

Metodyka modelowania danych dla systemów informatycznych do obsługi B2C

Paweł Gepner
Intel Corporation
e-mail: pawel.gepner@intel.com

Abstrakt. W artykule przedstawiono metodę konstruowania modeli danych dla systemów Internetowej sprzedaży w obrębie segmentu B2C. Umiejętność budowy modelu danych nie jest sprawą łatwą; sformalizowanie sposobu konstruowania takich modeli pozwala w efektywny sposób panować nad procesem jego powstawania i stwarza nadzieję, że wszystkie niezbędne pytania, aby lepiej zrozumieć system zostaną zadane. Zaprezentowany przykładowy - hipotetyczny model danych zbudowany z wykorzystaniem tej metody wskazuje na jakie aspekty należy zwrócić uwagę podczas budowy modelu danych dla systemów internetowej sprzedaży.

1. Wprowadzenie

Duże przedsiębiorstwa już jakiś czas temu zaczęły korzystać z transakcji elektronicznych w prowadzeniu biznesu między sobą. Systemy Electronic Data Interchange (EDI) są już eksploatowane od początku lat 60'tych. Banki zaczęły używać Electronics Funds Transfer (EFT) mniej więcej w tym samym czasie. Obecnie, wraz ze wzrostem popularności Internetu, elektroniczne transakcje stają się codziennością zarówno dla odbiorcy, klienta końcowego, jak i dla firm każdej wielkości. Internet zmienił rodzaj i formę prowadzenia biznesu dla wielu firm np. w 2000 r. ponad 90% wszystkich transakcji realizowanych przez firmę Intel Corporation zrealizowano za pośrednictwem Internetu [1]. To medium informacyjne, stając się coraz bardziej popularnym, stwarza możliwości współpracy coraz to nowym firmom i użytkownikom. Nie osiągnięto jeszcze „masy krytycznej”, ale chwila, kiedy wszystkie firmy będą *firmami internetowymi*, nie jest odległa. W wielu publikacjach definiuje się E-Commerce jako sprzedaż i kupno produktów i usług poprzez Internet, ale zjawisko to ma więcej aspektów. Już od początku E-Commerce zajmowało się przetwarzaniem transakcji i transferem pieniędzy poprzez sieci komputerowe. Dzisiaj E-Commerce obejmuje także sprzedaż i kupno takich nowych wartości, jak elektroniczna informacja. Również skala tego zjawiska zmieniła się zupełnie; początkowo transakcje tego typu były prowadzone pomiędzy wielkimi korporacjami i bankami. Natomiast Internet pozwolił użytkownikom końcowym włączyć się w ten łańcuch. Ten wzrost zainteresowania klienta końcowego i klientem końcowym widzimy już dzisiaj, a lata następne pokażą jak istotny jest to element. W mniejszych firmach również zauważono, że można prowadzić biznes internetowy tak, jak to robią wielkie korporacje. Organizacje różnych wielkości zauważyły, że Internet obniża koszty zastępując inne sieci. Używając Internetu jako nowego medium komunikacyjnego firmy te przekształcają swoje dane w elektroniczny format i włączają je w systemy swoich kooperantów, partnerów i klientów. Zanim zajmiemy się budową modelu systemu dla gospodarki Internetowej należy zastanowić się jak działa tradycyjna forma sprzedaży. Klient najpierw identyfikuje swoje potrzeby niezależnie od tego, czy jest to fizyczny produkt, usługa czy informacja. Następnie szuka i porównuje informacje na temat produktu, usługi; znajdując odpowiedni obiekt swoich zainteresowań, dokonuje następnie analizy (ceny, serwisu, reputacji itp.) warunków. Proces ten może również zawierać takie dodatkowe aspekty, jak negocjacje: ceny (np. aukcje internetowe itp.), ilościowe (promocje – dwa w cenie jednego itp.), warunków dostawy, a może także pewnych aspektów prawnych (akceptacja praw autorskich). Jednak proces ten nie kończy się na etapie dostawy produktu. Wsparcie posprzedażne zapewnia korzyści obu stronom. Klientowi pozwala na ciągłą kontrolę swego produktu. Dostawca natomiast zdobywa informacje o zachowaniach, tendencjach i oczekiwaniach rynku. W tym czasie bank i inne instytucje finansowe dokonują transferu pieniędzy pomiędzy kupującym a sprzedającym, niezależnie od tego, czy jest to odbiorca indywidualny, czy wielkie korporacje. Kiedy uświadomimy sobie ile procesów jest zaangażowanych w tradycyjnym modelu sprzedaży,

