

Roman BOJARSKI  
Politechnika Śląska  
Wydział Organizacji i Zarządzania  
Instytut Ekonomii i Informatyki

## **MODELE ZWIĄZKÓW ENCJI W BAZACH DANYCH SYSTEMÓW WSPOMAGAJĄCYCH ZARZĄDZANIE WYTWARZANIEM**

**Streszczenie.** W artykule przedstawiono klasyfikację związków encji służących do modelowania konceptualnego predefiniowanych struktur danych w informatycznych bazach danych. Klasyfikacji tych związków dokonano poprzez wprowadzenie pojęcia stopnia związku. Na bazie związków pierwszego, drugiego i wyższych stopni odwołano się do przykładów mających swoje odniesienie w bazach danych systemów wytwarzania. Wskazano zarówno na składnię odczytywania tych związków w języku naturalnym jak i semantykę modeli związków.

## **ENTITY RELATIONSHIP MODELS IN DATABASES OF *IT* MANUFACTURING MANAGEMENT SYSTEMS**

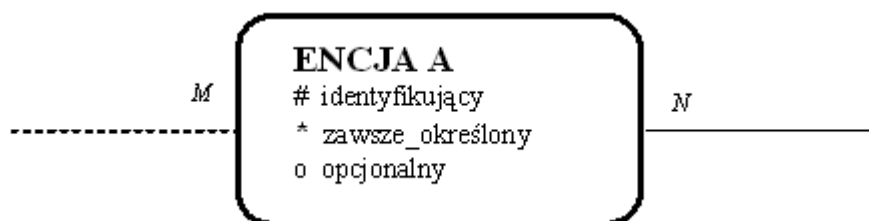
**Summary.** The paper presents classification of entity relationship diagrams used in conceptual modelling of data structures in databases of manufacturing management systems. Classification is based on the idea of a relationship degree. On that concept 1<sup>st</sup>, 2<sup>nd</sup> and higher degree entity relationships have been presented as well as their examples encountered in manufacturing IT systems. An attention has been brought to the formal interpretation of the relationships as well as to their semantics. At the end of the paper semistructured data have been discussed within the context of entity relationships.

## 1. Wstęp

Wdrażanie systemów informatycznych zarządzania jest nierozłącznie związane z zagadnieniami modelowania. Szczególne miejsce w tych systemach zajmują bazy danych, które gromadzą niezbędne informacje wykorzystywane we wszelkich procesach zarządzania obejmujących zarówno poziom operacyjny jak i taktyczny zarządzania a także są pomocne w podejmowaniu decyzji na poziomie strategicznym przedsiębiorstw. Modelujemy nie tylko po to aby wyrazić architekturę bądź strukturę systemu, ale także po to aby ogarnąć i zrozumieć złożoność wdrażanego systemu a następnie na etapie eksploatacji móc dokonywać niezbędnych modyfikacji dostosowawczych wynikających ze zmian środowiska, w którym funkcjonuje system informatyczny. Dzięki uniwersalnym narzędziom do których należy zaliczyć standard języka SQL (ang. *Structured Query Language*) dokonywanie tych zmian przez użytkowników wdrożonych systemów zarządzania stało się powszechną praktyką, stąd szczególnego znaczenie nabiera modelowanie baz danych.

## 2. Encje oraz związki encji 1. i 2. stopnia

Na temat encji ukazało się sporo publikacji, które w różny sposób prezentują to pojęcie; również nie ma ujednoczonego sposobu interpretowania linii związków [1], [2], [5]. Na potrzeby niniejszych rozważań **encja** jest obiektem służącym do reprezentowania typu bądź klasy przedmiotów charakteryzowanych za pomocą określonych cech a nie do reprezentowania jej konkretnych instancji (wystąpienia). Encja przedstawiana jest w postaci graficznej ramki i służy do opisu danych i związków między danymi. Zgodnie z konwencją Barkera [1], poniżej nazwy encji zapisywane są nazwy atrybutów i poprzedzone, jak na rys. 1, odpowiednimi symbolami służącymi do ich kwalifikacji (identyfikujący, zawsze określony, opcjonalny). Litera  $M$  i  $N$  oznaczają liczby naturalne i definiują liczebność (liczność) związku.

