

WYKORZYSTANIE ONTOLOGII W WYMIAROWANIU PROJEKTÓW INFORMATYCZNYCH

Przemysław PLECKA, Krzysztof BZDYRA

Streszczenie: W trakcie procesów sprzedaży systemów ERP okazuje się, że zbiór standardowej funkcjonalności musi być rozszerzony lub zmieniony (zmodyfikowany) zgodnie z wymaganiami klienta. Dostawcy stoją zatem przed problemem określenia kosztów dodatkowych prac. Znane metody szacowania zadawałające wyniki osiągają dopiero po etapie analizy przedwdrożeniowej. W artykule zaproponowano metodę specyfikacji wymagań użytkownika w oparciu o ontologiczny model wiedzy, zawierający informacje o systemie informatycznym oraz projekcie wdrożenia. Przedstawiono również algorytm szacowania kosztów projektu informatycznego w oparciu o opisaną metodę.

Słowa kluczowe: ERP, szacowanie oprogramowania, metryki, ontologia, wdrożenie, specyfikacja wymagań.

Wstęp

Pierwsze systemy informatyczne klasy ERP (ang. Enterprise Resource Planning) dostępne były jedynie dla bardzo dużych przedsiębiorstw. Bariery w ich upowszechnianiu były wysokie koszty wdrożenia i utrzymania spowodowane tym, że producenci oprogramowania pierwsze systemy tworzyli na zamówienie konkretnych klientów. Zbierając doświadczenie ze współpracy z wieloma klientami, producenci wyodrębnili zbiór funkcjonalności, który powtarzał się najczęściej i utworzyli z niego wersję standardową swojego produktu. Koszty wdrożenia i utrzymania zostały zmniejszone, do tego stopnia, że nawet średniej wielkości przedsiębiorstwa mogą sobie pozwolić na takie systemy [1]. Standaryzacja wiąże się jednak z tym, że istnieją grupy procesów w przedsiębiorstwie, którym nie odpowiada żadna funkcjonalność w standardowym systemie ERP. Stąd pojawia się potrzeba przystosowania systemu informatycznego. Koszt modyfikacji systemu podnosi wartość kontraktu (wdrożenia). W niektórych przypadkach alternatywą jest przystosowanie przedsiębiorstwa do systemu informatycznego, jednakże koszty zmiany organizacji ponosi wówczas klient [2].

Zakłada się, że dostawca dysponuje systemem informatycznym klasy ERP o określonym zbiorze funkcjonalności oraz doświadczeniem dotyczącym kosztów wdrożeń tegoż systemu. Znane są wymagania średniego przedsiębiorstwa dotyczące funkcjonalności wdrażanego SI (System Informatyczny). Wymagania te ujawniane są sukcesywnie w czasie rozmów handlowych i analizy przedwdrożeniowej. Wynik szacowania oczekiwany jest już od wczesnych etapów wdrożenia. Poszukiwana jest metoda szacowania oprogramowania możliwa do stosowania od etapu rozmów handlowych, dająca nie gorsze wyniki, niż inne metody na tym i na kolejnym etapie.

Metody pozwalające wycenić koszty wykonania oprogramowania są znane i opisane w literaturze, np. przez McConella [3]. Przykłady zastosowania metod algorytmicznych dowodzą ich lepszej od pozostałych metod dokładności. Jednak w praktyce szacowania kosztów zmian SI [4], dostawcy stosują metody niealgorytmiczne, jako szybsze (tzn.

tańsze) i łatwiejsze do realizacji, mimo, że stosowane w początkowych etapach wdrożenia (rozmowy handlowe) dają wyniki obciążone dużymi błędami.

Z wcześniejszych badań wynika [5], że podstawą do dokładniejszych szacowań kosztów wdrożenia jest kompletny zbiór wymagań szczegółowych klienta. Analiza wymagań jest częścią analizy przedwdrozeniowej i jej wyniki znane są dopiero po zakończeniu etapu. Aby adekwatną informację pozyskać wcześniej, na etapie rozmów handlowych, zaproponowano wykorzystanie ontologii, na podstawie której algorytm szacujący określa koszty zmian oprogramowania. Problem dotyczy wyłącznie systemów klasy ERP [6] lub podobnych, gdyż jednodziedziczne systemy informatyczne (np. sprzedaż) o ograniczonej i zamkniętej funkcjonalności nie muszą być modyfikowane. Potencjalni nabywcy (małe przedsiębiorstwa), wybierając gotowe programy komputerowe, kierują się zbieżnością funkcjonalności SI z organizacją procesów swojego przedsiębiorstwa.

W pierwszym rozdziale przeanalizowano znane sposoby specyfikacji wymagań użytkowników, które zmieniają standardową funkcjonalność SI oraz metody, które w oparciu o te specyfikacje pozwalają wycenić koszty zmian. Następnie zaproponowano zastosowanie ontologii do modelowania standardowego i zmodyfikowanego SI z uwzględnieniem kosztów wdrożenia. Dla przedstawionego modelu ontologicznego zaproponowano algorytm postępowania, który z jednej strony umożliwia uzyskiwanie dokładnych szacowań już na pierwszych etapach rozmów handlowych dając wyniki nie gorsze, niż znane metody wykorzystywane na późniejszych etapach wdrożenia.

