacji może posiadać tylko jeden klucz podstawowy, definiowany za pomocą klauzuli PRIMARY KEY. System zarządzania bazą danych automatycznie weryfikuje unikalność klucza podstawowego.

Pozostałe klucze potencjalne schematu relacji, nazywane kluczami drugorzędnymi, definiujemy za pomocą klauzuli UNIQUE.

Wprowadzimy następującą klasyfikację atrybutów. Atrybuty dzielimy na atrybuty podstawowe i atrybuty wtórne. Atrybut X nazywamy atrybutem podstawowym w schemacie R jeżeli należy do któregoś z kluczy schematu R. Atrybut X nazywamy atrybutem wtórnym w schemacie R jeżeli nie należy do żadnego z kluczy schematu R. Obecnie przejdziemy do przedstawienia kolejnych postaci normalnych.



Pierwsza postać normalna 1NF (1)

- Definicja:

Schemat relacji R znajduje się w pierwszej postaci normalnej (1NF), jeżeli wartości atrybutów są atomowe (niepodzielne)

- Tablica Płeć:

Płeć	Imię
Męska	Jan, Piotr, Zenon
Żeńska	Anna, Eliza, Maria

- Relacja Płeć w 1NF

Płeć	Imię
Męska	Jan
Męska	Piotr
Męska	Zenon
Żeńska	Anna
Żeńska	Eliza
Żeńska	Maria

Mówimy, że schemat relacji R znajduje się w pierwszej postaci normalnej (1NF), jeżeli wartości atrybutów są atomowe (niepodzielne).

Rozważmy tabelę Płeć przedstawioną na slajdzie. Zauważmy, że atrybut Imię jest atrybutem typu zbiorowego. Normalizacja tabeli Płeć do 1NF polega na utworzeniu dla każdej atomowej wartości atrybutu Imię osobnej krotki. W wyniku uzyskujemy tabelę Płeć w 1NF, jak przedstawiono na slajdzie.



Pierwsza postać normalna 1 NF (2)

- Pierwsza postać normalna zabrania definiowania złożonych atrybutów, które są wielowartościowe
- Relacje, które dopuszczają definiowanie złożonych atrybutów nazywamy **relacjami zagnieżdżonymi** (ang. *nested relations*)
- W relacjach zagnieżdżonych każda krotka może zawierać inną relację
- Pracownicy (idPrac, Nazwisko, {Projekty (nr, godziny)})

Pierwsza postać normalna zabrania definiowania złożonych atrybutów, które są wielowartościowe. Relacje, które dopuszczają definiowanie złożonych atrybutów nazywamy **relacjami zagnieżdżonymi** (ang. *nested relations*). W relacjach zagnieżdżonych każda krotka może zawierać inną relację. Rozważmy przykład relacji Pracownicy przedstawionej na kolejnym slajdzie.



Pierwsza postać normalna 1NF (3)

Pracownicy

IdPrac	Nazwisko	Projekty	
		nr	godziny
1234567	Kowalski	1	32,5
		2	7,5
6655443	Nowak	3	40,5
4343435	Kruczek	1	20
		2	20
3333333	Morzy	1	10
		2	10
		3	10
		4	10

Relacja
zagnieżdżona

Relacja zewnętrzna

BD – wykład 5 (12)

Zauważmy, że relacja Pracownicy zawiera zagnieżdżoną w niej relację Projekty składającą się z atrybutów: Nr i Godziny. Trywialna normalizacja relacji Pracownicy do 1NF polegałaby na utworzeniu dla każdej krotki relacji zagnieżdżonej osobnej krotki relacji znormalizowanej. W wyniku uzyskalibyśmy przykładowo, 2 krotki postaci <1234567; Kowalski; 1; 32,5> i <1234567; Kowalski; 2; 7,5>.

Zasadniczą wadą tego sposobu normalizacji relacji Pracownicy jest duża redundancja danych, tzn. informacje dot. identyfikatora pracownika i jego nazwiska będą występowały wielokrotnie w kolejnych krotkach znormalizowanej relacji. Zalecany sposób normalizacji schematów relacji nie będących w 1NF opiera się na rozróżnieniu relacji zagnieżdżonej i relacji zewnętrznej. Do relacji zewnętrznej należą wszystkie atrybuty, które nie wchodzi w skład relacji zagnieżdżonej.

Przedstawiony slajd ilustruje podział relacji pracownicy na relację zewnętrzną i relację zagnieżdżoną.



Pierwsza postać normalna 1NF (4)

- Dana jest relacja R, zawierająca inną relację P
- Dekompozycja relacji R do zbioru relacji w 1NF:
 - Utwórz osobną relację dla relacji zewnętrznej
 - Utwórz osobną relację dla relacji wewnętrznej (zagnieżdżonej), do której dodaj klucz relacji zewnętrznej
 - Kluczem nowej relacji wewnętrznej (klucz relacji wewnętrznej + klucz relacji zewnętrznej)
- Dekompozycja relacji Pracownicy:
Pracownicy (IdPrac, Nazwisko)
Uczestniczy (IdPrac, Nr, Godziny)

BD – wykład 5 (13)

Zalecany sposób normalizacji schematów relacji do 1NF ma następującą postać.

