

Wspomaganie procesu definiowania zadań wykonywanych w ramach projektu informatycznego

Tomasz Gratkowski

Streszczenie: Stosowane obecnie zwinne metodyki projektowania wymagają ciągłego dopasowywania procesu wytwórczego w trakcie tworzenia systemu informatycznego. W artykule tym została zaprezentowana koncepcja wykorzystania definicji zadań wielokrotnego użytku, których użycie uprości i przyspieszy proces adaptacji metodyki do zmieniających się wymagań projektowych.

Słowa kluczowe: metodyki procesu wytwórczego oprogramowania, wspomaganie dostosowywania procesu wytwórczego, Open Unified Process

1. WPROWADZENIE

Znaczący wzrost złożoności wytwarzanego oprogramowania komputerowego wymusił na firmach informatycznych stosowanie procesów wytwórczych oprogramowania (ang. software development process, SDP). Przyczyniło się to do zredukowania problemu przekraczania ustalonego budżetu oraz pozwoliło na dotrzymanie terminu wyznaczonego na budowę systemu. Łatwiej również zarządza się ryzykiem niepowodzenia inwestycji informatycznej. Zdefiniowany szkielet SDP zawierający określoną strukturę, plan oraz kontrolujący wykonywane zadania nazywany jest metodyką wytwarzania oprogramowania (ang. software development methodology, SDM) [13].

Każdy zespół informatyczny lub nawet projekt może wymagać stosowania innej SDM. Dlatego rozpoczęto opracowywanie iteracyjnej i przyrostowej metodyki, którą można zaadaptować na potrzeby każdego projektu. Jedną z pierwszych metodyk spełniających owe kryteria jest Rational Unified Process [4], jednak ze względu na złożoność i duży rozmiar jest uważana za trudną do wykorzystania. Z drugiej strony istnieją metodyki zwinne typu Agile [6] takie jak Scrum [5] lub Extreme Programming [1], które z racji przyjętego niesformalizowanego podejścia, spotykają się często z nieprzychylnością zarządzających procesem wytwórczym. Wychodząc naprzeciw oczekiwaniom twórców systemów informatycznych, opracowano metodykę Open Unified Process (OpenUP) [8], która zawiera najlepsze praktyki pochodzące zarówno z RUP, jak i z metodyk zwinnych (ang. agile).

Proces zarządzania procesem wytwórczym wymaga ciągłej adaptacji do zmieniających się wymagań

i specyfiki projektu. Dlatego nasuwa się pytanie: czy istnieje możliwość wspomaganie procesu adaptacji metodyki? Adaptacja metodyki jest tutaj rozumiana jako dopasowanie zadań wykonywanych w ramach metodyki, tak aby ich późniejsza realizacja pozwoliła na utworzenie systemu informatycznego zgodnego z wymaganiami funkcjonalnymi. W poniższym artykule zaprezentowano ideę bazującą na wykorzystaniu zadań wielokrotnego użytku, które mogą być zaadaptowane w ramach różnych projektów. Wprowadzenie wspomaganie przyspieszy i uprości proces adaptacji metodyki. Na obecnym etapie, badania dotyczą zadań z dyscypliny zarządzanie projektem z metodyki OpenUP. W szczególności badania dotyczą zarządzania iteracjami oraz przypisanymi do iteracji zadaniami wykonywanymi w ramach pozostałych dyscyplin. Artefaktem przechowującym listę iteracji i zadań jest plan projektu.

2. OPENUP

Metodyka OpenUP oparta jest na czterech głównych zasadach:

- 1) współpraca, w celu lepszego działania i zrozumienia w ramach zespołu;
- 2) równoważenie sprzecznych priorytetów, w celu maksymalizacji korzyści dla odbiorcy systemu;
- 3) jak najszybsza koncentracja na architekturze systemu, w celu minimalizacji ryzyka oraz organizacji fazy wytwarzania;
- 4) rozwój systemu w ramach małych częstych iteracji na podstawie zbieranych opinii od odbiorcy systemu.