staje się oczywiste, że definicja E-Commerce nie może ograniczyć się do kupna i sprzedaży produktów w sieciach komputerowych zastępujących tylko tradycyjne faktury, telefony, kurierów, ciężarówki itp. Mówiąc o E-Commerce musimy pamiętać, że nie możemy ograniczać jego procesów jedynie do procesów generujących przychody, ale również należy pamiętać o takich procesach towarzyszących i wspierających generowanie zysków, jak: kreujące popyt, wsparcie i serwis posprzedażny, etc. E-Commerce zbudowana na strukturze tradycyjnego biznesu oraz wzbogacona o wszystkie zalety oferowane przez sieci komputerowe tworzy całkowicie nową jakość. Operacja na danych cyfrowych stwarza możliwość interakcji w prosty sposób z kooperantami, usuwa również bariery, które są znaczącym ograniczeniem w przypadku tradycyjnych modeli – na przykład klient ma dostęp do produktu, usługi, wsparcia posprzedażnego przez 24 godziny na dobę, 7 dni w tygodniu, pod każdą szerokością geograficzną. Doświadczenia USA pokazują, że nowe technologie informatyczne mogą wspomagać wzrost gospodarczy i tworzyć nowe miejsca pracy. Okazało się, że w roku 1999 firmy bezpośrednio związane z Internetem dawały w USA 2,3 miliona miejsc pracy w porównaniu z 1,6 miliona - w 1998 [3]. Według raportu Unii Europejskiej w krajach piętnastki dokonano transakcji za pomocą Internetu na sumę około 17 milionów €, a szacunki na rok 2003 mówią o 340 milionach € [2]. E-Commerce otwiera nowe drogi i nowe formy robienia biznesu. Jako przykład weźmy największą księgarnię internetową świata – www.amazon.com. Siedziba firmy znajduje się w Seattle w stanie Washington. Firma ta nie ma fizycznych sklepów, a książki, CD i kasyety sprzedaje przez Internet koordynując wysyłki bezpośrednio z magazynów wydawcy. Nie musi, więc posiadać żadnych zapasów, czy zasobów magazynowych. Firmy takie jak: www.double-eye.com, czy www.heraut.com poszły jeszcze dalej. Ponieważ ich produkty są w pełni numeryczne, to ich magazyny są również cyfrowe i znajdują się na komputerach (serwerach), na których realizowane są zadania odpowiedzialne za obsługę zamówień. Trudno, więc określić definicję E-Commerce, ponieważ pewne możliwości nie są jeszcze całkowicie odkryte i nie dotyczy to tylko tych zadań, które znamy już dziś i realizujemy je za pomocą sieci komputerowych, ale zwłaszcza nowych zjawisk, które pojawią się w Internecie w przyszłości. Ujmując zagadnienie z punktu widzenia inżynierii systemów informacyjnych możemy przyjąć następującą definicję: E-Commerce to suma procesów i danych zaangażowanych w prowadzenie biznesu (dokonywanie transakcji) za pośrednictwem Internetu.

2. Budowa modelu

Podczas tworzenia modelu danych systemów informatycznych należy pamiętać, że modelowanie danych oraz modelowanie procesów nie są działaniami rozłącznymi, nie muszą też być wykonywane jedno po drugim [4]. W modelu encja - związek (ERD, entity relationship diagram) obszar analityczny jest opisywany za pomocą encji, związków między encjami i atrybutów. Atrybuty są związane z encjami. Związki również mogą posiadać atrybuty, ale muszą się one odnosić do samych relacji, a nie do przypisanych do nich encji. Pierwszą wersję modelu danych możemy zbudować na podstawie wiedzy, jaką posiadamy już na początku projektu. Kolejne kroki metody pozwolą uszczegółowić i rozbudować model wejściowy tak, aby w końcowej fazie był on pełny i dokładny.

2.1. Określenie głównych encji

Zaproponowany model danych systemu internetowej sprzedaży pozwala na jego rozbudowę zgodnie ze specyfiką i potrzebami klienta. Zakładała się, że system realizuje sprzedaż towarów do klientów indywidualnych, ale obsługa klientów instytucjonalnych wymaga jedynie minimalnych modyfikacji. Rozpatrywany model wykorzystuje również zewnętrzny system obsługi płatności. Wyszukiwanie encji, jak i budowa całego modelu danych to umiejętność stawiania konstruktywnych pytań zarówno przez analityków, jak i projektantów. Odpowiedzi na te pytania pozwolą wyodrębnić główne encje modelu. Poniżej przedstawiono kilka praktycznych reguł, które pomogą wyodrębnić te encje.