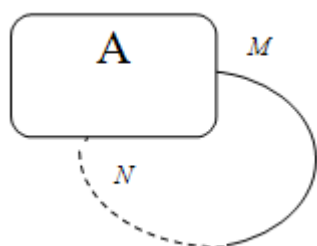


Rys. 1. Prezentacja Encji wraz z powiązaniem i liczebnością związku  
 Fig. 1. Diagram of entity with its relations and their multiplicity

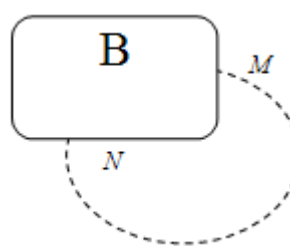
**Związek** między encjami reprezentuje łączące je powiązanie, mające postać linii przerywanej (opcjonalność związku) bądź linii ciągłej (wymuszenie związku). Z kolei **stopień związku** encji to liczba encji wchodząca w skład tego związku.

## 2.1. Związki 1. stopnia. Rekurencja

Związek 1. stopnia jest powiązaniem rekurencyjnym. Związek ten wprowadza relację porządkującą wśród danych reprezentowanych przez encję.



Rys.2 Związek typu: wymagane - opcjonalne  
Fig. 2. Relationship required-optional.



Rys.3. Związek typu: opcjonalne – opcjonalne  
Fig. 3. Relationship optional-optional.

Związek, który w części bądź całości ma linię ciągłą nie odzwierciedla rzeczywistości bowiem określa pętlę bez elementu najwyższego bądź najniższego. Związek z rys. 3 jest często stosowany (relacja typu  $M:N$ ) i służy do modelowania struktury sieciowej danych. Taki związek można przykładowo odczytać w sposób następujący: *Każda składowa* (element encji B) *może być złożona z jednej bądź więcej* (innych) *składowych* oraz *każda składowa może być użyta do złożenia jednej bądź więcej* (innych) *składowych*. Tego typu związek może posłużyć do reprezentowania grafu Gozinto [2] struktur wyrobów w systemach informatycznych wytwarzania [2].

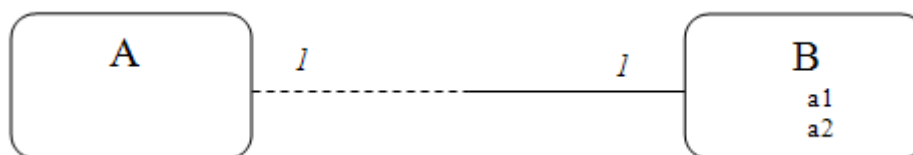
Dla przypadku gdy jedna ze zmiennych  $M$ ,  $N$  przyjmie wartość 1 (związek typu  $1:M$  bądź  $1:N$ ) taka relacja służy do reprezentowania prostej wielopoziomowej hierarchii i może być używana do modelowania struktur organizacyjnych przedsiębiorstw a także do prezentowania struktur wyrobów bądź klasyfikacji produktów.

## 2.2. Związki encji stopnia 2

Związki te dotyczą dwóch różnych połączonych ze sobą encji. Najprostszym przypadkiem jest tu związek typu  $1:1$  (rys. 4). W przypadku, gdy jest to związek wymuszony z obu stron powiązania (linia ciągła na całej długości związku), mamy wtedy do czynienia z encja-

mi, które nie mogą istnieć niezależnie od siebie, czyli że nie można utworzyć instancji (elementu) encji A bez jednoczesnego utworzenia instancji encji B. Tego typu związek jest bardzo rzadko spotykany i może służyć do prezentacji dekompozycji wcześniej utworzonej encji, która składa się z dwóch omawianych obiektów. A i B.

Na rys. 4 przedstawiono związek typu 1:1 z powiązaniem opcjonalnym z jednej strony. Ten rodzaj konstrukcji może być wykorzystywany do poszerzenia atrybutów encji A o atrybuty zapisane w encji B i może mieć zastosowanie, gdy pewna niewielka liczba instancji np. 10% będzie miała większą ilość atrybutów w stosunku do pozostałej populacji; w tym przypadku chodzi o atrybuty a1 i a2.



