

Wykład 4. Związki encji. Związki niejednoznaczne i związki rekurencyjne.

Wykład 4 zawiera dokładne omówienie pojęcia związku encji. Przedstawiony jest problem związku niejednoznacznego oraz rekurencyjnego. Na końcu wykładu zamieszczone są dwa, stosunkowo proste, przykłady diagramów związków encji modelujące zadane problemy.

Związek niejednoznaczny

Każdy związek niejednoznaczny, czyli taki związek, który nie da się go opisać funkcją, trzeba sprowadzić do związków jednoznacznych. W modelu relacyjnym nie istnieje inne rozwiązanie tego problemu. Związek niejednoznaczny jest nazywany również związkiem wiele – do – wiele.

Transformacja związku niejednoznacznego

Transformację związku niejednoznacznego można przedstawić w postaci następującego algorytmu:

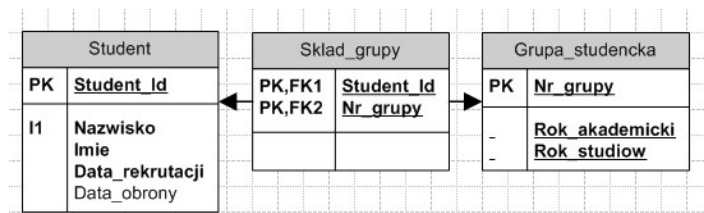
- Dla niejednoznacznego związku binarnego $Z(E1,E2)$ wprowadzamy nową encję $E0$ i dwa związki jednoznaczne (binarne) $Z1(E0,E1)$ oraz $Z2(E0,E2)$, łączące nową encję z encjami starymi. W utworzonych związkach encja nowa pozostaje po stronie „wiele” związku, encje wyjściowe po stronie „jeden”. Kluczem encji $E0$ jest suma kluczy encji $E1$ i $E2$
- Dla związku o liczbie argumentów większej niż dwa $Z(E1,...,En)$, $n > 2$ wprowadzamy nową encję $E0$ i n jednoznacznych związków binarnych $Zi(E0,Ei)$, łączących nową encję z encjami już istniejącymi. Klucz encji $E0$ jest sumą kluczy encji $E1,...,En$.

Wprowadzana encja reprezentująca związek, nazywa się encją asocjacyjną. Jest ona zawsze encją słabą (zależną), bo związki łączące ją z encjami - argumentami związku niejednoznacznego, są związkami identyfikującymi.

Nawiązując do poprzedniego przykładu związku jednoznacznego, zachodzącego pomiędzy encjami **Student** i **Grupa_studencka**, zmodyfikujmy jego założenia. Przyjmijmy, że każdy student zapisany jest do kilku grup studenckich (ćwiczeniowej, laboratoryjnej, wykładowej, językowej itd.) Oczywiście, do każdej z tych grup zapisanych jest wielu studentów. Taki związek, niedający opisać się funkcją, jest związkiem niejednoznacznym (wiele – do – wiele) i w modelu relacyjnym wymaga rozłożenia na dwa związki jednoznaczne.

Nawiązując do poprzedniego przykładu związku jednoznacznego, zachodzącego pomiędzy encjami **Student** i **Grupa_studencka**, zmodyfikujmy jego założenia. Przyjmijmy, że każdy student zapisany jest do kilku grup studenckich (ćwiczeniowej, laboratoryjnej, wykładowej, językowej itd.) Oczywiście, do każdej z tych grup zapisanych jest wielu studentów. Taki związek, niedający opisać się funkcją, jest związkiem niejednoznacznym (wiele – do – wiele) i w modelu relacyjnym wymaga rozłożenia na dwa związki jednoznaczne.

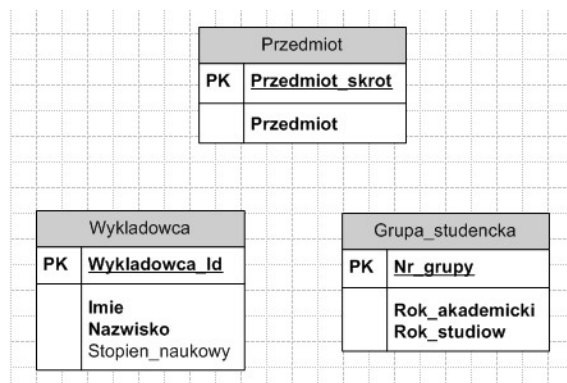
Stosując opisaną metodę wprowadzimy nową encję asocjacyjną, której zadaniem będzie opisanie związku, jaki zachodzi pomiędzy studentami i grupami. Encja asocjacyjna **Sklad_Grupy** zostanie powiązana z encjami wyjściowymi poprzez związki jednoznaczne, w których encje wyjściowe pozostają po stronie „jeden”, a encja asocjacyjna po stronie „wiele”.