- Encja ma określony cel w przedsiębiorstwie. Jeżeli nie potrafimy podać znaczenia encji, to nie należy jej wprowadzać do modelu.
- Encja musi mieć co najmniej jeden atrybut
- Encja ma więcej niż jedna instancję w przeciwnym wypadku to stała systemu.
- Encją często bywa terminator z modelu kontekstowego
- Rzeczowniki z opisu działania modeli procesów (funkcji) przedsiębiorstwa często bywają encjami.

W strukturze systemu sprzedaży internetowej wyodrębniono trzy główne obiekty informacyjne (encje), które posiadają najistotniejsze znaczenie dla analizowanego obszaru konceptualnego. Główne encje naszego modelu to: klient, produkt, transakcja (rys 1.). Encje te stanowią model wyjściowy do dalszych rozważań dotyczących modelu danych systemu internetowej sprzedaży i należy je traktować w proponowanym modelu, jako encje klasy „nadtyp”[4]. W takich systemach, jak Intershop 4.0, czy Microsoft E-commerce Server wykorzystywany model danych jest oparty na „podtypach” encji przedstawionych na rys.1.



Rys. 1. Główne encje rozpatrywanego modelu danych systemu internetowej sprzedaży

2.2. Określenie relacji

Kolejnym krokiem w budowie modelu danych systemu internetowej sprzedaży jest ustalenie powiązań pomiędzy głównymi obiektami informacyjnymi systemu. Przy ustalaniu tych związków pomocne mogą się okazać następujące reguły:

- Relacja to „pamiętane” w systemie współdziałanie pomiędzy dwiema (lub kilkoma) encjami.
- Czasownik często oznacza relacje.
- Relacja może posiadać atrybuty.
- Istotne są tylko te relacje, które znajdują się w obrębie obszaru kontekstowego. Należy pomijać powiązania spoza zakresu działania systemu.

W rozpatrywanym modelu określono następujące związki (relacje) między encjami systemu. Klient może dokonywać (składać) transakcji. Transakcja zaś może składać się z wielu produktów (towarów lub usług), które opisują zawartość koszyka (ang. shopping cart) czy formularza zakupów, a dany produkt może być elementem składowym transakcji.

2.2.1. Określenie liczebności relacji

Określenie liczebności relacji wskazuje, ile encji każdego typu uczestniczy w związku. Określenie liczebności to lepsze zrozumienie swoich danych dla modelowanego systemu, to postawienie konkretnych pytań i odpowiedzenie na nie:

- Czy klient może składać wiele transakcji?
- Czy dana transakcja może dotyczyć tylko jednego klienta (nawet, jeżeli jest to klient zbiorowy)?
- Czy dana transakcja może dotyczyć tylko jednego produktu?
- Czy dany produkt może występować w jednym czy wielu transakcjach?

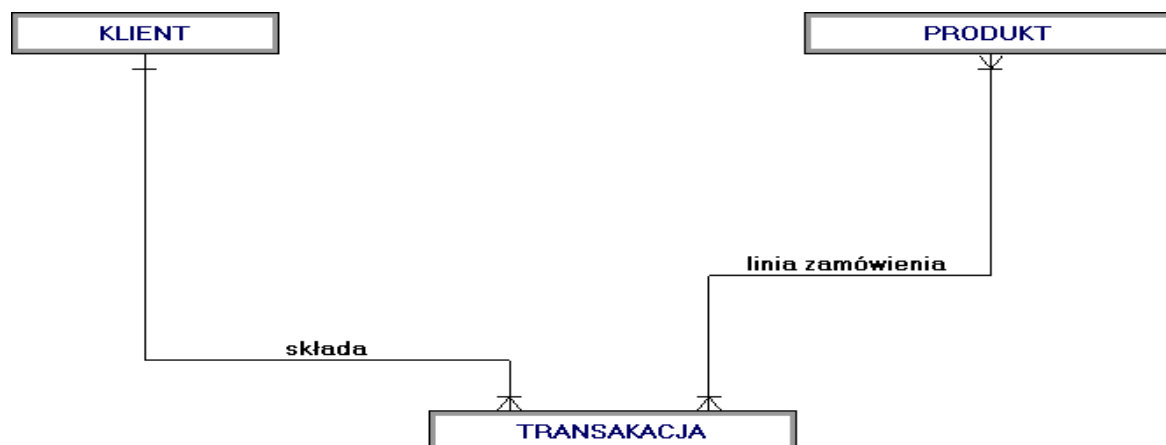
Odpowiedź na tak stawiane pytania pozwala lepiej zrozumieć naturę relacji i określić liczebność poszczególnych związków. Dla opracowanego modelu liczebność związków (relacji) została określona następująco. Relacja **składa** jest typu 1: N. Klient może składać wiele zamówień, ale konkretna transakcja dotyczy tylko jednego klienta. Relacja **linia zamówienia** jest typu M: N, gdyż dany produkt może występować w wielu transakcjach, a transakcja może składać się z wielu produktów.