Proponowane podejście zilustrowano na przykładzie rzeczywistego wdrożenia. W ostatnim rozdziale dokonano oceny wyników osiągniętych za pomocą proponowanej metody oraz wskazano kierunki dalszych badań.

1. Specyfikacja wymagań użytkownika (klienta)

We wczesnych etapach wdrożenia systemów informatycznych zachodzi potrzeba zrozumienia określonego fragmentu rzeczywistości, która będzie objęta implementacją, a obca jest dostawcy. Różne metodyki projektowania systemów informatycznych (np. Select Perspective [7], ARIS (ang. Architecture of Integrated Information Systems) [8] wymagają różnych modeli biznesowych tej rzeczywistości. W każdym przypadku, w sformalizowany sposób opisują one zdobytą wiedzę o organizacji. Głównym jej źródłem są wywiady z przedstawicielami przedsiębiorstwa lub odpowiednie dokumenty tekstowe (np. zapisy z Księgi Jakości). Proces pozyskiwania wiedzy z takich źródeł narażony jest na trudności związane z obszernością treści oraz ich specjalistycznym językiem, nieprecyzyznością, a często sprzecznością opisu.

Podstawą szacowania dla większości metod jest specyfikacja wymagań klienta - jedna ze składowych analizy przedwdrozeniowej. Najbardziej popularnym sposobem jej sformalizowania jest zapis strukturalny. Polega on na wyodrębnieniu wymagań ogólnych, a następnie zdekomponowaniu ich. Proces dekompozycji przeprowadzany jest parokrotnie, aż do osiągnięcia poziomu szczegółowości, który pozwoli na wyodrębnienie obiektów danych (encji) i operacji jednostkowych. Taka drzewiasta struktura nie pozwala jednak na kontrolę spójności między wymaganiami. Na przykład, jeśli jedno wymaganie określi potrzebę wykorzystania słownika klas ochrony przepięciowej w uwagach montażowych zlecenia produkcyjnego, a drugie wymaganie również określi wykorzystanie takiego słownika, ale w wydruku certyfikatu bezpieczeństwa wyrobu gotowego, to identyfikacja

takiej sytuacji (jeden wspólny słownik zamiast dwóch oddzielnych) zależy jedynie od percepcji projektanta oprogramowania.

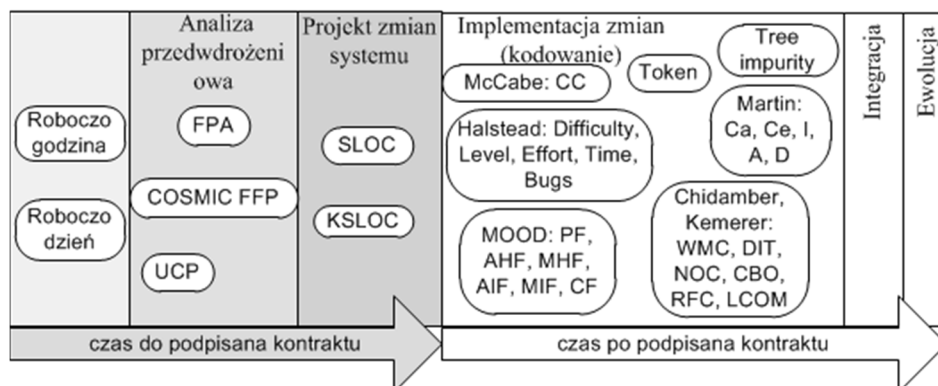
Istnieją inne, bardziej zaawansowane sposoby reprezentacji wymagań takie, jak diagram przypadków użycia, np. w UML (ang. Unified Modeling Language) [9], czy model procesów w BPMN (ang. Business Process Model and Notation) [10]. Specyfikacja wymagań w UML wykorzystuje jednocześnie wiele perspektyw opisu danych i procesów, przez co jej złożoność wyklucza wspólną weryfikację przez dostawcę i klienta. Z kolei modele BPMN, mimo, że z odpowiednim poziomem szczegółowości opisują procesy i są czytelne dla klientów, to zawierają jedynie szcątkowe informacje o obiektach danych.

Istotność tego etapu wdrożenia (specyfikacja wymagań klienta) wynika z tego, że jest on podstawą dalszych prac dostawcy, począwszy od szacowania kosztów, poprzez projektowanie, a skończywszy na implementacji systemu.

