

# 1 Modelowanie konceptualne

Proces tworzenia bazy danych dla zagadnienia rzeczywistego składa się z kilku etapów. W każdym z nich konstruujemy model bazy danych w innym języku (standardzie) zbliżając się stopniowo do fizycznej implementacji bazy danych.

**Modelowanie konceptualne** to przedstawienie zagadnienia rzeczywistego w postaci schematu (diagramu) obrazującego elementy występujące w rzeczywistości oraz związki pomiędzy nimi. Schemat ten jest zazwyczaj uzupełniany komentarzami słownymi opisującymi zależności pomiędzy danymi, których nie można ująć w postaci graficznej. W ramach modelowania konceptualnego wyróżniamy także role użytkowników w bazie danych, zakres ich zainteresowań i potrzebne im funkcjonalności.

Stosowane systemy zapisu modelu konceptualnego to diagramy E-R, UML oraz ODL. W niniejszych notatkach opisane są diagramy E-R oraz UML. O języku ODL można przeczytać w książce Ullmana i Widom.

**Modelowanie logiczne** wymaga podjęcia decyzji, jakiego rodzaju bazę danych będziemy tworzyć: relacyjną, obiektową, relacyjno-obiektową, sieciową, hierarchiczną itp. Następnie przedstawiamy model konceptualny za pomocą struktur dostępnych w danym rodzaju bazy danych. W przypadku relacyjnych baz danych przedstawienie to polega na zaprojektowaniu tabel przeznaczonych na dane. Kierujemy się przy tym zasadami normalizacji bazy danych, by ograniczyć redundancję i wystąpienia „wartości” NULL w bazie. Następnie możemy dokonać denormalizacji, obniżając w kontrolowany sposób stopień normalizacji bazy, wprowadzając redundancję i wartości NULL. Celem denormalizacji jest ułatwienie wykonywania trudnych i częstych operacji w bazie.

Na tym etapie tworzymy także „podschematy” bazy danych dla ról zaplanowanych dla bazy.

**Modelowanie fizyczne** odbywa się w konkretnym języku definiowania bazy danych.

W przypadku baz relacyjnych polega na utworzeniu tabel, określeniu dziedzin, zdefiniowaniu więzów i wszelkich struktur dodatkowych (indeksów, sekwencji, wyzwalaczy itp.). Na tym etapie definiujemy także perspektywy tworzące podschematy bazy dla poszczególnych użytkowników i określamy ich uprawnienia.

Zgodnie z zapowiedzią w dalszej części notatki zajmiemy się diagramami E-R i UML. Najpierw przedstawimy struktury dostępne w diagramach E-R. Następnie krótko opiszemy analogiczne do nich struktury UML.

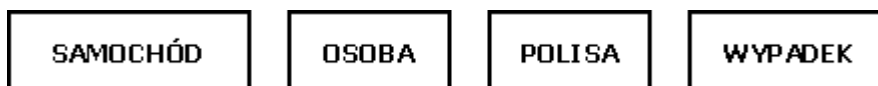
## 2 Diagramy związków encji (E-R)

W niniejszym rozdziale będziemy bazować głównie na przykładzie firmy zajmującej się ubezpieczeniami komunikacyjnymi. Nie będzie to jednak proces budowy kompletnego modelu dla tego zagadnienia. W razie potrzeby będziemy także odwoływać się do innych zagadnień, by lepiej zademonstrować określone elementy diagramów.

Przystąpmy teraz do przedstawienia elementów występujących w diagramach E-R.

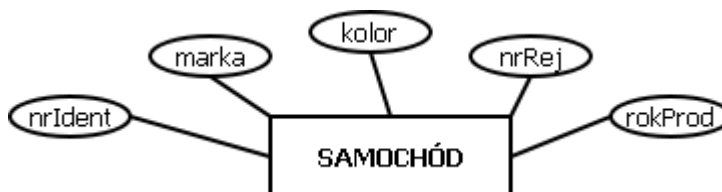
### 2.1 Zbiór encji

Zbiór encji, to zbiór jednorodnych elementów, o skończonej, regularnej strukturze, które można wyróżnić w zagadnieniu rzeczywistym (np. grupa obiektów o takiej samej strukturze i znaczeniu). Zakładamy, że w bazie będziemy mogli przechowywać dowolną liczbę encji danego typu i każdą z nich będziemy opisywać podobnym zestawem atrybutów. Zbiór encji zaznaczamy na diagramie jako prostokąt z nazwą zbioru encji w środku.