Rys. 4. Relacja typu 1:1

Fig. 4. Relationship one-to-one

Z koniecznością poszerzenia atrybutów możemy się spotkać w przypadku tabeli pozycji asortymentowych systemu klasy ERP (ang. *Enterprise Resource Planning*), które z punktu widzenia użytkownika stanowią pewną zamkniętą listę.



Rys. 5. Relacja typu 1:N (silna)

Fig. 5. Relationship 1:N (one-to-many).

Z kolei na rys. 5 przedstawiono relację typu 1:N z wymuszeniem związku między encjami. Z konstrukcji tej wynika, że obie encje nie mogą istnieć samodzielnie. I tak dla przedstawionego przykładu, który modeluje wielopozycyjne zlecenie (produkcyjne bądź zakupowe) wynika, że integralną częścią zlecenia jest zarówno nagłówek jak i związane z nim specyfikacje.



Rys. 6. Relacja z wymuszeniem więzów integralności referencyjnej  
 Fig. 6. Relationship with enforced referential integrity.

Na rys. 6 przedstawiono inny rodzaj związku 1:N, zwany często związkiem z wymuszeniem więzów integralności. Jest to najczęściej spotykany rodzaj związku w systemach informatycznych klasy ERP. Przedstawiona konstrukcja na powyższym rysunku może być odczytywana następująco:

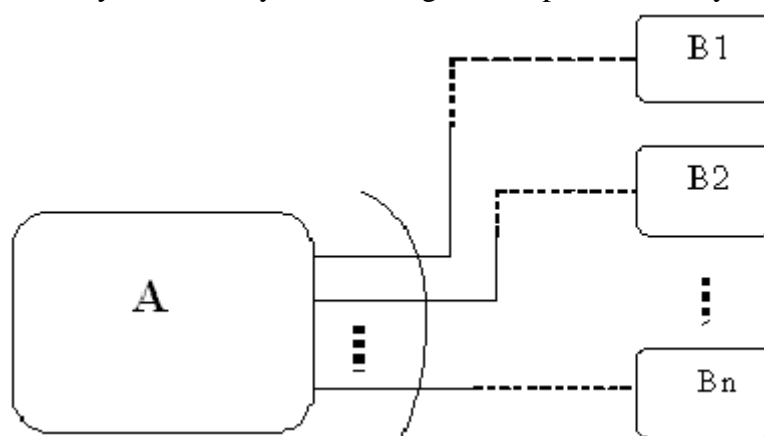
*Każdy określony Produkt może mieć jeden bądź więcej Wariantów*

a odczytywany w drugą stronę:

*Każdy Wariant należy do jednego i tylko jednego określonego Produktu.*

Aby zapewnić jednoznaczność identyfikatora *Wariantu* dla różnych produktów (słaba encja) wtedy należałoby powiązać ten identyfikator z atrybutem *nr\_produktu*, co zostało pokazane na rysunku w postaci pionowej kreski przecinającej linię związku w pobliżu encji *Wariant*. **Encje słabych** typów są identyfikowane w oparciu o kombinacje powiązań z innymi encjami.

W przypadku, gdy jedna encja jest powiązana bezpośrednio z wieloma innymi encjami, wtedy możemy mieć do czynienia z diagramem przedstawionym na rys.7.

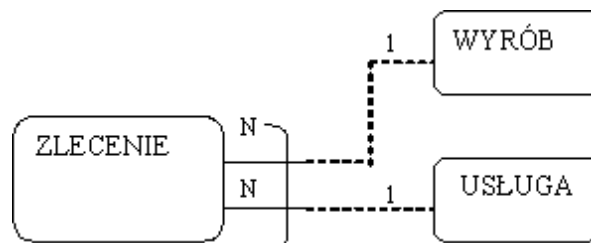


Rys. 7. Związki wzajemnie wykluczające się  
 Fig. 7. Exclusive OR relationships.