Dana jest relacja R, zawierająca inną relację zagnieżdżoną P. Dekompozycja relacji R do zbioru relacji w 1NF:

- Utwórz osobną relację dla relacji zewnętrznej
- Utwórz osobną relację dla relacji wewnętrznej (zagnieżdżonej), do której dodaj klucz relacji zewnętrznej
- Kluczem nowej relacji wewnętrznej (klucz relacji wewnętrznej + klucz relacji zewnętrznej)

Przykładowo, dekompozycja relacji Pracownicy do zbioru relacji w 1NF prowadzi do 2 relacji następujących postaci: Pracownicy (IdPrac, Nazwisko) i Uczestniczy (IdPrac, Nr, Godziny).



Druga postać normalna 2NF (1)

- Pełna zależność funkcyjna

Zbiór atrybutów Y jest w pełni funkcyjnie zależny od zbioru atrybutów X w schemacie R , jeżeli $X \rightarrow Y$ i nie istnieje podzbiór

$X' \subset X$ taki, że $X' \rightarrow Y$

Zbiór atrybutów Y jest częściowo funkcyjnie zależny od zbioru atrybutów X w schemacie R , jeżeli $X \rightarrow Y$ i istnieje podzbiór

$X' \subset X$ taki, że $X' \rightarrow Y$

- Druga postać normalna

Dana relacja r o schemacie R jest w drugiej postaci normalnej (2NF), jeżeli żaden atrybut wtórny tej relacji nie jest częściowo funkcyjnie zależny od żadnego z kluczy relacji r

Łatwo zauważyć, że 1NF nie rozwiązuje problemu anomalii wymienionych wcześniej. Przejdziemy zatem do przedstawienia definicji drugiej postaci normalnej (2NF). W tym celu wprowadzimy definicje pełnej i częściowej zależności funkcyjnej.

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

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

Możemy obecnie wprowadzić definicję drugiej postaci normalnej. Mówimy, że dana relacja r o schemacie R jest w drugiej postaci normalnej (2NF), jeżeli żaden atrybut wtórny tej relacji nie jest częściowo funkcyjnie zależny od żadnego z kluczy relacji r .



Druga postać normalna 2NF (2)

- **Uczestnictwo**

IdPrac	NrProj	Funkcja	Nazwisko	NazwaProj	Lokalizacja
--------	--------	---------	----------	-----------	-------------

$fd1: \{IdPrac, NrProj\} \rightarrow Funkcja$

$fd2: \{IdPrac, NrProj\} \rightarrow Nazwisko$

$fd3: \{IdPrac, NrProj\} \rightarrow NazwaProj$

$fd4: \{IdPrac, NrProj\} \rightarrow Lokalizacja$

$fd5: \{IdPrac\} \rightarrow Nazwisko$

$fd6: \{NrProj\} \rightarrow NazwaProj$

$fd7: \{NrProj\} \rightarrow Lokalizacja$

Zależności $fd2, fd3, fd4$ są zależnościami niepełnymi

Rozważmy następujący przykład ilustrujący definicję drugiej postaci normalnej. Dana jest relacja Uczestnictwo składająca się z atrybutów: IdPrac, NrProj, Funkcja, Nazwisko, NazwaProj, Lokalizacja. Relacja Uczestnictwo opisuje udział pracowników o identyfikatorze (IdPrac) w realizacji projektów o numerze NrProj. Kluczem schematu relacji Uczestnictwo jest para atrybutów IdPrac i NrProj. W schemacie relacji Uczestnictwo występuje 7 zależności funkcyjnych $fd1, \dots, fd7$, z których 4 pierwsze są zależnościami od klucza. Zależność funkcyjna atrybutu od klucza oznacza, że każdy atrybut jest funkcyjnie zależny od klucza schematu relacji. Zauważmy, że zależności $fd2, fd3, fd4$ są zależnościami niepełnymi. Przykładowo, zależność funkcyjna $fd2: \{IdPrac, NrProj\} \rightarrow Nazwisko$ jest częściową zależnością funkcyjną gdyż istnieje podzbiór lewej strony zależności funkcyjnej (IdPrac), który wyznacza funkcyjnie prawą stronę zależności. Podobnie jest w przypadku zależności $fd3$ i $fd4$. Łatwo zauważyć, że schemat relacji uczestnictwo nie jest w 2NF, gdyż istnieją atrybuty wtórne (Nazwisko, NazwaProj, Lokalizacja), które są częściowo zależne od klucza. Zachodzi zatem konieczność dekompozycji schematu relacji Uczestnictwo na mniejsze relacje.



Druga postać normalna 2NF (3)

Uczestnictwo'

<u>IdPrac</u>	NrProj	Funkcja
---------------	--------	---------

$fd1: \{IdPrac, NrProj\} \rightarrow Funkcja$

Pracownicy

<u>IdPrac</u>	ENAME
---------------	-------

$fd5: \{IdPrac\} \rightarrow Nazwisko$

Projekty

<u>NrProj</u>	NazwaProj	Lokalizacja
---------------	-----------	-------------

$fd6: \{NrProj\} \rightarrow NazwaProj$

$fd7: \{NrProj\} \rightarrow Lokalizacja$

$\{fd1, fd2, fd3, fd4, fd5, fd6, fd7\}^+ \equiv \{fd1, fd5, fd6, fd7\}^+$
bo:

$fd1 \Rightarrow fd2, fd3, fd4$, zgodnie z regułą poszerzenia

Zależnością funkcyjną występującą w schemacie Uczestnictwo, która narusza definicję 2NF jest zależność $fd5$. W związku z tym tworzymy nowy schemat relacji Pracownicy zawierający lewą i prawą stronę zależności funkcyjnej $fd5$ i usuwamy ze schematu relacji Uczestnictwo prawą stronę zależności funkcyjnej $fd5$. Zmodyfikowany schemat Uczestnictwo nadal nie spełnia definicji 2NF ze względu na zależności funkcyjne $fd6$ i $fd7$. Podobnie jak poprzednio, tworzymy nowy schemat relacji Projekty zawierający zależności funkcyjne $fd6$ i $fd7$ i usuwamy ze schematu relacji Uczestnictwo prawe strony zależności funkcyjnych $fd6$ i $fd7$. Uzyskany schemat Uczestnictwo' składa się z atrybutów: $IdPrac$, $NrProj$, $Funkcja$ i, co łatwo zauważyć, spełnia definicję 2NF. Ostatecznie, w wyniku dekompozycji schematu relacji Uczestnictwo otrzymujemy 3 schematy relacji: Uczestnictwo', Pracownicy, Projekty, wszystkie w 2NF.



Trzecia postać normalna 3NF (1)

Pracownicy-PP

Nazwisko	Instytut	Wydział
Brzeziński	I.Informatyki	Elektryczny
Morzy	I.Informatyki	Elektryczny
Koszłajda	I.Informatyki	Elektryczny
Królikowski	I.Informatyki	Elektryczny
...
Babij	ElektroEnerg.	Elektryczny
Kordus	ElektroEnerg.	Elektryczny
Sroczan	ElektroEnerg.	Elektryczny

Klucz: Nazwisko

Zależności funkcyjne: *Nazwisko* → *Instytut*

Nazwisko → *Wydział*

Instytut → *Wydział*

BD – wykład 5 (17)

Rozważmy przykład relacji Pracownicy-PP przedstawiony na slajdzie. Relacja składa się z 3 atrybutów: Nazwisko, Instytut, Wydział. Załóżmy, że kluczem schematu relacji jest atrybut Nazwisko. Łatwo zauważyć, że schemat relacji Pracownicy-PP jest w 2NF (gdyż klucz jest jednoatrybutowy). Niestety, w schemacie relacji Pracownicy-PP występują wszystkie wymienione wcześniej typy anomalii. Fakt, że Instytut Informatyki należy do Wydziału Elektrycznego jest powielony tyle razy ilu pracowników jest zatrudnionych w instytucie (redundancja danych i anomalia aktualizacji). Występuje zjawisko anomalii wstawiania – do relacji Pracownicy-PP nie można wstawić informacji o nowoutworzonym na Wydziale Elektrycznym Instytucie Sterowania, tak długo jak długo nie zostanie zatrudniony pierwszy pracownik w tym instytucie. Wreszcie, występuje w tym schemacie również anomalia usuwania – usuwając kolejno pracowników Babij, Kordus, ..., z Instytutu Elektroenergetyki mimo woli usuniemy informacje o przypisaniu Instytutu Elektroenergetyki do Wydziału Elektrycznego.



Trzecia postać normalna 3NF (2)

- **Przechodnia zależność funkcyjna**

Zbiór atrybutów Y jest przechodnio funkcyjnie zależny od zbioru atrybutów X w schemacie R , jeżeli $X \rightarrow Y$ i istnieje zbiór atrybutów Z , nie będący podzbiorem żadnego klucza schematu R taki, że zachodzi $X \rightarrow Z$ i $Z \rightarrow Y$

Zależność funkcyjna $X \rightarrow Y$ jest zależnością przechodnią jeżeli istnieje podzbiór atrybutów Z taki, że zachodzi $X \rightarrow Z$, $Z \rightarrow Y$ i nie zachodzi $Z \rightarrow X$ lub $Y \rightarrow Z$

Wszystkie wymienione problemy wynikają z faktu występowania w schemacie relacji Pracownicy-PP przechodniej zależności funkcyjnej. Mówimy, że zbiór atrybutów Y jest przechodnio funkcyjnie zależny od zbioru atrybutów X w schemacie R , jeżeli $X \rightarrow Y$ i istnieje zbiór atrybutów Z , nie będący podzbiorem żadnego klucza schematu R taki, że zachodzi $X \rightarrow Z$ i $Z \rightarrow Y$. Innymi słowy, mówimy, że zależność funkcyjna $X \rightarrow Y$ jest zależnością przechodnią jeżeli istnieje podzbiór atrybutów Z taki, że zachodzi $X \rightarrow Z$, $Z \rightarrow Y$ i nie zachodzi $Z \rightarrow X$ lub $Y \rightarrow Z$.



Trzecia postać normalna 3NF (3)

- Dana relacja r o schemacie R jest w trzeciej postaci normalnej (**3NF**), jeżeli dla każdej zależności funkcyjnej $X \rightarrow A$ w R spełniony jest jeden z następujących warunków:
 - X jest nadkluczem schematu R , lub
 - A jest atrybutem podstawowym schematu R

Wprowadzimy obecnie definicję trzeciej postaci normalnej. Dana relacja r o schemacie R jest w trzeciej postaci normalnej (**3NF**), jeżeli dla każdej zależności funkcyjnej $X \rightarrow A$ w R spełniony jest jeden z następujących warunków:

- X jest nadkluczem schematu R , lub
- A jest atrybutem podstawowym schematu R .