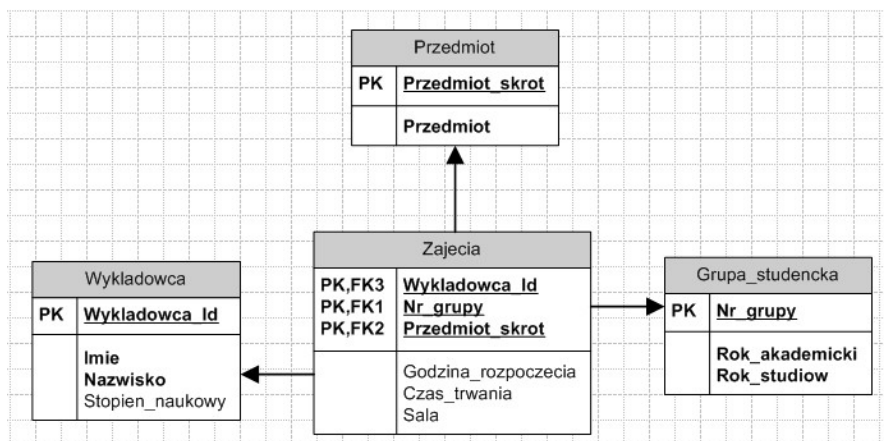
Jednoznaczny identyfikator (klucz) encji asocjacyjnej **Sklad_Grupy** stanowi suma kluczy obcych pochodzących z encji, które wchodzą w związek niejednoznaczny. Ten klucz może, (ale nie musi!) zostać kluczem głównym encji asocjacyjnej.

Obecnie, każda instancja encji po „stronie” jeden zostaje powiązana, z co najwyżej jedną instancją encji po stronie „wiele”. Zawartość tabeli, której encja **Sklad_Grupy** jest modelem, stanowią będą pary wartości: **Id_studenta** i **Id_grupy**.

Powyższy przykład dotyczy związku niejednoznacznego binarnego, w który wchodzi dwie encje. Jeżeli w związek niejednoznaczny wchodzi więcej niż dwie encje, jest on rozkładany na tyle związków jednoznacznych, ile jest encji w wyjściowym związku niejednoznacznym. Poniżej przedstawiamy przykład transformacji dla związku trójargumentowego. Związek można wyartykułować następująco: wykładowca prowadzi zajęcia dla grupy studenckiej, z przedmiotu nauczania. Encje tworzące związek to **Grupa_studencka**, **Wykladowca**, **Przedmiot**.



Stosując opisaną wyżej metodę, wprowadzimy nową encję asocjacyjną, której zadaniem będzie opisanie zależności, jakie zachodzą pomiędzy wykładowcami, przedmiotami nauczania i grupami. Encje wyjściowe zostaną powiązane z encją asocjacyjną związkami jednoznacznymi, w których encja asocjacyjna pozostaje po stronie „wiele”.



Kluczem encji asocjacyjnej jest, zgodnie z definicją, suma kluczy encji wchodzących w związek niejednoznaczny, z czego wynika, iż związki encji wyjściowych z encją asocjacyjną są związkami identyfikującymi.