### 2.2 Atrybut

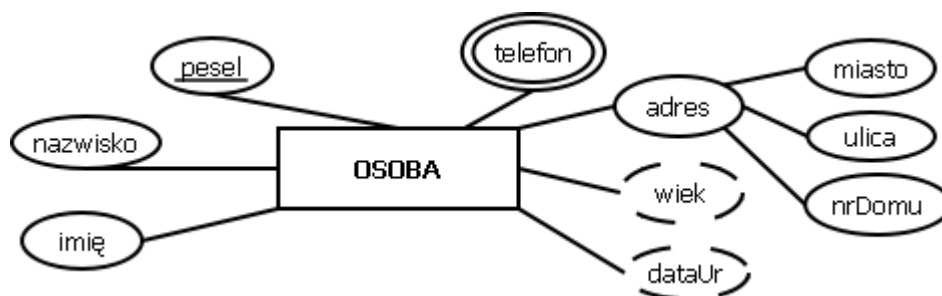
Atrybut to cecha encji. Dla encji należących do tego samego zbioru określamy takie same atrybuty. Atrybutów tych jest ustalona liczba i muszą one w pełni opisywać każdą z encji (nie będziemy pamiętać innych informacji o encjach niż wartości ich atrybutów). Atrybuty zaznaczamy na schemacie w owalach i przyłączamy je do prostokąta ze zbiorem encji, który atrybut opisuje.



**Dziedzina atrybutu.** Dla każdego atrybutu możemy określić dziedzinę. Może być to dziedzina standardowa (liczby, znaki, daty,...), okrojona standardowa, dziedzina wyliczana. Na diagramie możemy dziedzinę podać po nazwie atrybutu lub opisać w komentarzu. Często na tym poziomie pomijamy dziedzinę.

**Rodzaje atrybutów.** Podstawowy rodzaj to atrybuty *proste*. Ich wartości to pojedyncze wartości z dziedziny. Atrybuty te nie mają wewnętrznej struktury i są „udostępniane” w całości. Analizując atrybuty możliwe dla zbioru encji *OSOBA*, jako przykłady atrybutów prostych możemy wymienić nazwisko, imię, pesel. Atrybuty mogą być także

- *złożone*, czyli posiadające strukturę, w której można wyróżnić atrybuty składowe; przykładem atrybutu złożonego może być `adres`, w którym można wyróżnić `miasto`, `ulicę`, `nr_domu`; części składowe atrybutu złożonego zapisujemy w owalnych wierzchołkach połączonych z wierzchołkiem atrybutu złożonego;
- *wielokrotne*, gdy zakładamy, że jedna encja może mieć wiele wartości danego atrybutu, przykładem może być `telefon`, jeśli zakładamy, że każdy może podać ich kilka; atrybuty wielokrotne zapisujemy w owalu zakreślonym podwójną linią; przy atrybucie możemy podać w formie komentarza dostępne i przewidywane liczby powtórzeń;
- *wyliczane*, czyli możliwe do wyznaczenia na podstawie wartości innych atrybutów danej encji lub pozostałych informacji w bazie; należy przyjąć, że wartość takiego atrybutu musi zawsze być aktualna — odpowiadać aktualnej wartości atrybutów, od których jest zależna, np. atrybut `wiek` może być wyliczany na podstawie `daty_urodzenia`, która z kolei może być wyznaczana z numeru `pesel`; atrybut wyliczany wpisujemy w owalu zakreślonym linią przerywaną; formułę wyliczania podajemy w komentarzach, jeśli nie jest oczywista.

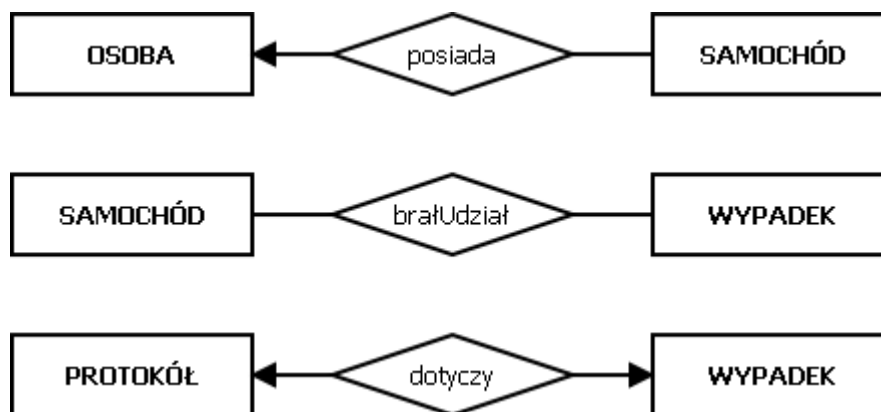


**Klucze zbioru encji.** Klucz to minimalny podzbiór atrybutów zbioru encji, który zawsze będzie miał unikalną wartość dla każdej encji z tego zbioru w bazie (i tym samym zawsze pozwala jednoznacznie zidentyfikować encję po podaniu wartości tychże atrybutów). Zbiór encji może mieć wiele kluczy. Nazywamy je *kluczami kandydującymi*. Tylko jeden z nich wyróżniamy jako *klucz główny* — jego atrybuty nazywamy *atributami głównymi*. Atrybuty główne podkreślamy na diagramie, jak `pesel` na przykładzie. Pozostałe klucze (alternatywne), których unikalność powinna być zagwarantowana w komentarzu do diagramu.