Trzecia postać normalna 3NF (4)

Pracownicy-PP-1

Nazwisko	Instytut
Brzeziński	I.Informatyki
Morzy	I.Informatyki
Koszalajda	I.Informatyki
Królikowski	I.Informatyki
...	...
Babij	ElektroEnerg.
Kordus	ElektroEnerg.
Sroczan	ElektroEnerg.

Pracownicy-PP-2

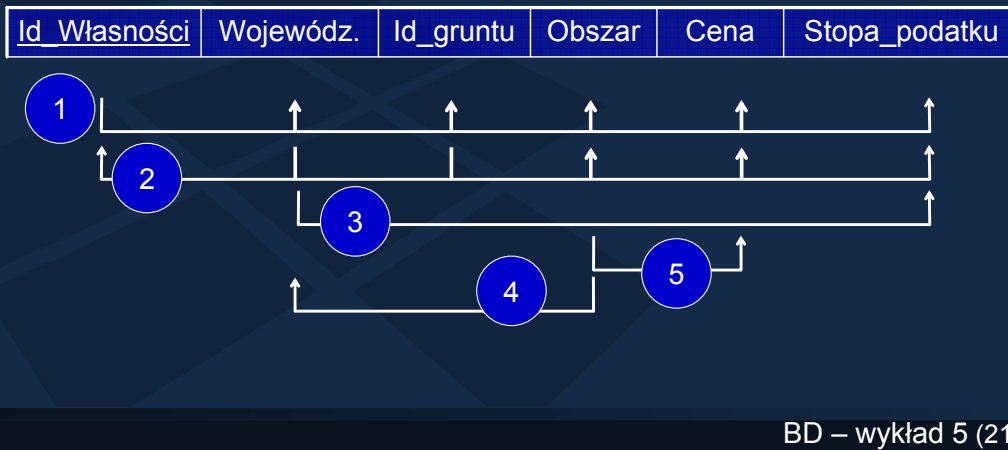
Instytut	Wydział
I.Informatyki	Elektryczny
...	Elektryczny
ElektroEnerg.	Elektryczny

Zauważmy, że wszystkie problemy związane z występowaniem anomalii znikną jeżeli zdekomponujemy relację Pracownicy-PP na dwie relacje Pracownicy-PP1 i Pracownicy-PP2. Relacja Pracownicy-PP1 zawiera informacje o pracownikach, natomiast relacja Pracownicy-PP2 zawiera informacje o przypisaniu instytutów do wydziałów. Zauważmy, że w przypadku relacji Pracownicy-PP2 przynależność instytutu do wydziału jest pamiętana tylko w jednej krotce – brak redundancji danych. Zauważmy również, że dekompozycja rozwiązuje problem anomalii wstawiania – informacje o nowym instytucie możemy wstawić do relacji Pracownicy-PP2, nawet jeżeli instytut ten nie zatrudnia żadnego pracownika. Dekompozycja ta rozwiązuje również problem anomalii usuwania – usunięcie informacji o pracownikach z relacji Pracownicy-PP1 nie pociąga za sobą usunięcia informacji o przypisaniu instytutów do wydziałów. Dekompozycja rozwiązuje również problem anomalii aktualizacji – zmiana przypisania instytutu do wydziału, np. Instytut Informatyki przeniesiony do Wydziału Informatyki i Zarządzania, dotyczy wyłącznie jednej krotki. Zauważmy, że dekompozycja relacji Pracownicy-PP na relacje Pracownicy-PP1 i Pracownicy-PP2 jest dekompozycją bez utraty informacji w tym sensie, że łącząc relację Pracownicy-PP1 i Pracownicy-PP2 wg. atrybutu połączeniowego Instytut możemy odtworzyć oryginalną zawartość relacji Pracownicy-PP.



Postać normalna Boyce-Codd (BCNF) (1)

- Postać normalna *Boyce-Codd*'a stanowi warunek dostateczny 3NF, ale nie konieczny



Rozważmy przykład relacji grunty przedstawiony na slajdzie. Schemat relacji składa się z 6 atrybutów: *Id_Własności*, *Województwo*, *Id_gruntu*, *Obszar*, *Cena*, *Stopa_podatku*. Schemat relacji posiada 2 klucze. Pierwszym z nich jest atrybut *Id_Własności*, a drugim – para atrybutów: *Województwo* i *Id_gruntu*. Atrybutami podstawowym relacji są: *Id_Własności*, *Województwo* i *Id_gruntu*. Atrybutami wtórnymi są: *Obszar*, *Cena*, *Stopa_podatku*.

Zbiór zależności funkcyjnych związanych ze schematem relacji został przedstawiony na slajdzie. Zależność nr 1 i nr 2 są zależnościami od klucza. Zależność nr 3 stwierdza, że atrybut *Stopa_podatku* zależy od atrybutu *Województwo*. Zależność nr 4 oznacza, że atrybut *Województwo* zależy od atrybutu *Obszar*. Zależność nr 5 oznacza, że atrybut *Cena* zależy od atrybutu *Obszar*.