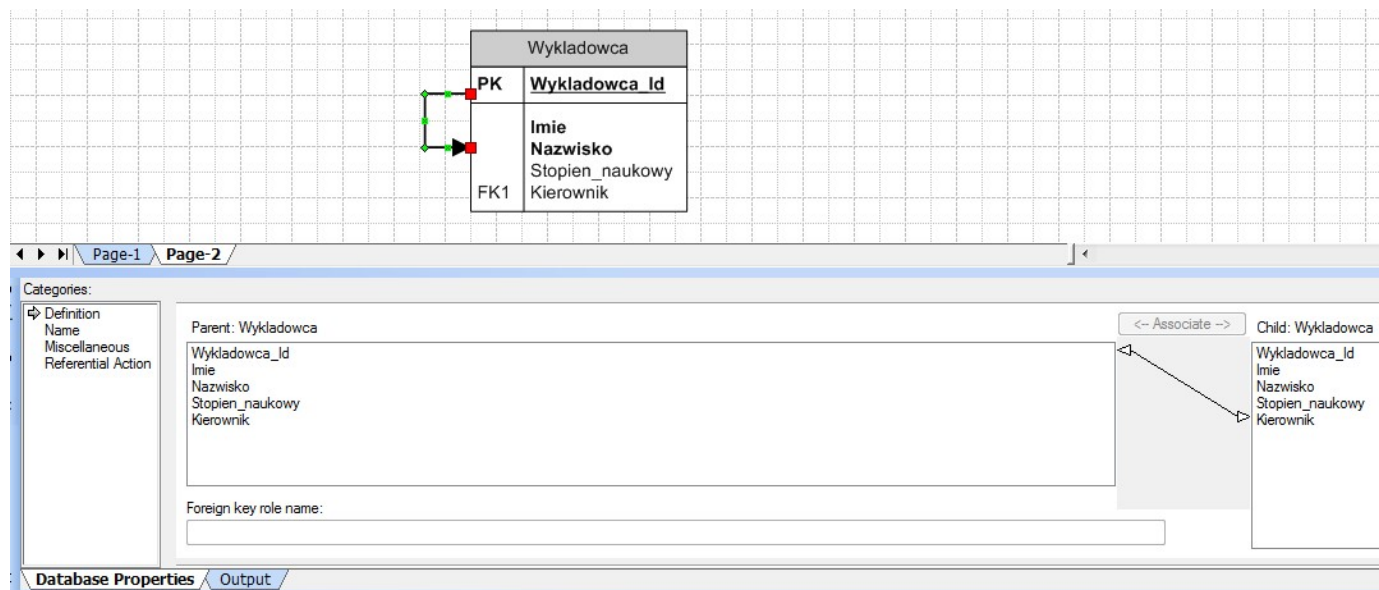
Encja asocjacyjna reprezentuje związek, zatem mogą być w niej umieszczane atrybuty opisujące ten związek. W powyższym przykładzie rola związku została przedstawiona, jako opis zajęć prowadzonych przez wykładowców, z przedmiotów nauczania, dla grup studenckich. Dodatkowymi atrybutami tego związku mogą być (tak, jak w przykładzie) atrybuty *Godzina_rozporoczenia*, *Czas_trwania*, *Sala*. Oczywiście, tak jak w przypadku encji opisujących klasy obiektów, zestawy atrybutów opisujących związki, mogą się różnić, w zależności od koncepcji bazy danych. W tym przykładzie można *Czas_trwania* zastąpić atrybutem *Godzina_zakonczenia*, albo dodać atrybut *Typ_zajec*, odpowiedzialny za zidentyfikowanie wykładów, zajęć laboratoryjnych i ćwiczeń.

Związek rekurencyjny

Związek rekurencyjny to związek, w którym jedna encja występuje więcej niż raz. Jednoznaczny związek rekurencyjny to taki związek, w którym po stronie „jeden”, jak i po stronie „wiele”, występuje ta sama encja. Inaczej mówiąc, encja „odwołuje się” do samej siebie.

Jednoznaczny związek rekurencyjny

Związki rekurencyjne wykorzystywane są do modelowania hierarchii lub podległości. Klasycznym przykładem jest umieszczenie w encji opisującej pracowników atrybutu wskazującego na przełożonego pracownika. W pokazanym poniżej przykładzie, związek rekurencyjny zostanie wykorzystany do przedstawienia zależności służbowej (podległości) wykładowców.



Utworzenie w narzędziu CASE związku rekurencyjnego, powoduje powstanie charakterystycznego „ucha”. Należy zwrócić uwagę na pewne szczegóły techniczne, związane z wykonaniem modelu takiego związku. Po poprowadzeniu związku rekurencyjnego, wśród atrybutów niekluczowych nie pojawia się nowy atrybut klucza obcego. Wynika to z tego, że zgodnie z postulatami Codd’a, nazwy atrybutów encji nie mogą się powtarzać. Należy zatem dopisać nowy atrybut o unikalnej nazwie, który będzie pełnił rolę klucza obcego. W naszym przykładzie jest to atrybut *Kierownik*. Następnie w oknie edycji właściwości związku należy powiązać ze sobą atrybut klucza głównego i klucza obcego (przycisk Associate na powyższym rysunku).

Jednoznaczny związek rekurencyjny nie może być związkiem identyfikującym, dodatkowo klucz obcy związku rekurencyjnego musi dopuszczać wartość NULL. Niespełnienie tych warunków uniemożliwiłoby wpisanie jakiegokolwiek rekordu do przyszłej tabeli.