2.2.2. Określenie opcjonalności związku

Opcjonalność dotyczy udziału encji w relacji. Uczestnictwo encji w związku jest opcjonalne, jeżeli istnieje przynajmniej jedna instancja encji, która nie bierze udziału w relacji. Określając opcjonalność związków w zaproponowanym modelu danych musimy opowiedzieć na pytania:

- Czy może istnieć transakcja bez klienta?
- Czy każdy klient musi dokonać jakiejś transakcji?
- Czy transakcja musi zawierać produkty?
- Czy istnieje produkt, który nie występuje w żadnej transakcji?

Odpowiedź na tak skonstruowane pytania pozwala na określenie opcjonalności relacji w modelu danych i zapewnienie lepszego zrozumienia systemu. Przedstawiony poniżej (rys.2) diagram modelu ukazuje główne encje (obiekty informacyjne) i związki pomiędzy nimi, występujące w zaproponowanym modelu danych systemu internetowej sprzedaży.



Rys. 2. Zaznaczone główne encje i związki modelu danych

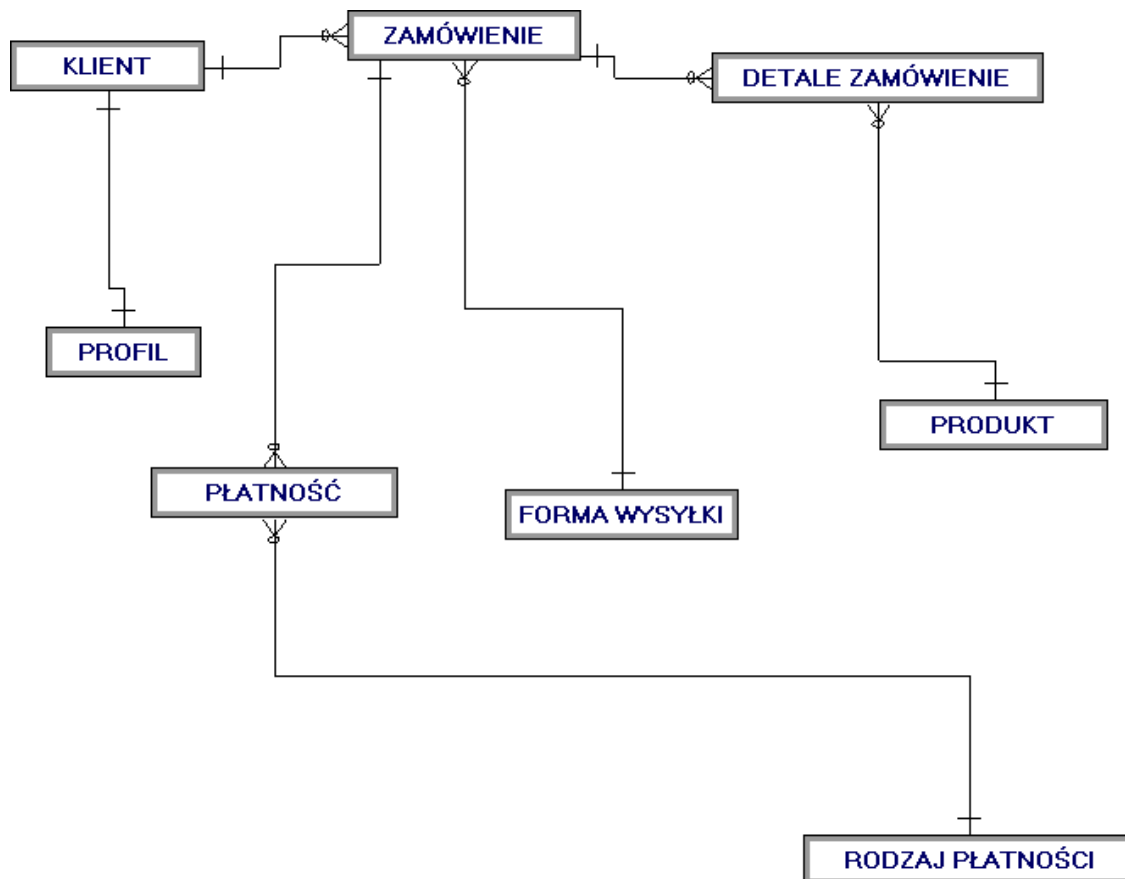
3. Podtypy encji i rozbudowa relacji wejściowych