Łatwo zauważyć, że schemat relacji jest w 1NF, nie jest natomiast w 2NF. Wynika to z faktu, że atrybut wtórny *Stopa_podatku* jest częściowo funkcyjnie zależny do klucza *Województwo* i *Id_gruntu* (zależność nr 3).



Postać normalna Boyce-Codd (BCNF) (3)

Grunty

<u>Id_Własności</u>	Wojewódz.	Id-gruntu	Obszar	Cena	Stopa_produkту
---------------------	-----------	-----------	--------	------	----------------



Grunty-1

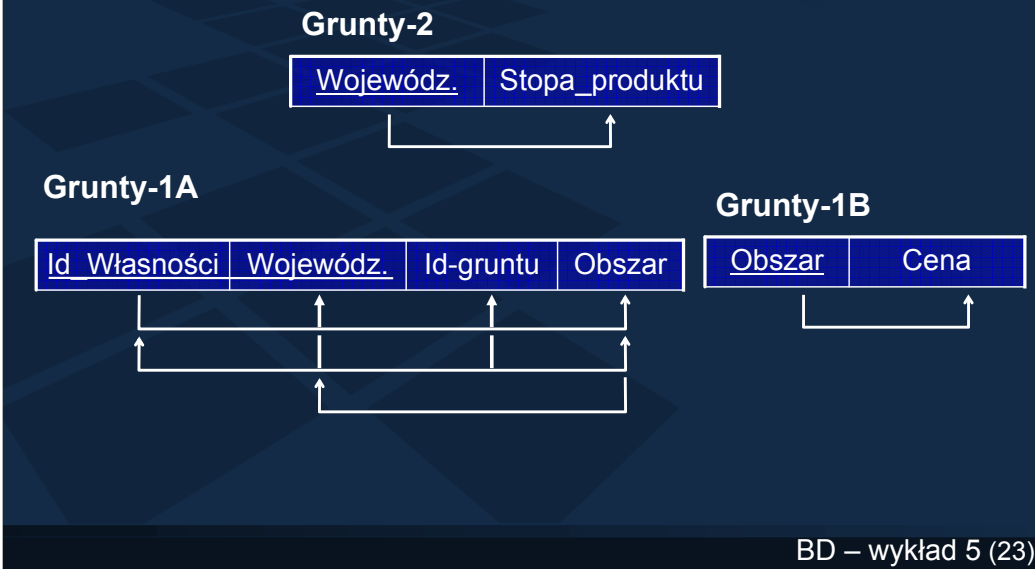
<u>Id_Własności</u>	Wojewódz.	Id-gruntu	Obszar	Cena
---------------------	-----------	-----------	--------	------



Dekomponujemy schemat Grunty na dwa schematy Grunty-1 i Grunty-2. Relacja Grunty-2 jest w 2NF i 3NF. Relacja Grunty-1 jest w 2NF, nie jest natomiast w 3NF ze względu na zależność funkcyjną nr 5 (Obszar -> Cena).



Postać normalna Boyce-Codd (BCNF) (4)



BD – wykład 5 (23)

Dokonajmy dekompozycji schematu Grunty-1 do schematów Grunty-1A i Grunty-1B. Ta dekompozycja kończy proces normalizacji schematu Grunty do zbioru schematów relacji w 3NF.



Postać normalna Boyce-Codd (BCNF) (2)

- Załóżmy, że w relacji *Grunty* mamy tylko dwa województwa. Co więcej, załóżmy, że działki w pierwszym województwie mają rozmiar 0.5, 0.6, 0.7 h; natomiast działki w drugim województwie mają obszar 1, 1.2, 1.4 h. Ta informacja może być powielona w tysiącach krotek relacji *Grunty* oraz, po dekompozycji, w relacji *Grunty-1A*
- Relacja *Grunty-1A* jest w trzeciej postaci normalnej (*Wojewódz.* jest atrybutem podstawowym)

Zależność funkcyjna nr 4 (Obszar \rightarrow Województwo) modeluje następującą sytuację rzeczywistą. Załóżmy, że w relacji *Grunty* mamy tylko dwa województwa. Co więcej, załóżmy, że działki w pierwszym województwie mają rozmiar 0.5, 0.6, 0.7 h; natomiast działki w drugim województwie mają obszar 1, 1.2, 1.4 h. Ta sytuacja jest opisana zależnością funkcyjną nr 4. Informacja o zależności województwa od obszaru jest powielona w tysiącach krotek relacji *Grunty* oraz, po dekompozycji, w relacji *Grunty-1A*. Relacja *Grunty-1A* jest w trzeciej postaci normalnej (*Województwo* jest atrybutem podstawowym). Część projektantów schematów baz danych traktuje to jako istotną wadę 3NF. Proponują oni dekompozycję schematów relacji do zmodyfikowanej 3NF, nazywanej postacią normalną Boyce'a-Codd'a. Otóż definicja postaci Boyce'a-Codd'a jest następująca:

Dana relacja r o schemacie R jest w postaci normalnej Boyce'a-Codd'a (**BCNF**), jeżeli dla każdej zależności funkcyjnej $X \rightarrow A$ w R spełniony jest następujący warunek: X jest nadkluczem schematu R . W tym przypadku, zachodzi konieczność dekompozycji relacji *Grunty-1A* na dwa schematy relacji: *Grunty-1A1* (*Id_Własności*, *Id_Gruntu*, *Obszar*) oraz *Grunty-1A2* (*Obszar*, *Województwo*).