2. Wymiarowanie oprogramowania

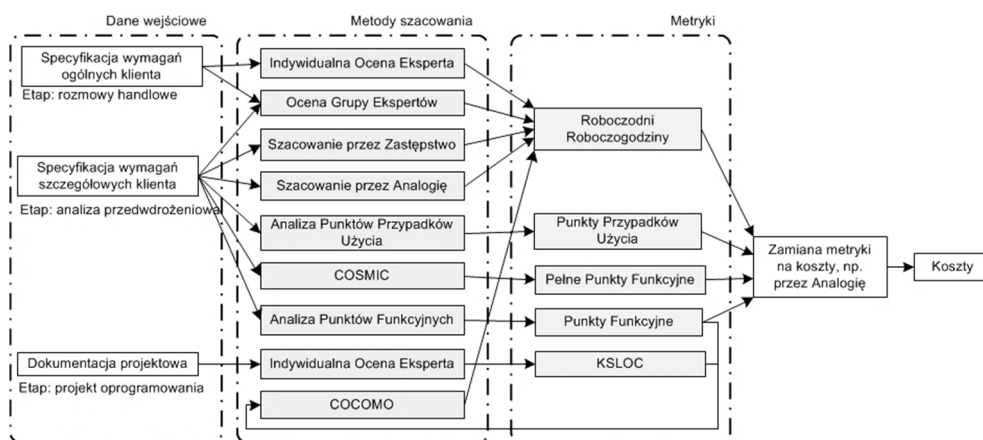
Efektom większości znanych metod wymiarowania oprogramowania [3] są metryki, które następnie dostawcy przeliczają na wartość kosztów po uwzględnieniu dodatkowych specyficznych dla danego projektu czynników. Metryki są formalnym zapisem pewnych własności oprogramowania [11]. Ich wartości powinny obiektywnie opisywać daną cechę mierzonego oprogramowania. Oznacza to, że muszą być niezależne od metody i osoby prowadzącej proces mierzenia. Tylko niektóre z metryk odzwierciedlają koszty wytworzenia tegoż oprogramowania. Na przykład, ilości SLOC (ang. Source of Line Code) lub KSLOC (ang. Kilo Source of Line Code) odpowiada pewnej pracochłonności przeznaczonej na wytworzenie kodu, a ta przekłada się na koszty. Z kolei popularny zestaw metryk Chidamera i Kramera [12] opisujący właściwości programów obiektowych takich, jak WMC (waga metody na klasę) czy DT (głębokość drzewa dziedziczenia) nie odpowiadają bezpośrednio kosztom realizacji danego fragmentu oprogramowania, którego dotyczą.

Wśród tych metryk, które odpowiadają kosztochłonności, tylko nieliczne dają się wyznaczyć przed etapem wytworzenia oprogramowania (implementacji zmian, kodowaniem), a tylko takie metryki są istotne dla procesu szacowania kosztów na wczesnych etapach. Na przykład, metryki McCabe'a wyznaczyć można dopiero, kiedy znany jest kod gotowego programu, co w przypadku szacowania kosztów oprogramowania, dyskwalifikuje ją. Interesujące metryki to wspomniana wcześniej SLOC oraz grupa metryk punktów funkcyjnych (FP [13], COCMIC FFP [14], UCP [15]). Określenie ilości kodu źródłowego SLOC możliwe jest dopiero na etapie projektu oprogramowania. W praktyce wdrożeń, nie często można tak bardzo opóźnić moment określenia wartości prac i podpisania kontraktu. Ilości punktów funkcyjnych wyznaczone są wcześniej, bo już na etapie analizy przedwdrozeniowej, na podstawie wymagań klienta. Dla dostawców tworzących oprogramowanie metodami zwinnymi (ang. agile), metryką mogą być historie użytkownika (ang. user stories). Określenie złożoności oprogramowania za pomocą tej metody jest znacznie mniej obiektywne niż za pomocą SLOC lub punktów funkcyjnych. Na etapie rozmów handlowych nie istnieją żadne metryki, dlatego wykorzystywane są bezpośrednio miary czasochłonności takie, jak *roboczogodzina* czy *roboczodzień*. Dostawcy właśnie w tym okresie oczekują określenia kosztów wdrożenia (w tym zmian oprogramowania). Możliwość wyznaczenia metryk na poszczególnych etapach cyklu życia przedstawiono na Rysunku 1.



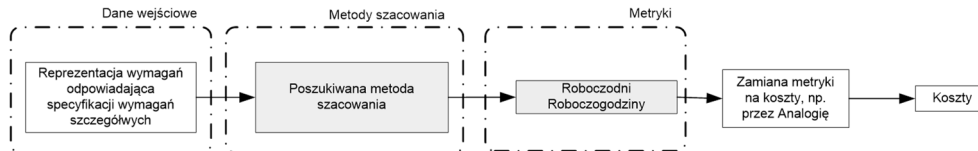
Rys. 1. Możliwość wyznaczenia metryk na poszczególnych etapach cyklu życia oprogramowania

Zauważyć należy, że dla procesu szacowania kosztów istotne są jedynie trzy pierwsze etapy cyklu życia oprogramowania, w których dochodzi do podpisania kontraktu. Później, mimo, że znacznie przybywa metryk, szacowanie już się nie odbywa - kontrakt został podpisany lub negocjacje zerwane. Powiązanie metryk i odpowiednich metod szacowania pokazano na Rysunku 2.



Rys. 2. Powiązanie metryk i metod szacowania

Wracając do określonego we wstępie problemu poszukiwania metody, można zauważyć, że powinna opierać się na takiej reprezentacji wymagań, która zapewni nie gorsze, niż w innych metodach, wynik szacowania. Ważną cechą wymagań na etapie rozmów handlowych jest ich szczegółowość i kompletność. Oczekiwaną metryką w tym wypadku jest *roboczegodzina* lub *roboczo dzień*. Na Rysunku 3 zaprezentowano model poszukiwanego rozwiązania.