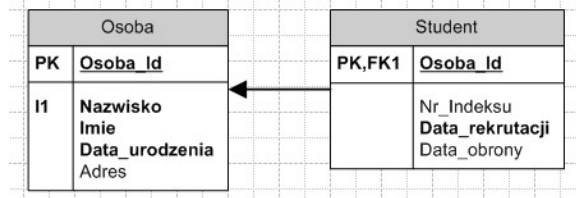
Niejednoznaczny związek rekurencyjny

Istnieją związki niejednoznaczne, w których występuje tylko jedna encja. Taki związek, jak każdy związek niejednoznaczny, jest rozkładany na dwa związki jednoznaczne z nową encją asocjacyjną. W obu związkach, po stronie „jeden” występuje ta sama encja. Przykład takiego związku, ze szczegółowym omówieniem zostanie przedstawiony w dalszej części wykładu.

Związek jedno – jednoznaczny (związek jeden – do – jeden).

Związek jedno – jednoznaczny, to związek jednoznaczny, którego instancja jest różnowartościową funkcją częściową. Inaczej mówiąc, każdej instancji encji po stronie „wiele” odpowiada, co najwyżej jedna instancja encji po stronie „jeden”. Przykładem może być związek encji Student i Osoba, dający się opisać: "Każdy student jest osobą" (Osoba może być studentem, ale nie musi nim być).

Należy zwrócić uwagę na to, że encje po obu stronach związku są identyfikowane przez ten sam atrybut klucza głównego, a instancje encji przez tą samą wartość klucza głównego.



Sposoby odwzorowania związku jednoznaczności na tabele.

Związek jedno-jednoznaczny może zostać odwzorowany w tabele bazy danych na jeden z trzech sposobów (na przykładzie związku między encjami Osoba i Student).

Metoda bezpośredniego przekształcenia encji w tabele.

(Metoda stosowana przez MS Visio) Tworzymy dwie tabele: **Student** i **Osoba**. W tabeli **Osoba** zapisujemy atrybuty wspólne dla wszystkich osób. W tabeli **Student** zapisujemy klucz główny z tabeli **Osoba** (identyfikujący studenta, jako osobę) oraz atrybuty charakterystyczne tylko dla studentów.

Wadą tego rozwiązania jest konieczność używania złączenia dwóch tabel w celu odczytania pełnych danych studenta. Zaletą jest elegancja schematu i brak wad rozwiązań alternatywnych.

Metoda przekształcenia w jedną tabelę

Używamy tylko jednej tabeli, w której są przechowywane wszystkie możliwe atrybuty dotyczące osób. Jeśli osoba nie jest studentem, wartości atrybutów charakterystycznych tylko dla studentów pozostają NULL.

Wadą tego rozwiązania jest istnienie kolumn, które z założenia nie będą wypełnione dla wszystkich rekordów, oraz niemożność utworzenia niektórych więzów integralności w sposób prosty (np. na kolumnie *Data_rekrutacji* nie można ustanowić więzów NOT NULL, gdyż wartość ta z założenia nie występuje w rekordach nie-studentów). Zaletą jest brak konieczności odczytu danych z tabel połączonych.

Metoda przekształcenia w dwie niezależne tabele

Używamy tylko jednej tabeli Student. W przypadku, gdy znajdzie potrzeba reprezentowania jeszcze innego typu osób jak np. Pracownik – dla tej kategorii osób definiujemy osobną tabelę. Gdy jest potrzebna informacja obejmująca wszystkie osoby, trzeba wtedy stosować sumowanie zawartości tabel.

Metoda ta jest dobra, gdy podział na kategorie osób jest rozłączny. Gdy podział nie jest rozłączny, ta sama informacja będzie powtarzana, np. informacja o pracownikach będących jednocześnie studentami. Efektem jest niekontrolowana redundancja – nie istnieje możliwość jednoznacznego stwierdzenia, czy osoby o takich samych danych, ale zapisanych w dwóch różnych tabelach, są faktycznie różnymi osobami. Rozwiązaniem tego problemu jest stosowanie kluczy naturalnych (np. PESEL, przy założeniu, że jest unikalny).