Zależności wielowartościowe (1)

Loty

Lot	Dzień_tygodnia	Typ_samolotu
106	poniedziałek	134
106	czwartek	154
106	poniedziałek	154
106	czwartek	134
206	środa	747
206	piątek	767
206	środa	767
206	piątek	747

Języki

Nazwisko	Język_obcy	Język_prog.
Nowak	angielski	Basic
Nowak	włoski	Fortran
Nowak	angielski	Fortran
Nowak	włoski	Basic
Nowak	czeski	Basic
Nowak	czeski	Fortran

Rozważmy przykładowe relacje Loty i Języki przedstawione na slajdach. Relacja Loty składa się z 3 atrybutów: Lot, Dzień_tygodnia, Typ_samolotu. Opisuje ona typ samolotu i dzień tygodnia, w którym odbywają się określone loty. Kluczem schematu relacji Loty są wszystkie trzy wymienione atrybuty. Stąd, schemat relacji Loty jest w 3NF i BCNF. Niestety, schemat ten posiada dość istotną wadę – występuje w nim problem modyfikacji zależnej od stanu bazy danych.

Podobny problem występuje w schemacie relacji Języki składającej się również z 3 atrybutów: Nazwisko, Język_obcy, Język_programowania, które również stanowią klucz schematu relacji.



Modyfikacja relacji z zależnościami wielowartościowymi

- Lot 106 będzie dodatkowo odbywał się w Środę i na tę linię wprowadzamy, dodatkowo, nowy typ samolotu – 104

Loty

Lot	Dzień-tygodnia	Typ-samolotu
106	poniedziałek	134
106	czwartek	154
106	poniedziałek	154
106	czwartek	134
106	poniedziałek	104
106	czwartek	104
106	środa	134
106	środa	154
106	środa	104

BD – wykład 5 (26)

Rozważmy prostą modyfikację relacji Loty. Załóżmy, że lot 106 będzie dodatkowo odbywał się w środę i na tę linię wprowadzamy, dodatkowo, nowy typ samolotu – 104. Zauważmy, że ta stosunkowo prosta modyfikacja wymaga wprowadzenia aż pięciu nowych krotek do relacji Loty: <106, poniedziałek, 104>, <106, czwartek, 104>, <106, środa, 134>, <106, środa, 154>, <106, środa, 104>. Dwie pierwsze krotki wiążą się z faktem, że zarówno w poniedziałek jak i czwartek lot 106 będzie obsługiwał nowy typ samolotu 104, pozostałe 3 krotki wiążą się z faktem, że lot 106 będzie dodatkowo odbywał się w środę. Liczba wprowadzanych krotek zależy od aktualnego stanu bazy danych. Ta własność schematu relacji Loty utrudnia pielęgnację tej relacji przez osoby nie będące informatykami.

Podobny problem występuje w odniesieniu do relacji języki. Załóżmy, że Nowak nauczył się języka obcego francuskiego i języka programowania C++. Wprowadzenie tej modyfikacji do relacji języki wymaga wprowadzenia 6 nowych krotek.



Dekompozycja

Lot-1

Lot	Dzień-tygodnia
106	poniedziałek
106	czwartek
206	środa
206	piątek
...	...
106	środa

Lot-2

Lot	Typ-samolotu
106	134
106	154
206	747
206	767
...	...
106	104

Język-1

Nazwisko	Język_obcy
Nowak	angielski
Nowak	włoski
Nowak	czeski

Język-2

Nazwisko	Język_prog.
Nowak	Basic
Nowak	Fortran

Wymieniony wyżej problem modyfikacji relacji Loty i Języki znika jeżeli oba schematy zdekomponujemy odpowiednio na: Lot-1 i Lot-2 oraz Język-1 i Język-2. Przykładowo, wprowadzenie modyfikacji „lot 106 będzie dodatkowo odbywał się w Środę i na tę linię wprowadzamy, dodatkowo, nowy typ samolotu – 104” wymaga, po dekompozycji, wprowadzenia jednej krotki <106, środa> do relacji Lot-1 oraz jednej krotki <106, 104> do relacji Lot-2. Zauważmy, że teraz modyfikacja ta nie zależy od stanu bazy danych.

Podobnie jest w przypadku modyfikacji relacji Języki. Wprowadzenie modyfikacji „Nowak nauczył się języka obcego francuskiego i języka programowania C++” wymaga wprowadzenia jednej krotki <Nowak, francuski> do relacji Język-1 i <Nowak, C++> do relacji Język-2.



Zależności wielowartościowe (2)

- Zależności wielowartościowe są konsekwencją wymagań pierwszej postaci normalnej, która nie dopuszcza, aby krotki zawierały atrybuty wielowartościowe
- Zależność wielowartościowa występuje w relacji $r(R)$ nie dlatego, że na skutek zbiegu okoliczności tak ułożyły się wartości krotek, lecz występuje ona dla dowolnej relacji r o schemacie R dlatego, że odzwierciedla ona ogólną prawidłowość modelowanej rzeczywistości

Lot → → *Dzień-tygodnia*

Lot → → *Typ-samolotu*

Nazwisko → → *Język-obcy*

Nazwisko → → *Język-programowania*

BD – wykład 5 (28)

Powyższe problemy z modyfikacją zależną od stanu bazy danych wynikają z faktu występowania w schemacie relacji *Loty* i *Języki* tzw. zależności wielowartościowych. Zależności wielowartościowe są konsekwencją wymagań pierwszej postaci normalnej, która nie dopuszcza, aby krotki zawierały atrybuty wielowartościowe. Zależność wielowartościowa jest własnością semantyczną schematu relacji.

