

Diagramy ERD.

Model struktury danych jest najczęściej tworzony z wykorzystaniem diagramów pojęciowych (konceptualnych). Najpopularniejszym konceptualnym modelem danych jest tzw. model związków encji (ERM - entity relationship model), którego graficznym odpowiednikiem jest diagram związków encji (ERD - entity relationship diagram). Diagram ten spotyka się w różnych notacjach, do których zaliczamy m.in. notacje Chena, Martina, Bachmana, IDEF1X. Diagram ERD to rodzaj graficznego przedstawienia związków pomiędzy encjami używany w projektowaniu systemów informacyjnych. ERD pozwala na zrozumienie struktury danych, przygotowania późniejszej strategii optymalizacji bazy oraz stanowi podstawową dokumentację systemu przechowywania informacji. ERD jest częścią wielu popularnych narzędzi CASE (Computer Aided Software Engineering), pozwalających na wygenerowanie fizycznej struktury bazodanowej na podstawie zaprojektowanego diagramu. Systemy CASE, które wspierają tworzenia tych diagramów, mogą na ich podstawie automatycznie tworzyć bazy danych odpowiadające relacjom na diagramie. Przykłady diagramów ERD w różnych notacjach:

Różne rodzaje notacji

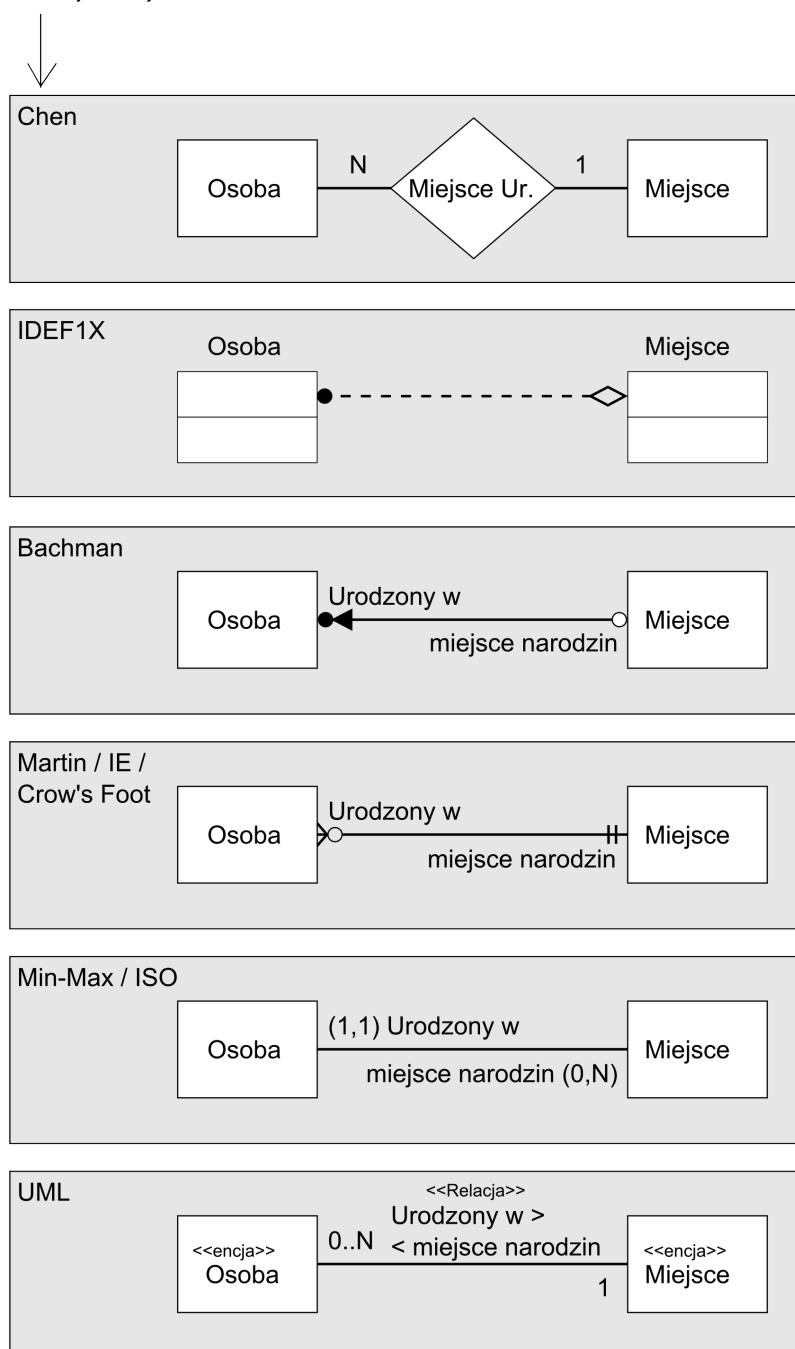
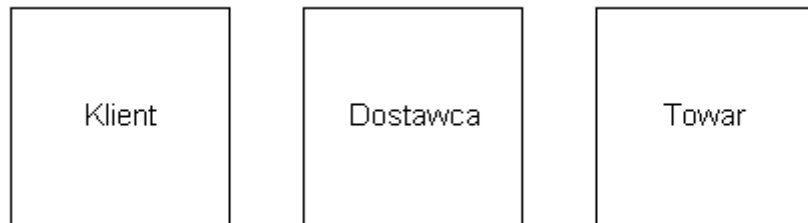


