Politechnika Warszawska
Wydział Elektroniki i Technik Informacyjnych

DZIEKAN I RADA WYDZIAŁU ELEKTRONIKI I TECHNIK INFORMACYJNYCH
POLITECHNIKI WARSZAWSKIEJ

zawiadamiają o

PUBLICZNEJ OBRONIE ROZPRAWY DOKTORSKIEJ

mgr. Szymona Kijasa

która odbędzie się w dniu 10 września 2019 r. o godzinie 11.00

na Wydziale Elektroniki i Technik Informacyjnych Politechniki Warszawskiej
Gmach im. Janusza Groszkowskiego, Warszawa ul. Nowowiejska 15/19, w sali nr 116

Tytuł rozprawy doktorskiej*:*

"Decyzje architektoniczne w ewolucji systemów o architekturze usługowej"

promotor: prof. dr hab. Włodzimierz Ogryczak z Wydział Elektroniki i Technik Informacyjnych Politechniki Warszawskiej,

recenzenci: dr hab. inż. Piotr Habela z Wydziału Informatyki Polsko-Japońskiej Akademii Technik Komputerowych,

dr hab. inż. Ilona Bluemke z Wydziału Elektroniki i Technik Informacyjnych Politechniki Warszawskiej.

Na stronie internetowej wydziału **www.elka.pw.edu.pl/Wydzial/Rada-Wydzialu/Harmonogram-obron-doktorskich-streszczenia-i-recenzje** znajdują się streszczenie rozprawy oraz recenzje, jak również dostęp do tekstu rozprawy umieszczonej w Bazie Wiedzy Politechniki Warszawskiej.

Dziekan

prof. dr hab. inż. Krzysztof Zaremba

Mgr Szymon Kijas

Promotor – prof. dr hab. inż. Krzysztof Sacha

Tytuł rozprawy doktorskiej:"Decyzje architektoniczne w ewolucji systemów o architekturze usługowej"

Streszczenie

Zasadniczym celem rozprawy było opracowanie oryginalnej metodyki ewolucji i rozwoju systemów usługowych MAD4SOA oraz stanowiącego integralna część tej metodyki oryginalnego modelu decyzji architektonicznych służącego dokumentowaniu ewolucji systemów usługowych. W ramach wykonanych prac badawczych rozwiązano szereg istotnych problemów badawczych stanowiących oryginalny wkład w dziedzinę inżynierii oprogramowania:

- Zdefiniowano proces ewolucji określający sposób wprowadzania zmiany do systemu usługowego wraz z produkowanymi w tym czasie artefaktami.
- Zdefiniowano sposób dokumentowania przebiegu ewolucji za pomocą decyzji architektonicznych opisujących rozwiązywane problemy architektoniczne, warianty rozwiązań rozważane przez projektantów oraz wybrane rozwiązania.
- Stworzono sformalizowany model prowadzenia ewolucji, obejmujący model systemu usługowego, model procesu wprowadzania zmiany i model decyzji architektonicznych. Formalny zapis elementów modelu ewolucji został wyrażony za pomocą deklaratywnego języka specyfikacji Alloy umożliwiającego modelowanie w oparciu o rachunek predykatów pierwszego rzędu, opracowanego przez naukowców z MIT.

Finalnie metodyka MAD4SOA została zweryfikowana na przykładzie bazującym na rzeczywistym systemie realizującym płatności natychmiastowe w sektorze bankowym w Polsce.

Słowa kluczowe: architektura systemów informatycznych, architektura usługowa, SOA, systemy usługowe, procesy biznesowe, usługi, decyzje architektoniczne.

Dr hab. inż. Ilona Bluemke
Instytut Informatyki, PW
Warszawa, Nowowiejska 15/19

Warszawa 22 stycznia 2019

Recenzja rozprawy doktorskiej

Tytuł rozprawy: **Decyzje architektoniczne w ewolucji systemów o architekturze usługowej**