Proste przykłady diagramów związków encji

Przykład 1 - Piwo i piwosze

Zaprojektuj diagram związków encji dla bazy danych, której zadaniem będzie rejestracja badań dotyczących spożycia piwa, prowadzonych na terenie miasta X. Badaniu będą podlegały marki piwa, sprzedawane w pubach i pite tam przez badaną, testową grupę piwoszy. W bazie należy przechowywać dane piwoszy (Imię, Nazwisko, telefon kontaktowy), pubów (nazwa, adres) oraz marek piwa (nazwa, producent). Wynikiem badań, koniecznym do zapisania w bazie będzie informacja o tym, który piwosz, jakie marki piw preferuje, do których pubów uczęszcza, oraz w których pubach, które marki piwa są serwowane i w jakiej cenie.

Zgodnie z zasadami projektowania baz danych, pracę rozpoczynamy od zidentyfikowania obiektów (klas obiektów), które „fizycznie” pojawiają się w opisie problemu. W tym przypadku będą to:

- piwosze (osoby)
- puby (lokale serwujące piwo)
- marki piwa

Ponadto, zgodnie z założeniami, należy znać producenta piwa, zatem wprowadzimy kolejną klasę obiektów

- browar (producent piwa)

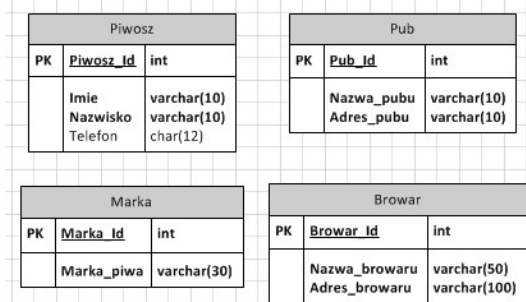
Zaprojektujmy teraz encje, które będą modelowały wymienione obiekty

Piwosz – atrybuty *Imie* (tekstowy), *Nazwisko* (tekstowy), *Telefon* (tekstowy). Ponieważ żadna kombinacja wymienionych atrybutów nie spełnia warunków klucza (nie jest unikalna), zatem należy dodać „sztuczny” atrybut, który będzie pełnił rolę klucza głównego *Piwosz_Id* (liczbowy).

Pub – atrybuty *Nazwa_pubu* (tekst), *Adres_pubu* (tekst) i także atrybut klucza głównego *Pub_Id* (liczbowy).

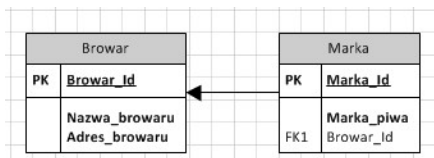
Marka – atrybuty *Marka_Id* (liczbowy), *Marka_piwa* (tekstowy).

Browar – *Browar_Id* (liczbowy), *Nazwa_browaru* (tekstowy), *Adres_browaru* (tekstowy).



Na powyższym rysunku pokazane zostały encje, modelujące obiekty wymienione w opisie problemu. Wyświetlono dodatkowo typy danych, co na dalszych rysunkach zostanie pominięte

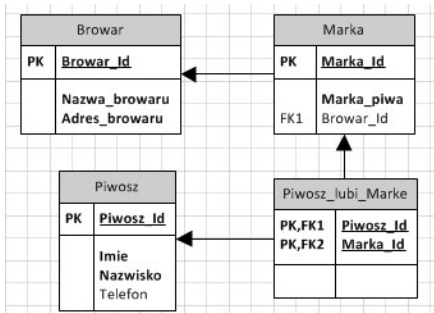
Po utworzeniu encji należy rozważyć i opisać zależności, jakie pomiędzy nimi zachodzą, aby na ich podstawie utworzyć związki encji. Najprostszą jest zależność pomiędzy encjami **Browar** i **Marka**. Możemy opisać ją następująco: browar produkuje kilka (wiele) marek piwa, każda marka powstaje w jednym browarze. Jest to zależność odpowiadająca związkowi jeden – do – wiele (jednoznaczny), gdzie po stronie „jeden” występuje browar, po stronie „wiele” – marka piwa.



Pozostałe związki można opisać następująco:

- Piwosz lubi piwo kilku marek, każda marka piwa ma wielu swoich wielbicieli.

Jest to związek pomiędzy encjami **Piwosz** i **Marka**. Związek jest niejednoznaczny (wiele – do – wiele) i będzie musiał zostać rozłożony na dwa związki jednoznaczne, z nową encją asocjacyjną **Piwosz_lubi_Marke**. Kluczem encji asocjacyjnej jest suma kluczy encji **Piwosz** i **Marka**, czyli dwuelementowy zbiór atrybutów {*Piwosz_Id*, *Marka_Id*}. Ten związek został przedstawiony na poniższym rysunku.