Diagram ERD przedstawia:

- obiekty, o których informacje są istotne z punktu widzenia realizacji celów strategicznych firmy;
- atrybuty obiektów;
- związki pomiędzy obiektami.

Wyodrębnione obiekty mogą być rzeczywiste lub mogą być pojęciami abstrakcyjnymi. Obiekty mające te same atrybuty łączy się w typy obiektów np. Towar, Klient, Dostawca, Zamówienie.

ENCJA jest rzeczą lub obiektem mającym dla nas znaczenie, rzeczywistym bądź wyobrażonym, o którym informacje muszą być znane lub przechowywane.



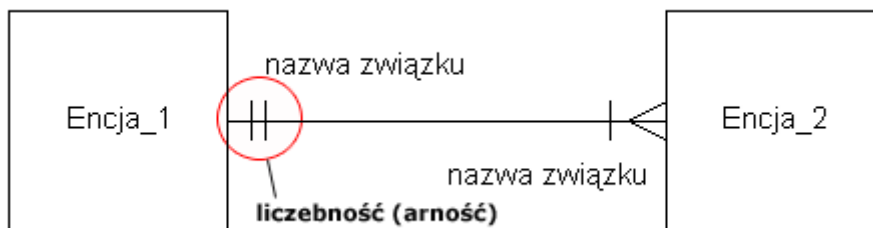
Graficzną reprezentacją ENCJI jest prostokąt z nazwą ENCJI zapisaną w liczbie pojedynczej.

ZWIĄZEK jest nazwanym, istotnym powiązaniem pomiędzy dwiema encjami. Związki przedstawiają zależności zachodzące pomiędzy obiektami.

Każdy związek ma dwa końce, z których każdy ma przypisane następujące atrybuty.

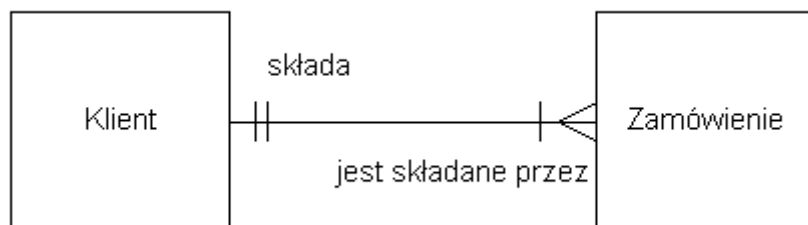
- nazwę;
- liczebność (jak wiele);
- opcjonalność (opcjonalny czy wymagany).

Związek jest reprezentowany za pomocą linii łączącej dwie encje.

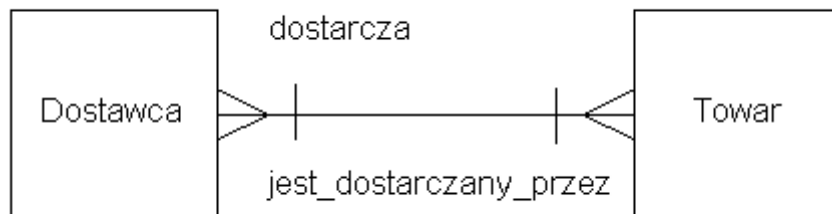


Na powyższym schemacie jest przedstawiony najczęściej występujący związek jeden-do-wiele.