Rys. 3. Model poszukiwanego rozwiązania

3. Ontologie w wymiarowaniu oprogramowania

Skoro, jak pokazano w rozdziale 1, wymagania klienta rozumiane jako ograniczony zakres wiedzy o przedsiębiorstwie dają się formalizować na wiele sposobów, jednym z nich może być ontologia, która jest uznanym i coraz częstszym sposobem formalizowania wiedzy.

Według T. Grubera [16]: ontologia to wyraźny, formalny opis konceptualizacji wybranej dziedziny, „tego co jest w rzeczywistości, w umysłach ludzi i zapisanego w postaci różnych symboli” [17]. Zawsze obejmuje pewien fragment, mniej lub bardziej dokładnie określony. *Wyższe ontologie* (ang. *upper ontologies*) opisują najbardziej podstawowe klasy bytów całej rzeczywistości, dziedzinowe - szczegółowe klasy niewielkiego wycinka rzeczywistości. Ontologie w informatyce tworzone są między innymi, aby umożliwić wielokrotne wykorzystanie wiedzy z danej dziedziny [18] (np. o systemach ERP), lub analizować wiedzę o konkretnej dziedzinie [19]. Do zapisu wiedzy używać można wielu narzędzi, które wykorzystują języki (np.: XML, RDF, OWL) [20] pozwalające na bezpośrednie wyrażanie pojęcia, organizowanie ich i łączenie, osadzanie pewnych aksjomatów, rozróżnianie, np. faktów od twierdzeń czy wnioskowania.

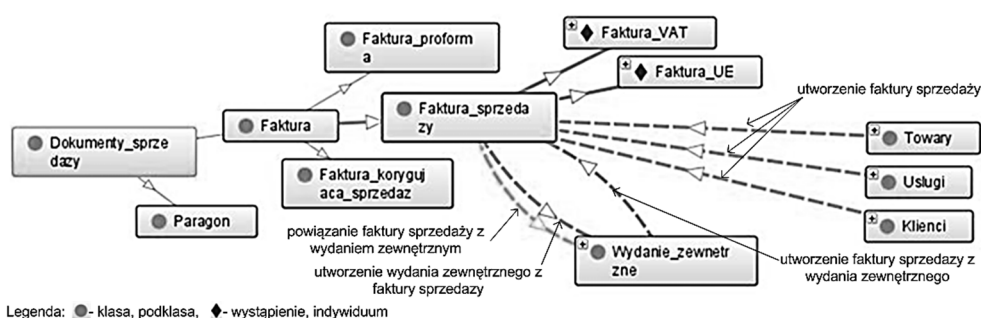
Wykorzystanie ontologii do implementacji systemów informatycznych można coraz częściej znaleźć w literaturze [21]. Użycie jej do wymiarowania kosztów oprogramowania implikuje to, że powinna zawierać informacje o samym systemie informatycznym, to znaczy o strukturach danych i procesach realizowanych przez system oraz związanych z nimi informacjami o kosztach wcześniejszych wdrożeń. Dodatkowo musi być uzupełniona o wymagania klienta, o ile nie odpowiadają im żadne wcześniej zdefiniowane elementy SI.

W tym kontekście ontologia pozwala zarówno na zapis klas SI wraz z relacjami między nimi (model statyczny), jak również na zapis procesów relacji między klasami (model dynamiczny). Z powodu kompletności i spójności informacji, zapis wymagań w ontologii jest pełniejszy, niż w pozostałych sformalizowanych postaciach opisanych w rozdziale 1.

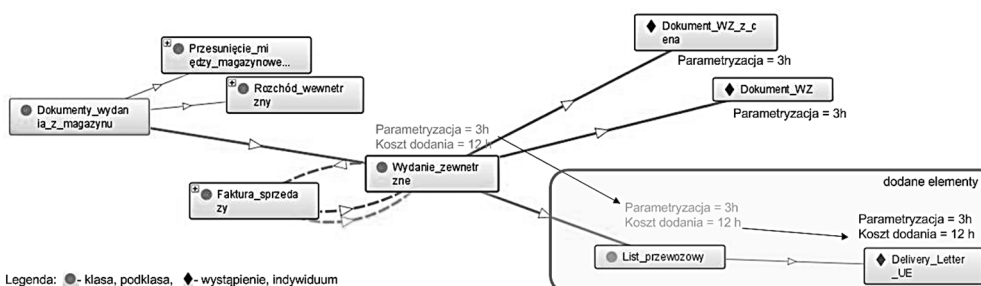
Na Rysunku 4 przedstawiono fragment ontologii zawierający wybrane wymagania klienta w zakresie dokumentów sprzedaży. Klient ewidencjonuje proces sprzedaży, między innymi fakturą VAT (indywidualnie, inaczej wystąpienie) *Faktura VAT*, który jest jednym z grupy dokumentów sprzedaży (klasa *Faktura_sprzedazy*). Faktura sprzedaży może zostać utworzona na podstawie danych o klientach (klasa *Klienci*), towarach (klasa *Towary*) i usługach (klasa *Uslugi*). Alternatywnym sposobem utworzenia faktury jest skopiowanie potrzebnych danych z istniejącego dokumentu wydania zewnętrznego (klasa *Wydanie_zewnetrzne*). Możliwa jest też odwrotna operacja: faktura sprzedaży jest podstawą do utworzenia dokumentu wydania zewnętrznego.