W pierwszym etapie budowy modelu zostały wyróżnione główne obiekty informacyjne, które należy traktować jako encje typu nadtyp modelu danych. Na obecnym etapie encje te poddamy analizie zastanawiając się, czy w ich obrębie można wyszczególnić jakieś podtypy czy podgrupy.

Podobnej analizie należy również poddać relacje. Fundamentalną zasadą podczas takiej dekompozycji (pogłębionej analizy) jest zapewnienie warunków minimalności i kompletności modelu. Należy dodawać tylko te obiekty, które mają istotne znaczenie dla systemu z punktu widzenia funkcjonowania biznesu. Kryterium rozbudowy musi być oparte na rzetelnej analizie potrzeb informacyjnych systemu jak i na zdrowo-rozsądkowym planowaniu. Na połączenie tych dwóch „pierwiastków” w planowaniu systemu zwraca uwagę w swojej pracy, poświęconej metodologii programowania, W.M. Turski [5]. Również Z. Szyjewski w pracy na temat tworzenia systemów informatycznych wyraźnie akcentuje te dwa nurty badawcze związane z automatyzacją i wspomaganie budowy systemów informatycznych [6]. Rekurencyjny charakter tej metody pozwala na analizę i rozbudowę encji powstałych z rozłożenia encji wejściowych (wyjściowe w następnym kroku stają się encjami wejściowymi) aż do momentu, gdy zaproponowany rozkład odzwierciedla optymalną budowę modelu danych i nie wymaga dalszego rozkładu, a przy tym zapewnia warunek kompletności i minimalności danych. Chociaż wymienione wyżej encje modelu wejściowego systemu internetowej sprzedaży opisują na poziomie ogólności większość systemów klasy B2C, to należy poddać rozbudowie ten mini-model, dostosowując go do specyfiki konkretnej implementacji. Poniższa lista przykładowych pytań pozwoli lepiej wyodrębnić nowe obiekty informacyjne w rozbudowywanym modelu:

- Czy grupujemy towary?
- Czy klienci posiadają różny status?
- Czy klienci posiadają indywidualne profile?
- Czy przechowujemy informacje o zawartości koszyka (shopping cart), nawet jeśli nie jest on zatwierdzony jako zamówienie?
- Czy przewidujemy możliwość obsługi promocji?
- Czy interesują nas informacje o formie dostawy zamówienia?
- Czy mają dla nas znaczenie informacje o sposobie płatności?

Odpowiedź na te i podobne pytania pozwoli dodać nowe encje tak, że będą one dostosowane do specyfiki działania systemu i zaspokoją jego potrzeby informacyjne. Podobnej analizie poddajemy relacje klasy M:N. Należy zadać sobie przy tym pytanie, czy daną relację należy zdekomponować na relacje klasy 1:N, czy z punktu widzenia potrzeb informacyjnych przedsiębiorstwa pierwotna postać jest bardziej naturalna i pełna. Wynikiem takiej dekompozycji modelu wejściowego może być model przedstawiony na rys 3.