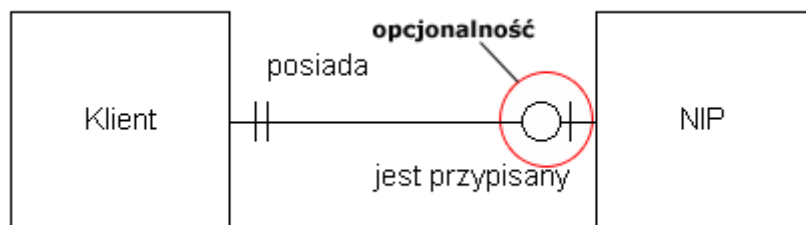
KLIENT składa **ZAMÓWIENIE**
Arność: Jeden do Jeden-lub-Wiele



DOSTAWCA dostarcza TOWAR
Arność: Wiele do Wiele



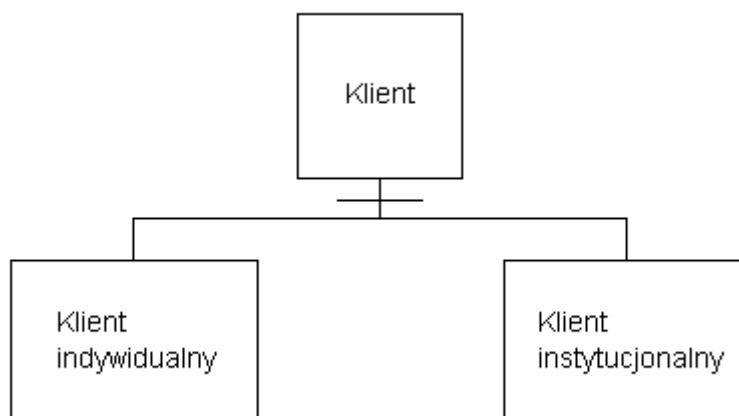
KLIENT posiada NIP
Arność: Jeden do Jeden (opcjonalnie)



Związek NADTYP-PODTYP

Związek nadtyp-podtyp obejmuje zwykle typ obiektu oraz jedną lub więcej podkategorii. Atrybuty nadtypu odnoszą się do wszystkich podtypów, natomiast atrybuty podtypów są unikalne w zakresie podtypu.

Poniżej znajduje się przykład związku nadtyp-podtyp opisujący klienta w rozbiciu na klienta indywidualnego oraz instytucjonalnego.



ATRYBUT jest dowolnym szczegółem służącym do kwalifikowania, identyfikowania, klasyfikowania, określania ilości lub wyrażania stanu ENCJI.

Lub

ATRYBUT jest dowolnym opisem mającym znaczenie dla ENCJI. **ATRYBUT** może być tekstem, liczbą, wartością logiczną lub obrazem.



Przykład atrybutów ENCJI klient.

Zasady tworzenia diagramów ERD

Poniżej przedstawiona jest metoda tworzenia diagramów ERD jako proces składający się z pięciu kroków:

- **Identyfikacja zbioru obiektów, o których informacje są istotne;**

Identyfikację obiektów rozpoczynamy od wyodrębnienia kluczy głównych poszczególnych obiektów, czyli identyfikatorów jednoznacznie identyfikujących obiekt. Wyróżnione identyfikatory pozwalają na uformowanie wstępnej listy obiektów;

- **Określenie związków zachodzących pomiędzy obiektami oraz rodzaju związków;**

Identyfikacja bezpośrednich zależności pomiędzy obiektami, które odzwierciedlają pewne reguły przyjęte w firmie. Należy zbadać istnienie bezpośrednich powiązań między wszystkimi parami obiektów w systemie;