Do zapisania doświadczenia dostawcy z wcześniejszych wdrożeń wykorzystane są właściwości obiektów w ontologii (klas, wystąpień, relacji). Można zapisać w nich metryki służące do późniejszego wymiarowania kosztów. Na przykład, niech w ontologii SI w

części logistycznej, występuje klasa *Wydanie_zewnetrzne*. Klasa ta posiada metrykę *Parametryzacja=3h*, oraz *Koszt_dodania = 12h*. Jeśli w trakcie rozmów z klientem dostawca pozyska informacje, że klient wystawia nie tylko dokument WZ (wydanie zawewnętrzne), ale dodatkowo dla odbiorców zagranicznych list przewozowy (ang. Delivery Letter UE), to w ontologii zostanie dodana klasa *List_przewozowy* i wystąpienie *Delivery_Letter_UE*, które odziedziczy metryki (właściwości) z klasy nadrzędnej. W procesie szacowania wdrożenia oznaczać to będzie, że czas potrzebny na odpowiednią modyfikację oprogramowania wyniesie 12 godzin, a parametryzacja utworzonego w ten sposób raportu (wydruku *List_przewozowy*) kolejne 3 godziny. Powyższy przykład zilustrowany jest na Rysunku 5.



Rys. 4. Przykład reprezentacji wiedzy o dokumentach sprzedaży (przygotowano w Protege) [22]



Rys. 5. Przykładowe właściwości klas i relacji wykorzystane jako metryki oprogramowania

W analogiczny sposób odbywa się szacowanie relacji. Wymiarowanie wszystkich wystąpień w klasach i relacji określa wymiar oprogramowania opisanego przez daną ontologię. Jeśli metryki przypisane do klas i relacji będą miały wymiary kosztów lub dające się bezpośrednio przeliczyć (np. czasochłonność), to ich suma odpowiadać będzie kosztom danego oprogramowania. Powyższy sposób prowadzi do utworzenia ontologii kosztów wdrożenia.

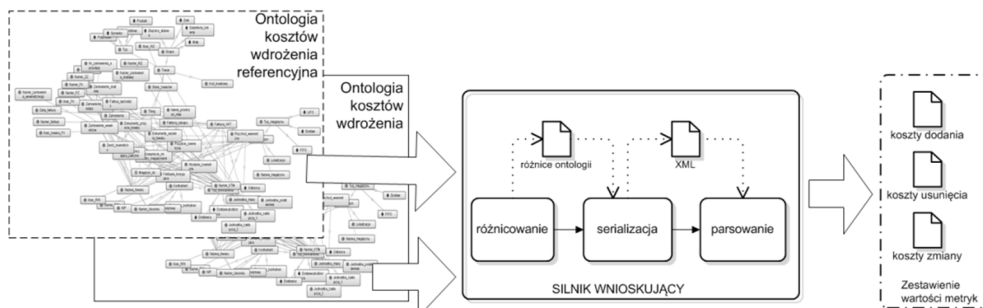
Celowym jest automatyczne dziedziczenie metryk do klas potomnych na etapie dodawania elementów ontologii, jednak dopuszczalne jest późniejsze modyfikowanie ich wartości na podstawie, np. *Indywidualnej Oceny Eksperta*.

4. Propozycja rozwiązania

Wykorzystując spostrzeżenia z poprzednich rozdziałów, proponuje się rozwiązanie (dalej uszczegółowione) wykorzystujące metodę szacowania w oparciu o ontologię kosztów wdrożenia. Efektem końcowym (wynikiem) powinien być zestaw metryk opisujących koszty zmian wdrażanego SI.

Zakładając, że dostawca posiada model ontologiczny oferowanego produktu wraz z metrykami niezbędnymi do szacowania, można przyjąć, że zna również dokładne koszty wdrożenia takiego systemu, a w szczególności koszty potencjalnych zmian oprogramowania. Zatem różnica w kosztach pomiędzy wdrożeniem SI uwzględniającego wymagania klienta i SI standardowego mają odzwierciedlenie w różnicach w ontologiach. Inaczej mówiąc zakłada się, że w ramach rozmów handlowych wystarczy wyspecyfikować i wprowadzić do znanej ontologii zmiany wynikające z wymagań klienta, aby móc na tej podstawie określić ich koszty.