Rys 3. Przykład rozbudowy modelu wejściowego

Formalizując metodę rozbudowy modelu danych możemy zapisać ją w postaci następującej procedury:

```

Dekompozycja modelu poziomu(n)
{
    model poziomu (n+1) otrzymujemy w rezultacie rozbudowy modelu wejściowego o
    nowe encje i relacje powstałe z analizy encji i relacji modelu poziomu(n);
    sprawdzamy czy model jest minimalny i wystarczający; jeżeli tak to koniec
    procedury;
    w przeciwnym wypadku:
    Dekompozycja modelu poziomu(n+1)
}
    
```

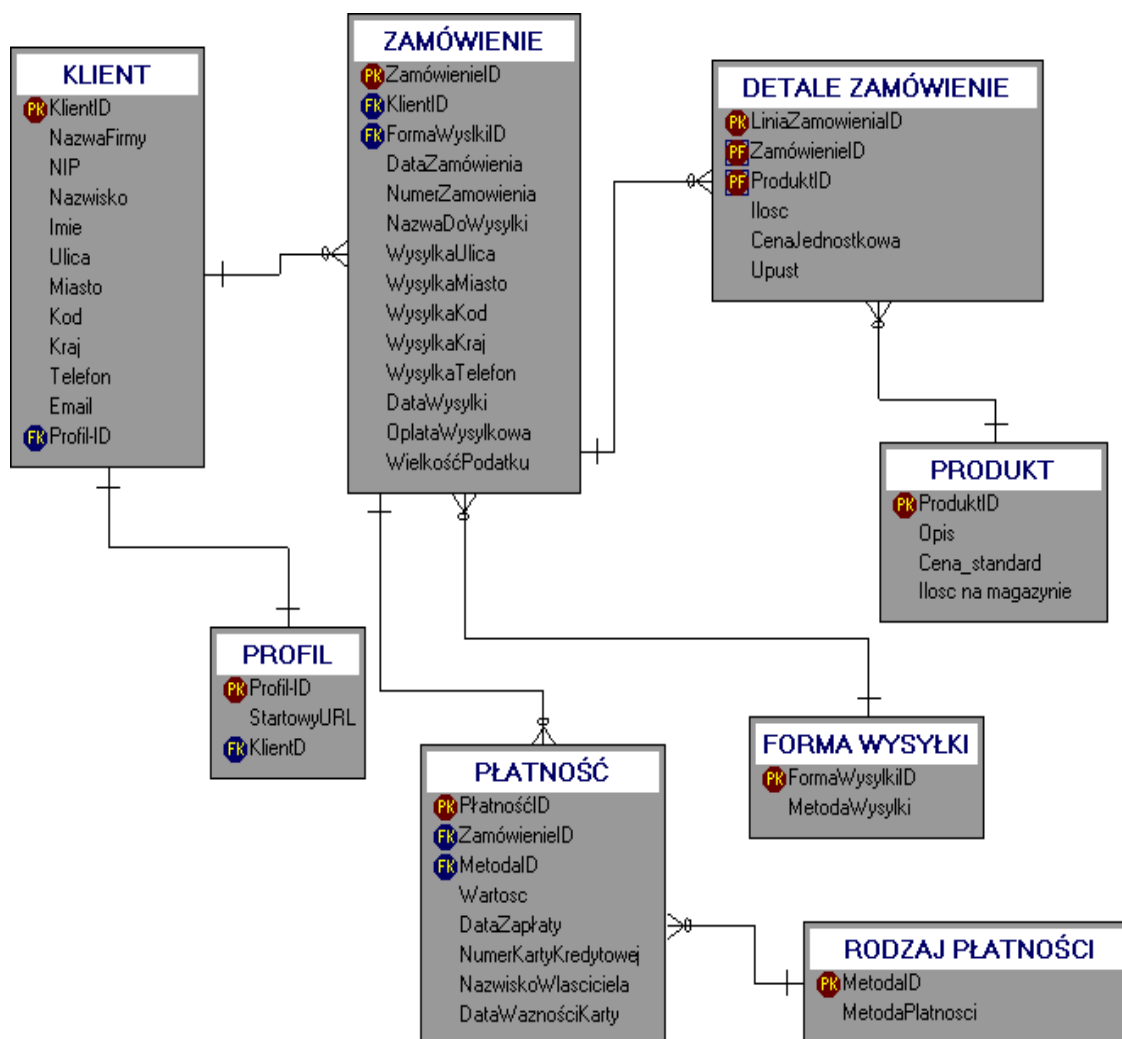
W praktyce w systemach sprzedaży internetowej klasy B2C rozbudowa modelu wyjściowego już po pierwszej dekompozycji w sposób pełny zapewnia powstanie modelu, który odpowiada potrzebom informacyjnym systemu.