Diagram ten przedstawia sytuację **albo/albo** (alternatywa wykluczająca), wymuszoną przez zastosowanie łuku wykluczającego obejmującego linie związków wychodzących z encji A. Diagram ten może być odczytywany od lewej strony następująco:

Każda encja  $A$  musi być albo dla jednej i tylko jednej encji  $B1$ , albo dla jednej i tylko jednej encji  $B2$ , ... albo dla jednej i tylko jednej encji  $Bn$ .

Przykładem takiego związku encji spotykanym w systemach zarządzania może być diagram przedstawiony na rys. 8.



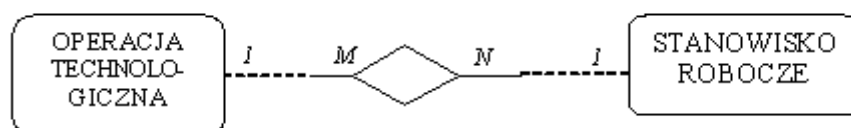
Rys. 8. Przykład związku z łukiem wykluczającym.

Fig. 8. Example of diagram with exclusive OR relationship.

Powyższa konstrukcja odczytywana od lewej strony oznacza, że każde zlecenie *musi być albo* na Wyrób *albo* na Usługę co oznacza, że instancja Zlecenia nie może zawierać jednocześnie egzemplarza Wyrobu i Usługi. Natomiast czytany od strony prawej oznacza, że każdy Wyrób  *bądź*  każda Usługa  *może być*  przedmiotem jednego bądź wielu zleceń.

### 2.3. Związki stopnia 2. typu M:N

Omawiane w tym podrozdziale związki są relacjami binarnymi o współczynniku licznosci  $M:N$ , gdzie  $M \geq 1$  oraz  $N \geq 1$ . Związków tych nie da się bezpośrednio zaimplementować w systemach baz danych dlatego na etapie analizy powinny zostać wyeliminowane przez wprowadzenie tzw. encji intersekcji między dwa końce związków [1], [2]. Tak nowo utworzonej encji należy nadać nazwę, którą może być rzeczownik. Równie dobrym rozwiązaniem może być nadanie takiemu związkowi formy czasownika [5]. Encję intersekcji, dla jej odróżnienia od pozostałych, dobrze jest przedstawić w postaci rombu, jak na poniższym rysunku [5].



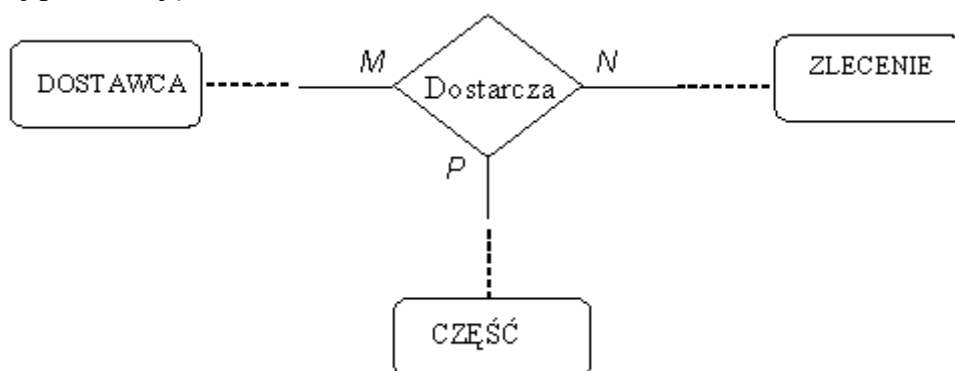
Rys. 9. Przykład związku typu M:N

Fig. 9. Example of relationship M:N (many-to-many)

Powyższy związek jest typowym przykładem związku binarnego o liczebności M:N reprezentujący strukturę sieciową danych. Z przedstawionego diagramu wynika, że każda Operacja Technologiczna może być powiązana z jednym bądź wieloma Stanowiskami Roboczymi a z odczytywania tego związku od strony prawej będzie wynikać, iż każde Stanowisko Robocze może być powiązane z jedną bądź wieloma Operacjami Technologicznymi. W tej konwencji odczytywania związku rzeczowniki stanowią nazwy typów encji, natomiast czasowniki pełnią rolę nazwy związku.