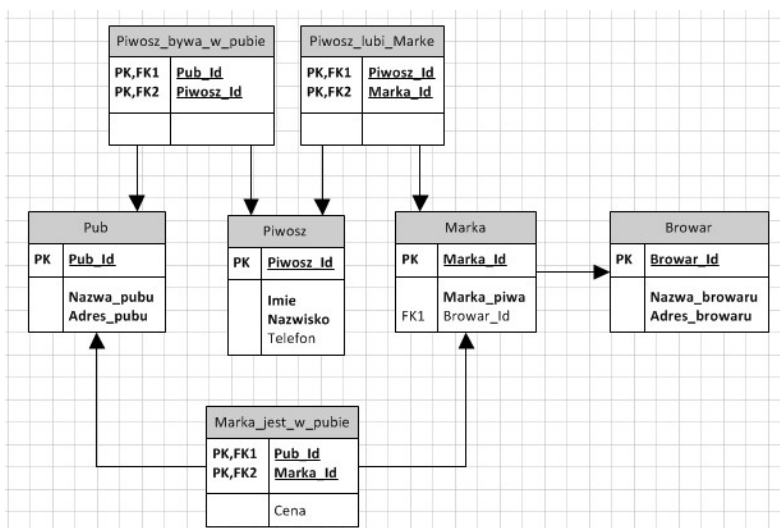
- Piwosz odwiedza kilka ulubionych pubów, a każdy pub ma wielu stałych bywalców.

Jest to związek encji **Piwosz** i **Pub**. Związek jest również niejednoznaczny (wiele – do – wiele) i będzie musiał zostać rozłożony na dwa związki jednoznaczne, z nową encją asocjacyjną **Piwosz_bywa_w_Pubie**.

- W każdym pubie serwowane jest kilka marek piwa, a każda marka jest dostępna w wielu pubach.

Tak jak poprzednie jest to związek niejednoznaczny encji **Pub** i **Marka**, reprezentować go będzie encja **Marka_jest_w_pubie**. Dodatkowym atrybutem tego związku jest *Cena* marki piwa – możemy ją określić tylko dla danej marki w danym pubie.

Cały diagram jest przedstawiony na poniższym rysunku.



Przykład 2 – Biblioteka PJWSTK

Zaprojektuj diagram związków encji dla biblioteki PJWSTK. W bazie będą przechowywane dane osób, książek, wydawnictw. Baza musi zapewnić kontrolę nad wypożyczaniem książek z biblioteki.

Spróbujmy dokładniej zanalizować te pozornie proste wymagania.

Książka, dopóki nie zostanie po raz pierwszy wydana, jest tylko tytułem stworzonym przez autora, lub autorów. Z tytułem możemy związać (poza danymi autorów) rok pierwszego wydania oraz język oryginału.

Osoba może być autorem książki, jej tłumaczem, lub osobą wypożyczającą książkę z biblioteki. W dodatku każda z osób będzie mogła wystąpić w roli autora, tłumacza lub wypożyczającego. Jedna osoba może być autorem wielu książek, jeden tłumacz może dokonać wielu tłumaczeń, jeden wypożyczający może wypożyczać wiele egzemplarzy, a każdy egzemplarz może zostać wypożyczony wielokrotnie, przez różne osoby.

Wydawnictwo to firma, która uzgodniwszy z autorem tytułu warunki, realizuje wydanie. Tytuł może zostać wydany w języku oryginału, albo może zostać przetłumaczony na potrzeby wydania. Każdy tytuł może być wielokrotnie wydawany przez różne wydawnictwa.

W wyniku wydania powstają egzemplarze książek, z których kilka może (nie musi) trafić do naszej biblioteki i zostać oznaczone sygnaturami. Egzemplarz może zostać zapisany na różnych nośnikach (papier, DVD, MP3 itd.) W bazie chcemy przechowywać dane zarówno egzemplarzy, jak i tytułów, których egzemplarze nie są w posiadaniu naszej biblioteki.

Dla ułatwienia wyszukiwania, tytuły zostały podzielone na tematy.

Na zakończenie uwaga. Przyjęty model działania biblioteki nie ma nic wspólnego z rzeczywistym modelem katalogowania wydawnictw, przyjętym, jako standard w bibliotekarstwie. Nasz model pełni rolę czysto dydaktyczną

Model danych. Encje.