Powyższe zasady odpowiadają regułom zaproponowanym w ramach manifestu Agile (Tabela 2).

Tabela 2.

Zestawienie zasad OpenUP z regułami manifestu Agile

Zasady OpenUP	Reguły manifestu Agile
1	ważniejsi są ludzie i ich wzajemne interakcje (współdziałanie) ponad procedury i narzędzia
2	należy przedkładać współpracę z klientem nad negocjację umów

3	istotniejsze jest działające oprogramowanie nad wyczerpującą dokumentację
4	należy reagować na zmiany niż realizować wcześniej założony planu

OpenUP jest zorganizowana w dwóch różnych skorelowanych obszarach: zawartości metody oraz zawartości procesu. Zawartość metody definiuje elementy metodyki (role, zadania, artefakty oraz porady), bez wskazywania jak zostaną użyte. Zawartość procesu składa się z szablonów zastosowania poszczególnych elementów metodyki. Wiele różnych projektów może być budowanych na podstawie tego samego zbioru elementów metody.

OpenUP opisuje organizację pracy na trzech poziomach: osobistym, zespołu oraz odbiorcy systemu. Na poziomie osobistym członek zespołu wykonuje mikro przyrosty, które reprezentują uzyskane wyniki kilkugodzinnej lub kilkudniowej pracy. Zespół realizuje kolejne zaplanowane iteracje, w ramach których dostarczane są odbiorcy systemu kolejne elementy. Na poziomie odbiorcy systemu OpenUP została podzielona na cztery fazy:

- rozpoczęcie (ang. inception);
- opracowanie (ang. elaboration);
- wytworzenie (ang. construction);
- przekazanie (ang. transition).

Tak zdefiniowany proces wytwórczy pozwala odbiorcy systemu na prowadzenie nadzoru nad finansami, zakresem oraz ryzykiem projektu.

Metodyka OpenUP posiada również zdefiniowane role członków zespołu, którzy wykonują składające się z kroków zadania, produkując artefakty. Zadania są pogrupowane w dyscypliny, natomiast artefakty w dziedziny.

W ramach fazy „rozpoczęcie” menadżer projektu definiuje plan projektu, natomiast w trakcie trwania projektu odpowiedzialny jest za tworzenie, zarządzanie oraz kontrolowanie wykonania iteracji. Rola menadżera projektu jest kluczowa z punktu widzenia wspomagania adaptacji metodyki, ponieważ omawiany mechanizm ma uprościć oraz częściowo zautomatyzować pracę zdefiniowaną w OpenUP w ramach tej roli.

Do definicji OpenUP zastosowano specyfikację Software Process Engineering Meta-Model (SPEM) [9], dzięki czemu metodyka OpenUP może być w dowolny sposób dostosowywana do potrzeb projektu lub firmy. Dodatkowo specyfikacja SPEM posiada mechanizm umożliwiający rozszerzenie formy opisu metodyki o niezbędne z punktu widzenia prezentowanego rozwiązania formalizmy.

3. SOFTWARE PROCESS ENGINEERING META-MODEL

Specyfikacja wychodzi naprzeciw oczekiwaniom ujednolicenia definiowania procesu wytwarzania oprogramowania. SPEM jest meta-modelem, pozwalającym definiować procesy konstruowania i budowania systemów informatycznych oraz wchodzące