Zależność wielowartościowa występuje w relacji $r(R)$ nie dlatego, że na skutek zbiegu okoliczności tak ułożyły się wartości krotek, lecz występuje ona dla dowolnej relacji r o schemacie R dlatego, że odzwierciedla ona ogólną prawidłowość modelowanej rzeczywistości. W przykładowych relacjach *Loty* i *Języki* występują 4 zależności wielowartościowe:

Lot → → *Dzień-tygodnia*

Lot → → *Typ-samolotu*

Nazwisko → → *Język-obcy*

Nazwisko → → *Język-programowania*



Zależności wielowartościowe (3)

- Wystąpienie zależności wielowartościowej $X \twoheadrightarrow Y$ w relacji o schemacie $R = XYZ$ wyraża dwa fakty:
- Związek pomiędzy zbiorami atrybutów X i Y ;
- Niezależność zbiorów atrybutów Y, Z . Zbiory te są związane ze sobą pośrednio poprzez zbiór atrybutów X

Lot-3

Lot	Dzień-tygodnia	Typ-samolotu
106	poniedziałek	134
106	czwartek	154
106	czwartek	134
206	środa	747
206	piątek	767

BD – wykład 5 (29)

Wystąpienie zależności wielowartościowej $X \twoheadrightarrow Y$ w relacji o schemacie $R = XYZ$ wyraża dwa fakty:

- związek pomiędzy zbiorami atrybutów X i Y ;
- niezależność zbiorów atrybutów Y, Z ; zbiory te są związane ze sobą pośrednio poprzez zbiór atrybutów X .

W relacji Lot-3 przedstawionej na slajdzie występuje jedna zależność wielowartościowa: $\text{Lot} \twoheadrightarrow \{\text{Dzień_tygodnia}, \text{Typ_samolotu}\}$. Dekompozycja schematu Lot-3, podobnie jak w przypadku relacji Loty, prowadziłaby do utraty informacji, że lot 206 w środę jest obsługiwany przez typ samolotu 747 i lot 206 w piątek jest obsługiwany przez typ samolotu 767. Innymi słowy, schemat Lot-3 jest niedekomponowalny bez utraty informacji.



Definicja własności zależności wielowartościowych

- Niech R oznacza schemat relacji, natomiast X , Y są rozłącznymi zbiorami atrybutów schematu R i $Z = R - (XY)$
- Relacja $r(R)$ spełnia zależność wielowartościową $X \twoheadrightarrow Y$, jeżeli dla dwóch dowolnych krotek t_1 i t_2 z $r(R)$ takich, że $t_1[X] = t_2[X]$, zawsze istnieją w $r(R)$ krotki t_3 , t_4 takie, że spełnione są następujące warunki:

$$\begin{aligned} t_1[X] &= t_2[X] = t_3[X] = t_4[X] \\ t_3[Y] &= t_1[Y] \text{ i } t_4[Y] = t_2[Y] \\ t_3[R - X - Y] &= t_2[R - X - Y] \\ t_4[R - X - Y] &= t_1[R - X - Y] \end{aligned}$$

- Z symetrii powyższej definicji wynika, że jeżeli w relacji $r(R)$ zachodzi $X \twoheadrightarrow Y$, to zachodzi również: $X \twoheadrightarrow [R - X - Y]$
- Ponieważ $R - X - Y = Z$, to powyższy fakt zapisujemy czasami w postaci: $X \twoheadrightarrow Y / Z$

BD – wykład 5 (30)

Teraz krótko scharakteryzujemy własności zależności wielowartościowych. Niech R oznacza schemat relacji, natomiast X , Y są rozłącznymi zbiorami atrybutów schematu R i $Z = R - (XY)$.

Relacja $r(R)$ spełnia zależność wielowartościową $X \twoheadrightarrow Y$, jeżeli dla dwóch dowolnych krotek t_1 i t_2 z $r(R)$ takich, że $t_1[X] = t_2[X]$, zawsze istnieją w $r(R)$ krotki t_3 , t_4 takie, że spełnione są następujące warunki, przedstawione na slajdzie:

$$\begin{aligned} t_1[X] &= t_2[X] = t_3[X] = t_4[X] \\ t_3[Y] &= t_1[Y] \text{ i } t_4[Y] = t_2[Y] \\ t_3[R - X - Y] &= t_2[R - X - Y] \\ t_4[R - X - Y] &= t_1[R - X - Y] \end{aligned}$$

Z symetrii powyższej definicji wynika, że jeżeli w relacji $r(R)$ zachodzi $X \twoheadrightarrow Y$, to zachodzi również: $X \twoheadrightarrow [R - X - Y]$. Ponieważ $R - X - Y = Z$. Powyższy fakt zapisujemy czasami w postaci: $X \twoheadrightarrow Y / Z$.