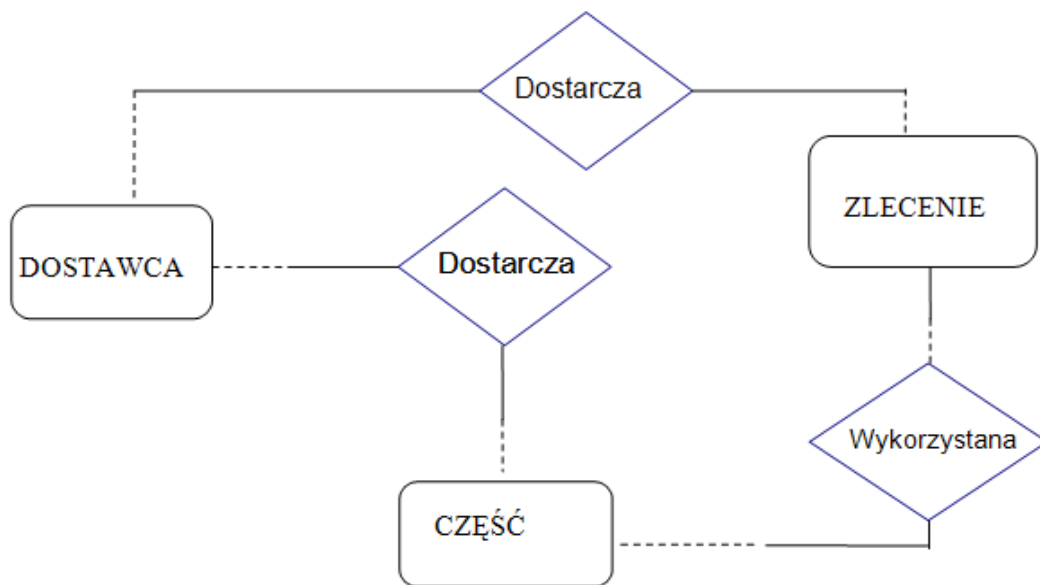
### 3. Typy związków wyższego stopnia

Na rys. 10 przedstawiono przykład notacji diagramu związków encji dla typu trójskładnikowego. Związek ten odczytywany od strony lewej oznacza, że Dostawca *dostarcza* Części dla Zlecenia. W tym przypadku zbiór związków Dostarcza jest zbiorem instancji  $(d,c,z)$ , gdzie  $d$  reprezentuje encję Dostawca, który aktualnie dostarcza Część  $c$  dla instancji  $z$  reprezentowanej przez encję Zlecenie.



Rys. 10. Typ związku trójskładnikowego  
Fig. 10. Example of multiple relationship.

W powyższym diagramie można wyróżnić następujące zbiory egzemplarzy związków:  $(d, c)$ ,  $(c, z)$ ,  $(d, z)$  oraz  $(d, c, z)$ . Próba przekształcenia omawianego diagramu do postaci związków trójskładnikowych, jak na rys. 11, bazująca na tym samym zbiorze egzemplarzy związków, może się okazać nierównoważna.



Rys. 11. Związki binarne nierównoważne związkowi z rys. 10.

Fig. 11. Binary relationships not equivalent to those from fig. 10.

Na powyższym diagramie również istnieją trzy typy związków binarnych jak na poprzednim rysunku, jednak istnienie tutaj egzemplarza  $(d, c, z)$  nie musi oznaczać, że taki sam egzemplarz istnieje w związku typu trójskładnikowego (Dostawca może dostarczać Część ale to nie oznacza, że ta Część będzie wykorzystana w Zleceniu). Stąd w procesie projektowania bazy danych decydowanie o przedstawieniu związków encji w postaci diagramu  $n$ -tego stopnia czy też jego dekompozycja na związki niższych stopni powinno być gruntownie przemyślane, aby nie stracić z pola widzenia semantyki powiązań. Jeśli narzędzia wspomagające projektowanie logiczne baz danych bazuje wyłącznie na związkach binarnych, wówczas związek trójskładnikowy (np. Dostarcza) powinien być reprezentowany za pomocą słabego typu encji oraz trzech związków identyfikujących, które eliminują konieczność wyznaczenia klucza częściowego w tym słabym typie encji [5].