4. Przypisanie atrybutów

Gdy mamy już określone obiekty informacyjne modelu (encje) i powiązane z nimi relacje, przystępujemy do przypisywania im atrybutów. Przypisywanie atrybutów polega na przyporządkowaniu elementów danych do właściwej encji. Należy również pamiętać, że atrybuty

mogą opisywać relacje. Atrybuty są cechami mierzalnymi i niemierzalnymi obiektu, które w sposób bezpośredni charakteryzują dany obiekt informacyjny. Dla każdego atrybutu musimy zadać sobie pytanie: Jaką encję lub związek opisuje konkretny element danych [4]. Czasami po przypisaniu wszystkich atrybutów okazuje się, że istnieje encja, która nie posiada żadnych atrybutów. Praktyczna zasada mówi, że jeśli nie ma danych opisujących encję, to znaczy, że encja ta nie jest potrzebna. W trakcie przypisywania atrybutów poznajemy modele coraz dokładniej i pełniej. Należy mieć jednocześnie na uwadze zasadę mówiącą o tym, że każdy element danych pamiętany w systemie musi być przypisany do jednej i tylko jednej encji lub relacji. Wracając do naszego modelu (rys. 3) określenie atrybutów dla encji **klient** powinno pozwolić na przechowywanie minimum niezbędnych informacji o poszczególnych klientach, a z drugiej strony znajdują się tam informacje wystarczające np. do wystawienia rachunku. Każdy **klient** posiada unikalny numer identyfikacyjny **KlientID**, który jest jednocześnie kluczem głównym. System uwzględnia obsługę firm i atrybut **NazwaFirmy** przechowuje informacje o kliencie, gdy ten jest firmą. Jest to atrybut opcjonalny - to znaczy, że w przypadku gdy klient jest osobą fizyczną, atrybut ten nie przechowuje żadnych danych. **NIP** - to również atrybut opcjonalny; jest on niezbędny tylko dla klientów będących firmami. **Nazwisko, Imię, Ulica, Miasto, Kod, Kraj, Telefon** - to atrybuty, które opisują każdego klienta. **Email** - to atrybut, w którym zawarta informacja pozwala na komunikowanie się z klientem w celu, na przykład, potwierdzenia przyjęcia zamówienia. Encja **klient** została rozwinięta tak, aby przechowywać profil klienta, gdzie klient mówi nam o swych preferencjach, czy dziedzinach zainteresowań. Encja **profil** może być zasilana informacjami bądź z ankiet wypełnianych przez klientów, bądź z analizy danych historycznych (strony, do których klient sięga najczęściej). Atrybut **Profil-ID** w encji **klient** - to klucz obcy, który łączy klienta z jego ewentualnym profilem. Encja **profil** opisana jest za pomocą atrybutu **StartowyURL**, gdzie przechowywany jest adres profilowanej strony przypisanej do konkretnego klienta. Klient może mieć tylko jeden profil, stąd relacja pomiędzy encją **klient**, a encją **profil** jest typu 1:1. Klient może składać wiele zamówień, ale konkretne zamówienie może dotyczyć tylko jednego klienta, tak więc relacja pomiędzy encją **klient**, a encją **zamówienie** jest typu 1:N. Encja **zamówienie** wprowadza atrybuty dla zapisu informacji kto, kiedy i na co złożył zamówienie. Oprócz danych związanych z numerem zamówienia i klientem, który je złożył przechowywane są informacje gdzie i kiedy przesyłka ma być dostarczona (adres klienta nie musi być adresem dostawy, np. serwis kwiatowy). W atrybucie **OplataWysylkowa** przechowuje się informacje o kwocie pobranej za dostarczenie. **FormaWysylkiID** - to klucz obcy związany z encją **forma wysyłki**, która zawiera informacje o dostępnych formach dostarczenia zamówionego towaru. Może istnieć wiele sposobów dostarczenia towaru, ale dane zamówienie może zostać tylko jednym sposobem dostarczone tak, jak ukazuje relacja pomiędzy tymi encjami. W encji **płatność** przechowuje się takie informacje o dokonanych płatnościach za określone **zamówienie**, jak: wielkość danej płatności i data jej dokonania. Tu również przechowywane są informacje o karcie kredytowej, którą dokonywana jest płatność (system płatności oparty o obsługę kart kredytowych), jej właścicielu i dacie ważności karty. Atrybut **MetodaID** - to klucz obcy związany z encją **rodzaj płatności**, gdzie znajdują się informacje o metodach płatności za pomocą kart kredytowych (przedpłata, w momencie dostawy, raty, siedem dni po otrzymaniu, etc.). Ponieważ zamówienie może składać się z wielu pozycji, encja **detale zamówienia** służy przechowywaniu tych informacji. Atrybut **LiniaZamowieniaID** identyfikuje każdą linię w zamówieniu przyporządkowaną przez klucz obcy **ZamowienieID** konkretnemu zamówieniu i produktowi **ProduktID**. Atrybut **Ilość** określa ilość danego produktu w danej linii zamówienia; jego cenę jednostkową określa atrybut **CenaJednostkowa**, a ewentualny upust zawiera atrybut **Upust**. Relacja wiążąca encję **zamówienie** z encją **detale zamówienia** jest typu 1:N. Taka liczebność związku wydaje się oczywista, ponieważ zamówienie może składać z wielu detali zamówienia, ale konkretny detal zamówienia dotyczy tylko jednego konkretnego zamówienia. Encja **produkt**, która służy do przechowywania informacji o stanie i cechach towarów składa się z następujących atrybutów: **ProduktID, Opis, Ilość na magazynie, Cena_standard**. Atrybut **ProduktID** - to jednocześnie klucz główny encji produkt, który służy jednoznacznemu identyfikowaniu produktów. Definicja takiego klucza może w łatwy sposób nieść również informacje, do jakiej kategorii produktów jest dany obiekt zaliczany, gdzie fizycznie się on znajduje. **Opis** - to atrybut, w którym opisane są cechy fizyczne **produktu**; może on również