## 2.3 Związki

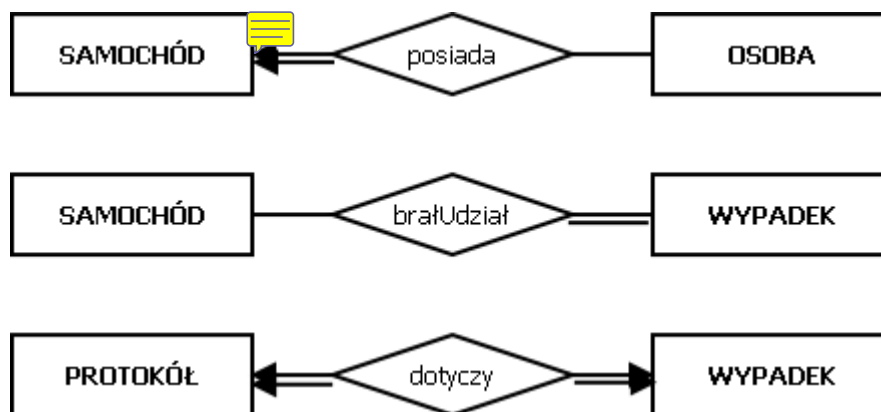
Związek to relacja pomiędzy zbiorami encji. Wyróżniamy związki *wzajemnie jednoznaczne* (1-1 lub jeden-do-jeden), *funkcyjne*, czyli *jednoznaczne* (1-n lub jeden-do-wielu) i *wieloznaczne* (n-m lub wiele-do-wielu). Na poniższym przykładzie pokazany jest sposób przedstawienia każdego rodzaju związku, a przyjęte założenia oznaczają, że:

- *związek funkcyjny* — samochód jest własnością najwyżej jednej osoby, ale jedna osoba może mieć wiele samochodów (lub żadnego);
- *związek wieloznaczny* — samochód może brać udział w wielu wypadkach i w jednym wypadku może brać udział wiele samochodów;
- *związek wzajemnie jednoznaczny* — protokół dotyczy jednego wypadku i dla wypadku jest sporządzany jeden protokół.

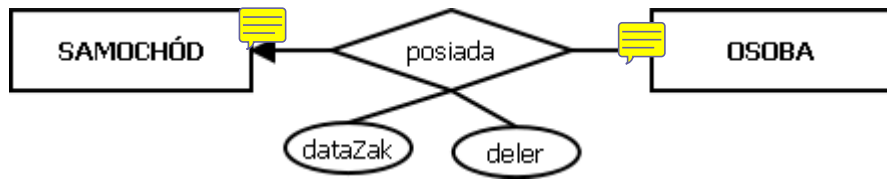


**Wymuszenie związku.** Związek łączący zbiory encji  $E$  i  $F$  oznacza, że każda encja ze zbioru  $E$  *może*, ale nie musi wystąpić w związku z pewną encją zbioru  $F$  (i odwrotnie). Jeśli chcemy zaznaczyć, że każda encja ze zbioru  $E$  *musi* być w związku z pewną encją zbioru  $F$ . Wówczas na diagramie łączymy prostokąt encji  $E$  z rombem związku podwójną linią. W poniższych przykładach w ten sposób zaznaczyliśmy, że:

- *związek funkcyjny* — samochód musi mieć właściciela, ale osoba z bazy nie musi mieć samochodu;;
- *związek wieloznaczny* — samochód nie musi brać udziału w jakimkolwiek wypadku, ale w wypadku musiał brać udział przynajmniej jeden saochód;
- *związek wzajemnie jednoznaczny* — protokół musi dotyczyć wypadku i dla wypadku musi być sporządzony protokół.



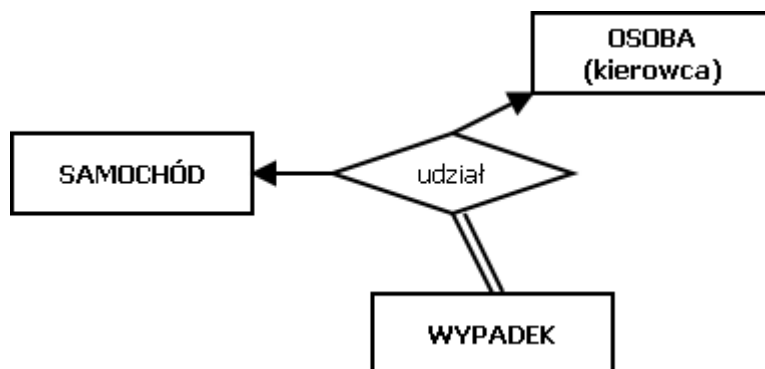
**Atrybuty związku.** Związek może mieć swoje atrybuty. Zaznaczamy je jako owalne wierzchołki połączone rombem związku. Na poniższym przykładzie widzimy dodatkowe informacje o nabyciu samochodu przy związku pomiędzy osobą i samochodem.