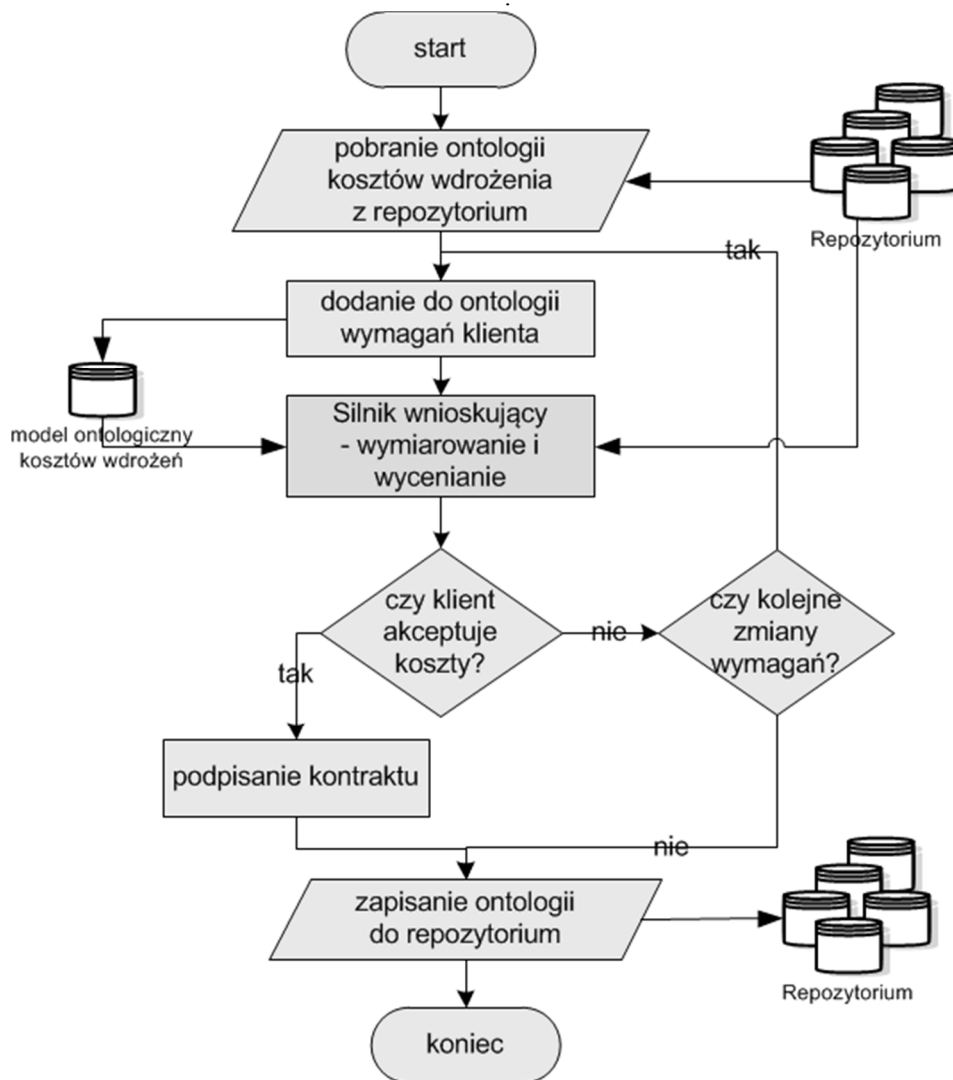
W szczególności, proponowane podejście polega na wyznaczeniu różnic między ontologią zawierającą dodatkową wiedzę o wymaganiach klienta, a ontologią referencyjną. Efektem różnicowania jest zbiór pojedynczych lub powiązanych między sobą obiektów takich jak, klasy, relacje lub wystąpienia. Następnie realizowana jest zmiana zapisu różnic do formatu szeregowego danych (serializacja) i parsowanie (analiza struktury gramatycznej), które doprowadza do wybrania i zsumowania wartości metryk istotnych dla kosztów wdrożenia. Wyznaczanie sumarycznej wartości odpowiednich metryk musi uwzględniać (odejmować) wartości metryk związanych z usuniętymi klasami i relacjami. Model silnika wnioskującego przedstawiono na Rysunku 6.



Rys. 7. Model silnika wnioskującego

Przedstawione rozwiązanie, wykorzystujące model ontologiczny jako formalizację wymagań klienta, pozwala na prowadzenie szacowania począwszy od etapu rozmów handlowych. Koszt przeprowadzenia wyceny jest na tyle znikomy, że wielokrotne szacowanie, po każdej zmianie (uzupełnieniu) wymagań klienta, nie obciążą znacząco budżetu dostawcy. Te cechy powodują, że można aktywnie kontrolować koszt oprogramowania w powiązaniu z budżetem zadanym przez klienta.

Opisany sposób szacowania dostawca może stosować w negocjacjach, w których klient najpierw chce znać koszt wdrożenia wersji standardowej, a później koszty przystosowania oprogramowania do jego organizacji. Scenariusz ten przedstawiony jest za pomocą algorytmu na Rysunku 7.



Rys. 8. Scenariusz negocjacji z klientem przy użyciu silnika wnioskującego

Reasumując, proponowane rozwiązanie polega na przygotowaniu danych wejściowych w postaci ontologii zawierającej wiedzę o systemie informatycznym, kosztach wcześniejszych wdrożeń oraz wymaganiach klienta i na tej podstawie rozpoczęciu procesu wnioskowania o kosztach przyszłego wdrożenia.