- **Identyfikacja atrybutów obiektów;**

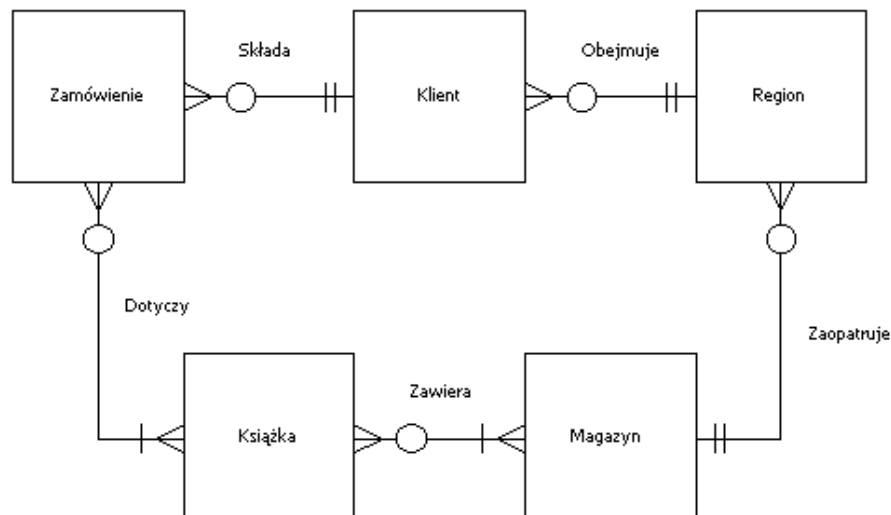
Po wyodrębnieniu istotnych obiektów oraz związków zachodzących pomiędzy nimi można uzupełnić opis obiektów identyfikując potrzebne atrybuty tzn. dane istotne dla firmy opisujące konkretny obiekt;

- **Stworzenie wstępnego modelu pojęciowego;**
- **Sprawdzenie poprawności otrzymanej struktury.**

Na podstawie zidentyfikowanych obiektów oraz powiązań między nimi możemy narysować wstępny diagram związków encji.

PRZYKŁAD

Weryfikacja



- **WERYFIKACJA POD KĄTEM WYMAGAŃ UŻYTKOWNIKA**

Otrzymany na podstawie opisu organizacji model należy zweryfikować, pod kątem spełnienia wymagań użytkownika. Jest to oczywiście wstępna weryfikacja, gdyż na etapie analizy próbujemy formalnie naszkicować część modelowanej rzeczywistości, natomiast podczas etapu projektowania modele te zostaną rozwinięte i uszczegółowione.

Jeżeli utworzony model danych nie spełnia wymagań użytkownika, to należy uzupełnić go o dodatkowe obiekty i powiązania.

- **WERYFIKACJA POD KĄTEM SPÓJNOŚCI PERSPEKTYW PRZEDSIĘBIORSTWA**

Jeżeli modelujemy kilka fragmentów firmy, zawarcie wszystkiego w jednym diagramie jest właściwie niemożliwe. Nawet w przypadku stworzenia takiego diagramu jest on nieczytelny i trudny do analizy. W takich przypadkach można zastosować rozbicie wg perspektyw przedsiębiorstwa. Perspektywa przedsiębiorstwa jest to zbiór obiektów zawartych w polu zainteresowań wybranej grupy pracowników firmy. Oczywiście perspektywy mogą się częściowo nakładać. Wynika to z faktu, że różne grupy pracowników mogą korzystać z podobnych informacji.

Projektowanie baz danych za pomocą narzędzi CASE

Metody tworzenia systemów informatycznych w tym, także rozbudowanych baz danych są komputerowo wspomagane przez narzędzia CASE (ang. Computer Aided Software Engineering).

Dzięki narzędziom CASE, proces tworzenia systemu staje się o wiele krótszy pociągając za sobą mniejsze koszty. Ponadto narzędzia CASE zapewniają wyższą jakość tworzonego systemu.

Narzędzia CASE zawierają w sobie inżynierię dwustronną: inżynieria do przodu oraz inżynieria wstecz. Inżynieria do przodu jest niczym innym jak generacją kodu na podstawie modelu.

Inżynieria wstecz to proces analizy oprogramowania, którego celem jest odtworzenie projektu i specyfikacji. W tym procesie program nie ulega zmianie. Kod źródłowy oprogramowania jest zwykle dostępny i stanowi daną wejściową dla procesu inżynierii wstecz. Celem inżynierii wstecz jest określenie projektu i specyfikacji systemu na podstawie kodu źródłowego.

Przykładem tego typu narzędzia jest program DBDesigner4. Jest to system wizualnego projektowania, modelowania i tworzenia baz danych. Program jest w całości darmowy i można go pobrać ze strony internetowej: <http://fabforce.net/index.php>. DBDesigner4 jest w wersji zarówno dla systemu Windows jak i Linux.