w ich skład komponenty. SPEM jest również profilem UML [10], co pozwoliło na zdefiniowanie stosowanych w SPEM stereotypów oraz na wykorzystanie diagramów UML do opisu poszczególnych elementów specyfikacji SPEM. Wersję 2.0 dopasowano do obowiązujących standardów, tworząc architekturę zgodnie ze standardem MOF 2.0 [11], który służy do definiowania gramatyki języków graficznych. W specyfikacji SPEM położono duży nacisk na elastyczność meta-modelu, tak aby w łatwy sposób można go było wykorzystywać w procesie implementacji narzędzi wspierających proces wytwarzania oprogramowania. Profil został dopasowany w szczególności do metodyk wywodzących się z Unified Software Development Process [3] takich jak RUP oraz OpenUP, gdzie inżynier zarządzający procesem wytwórczym dopasowuje przepływ zadań do konkretnego projektu. W ramach specyfikacji SPEM wykorzystano czterowarstwową hierarchię meta-modelu, gdzie w warstwie M0 znajduje się konkretna instancja projektu systemu (tabela 1). Na warstwie M1 umieszczana jest definiowana metodologia projektowania systemu informatycznego. Natomiast na warstwie M2 i M3 umiejscowione są odpowiednio UML i MOF. UML jest meta-modelem opisującym dozwolone konstrukcje na warstwie M1, natomiast MOF jest meta-modelem dla warstwy M2.

Tabela 1.
Czterowarstwowa hierarchia meta-modelu OMG dla specyfikacji SPEM

Warstwa	Opis	Użyte elementy
M3	meta-meta-model	MOF
M2	meta-model	UML+SPEM
M1	model	metodologia projektowania systemu informatycznego (np. OpenUP)
M0	implementacja	instancja projektu systemu informatycznego

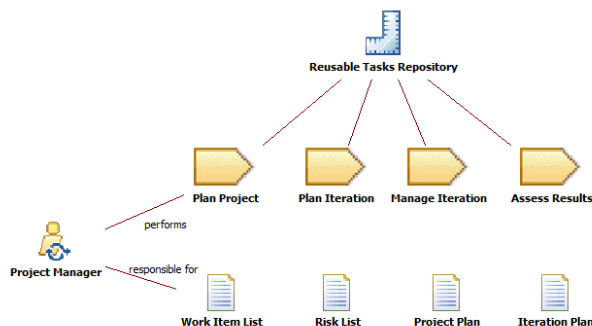
4. ZADANIA WIELOKROTNEGO UŻYTKU DLA DISCYPLINY ARCHITEKTURA Z METODYKI OPENUP

Zadanie utworzenia planu projektu posłuży do zaprezentowania opisywanej idei, która zostanie następnie zaadaptowana do innych zadań przypisanych dla roli menadżera projektu. W ramach zadania na podstawie technicznej specyfikacji systemu oraz listy czynności do wykonania tworzony jest artefakt wyjściowy plan projektu. Menadżer projektu w ramach zadania: identyfikuje potrzebnych członków zespołu, oszacowuje wielkość projektu, ocenia ryzyko, określa czas trwania projektu, organizuje iteracje w fazy, oszacowuje cenę oraz planuje proces wytwórczy. Krok poświęcony organizacji iteracji w fazy definiuje proces wytwórczy projektu. Wykonanie fazy ma na celu realizację kamienia milowego

(ang. milestone). Na podstawie wymagań funkcjonalnych zawartych w technicznej specyfikacji systemu należy określić kamienie milowe, które muszą zostać uzyskane w ramach projektu. Dla przedstawionego zadania proces adaptacji polega na zdefiniowaniu celów, jakie zostaną zrealizowane w ramach każdego kamienia milowego. Realizacja celów odbywa się dzięki wykonaniu przypisanych zadań. Otrzymany proces wytwórczy zawarty w planie projektu powinien być na bardzo wysokim poziomie abstrakcji, a jego uszczegółowienie odbywa się w kolejnych zadaniach przypisanych do roli menadżera projektu.

W proponowanym rozwiązaniu proces wyboru zadań, jakie powinny być wykonane w celu realizacji wymagań funkcjonalnych, został uzupełniony o mechanizm wspomagający. W repozytorium zadań wielokrotnego użytku (ang. Reusable Tasks Repository, RTR) odszukiwane będą zadania na podstawie zdefiniowanych kryteriów. Na rysunku 1 przedstawiono diagram w specyfikacji SPEM, który wskazuje zadania wykorzystujące RTR jako „podpowiedź” (ang. guideline), w celu realizacji przypisanych zadaniom artefaktów wyjściowych.