**Związki między wieloma zbiorami encji.** Możliwe są związki pomiędzy więcej niż dwoma zbiorami encji (związki 3-arne, 4-arne i ogólnie  $k$ -arne). Zaznaczamy je podłączając prostokąty wielu zbiorów encji do jednego rombu związku.

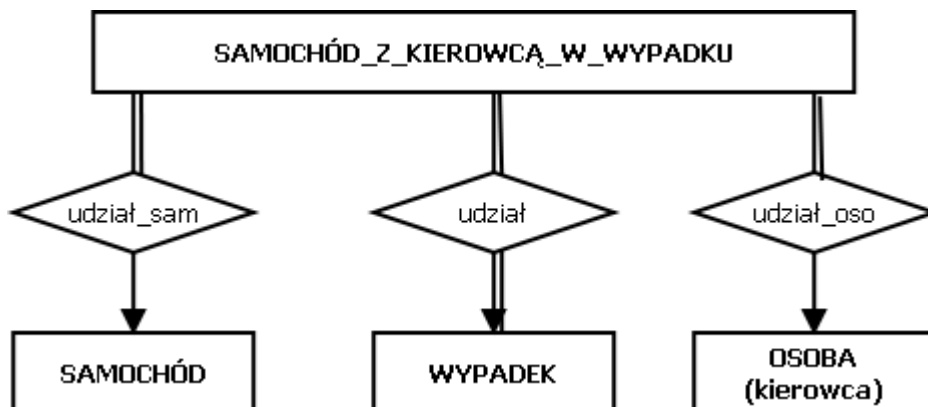
W związku  $k$ -arnym możemy także korzystać ze strzałek do zaznaczania jednoznaczności związku i podwójnych linii do zaznaczania wymuszenia. Trzeba jednak zachować przy tym szczególną ostrożność, ponieważ interpretacja jest następująca:

- w związku pomiędzy  $E_1, E_2, \dots, E_k$ , zbiór encji  $E_i$  wskazujemy strzałką jako wyznaczony jednoznacznie, jeżeli przy ustalonym wyborze pozostałych encji  $\mathcal{A} = \{e_j \in E_j \mid 1 \leq j \leq k, j \neq i\}$  istnieje najwyżej jedna encja  $e_i \in E_i$  będąca z związku z encjami ze zbioru  $\mathcal{A}$ ; w poniższym przykładzie oznacza to, że dla samochodu uczestniczącego w wypadku jest ustalony najwyżej jeden kierowca tego samochodu i osoba uczestnicząca w wypadku mogła kierować najwyżej jednym samochodem w tym wypadku;
- w związku pomiędzy  $E_1, E_2, \dots, E_k$ , zbiór encji  $E_i$  łączymy z rombem związku podwójną linią, jeżeli dla każdej wartości  $e_i \in E_i$  musi istnieć przynajmniej jeden zbiór wartości  $\mathcal{A} = \{e_j \in E_j \mid 1 \leq j \leq k, j \neq i\}$  będący w związku z  $e_i$ ; na diagramie zaznaczono, że w wypadku musiał uczestniczyć przynajmniej jeden samochód z kierowcą.

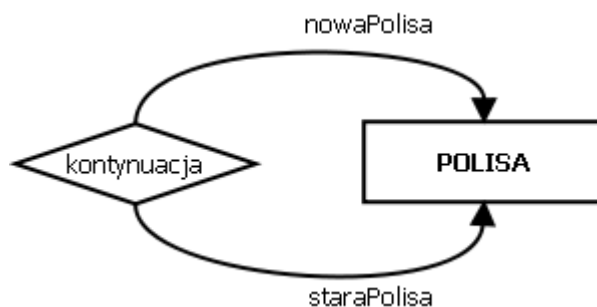


**Alternatywne przedstawienia.** Tę samą sytuację można wyrazić jako zbiór encji lub jako związek. Reprezentację w postaci zbioru encji wybieramy zazwyczaj, gdy związek ma charakter trwały, jest wielokrotny i ma skomplikowane zależności funkcyjne

i wymuszenia, ma atrybuty. Poniżej widzimy przedstawienie udziału w wypadku w postaci zbioru encji (z wymuszeniami i jednoznacznością, jak na ostatnim diagramie dla tego przykładu).