Encjami opisującymi obiekty (klasy obiektów) w modelu są:

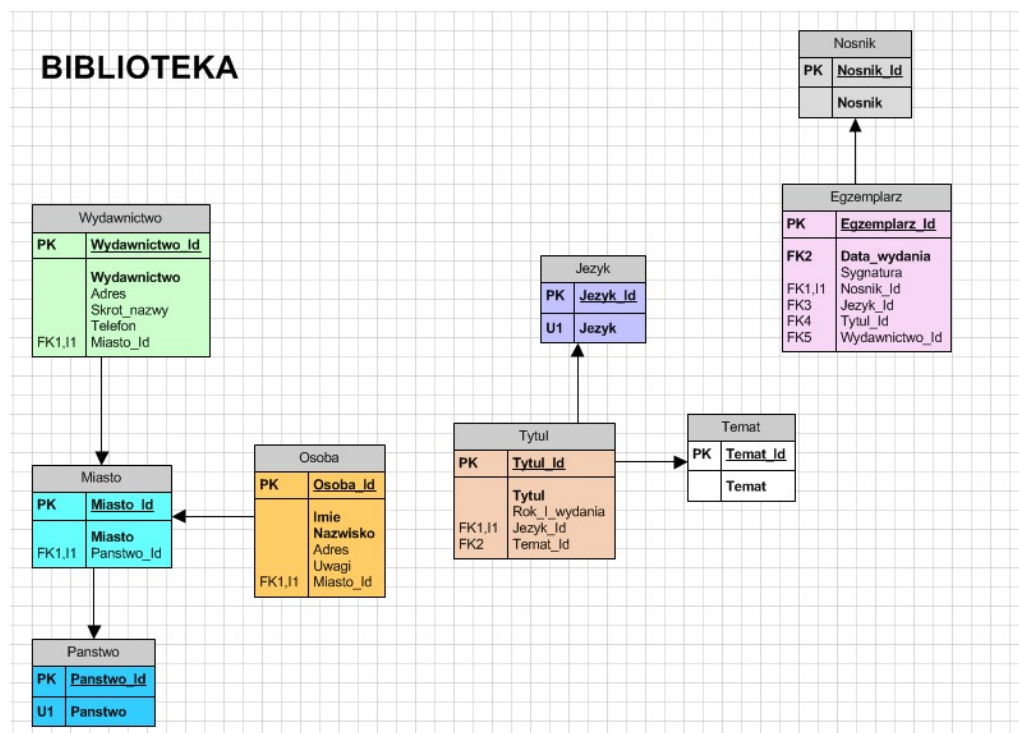
Osoba, Tytuł, Wydawnictwo, Egzemplarz

Dodatkowo wprowadzimy encje słownikowe, porządkujące dane w bazie:

Temat, Miasto, Państwo, Język, Nosnik

Encje słownikowe, to encje modelujące tabele, które będą zawierały stosunkowo niewielkie liczby rekordów, niepodlegających częstym zmianom. Tabele te wprowadza się je do bazy danych w celu uzyskania spójności danych (ograniczenia edycji przez użytkowników) i zachowania kontroli nad danymi.

W efekcie otrzymujemy model zawierający encje opisujące obiekty, encje słownikowe poza oraz ich związki jednoznaczne. Nie pokazano encji modelujących związki niejednoznaczne. Ten diagram został pokazany na kolejnym rysunku.



Drugim etapem pracy będzie werbalne zdefiniowanie związków pomiędzy encjami, oraz ich zamodelowanie. I tak:

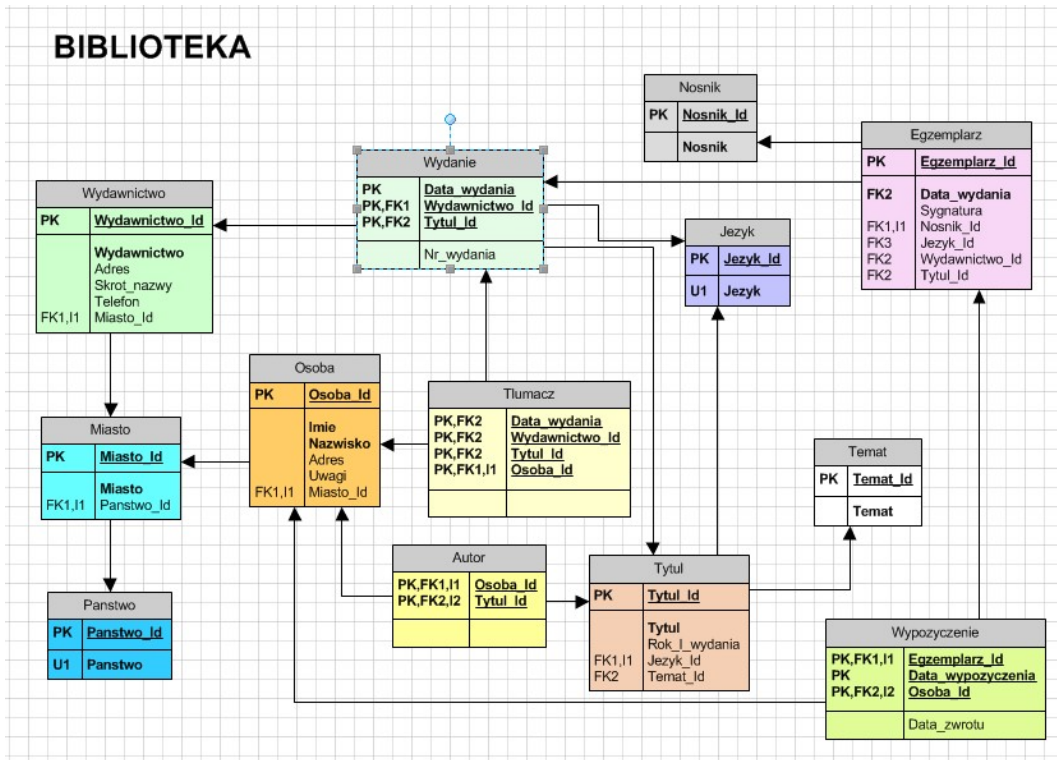
Związek encji **Osoba** i **Tytuł** to autor. Jedna osoba może być autorem kilku (wielu) tytułów, jeden tytuł może mieć kilku autorów (prace zbiorowe). Związek jest niejednoznaczny, wymaga rozłożenia na dwa związki jednoznaczne z encją asocjacyjną **Autor**.

Związek encji **Tytuł** i **Wydawnictwo** to wydanie tytułu przez wydawnictwo, w wyniku którego powstają egzemplarze (wiele egzemplarzy w wyniku jednego wydania). Jedno wydawnictwo wydaje wiele tytułów (niektóre wielokrotnie), jeden tytuł może zostać wydany przez wiele wydawnictw. Związek modeluje encję **Wydanie**. W skład klucza tej encji, zgodnie z definicją, wchodzi klucze encji **Tytuł** i **Wydawnictwo**. Ale z uwagi na fakt, że jedno wydawnictwo może wydać ten sam tytuł wielokrotnie, taki skład klucza głównego nie wystarczy. Należy do niego dodać albo datę wydania, albo numer wydania w danym wydawnictwie. Rozwiązanie używające daty jest zdecydowanie łatwiejsze do zaimplementowania, zatem je zastosujemy.

Związek encji **Osoba** i **Wydanie** opisuje tłumaczenie tekstu na język wydania. Związek jest niejednoznaczny, modeluje go encja **Tłumacz**.

Wreszcie związek encji **Osoba** i **Egzemplarz**, to możliwość wypożyczenia egzemplarza książki przez osobę. Związek niejednoznaczny, w dodatku wymagający dodatkowego atrybutu w kluczu głównym, opisującego datę wypożyczenia (jedna osoba może wielokrotnie wypożyczać ten sam egzemplarz).

Diagram utworzony według powyższego opisu przedstawia



Model bazy, przedstawiony na diagramie, realizuje wszystkie postawione w założeniach wymagania, można na jego podstawie wygenerować bazę danych. Ale przy realizacji aplikacji klienckich ujawni się pewien mankament, który może sprawiać kłopoty implementacyjne, zwłaszcza mniej doświadczonym programistom. Polega on na istnieniu w trzech encjach kluczy „bardzo złożonych” – w encjach **Wypozyczenie** i **Wydanie** z trzech atrybutów, a w encji **Tlumacz** z czterech. Jest to oczywiście rozwiązanie poprawne, jednak może zostać uproszczone, poprzez zamianę kluczy złożonych na sztuczny klucz prosty, w dodatku zrealizowany przez autonumerowanie (Int Identity w MS SQL Server). Ale w celu zachowania istniejących więzów integralności, na atrybutach wchodzących w skład dotychczasowych kluczy głównych należy założyć indeks unikalny (Unique Index).

Taka zamiana klucza, łącznie z utworzeniem unikalnego indeksu na atrybutach klucza, pokazana jest na ostatnim rysunku.