Trywialna zależność wielowartościowa (1)

- Zależność wielowartościowa $X \twoheadrightarrow Y$ w relacji $r(R)$ nazywamy zależnością trywialną, jeżeli
 - zbiór Y jest podzbiorem X , lub
 - $X \cup Y = R$
- Zależność nazywamy trywialną, gdyż jest ona spełniona dla dowolnej instancji r schematu R

Zanim przejdziemy do zdefiniowania czwartej postaci normalnej wprowadźmy pojęcie trywialnej zależności wielowartościowej. Zależność wielowartościowa $X \twoheadrightarrow Y$ w relacji $r(R)$ nazywamy zależnością trywialną, jeżeli zbiór Y jest podzbiorem X , lub $X \cup Y = R$. Zależność nazywamy trywialną, gdyż jest ona spełniona dla dowolnej instancji r schematu R .



Czwarta postać normalna 4NF

- Relacja r o schemacie R jest w czwartej postaci normalnej (**4NF**) względem zbioru zależności wielowartościowych MVD jeżeli jest ona w $3NF$ i dla każdej zależności wielowartościowej $X \twoheadrightarrow Y \in MVD$ zależność ta jest trywialna lub X jest nadkluczem schematu

Obecnie wprowadzimy definicję czwartej postaci normalnej. Mówimy, że relacja r o schemacie R jest w czwartej postaci normalnej (**4NF**) względem zbioru zależności wielowartościowych MVD jeżeli jest ona w $3NF$ i dla każdej zależności wielowartościowej $X \twoheadrightarrow Y \in MVD$ zależność ta jest trywialna lub X jest nadkluczem schematu.

Jak łatwo zauważyć, przedstawione uprzednio schematy relacji Loty i Języki nie są w 4NF. Przykładowo, schemat relacji Loty nie jest w 4NF gdyż zależność wielowartościowa, np. $Lot \twoheadrightarrow Typ_samolotu$ nie jest trywialna jak również Lot nie jest nadkluczem schematu Loty. Równie łatwo zauważyć, że schematy relacji Lot-1 i Lot-2, uzyskane w wyniku dekompozycji schematu Loty, są w 4NF gdyż każdy z tych schematów zawiera trywialną zależność wielowartościową. Podobnie jest w przypadku relacji Języki.



Dekompozycja relacji na relacje bez utraty informacji (1)

- **Dekompozycja na relacje w 3NF**

Dana jest relacja r o schemacie R , i dany jest zbiór F zależności funkcyjnych dla R . Niech relacje r_1 i r_2 o schematach, odpowiednio, R_1 i R_2 , oznaczają dekompozycję relacji $r(R)$. Dekompozycja ta jest dekompozycją bez utraty informacji, jeżeli co najmniej jedna z poniższych zależności funkcyjnych jest spełniona:

$$R_1 \cap R_2 \rightarrow R_1$$

$$R_1 \cap R_2 \rightarrow R_2$$

Na zakończenie podamy twierdzenia dotyczące dekompozycji schematów relacji na mniejsze schematy relacji, bez utraty informacji. Pierwsze twierdzenie dotyczy dekompozycji schematu relacji R na schematy relacji w 3NF.

Dana jest relacja r o schemacie R , i dany jest zbiór F zależności funkcyjnych dla R . Niech relacje r_1 i r_2 o schematach, odpowiednio, R_1 i R_2 , oznaczają dekompozycję relacji $r(R)$. Dekompozycja ta jest dekompozycją bez utraty informacji, jeżeli co najmniej jedna z poniższych zależności funkcyjnych jest spełniona:

- $R_1 \text{ ILOCZYN } R_2 \rightarrow R_1$,

- $R_1 \text{ ILOCZYN } R_2 \rightarrow R_2$.



Dekompozycja relacji na relacje bez utraty informacji (1)

- **Dekompozycja na relacje w 4NF**

Dana jest relacja r o schemacie R . Niech relacje r_1 i r_2 o schematach, odpowiednio, R_1 i R_2 , oznaczają dekompozycję relacji $r(R)$. Dekompozycja ta jest dekompozycją bez utraty informacji, jeżeli co najmniej jedna z poniższych zależności wielowartościowych jest spełniona:

$$R_1 \cap R_2 \twoheadrightarrow (R_1 - R_2)$$

$$R_1 \cap R_2 \twoheadrightarrow (R_2 - R_1)$$

Drugie twierdzenie dotyczy dekompozycji schematu relacji R na schematy relacji w 4NF.

Dana jest relacja r o schemacie R . Niech relacje r_1 i r_2 o schematach, odpowiednio, R_1 i R_2 , oznaczają dekompozycję relacji $r(R)$. Dekompozycja ta jest dekompozycją bez utraty informacji, jeżeli co najmniej jedna z poniższych zależności wielowartościowych jest spełniona:

$$- R_1 \text{ ILOCZYN } R_2 \twoheadrightarrow (R_1 - R_2),$$

$$- R_1 \text{ ILOCZYN } R_2 \twoheadrightarrow (R_2 - R_1).$$

Przykładowo, dekompozycja schematu relacji Loty na schematy Lot-1 i Lot-2 w 4NF jest dekompozycją bez utraty informacji, gdyż:

- Lot-1 ILOCZYN Lot-2 = {Lot} wyznacza wielowartościowo zarówno (Lot-1 – Lot-2) jaki (Lot-2 – Lot-1). (Lot-1 – Lot-2) = {Dzień_tygodnia}, natomiast (Lot-2 – Lot-1) = {Typ_samolotu}.