**Związki rekurencyjne.** Związki rekurencyjne łączą encje z tego samego zbioru. Definiując taki związek możemy określić rolę encji w związku, by mieć możliwość wyróżnienia każdej encji z pary powiązanej związkiem. Na diagramie nazwę roli umieszczamy przy linii łączącej zbiór encji z rombem oznaczającym związek.

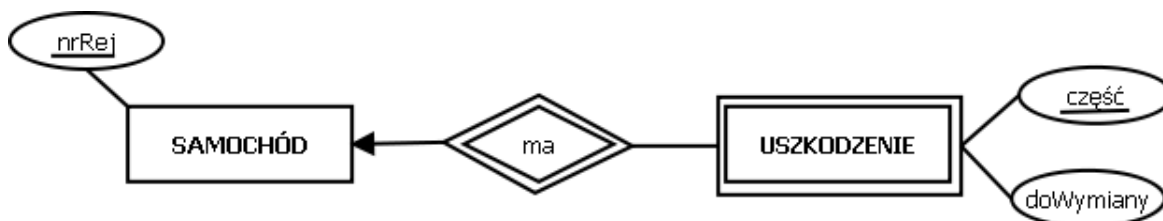


## 2.4 Słabe zbiory encji

Dotychczas tworząc zbiór encji zakładaliśmy, że elementy tego zbioru są podobne (ze względu na strukturę), ale rozróżnialne (ze względu na wartości atrybutów). Innymi słowy, spośród atrybutów zbioru encji zawsze można wyłonić klucz. Słaby zbiór encji nie musi posiadać klucza, a jego elementy mogą być identyczne. Słaby zbiór encji musi być natomiast powiązany związkiem funkcyjnym z pewnym nadrzędnym dla niego zbiorem encji (lub kilkoma zbiorami). Elementy słabego zbioru encji muszą być rozróżnialne lokalnie, to znaczy w grupie encji przyporządkowanych do tej samej encji nadrzędnej.

Słabe zbiory encji nie posiadają samodzielnie klucza — ich klucze składają się z kluczy nadrzędnych zbiorów encji i ich własnych atrybutów kluczowych. Prostokąty oznaczające słaby zbiór encji rysujemy na diagramie podwójną linią, podobnie jak romby związków funkcyjnych łączących je z nadrzędnym zbiorem (zbiorami) encji.

Na poniższym przykładzie widzimy zbiór encji **USZKODZENIE**, które opisują szczegółowe uszkodzenia samochodu. Globalnie są one powtarzalne, ale w ramach jednego auta są unikalne.

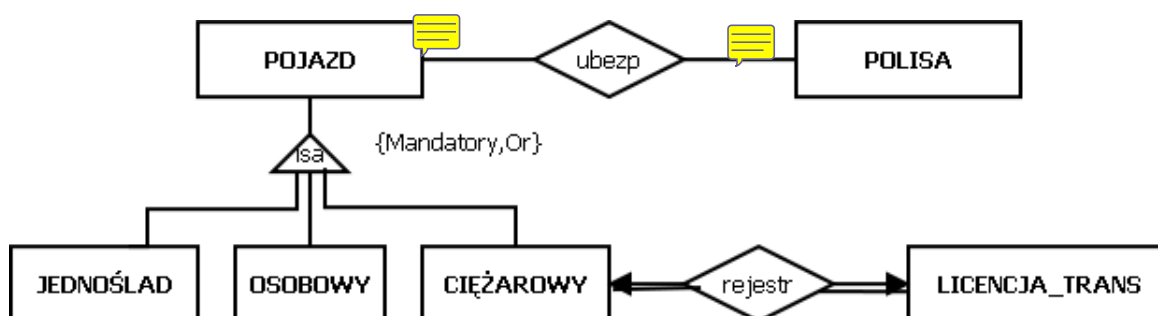


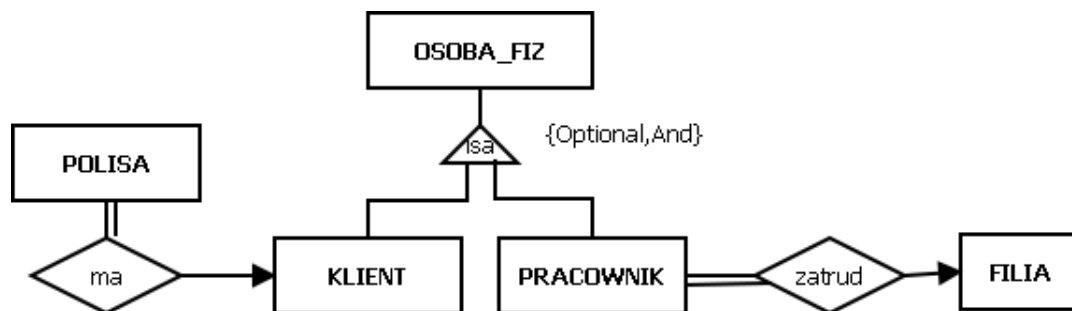
## 2.5 Związki typu klasa-podklasa

**Związki hierarchiczne („isa”).** Jeżeli jeden zbiór encji stanowi podklasę innego zbioru encji, to możemy to zaznaczyć wprowadzając związek hierarchiczny. Encja podrzędna ma wówczas wszystkie cechy (atrybuty i związki) encji nadrzędnej i ewentualnie swoje własne specyficzne cechy (może mieć dodatkowe atrybuty i/lub być w dodatkowych związkach). W związkach hierarchicznych możemy zaznaczyć:

- czy encja klasy nadrzędnej **musi** należeć do co najmniej jednej z klas podrzędnych (komentarz *Mandatory*) lub może nie należeć do żadnej (komentarz *Optional*); można to także zaznaczyć na diagramie E-R łącząc encję nadrzędną z symbolem związku linią podwójną oznaczającą wymuszenie;
- czy encja klasy nadrzędnej może należeć do wielu klas podrzędnych (komentarz *Or*) czy też najwyżej do jednej (komentarz *And*).

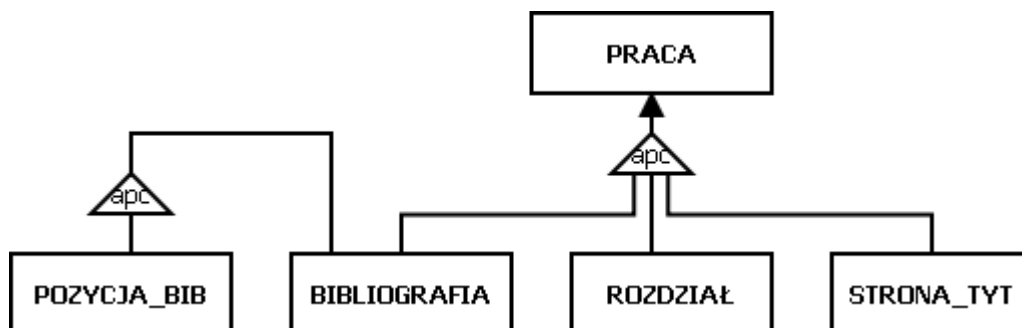
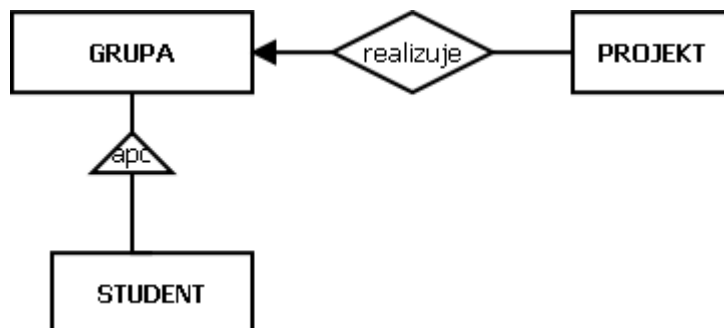
Związek hierarchiczny zaznaczamy wierzchołkiem z trójkątem skierowanym wierzchołkiem w stronę klasy nadrzędnej i podstawą w stronę klas podrzędnych. Na pierwszym z poniższych przykładów widzimy podział pojazdów ze względu na kategorię. W drugim przykładzie widzimy podział osób, ze względu na zatrudnienie w firmie. W pierwszym przykładzie podział jest wymuszony i rozłączny. W drugim przykładzie — dowolny i nierozłączny.





**Agregacja i kompozycja („apo”).** Związek „apo” pomiędzy nadrzędnym zbiorem encji  $E$  i podrzędnym zbiorem encji  $F$  oznacza, że encja  $F$  jest częścią składową encji  $E$ . Związek ten oznaczamy analogicznie jak poprzedni, tylko umieszczamy w trójkącie skrót APO (od *a part of*). W przypadku agregacji zakładamy, że element klasy podrzędnej może być związany z wieloma encjami klasy nadrzędnej. W przypadku kompozycji zakładamy, że jest integralną częścią składową tylko jednej encji klasy nadrzędnej. Rozróżnienie pomiędzy tymi rodzajami związków jest lepiej widoczne na diagramach UML. W notacji E-R możemy je zaznaczyć łącząc symbol związku z klasą nadrzędną strzałką oznaczającą związek funkcyjny.

Na poniższych przykładach widzimy najpierw przykład agregacji — studenci są członkami grup, a grupy realizują projekty. Student może należeć do wielu grup. W drugim przykładzie widzimy kompozycję — praca składa się z określonych elementów składowych, które mogą należeć tylko do niej (kwestię przynależenia pozycji bibliografii do pracy przedyskutowaliśmy na wykładzie).