5. Przykład zastosowania

Weryfikację proponowanej metody przeprowadzono na przykładzie szacowania kosztów zmian SI w średnim przedsiębiorstwie produkcyjnym z branży wyrobów dla przemysłu okrętowego. Po roku od wdrożenia standardowej wersji SI w obszarze produkcji, klient zgłasza dostawcy nowe wymaganie związane z certyfikatami bezpieczeństwa wyrobu wprowadzonego do produkcji i określa, że przeznacza na tę zmianę budżet w wysokości 10000 zł. W trakcie rozmów handlowych ujawnia między innymi że, zamawiane surowce z grypy **A** dostarczane są z certyfikatami bezpieczeństwa, które należy dołączyć do wyrobu gotowego sprzedawanego odbiorcy, a kopie w wersji elektronicznej przechowywać w archiwum. Na tym etapie, jedynymi metodami szacowania możliwymi do zastosowania są *Indywidualna Ocena Eksperta* lub *Ocena Grupy Ekspertów*. Dostawca szacuje wartość prac na 5600 zł, co oznacza możliwość podjęcia się realizacji prac. W trakcie analizy wdrożeniowej dostawca uszczegółowia wymagania o dokładną zawartość nowego dokumentu (ilość i typy pól) oraz uzyskuje informację, że certyfikaty zawierają dodatkowo informacje dotyczące sposobu montażu, które powinny zostać umieszczone na dokumencie RW (rozchód wewnętrzny), wydrukowane i przekazane pracownikowi na wydziale produkcji. Ponadto, klient w przypadku serwisu (naprawy) swojego wyrobu musi odszukać odpowiedni certyfikat. Po skompletowaniu wymagań, koszty zmian SI dostawca szacuje metodą Analizy Punktów Funkcyjnych i po skorygowaniu otrzymuje wartość 32 FP (ang. Function Point). Na podstawie wcześniejszych wycen, dostawca określa wartość 1 FP równą 230 zł, co daje wartość szacowania 7360 zł.

Jednocześnie przeprowadza się wycenę korzystając z modelu ontologicznego. Pierwsze szacowanie dokonuje się na etapie rozmów handlowych. Na podstawie wstępnie ujawnionych wymagań klienta, ontologię referencyjną zwiększa się o 2 tożsame klasy powiązane z klasami *Dokumenty_zakupu* i *Dokumenty_sprzedaży*, dwa wystąpienia (*Certyfikat_dok* i *Certyfikat_img*) oraz 12 relacji między dodanymi, a istniejącymi elementami. Klasy i wystąpienia odziedziczają metrykę *koszt_dodania* z obiektów nadrzędnych, natomiast w dodanych relacjach wartość tej metryki określa ekspert. Po wyznaczeniu różnicy między zmienioną ontologią i referencyjną, łączna wartość przypisana do metryki *koszt_dodania* wynosi 6400 zł. Dostawca na etapie rozmów handlowych może zapewnić klienta o możliwości realizacji prac w zadanym budżecie. Po analizie przedwdrożeniowej dodaje się kolejne 9 relacji i szacowanie zwiększa się do 7900 zł.

Po realizacji projektu, analiza raportów z wykonania wykazuje, że wartość prac przekazanych klientowi wynosi 8600 zł. Porównanie błędów szacowania każdej z metod przedstawiono w Tabeli 1.

Tab. 1. Porównanie błędów na pierwszych etapach wdrożenia

Metoda szacowania	Etap rozmów handlowych	Etap analizy przedwdrożeniowej
Ocena grupy ekspertów i Analiza Punktów Funkcyjnych	35%	14%
Metoda wnioskowania z ontologii	26%	8%

Na podstawie analizowanego przypadku, zauważyć można, że na obu etapach wdrożenia, szacowanie kosztów w oparciu o wnioskowanie z ontologii dało nie gorsze wyniki, niż znane metody.

Wnioski

Powyższe rozważania potwierdzają tezę, że metoda szacowania oprogramowania oparta na zapisie wymagań klienta za pomocą ontologii jest możliwa do stosowania od momentu rozpoczęcia rozmów handlowych i daje nie gorsze wyniki, niż inne znane metody na tym i kolejnym etapie. Dodatkowo, cecha aktywnego szacowania skraca i usprawnia proces wyceny kosztów w trakcie negocjacji z klientem. Należy zauważyć, że proces szacowania nie wymaga od wykonawcy specjalistycznej wiedzy, jak na przykład Analiza Punktów Funkcyjnych, przez co nie jest kosztowny. Jednakże dostawca musi ponieść pewien koszt utworzenia i pielęgnowania ontologii oraz utrzymania repozytorium.

Ponieważ sprawdzenie metody opisane w poprzednim rozdziale, zrealizowano na podstawie stosunkowo niewielkiego zakresu zmian oprogramowania, wskazane by było przeprowadzenie dalszej weryfikacji wyników wykorzystując większy zakres materiału badawczego.