Autor rozprawy: **mgr Szymon Kijas**

Zakres i cel rozprawy

Architektura oprogramowania, określająca jego podstawowe elementy i relacje występujące między nimi, jest projektowana na początku procesu budowy oprogramowania jednak decyzje, które wówczas są podejmowane mają kluczowy wpływ na działanie systemu i możliwość jego ewolucji. Decyzje architektoniczne są zwykle poprzedzone analizą istniejących w projekcie ograniczeń oraz możliwych do realizacji rozwiązań. Łańcuch decyzji architektonicznych wynika z wiedzy dotyczącej realizowanego projektu. Właściwa dokumentacja podjętych decyzji oraz racji, które przesądziły o ich wyborze, tworzy zasób wiedzy, który może ułatwić i przyśpieszyć procesy ewolucji systemu bez powracania do podobnych analiz. Recenzowana rozprawa zajmuje się tymi problemami. Autor skoncentrował się na architekturze usługowej (Service-Oriented Architecture – SOA). W systemach SOA poszczególne funkcje mogą być realizowane przez wyodrębnione systemy informatyczne a mechanizmy integracyjne umożliwiające współpracę w sposób zapewniający realizację wymaganych procesów biznesowych. Podstawowe decyzje architektoniczne podejmowane podczas tworzenia i ewolucji takich systemów dotyczą sposobów i mechanizmów współpracy usług, nie wchodząc w detale implementacyjne poszczególnych usług.

Celem pracy jest poszukiwanie metodyki ewolucji systemów usługowych i dokumentowania decyzji architektonicznych podejmowanych w ich toku.

Autor postawił sobie następujące **cele badawcze**:

- Opracować oryginalną metodykę ewolucji systemów usługowych

- Opracować model decyzji architektonicznych służący do dokumentowania ewolucji systemów usługowych.

Do realizacji powyższych celów Autor zdefiniował sposób wprowadzania zmiany do systemu usługowego wzorowany na abstrakcyjnym procesie zdefiniowanym w standardzie ISO/IEC 20000, oraz produkowane w tym czasie artefakty. Proces jest uruchamiany w skutek pojawienia biznesowego zapotrzebowania na zmianę i obejmuje: ocenę wpływu zmiany na bieżące wydanie systemu, zatwierdzenie i przekazanie zmiany do realizacji, implementację i wdrożenie zmiany oraz przegląd przeprowadzonej zmiany.

Autor zdefiniował także sposób dokumentowania przebiegu ewolucji za pomocą decyzji architektonicznych opisujących rozwiązywane problemy, warianty rozwiązań rozważane przez projektantów oraz wybrane rozwiązania. Model prowadzenia ewolucji, obejmujący model systemu usługowego, model procesu wprowadzania zmiany i model decyzji architektonicznych został wyrażony w języku Alloy i przeanalizowany za pomocą narzędzia Alloy Analyzer.

Zaproponowana w rozprawie metodyka ewolucji i rozwoju systemów usługowych została pokazana na przykładzie ciągu modyfikacji systemu płatności bankowych.

Teza rozprawy dobrze określa zakres pracy. Zarówno teza jak i cele badawcze są dobrze określone. Właściwe jest także ich wzajemne powiązanie. Wskazane cele reprezentują spójne, metodyczne podejście do uzasadnienia postawionej tezy.

Zawartość rozprawy

Rozprawa zawiera osiem rozdziałów, sto czterdzieści trzy strony i dwieście jedenaście pozycji literatury. Literatura jest nieaktualna. Najnowsza pozycja jest z 2015 roku (jedna), większość prac jest z lat 2003- 2012. Brak bardziej aktualnych pozycji w spisie literatury może wskazywać, że tematyka pracy jest już nieaktualna lub Autor nie zaktualizował literatury w końcowym etapie pracy. W rozprawie jest 66 rysunków, wiele z nich całostronicowych, według mnie nie wszystkie są potrzebne, do wielu z nich odwołania są lakoniczne.

Pierwsze dwa rozdziały zawierają wprowadzenie, cele oraz tezę pracy.

Rozdział trzeci jest wprowadzeniem w tematykę systemów usługowych, wprowadza podstawowe pojęcia i notacje dziedziny. Przedstawiony jest warstwowy model systemu usługowego firmy Oracle, wykorzystywany przez Autora w dalszych częściach pracy. Nie jestem przekonana o potrzebie pokazywania przykładu - rozdz. 3.3.

W rozdziale czwartym Autor zamieścił podstawowe pojęcia związane z decyzjami architektonicznymi, przedstawił modele decyzji zaproponowane w literaturze (Archium, RADM, taksonomie Kruchtena) a także krótko przedstawił narzędzia do graficznej reprezentacji relacji. Istniejące rozwiązania są kolejno opisane. W rozprawie doktorskiej oczekiwałabym bardziej krytycznego opisu metod, opisu na wyższym poziomie abstrakcji, wskazania zalet ale i ograniczeń. Autor wprowadził notację graficzną MAD, której jest współautorem.