zawierać zdjęcie, schemat, czy rysunek produktu. *Cena_standard* - to cena **produktu**, za jaką chcemy sprzedawać dany **produkt** bez upustów, czy promocji; nie należy jej mylić z ceną magazynową. *Ilość na magazynie* - to atrybut mówiący, jaką ilością towaru jeszcze dysponujemy. Oczywiście, dla pewnych produktów niektóre atrybuty encji **produkt** nie mają uzasadnienia. W przypadku sprzedaży produktów zdigitalizowanych takich, jak: oprogramowanie, muzyka MP3, e-book, porady prawne, horoskopy czy pliki wideo, atrybut *Ilość na magazynie* nie ma sensu, ponieważ natura tego typu produktów jest praktycznie niewyczerpywalna. Relacja pomiędzy encją **produkt**, a encją **detale zamówienia** została zdefiniowana jako relacja typu 1:N, gdyż dany produkt może być składową wielu linii zamówienia, ale konkretna linia zamówienia może dotyczyć tylko jednego produktu. Gdy przypiszemy już wszystkie atrybuty konkretnym encjom i relacjom modelowanie danych systemu wydaje się być zakończone. W dalszej fazie analizy modelu danych może okazać się, że pewne atrybuty wydają się być zbędne, a innych elementów brakuje, ale metodycznie przeprowadzone planowanie modelu danych pozwala uniknąć wielu błędów. W czasie realizacji rozwiązań dla potrzeb systemów do obsługi systemów sklepowych klasy B2C jednym z kluczowych czynników jest czas, a właściwie brak czasu na eksperymenty. Zaproponowana metoda konstruowania modelu danych usprawnia proces budowy informatycznych systemów sprzedaży internetowej, a także porządkuje panujący w dziedzinie budowy tego typu systemów chaos. Do budowy przykładowego modelu danych systemu Internetowej sprzedaży z zaznaczonymi encjami, relacjami i atrybutami przypisanymi do encji (patrz rys. 4), zastosowano zaproponowaną wyżej metodę.



Rys. 4. Przypisanie atrybutów modelu ERD systemu internetowej sprzedaży

Przedstawiony wyżej model danych jest przykładowym rozwinięciem modelu wyjściowego, gdzie wprowadzone rozszerzenia dostosowują go do wymagań klientów i dopasowują model tak, aby w zaproponowanej strukturze danych były przechowywane dane, wymagane przez konkretny system.

Bibliografia

1. Intel Annual Report, 2000
2. Andersen Consulting, "eEurope takes off", 2000
3. Jędrzejek C, Internet szansą dla wszystkich: zastosowania infrastruktura techniczna i software'owa. Stan obecny i przyszłość Internetu., Seminarium NetWorld, kwiecień 2000.
4. Robertson J, Robertson S, Pełna Analiza Systemowa, Wydawnictwa Naukowo-Techniczne , 1999, ISBN 83-204-2293-0
5. Turski W. M., *Metodologia programowania*. WNT, Warszawa 1978.
6. Szyjewski Z., *Zarządzanie projektami Informatycznymi-metodyka tworzenia systemów informatycznych*, Placet 2001, ISBN 83-85428-64-X