Aby możliwe było odszukiwanie zadań, należy określić kryteria wyboru, które we wstępnej fazie powinny bazować na zbiorach rozwiązań biznesowych dla wybranej dziedziny problemu informatycznego. Zastosowanie czterowarstwowej architektury w ramach specyfikacji SPEM, tak jak w przypadku UML, umożliwia jej rozszerzanie poprzez tworzenie profili. Zastosowanie profilu pozwala np. na wprowadzenie reguł do definicji zadań wykonywanych w ramach metodyki [2] oraz innych dodatkowych własności, które pozwolą oszukać zadania na podstawie zbioru kryteriów. Wybrane zadania wielokrotnego użytku stanowią szablony, które następnie



Rys. 1. Użycie repozytorium zadań wielokrotnego użytku jako wskazówek dla zadań przypisanych dla roli menadżera projektu

mogą być dopasowywane do potrzeb konkretnego projektu. Przykłady szablonów zadań są skutecznie wykorzystywane np. w środowisku Jazz w ramach narzędzia Rational Team Concert firmy IBM [7], które jest mocno rozwijane przez firmę IBM jako środowisko integrujące i wspomagające proces produkcji systemów informatycznych z wykorzystaniem między innymi metodyki OpenUp do zarządzania projektem informatycznym. Zaproponowana idea repozytorium została skutecznie wykorzystana w ramach projektu ReDSeeDS [12], gdzie odszukiwane są ponownie używalne „przypadki” (ang. cases), które stanowią

kompletny zestaw powiązanych artefaktów powstających w procesie wytwarzania oprogramowania.

5. PODSUMOWANIE

Przedstawiona idea proponuje nowe spojrzenie na proces adaptacji metodyk projektowania. Zastosowanie mechanizmu wspomagania adaptacji, na podstawie repozytorium zadań wielokrotnego użytku, uprości i przyspieszy proces adaptacji metodyki na potrzeby różnych projektów informatycznych. Dodatkowym celem prowadzonych badań będzie utworzenie otwartego repozytorium, które może zostać użyte w już istniejących środowiskach wspomagających zarządzanie projektem np. w ramach projektu Jazz.

LITERATURA

- [1] Beck K., Andres C., Extreme Programming Explained: Embrace Change, Addison Wesley 2004.
- [2] Gratkowski T., „Rozszerzenie SPEM precyzując definicję zadań wykonywanych w krokach”, Metody i narzędzia wytwarzania oprogramowania, Szklarska Poręba 2007, Oficyna Wydaw. Politechniki Wrocławskiej, 2007, ss. 87-98.
- [3] Jacobson I., Booch G., Rumbaugh J., „The Unified Software Development Process”, Addison-Wesley 1999.
- [4] Kruchten, P., The Rational Unified Process: An Introduction, Addison-Wesley, 2003.
- [5] Schwaber K., Beedle M., Agile Software Development with Scrum, Prentice Hall, 2002.
- [6] Shore J., The Art of Agile Development, O'Reilly 2007.
- [7] Jazz Home Page, jazz.net.
- [8] Introduction to OpenUP (Open Unified Process), www.eclipse.org/epf/.
- [9] OMG, Software & Software Process Engineering Meta-Model (SPEM 2.0), April 2008.
- [10] OMG, OMG Unified Modeling Language, Superstructure, Version 2.2, February 2009.
- [11] OMG, Meta Object Facility (MOF) Core Specification, Version 2.0, January 2006.
- [12] ReDSeeDS, ReDSeeDS Software Development Methodology, ReDSeeDS project, 07.07.2008.
- [13] The Centers for Medicare & Medicaid Services, Selecting a development approach, 27 Oct 2008.



dr inż. Tomasz Gratkowski

Uniwersytet Zielonogórski
Wydział Elektrotechniki, Informatyki
i Telekomunikacji
Instytut Informatyki i Elektroniki

ul. prof. Z. Szafrana 2
65-246 Zielona Góra

tel.: +48 68 32 82