W rozdziale piątym zamieszczono opis metod dotyczących systemów usługowych. Podobnie jak w rozdziale czwartym istniejące metodyki są kolejno opisywane. Wolałabym by opis był bardziej abstrakcyjny, by wskazano wspólne cechy i różnice. Na szczęście umieszczono tabelaryczne zestawienie zakresów poszczególnych metod (tabela 5.1). Następnie Autor opisuje swój model MAD4SOA wraz z relacjami spójności. Reguły te zostały zapisane w języku Alloy (rozdz.5.5) i sprawdzone w narzędziu Alloy Analyzer.

W kolejnym rozdziale Autor opisuje studium przypadku. Duża liczba mało czytelnych, całostronicowych rysunków nie ułatwia zrozumienia metodyki i jak dla mnie raczej zniechęca do praktycznego stosowania MAD4SOA.

W rozdziale siódmym Autor porównuje swoją metodykę MAD4SOa ze znanymi metodykami Kruchtena, Archium i RADM na prostym przykładzie.

Podsumowanie pracy znajduje się w rozdziale ósmym. Autor skopiował cele badawcze z rozdz. 1 „Wprowadzenie" i wymienił kierunki dalszych prac, niestety nie podsumował swojego podejścia wskazując wady, zalety proponowanej metodyki.

Uwagi szczegółowe

Mimo starannej edycji rozprawy Autor nie ustrzegł się pewnych drobnych usterek i błędów. Poniżej omawiam niektóre z nich:

- Pewne krótkie fragmenty tekstu są dokładnie, kilkukrotnie powtarzane np. rozdz. 1 Wprowadzenie str. 11-12 i rozdz. 2.2 Cel rozprawy str. 13-14, rozdz. 8 Podsumowanie str. 131.
- Kiepska jakość większości rysunków wynikająca zapewne z faktu, że w oryginale są one prawdopodobnie kolorowe a w pracy, która otrzymałam, czarno-białe oraz ze znacznego ich zmniejszenia np. Rysunek 6.21 –str. 122, Rysunek 6.14 - str. 114, Rysunek 6.6 –str. 106, Rysunek 6.4 –str. 104 - mało czytelne napisy, Rysunki 5.16- 5.21 –str. 91-97 – całkowicie nieczytelne , itd.

- Podając strony internetowe należy podać kiedy ta strona była aktualna np. 211 pozycja literatury.

Wyżej wymienione drobne usterki nie wpływają na moją pozytywna ocenę rozprawy.

Ocena rozprawy

Problematyka ewolucji systemów usługowych i jej dokumentowanie jest ważna i aktualna dla dziedziny inżynierii oprogramowania. Autor uzyskał postawione przez siebie cele badawcze. Wkład Autora w problematykę jest dostateczny.

Rozprawa jest poprawnie napisana, jej układ jest logiczny jednak mnie nie bardzo odpowiada jej struktura: elementy opisowe i własne pomysły są przemieszane. Gdyby nie ostatni akapit we wprowadzeniu zaczynający się od: „Autorski wkład twórcy...." trudno by było zorientować się na czym polega wkład Autora. Niezbyt jasne są też relacje treści rozprawy i innych publikacji Autora. W literaturze jest trzynaście publikacji ([191-203]) których jest współautorem z lat 2010-2013, a tytuły tych publikacji są bardzo zbliżone do tematyki rozprawy. To budzi mój niepokój jakie są relacje miedzy tymi pracami a rozdziałami rozprawy, nie jest to jasno napisane w pracy. Np. rozdział 4.4 Autor wskazuje (str. 12) jako „wkład własny" a w rozdz. 4.4 pisze ze MAD opublikowano w [200]. Zamiast kolejno opisywanych rozwiązań z literatury oczekiwałabym systematycznego przeglądu literaturowego.

Wkładem Autora w rozwój inżynierii oprogramowania jest zaproponowana metodyka ewolucji i rozwoju systemów usługowych (metodyka MAD4SOA), obejmująca więcej etapów procesu życia oprogramowania niż metodyki wcześniej proponowane. Jako jedyna metodyka ta obejmuje etap ewolucji oprogramowania usługowego. Propozycje Autora wydają się być poprawne i użyteczne. O użyteczności świadczy zamieszczone w rozdziale 6 studium przypadku. Nie sadzę jednak by propozycja Autora mogła by być wykorzystana w praktyce przez kogokolwiek, gdyż do MAD4SOA Autor nie zrealizował żadnego narzędzia. Metodyka oparta o symbole graficzne powinna być wspierana przez stosowne narzędzie. Trudno wyobrazić sobie „ręczne" rysowanie diagramów. Narzędzie powinno nie tylko ułatwiać rysowanie diagramów ale przede wszystkim wykrywać sprzeczności, niespójności projektu.