#### 4. Uwagi końcowe

Przedstawione do tej pory diagramy związków encji dotyczyły sytuacji, gdy dane prezentowane przez system są zgodne z narzuconym schematem (strukturą). Tymczasem nie zawsze dane są przechowywane w bazach o precyzyjnie określonej strukturze (dane strukturalne). Taka sytuacja może wystąpić, gdy będziemy chcieli zapisać w bazie dokument zlecenia produkcyjnego zawierającego strukturę wyrobu, która wcześniej nie została w tej bazie zdefiniowana.



W związku z potrzebą użycia sieci WWW jako źródła danych oraz możliwości operowania elastycznym formatem wymiany danych między odmiennymi bazami danych, pojawiło się

w bazach pojęcie danych **semistrukturalnych** [8]. Są to dane, które mogą być nieregularne bądź niekompletne, których struktura może się zmieniać w sposób nieprzewidywalny. W przypadku danych semistrukturalnych informacje powiązane ze schematem są umieszczane łącznie z danymi.

W tradycyjnych systemach relacyjnych wymagany jest predefiniowany schemat tabelowy a więc taki, któremu odpowiadają wcześniej omówione związki encji, gdzie przyjęto milczące założenie, że atrybuty są danymi prostymi. Mimo że obiektowe systemy zarządzania bazami danych dopuszczają bogatsze struktury w stosunku do baz relacyjnych [5], [7], nadal wymagają, aby dane odpowiadały wcześniej zdefiniowanemu schematom.

Implementacja języka XML (ang. *eXtended Markup Language*) w systemach informatycznych sprawiła, że stał się on standardem reprezentacji i wymiany danych w sieci WWW mimo pewnej niedogodności polegającej na tym, że dokument XML musi reprezentować strukturę hierarchiczną, co w przypadku związków encji o liczebności N:M będzie wymagało odpowiednich przekształceń i w konsekwencji zwielokrotnienia danych. Warto zauważyć, że wprowadzenie w ostatnich latach typu XML w relacyjnych bazach danych znacznie ułatwia definiowanie elastycznych struktur.

Rozwój potrzeb w zakresie obsługi danych semistrukturalnych będzie wymagać modyfikacji notacji ERD (ang. *Entity Relationship Diagram*), co powinno wyznaczyć kierunek zmian w zakresie rozwoju tego formalizmu będącego istotnym narzędziem w projektowaniu konceptualnym baz danych..

## BIBLIOGRAFIA

1. Barker R.: Case\* Method. Modelowanie związków encji. WNT, Warszawa 1996.
2. Bojarski R. i In. :Laboratorium a systemów informatycznych zarządzania przedsiębiorstwem przemysłowym, Wyd. politechnik Śląskiej, Gliwice 2010.
3. Booch G., Rumbaugh J., Jacobson I.: UML – Przewodnik użytkownika. WNT, Warszawa 2002.
4. Date C. J.: Wprowadzenie do systemów baz danych. WNT, Warszawa 2001.
5. Elmasri R., Navathe Sh.: Wprowadzenie do systemów baz danych. Wyd. Helion Gliwice 2005.

6. Fryźlewicz Z., Salomon A.: Podstawy architektury i technologii usług XML sieci WEB, Wyd. Mikom PWN, Warszawa 2008.
7. Graham Ian: Metody obiektowe w teorii i praktyce, WNT Warszawa 2004.
8. Turski W. M.: Struktury danych, WNT Warszawa, 1971.

### **Abstract**

The paper introduces classification of Entity Relationship Diagrams (ERD) used in conceptual modelling of data structures in databases related to manufacturing management systems. Classification is based on the idea of a relationship degree, depending on how many entities are included in the diagram.. On that concept 1<sup>st</sup>, 2<sup>nd</sup> and higher degree entity relationships have been presented as well as their examples encountered in manufacturing IT systems. An attention has been brought to the formal interpretation of the diagrams as well as to their semantics, especially when decomposing higher degree relationships to lower ones. Also the need for handling semistructured data in database systems has been discussed within the context of entity relationships and usefulness of XML language.