Ponadto, należy zwrócić uwagę, że podstawowym kryterium oceny każdej metody szacowania jest wielkość błędu jakim obarczone są wyniki. Błąd znany jest jednak dopiero po realizacji prac poprzez porównanie wartości szacowanej z rzeczywistymi kosztami. Dostawcy mają potrzebę określenia spodziewanego błędu lub zakresu wyniku szacowania wcześniej, już na etapie negocjacji kontraktu. Na przykład, jeśli z szacowania danego zakresu prac wyznaczona jest wartość 150 tys. zł, to do zawarcia kontraktu istotna jest dla dostawców wiedza, czy można oczekiwać kosztów realizacji w niewielkim zakresie odchylenia od szacowanej wartości (np. 140-160 tys. zł) czy znacznym (np. 120-220 tys. zł). Określenie zakresu niepewności szacowania spowoduje odpowiednie zwiększenie lub zmniejszenie wartości negocjowanego kontraktu. Z tego powodu, kierunkiem przyszłych badań powinna być modyfikacja proponowanej metody o zastosowanie rozmytych wartości metryk. Pozwoli to na otrzymanie rozmytych wyników szacowań, co z kolei pomoże dostawcom zorientować się w zakresie szacowanych wartości i odpowiednio negocjować kontrakt.

Literatura

1. Burns M.: How to select and implement an ERP System, 2005. dostępne: <http://www.180systems.com/ERPWhitePaper.pdf>, [Data uzyskania dostępu: 10.10.2013].
2. Kotarba M. :Problemy występujące podczas modyfikacji standardowego systemu ERP i sposoby ich pokonania na przykładzie wdrożenia systemu Oracle JD Edwards Enterprise One w przedsiębiorstwie z branży spożywczej. W Systemy Wspomagania Organizacji 2007, Katowice, 2007.
3. McConell S.: Szacowanie oprogramowania, Microsoft Press, 2006.
4. Plecka P.: Selected Methods of Cost Estimation of ERP Systems' Modifications, Zarządzanie Przedsiębiorstwem, nr 4, 2013, str. 27-34.
5. Plecka P., Bzdyra K.: Algorithm of Selecting Cost Estimation Methods for ERP Software Implementation. W Foundations of Management - International Journal, Warszawa, 2013, rekomendowane do druku.
6. Lech P.: Zintegrowane systemy zarządzania ERP/ERP II. Wykorzystanie w biznesie, wdrażanie. Difin, Warszawa, 2003.

7. Allen P., Frost S.: *Component-Based Development for Enterprise Systems, Applying the Select Perspective*. Cambridge: Cambridge University Press, 1998.
8. ARIS, dostępne: <http://www.softwareag.com>, [Data uzyskania dostępu: 27.12.2013].
9. UML Resource Page, Object Management Group , dostępne: <http://www.uml.org/>, [Data uzyskania dostępu: 27.12.2013].
10. BPMN, Object Management Group, dostępne: <http://www.bpmn.org/>, [Data uzyskania dostępu: 27.12.2013].
11. 1061-1998 - IEEE Standard for a Software Quality Metrics Methodology, Oversight Committee:C/S2ESC - Software & Systems Engineering Standards Committee, 1998.
12. Kamerer C. F., Chidamber S. R.: *Towards a metrics suite for object oriented design*. W OOPSLA '91 Conference proceedings on Object-oriented programming systems, languages, and applications., New York, 1991.
13. International Function Point Users Group , dostępne: <http://www.ifpug.org/>, [Data uzyskania dostępu: 27.12.2013].
14. Common Software Measurement International Consortium, dostępne: <http://www.cosmicon.com/>. [Data uzyskania dostępu: 16 11 2013].
15. Schneider G., Winters J. P.: *Applying Use Cases: A Practical Guide*. Addison Wesley Publishing Company Incorporated , 2001.
16. Gruber T. R.: *A Translation Approach to Portable Ontology Specifications*, Technical Report KSL 92-71. Knowledge Systems Laboratory, Computer Science Dept. Stanford University, Stanford, California, 1993.
17. Mealy G. H.: *Another look at data*. Fall Joint Computer Conference, Anaheim, California, 1967.
18. Noy N. F., McGuinness D. L.: *Ontology Development 101: A Guide to Creating Your First Ontology*. Stanford Knowledge Systems Laboratory Technical Report KSL-01-05, Stanford, California, 2001.
19. Czarnecki A., Sitek T.: *Ontologies vs. Rules — Comparison of Methods of Knowledge Representation Based on the Example of IT Services Management*. Information Systems Architecture and Technology: Intelligent Information Systems, Knowledge Discovery, Big Data and High Performance Computing., Wrocław, Oficyna Wydawnicza Politechniki Wrocławskiej, 2013, str. 99–109.
20. Goczyła K.: *Ontologie w systemach informatycznych*. Akademicka Oficyna Wydawnicza EXIT, Warszawa, 2011
21. Czarnecki A., Orłowski C.: *Możliwości zastosowania ontologii do oceny technologii informatycznych*. W *Komputerowo zintegrowane zarządzanie*. tom 2 pod red. Ryszarda Knosali, Opole, Oficyna Wydawnicza Polskiego Towarzystwa Zarządzania Produkcją, 2007, str. 143-152.
22. *Welcome to protégé*, Stanford Center for Biomedical Informatics Research, Stanford University School of Medicine, dostępne: <http://protege.stanford.edu/>. [Data uzyskania dostępu: 16 12 2013].

Mgr inż. Przemysław Plecka
 Dr. inż. Krzysztof Bzdyra
 Politechnika Koszalińska, Wydział Elektroniki i Informatyki
 ul. Śniadeckich 2, 75-453 Koszalin
 e-mail: przemek.plecka@gmail.com
krzysztof.bzdyra@tu.koszalin.pl