## 2.6 Sytuacje trudne

Do zrobienia...

## 2.7 Dodatkowe więzy integralności

Dodatkowe więzy integralności, to warunki, których nie da się zilustrować na diagramie, a niosą istotną wiedzę o zagadnieniu rzeczywistym. Dodanie ich w postaci komentarza do diagramu oznacza, że stworzona baza danych i związana z nią aplikacja powinny gwarantować zachowanie tych warunków. Można zaproponować następujące warunki dla bazy danych ubezpieczeń komunikacyjnych:

- procent odpowiedzialności za wypadek musi sumować się do  $\leq 100$ ;
- do polis obejmujących ubezpieczenie AC, OC i NW dochodzi zawsze ZIELONA KARTA;
- data wypadku musi zawierać się w okresie ważności polisy;
- protokół dla szkody powyżej 10 tys. zł może sporządzić tylko agent z trzyletnim stażem.

## 2.8 Opis ról i podschematów

Musimy określić jakiego typu użytkowników bazy danych przewidujemy (*role*), np. *agent ubezpieczeniowy*, *manager*, *policja*, *klient*. Następnie dla każdej z tych ról należy określić do jakich danych i jaki dostęp (do odczytu, prawo modyfikacji, prawo usuwania i dopisywania) powinien mieć określony użytkownik. Ponadto, często warto opisać aplikacje wymagane dla użytkowników każdego rodzaju, zwłaszcza, jeżeli będą to aplikacje nietypowe. Przykładem może być:

- podgląd historii szkód refundowanych danemu użytkownikowi i wyznaczanie na tej podstawie upustu w ubezpieczeniu;
- analiza szkód samochodu wspomagana rozpisaniem samochodu „na części” i katalogowe ceny tychże części.

## 3 UML (Unified Modeling Language)

W niniejszym rozdziale jest zawarty treściwy przegląd notacji UML pozwalającej przedstawić informacje o zagadnieniu rzeczywistym analogiczną do informacji zawartej w diagramach E-R.

## Zbiór encji i jego atrybuty

Nazwa	← nazwa zbioru encji
atrybut 1{PK}	← atrybut główny (klucza głównego)
atrybut 2	← “zwykły” atrybut
atrybut 3	← atrybut złożony
atrybut 3a	i jego atrybuty składowe
atrybut 3a	
atrybut 3a	
atrybut 4 [k..l]	← atrybut wielokrotny (od k do l wartości)
\ atrybut 5	← atrybut wyliczany

Jeżeli na pewnym etapie projektowania modelu konceptualnego nie określamy jeszcze wszystkich atrybutów zbioru encji, to w wykazie atrybutów podajemy tylko atrybuty klucza głównego; nie musimy ich wówczas oznaczać {PK}. Atrybuty składające się na inny klucz możemy oznaczyć etykietą {AK}.

Słaby zbiór encji zaznaczamy jak zwykły, ale przy żadnym z atrybutów nie umieszczamy oznaczenia klucza głównego.

## Związki

Encja 1	związek ▷	Encja 2
atrybut 1	$k_1 : l_1$	atrybut 2
	$k_2 : l_2$	

Kierunek strzałki przy nazwie związku nie ma większego znaczenia – pokazuje jedynie jak należy czytać w języku potocznym znaczenie związku: “encja 1 jest w związku z encja 2”, czy odwrotnie. Określenia krotności,  $k_1, l_1, k_2, l_2$ , mogą być liczbami naturalnymi (począwszy od 0) lub symbolem \* oznaczającym nieskończoność. Znaczenie tego określenia jest następujące: każda encja typu Encja 2 **musi** występować w związku z liczbą encji typu Encja 1 mieszczącą się w przedziale  $k_1..l_1$ .

Chcąc reprezentować związki  $k$ -arne, dla  $k > 2$ , dodajemy na linii romb z nazwą związku (podobnie jak w diagramach Chena). Przy związkach rekurencyjnych możemy umieścić przy każdym ze zbiorów encji nazwę roli, jaką pełni on w związku (także podobnie, jak w diagramach Chena).

Atrybuty związku umieszczamy w prostokącie analogicznym do opisującego zbiór encji, ale z pustym polem nazwy zbioru encji, i łączymy linią przerywaną z odcinkiem (lub rombem) opisującym związek.

**Związek hierarchiczny** (patrz związek pomiędzy *Właściciel* oraz *Właściciel prywatny* i *Właściciel instytucjonalny* na rysunku niżej)

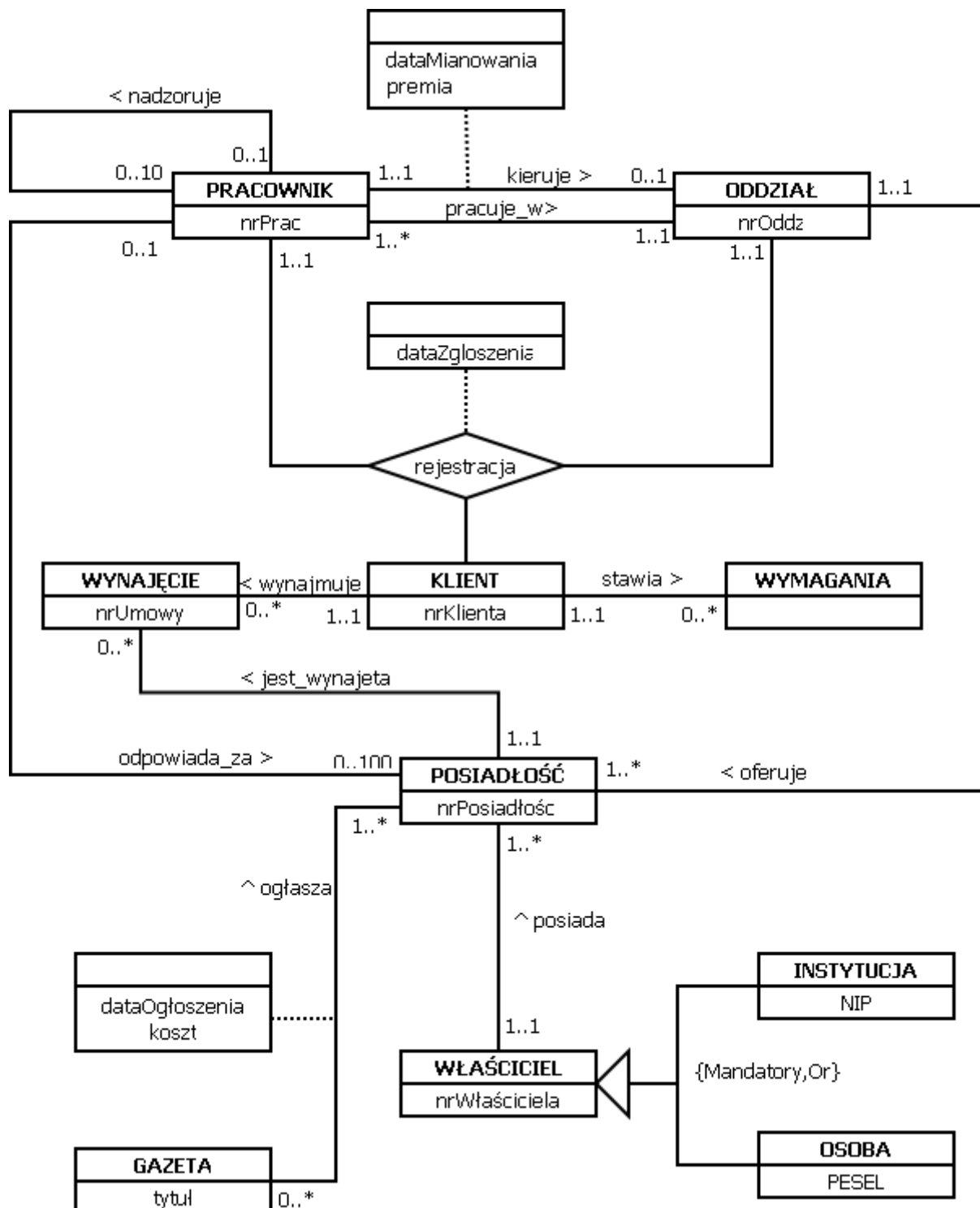
Komentarz umieszczony przy związku może brzmieć:

1. *Mandatory* – oznacza, że każda encja z nadklasy musi wystąpić także w (pewnej) podklasie;

2. *Optional* – jest “przeciwieństwem” poprzedniej opcji i oznacza, że każda encja nadklasy może, ale nie musi, wystąpić w jednej z podklas;
3. *Or* – oznacza, że podklasy są rozłączne i każda encja nadklasy może trafić najwyżej do jednej z podklas;
4. *And* – oznacza, że podklasy nie muszą być rozłączne i każda encja z nadklasy może trafić do dowolnej liczby podklas.

**Agregacja i kompozycja** analogiczne do związku APO z diagramów Chena (E-R) są przedstawiane za pomocą linii łączącej prostokąty zbiorów encji. Przy zbiorze nadrzędnym linia jest zakończona rombem: pustym dla agregacji i zamalowanym dla kompozycji. Związku kompozycji używamy, gdy części i całość praktycznie nie istnieją oddzielnie i jednocześnie są tworzone i usuwane. Jeżeli części mogą istnieć niezależnie od całości, to stosujemy związek agregacji.

### Przykład<sup>1</sup>



---

<sup>1</sup>na podstawie T.Connolly, C.Begg