Osiągnięciem Autora jest zapisanie reguł integralności w języku Alloy i przebadanie ich w analizatorze języka.

W pracy bardzo brakuje mi krytycznego spojrzenia na zaproponowaną metodykę a szczególnie wskazania jej ograniczeń i wad.

Podsumowując należy stwierdzić, że przedmiotem rozprawy jest poprawnie sformułowany i potencjalnie przydatny problem badawczy. Przedstawiony problem Autor rozwiązał stosując właściwe metody. Tematyka rozprawy jest jeszcze aktualna i praktycznie przydatna.

Wniosek końcowy

W opiniowanej rozprawie mgr Szymon Kijas pokazał umiejętność formułowania celów badawczych oraz warsztat badawczy prowadzący do rozwiązań stawianych problemów.

Uważam że spełnia wymagania stawiane rozprawom doktorskim przez obowiązującą ustawę o stopniach i tytułach naukowych, wnoszę zatem o dopuszczenie jej do publicznej obrony.

Irena Bluemke

Dr hab. inż. Piotr Habela
Wydział Informatyki
Polsko-Japońska Akademia Technik Komputerowych

Warszawa, 2019-02-18

KWESTIONARIUSZ- RECENZJA ROZPRAWY DOKTORSKIEJ DLA RADY WYDZIAŁU ELEKTRONIKI I TECHNIK INFORMACYJNYCH POLITECHNIKI WARSZAWSKIEJ

Tytuł rozprawy: **Decyzje architektoniczne w ewolucji systemów o architekturze usługowej**

Autor rozprawy: mgr Szymon Kijas

1. Jakie zagadnienie naukowe jest rozpatrzone w pracy /teza rozprawy/ i czy zostało ono dostatecznie jasno sformułowane przez autora? Jaki charakter ma rozprawa (teoretyczny, doświadczalny, inny)?

Przedłożona praca stanowi wkład w obszar praktycznej inżynierii oprogramowania w jej warstwie metodyczno-narzędziowej. Jako taka łączy charakter teoretyczny (analiza i krytyczna ocena istniejących środków modelowania i metodyk oraz opracowanie własnych propozycji udoskonaleń i rozwinięć) z dokonaną w niewielkim zakresie praktyczną weryfikacją na przykładzie rzeczywistego projektu (przedstawionego w rozprawie jako studium przypadku). Opracowane rozwiązanie obejmuje język modelowania zagadnień związanych z ewolucją rozwiązań architektonicznych w systemach usługowych oraz kreśli proces przeprowadzania takiej ewolucji w toku którego ów model jest wykorzystywany.

2. Czy w rozprawie przeprowadzono w sposób właściwy analizę źródeł / w tym literatury światowej, stanu wiedzy i zastosowań w przemyśle /świadczący o dostatecznej wiedzy autora. Czy wnioski z przeglądu źródeł sformułowano w sposób jasny i przekonywujący?

Autor poparł swoje rozważania obszerną analizą stanu sztuki, zarówno w sferze artykułów naukowych jak i standardów przemysłowych, i przekonująco uzasadnił postulowane rozwiązania zestawiając je z istniejącymi odpowiednikami.

3. Czy autor rozwiązał postawione zagadnienia, czy użyl właściwej do tego metody i czy przyjęte założenia są uzasadnione?

Opracowane rozwiązanie wydaje się trafną odpowiedzią na postawiony problem. Nie jest wprawdzie zwieńczone kompletnym rozwiązaniem narzędziowym w postaci oprogramowania implementującego modele i proces zaproponowanej metodyki, jednak zważywszy na ramy tego typu rozprawy jest satysfakcjonujące.

4. Na czym polega oryginalność rozprawy, co stanowi samodzielny i oryginalny dorobek autora, jaka jest pozycja rozprawy w stosunku do stanu wiedzy czy poziomu techniki reprezentowanych przez literaturę światową?

Na uznanie zasługuję dokonana przez Autora trafna diagnoza problemów zarządzania wymaganiami i projektowania architektury systemów usługowych, podbudowana, jak się wydaje, istotnym doświadczeniem praktycznym. W połączeniu ze zbadaniem stanu sztuki i identyfikacją relewantnych rozwiązań zbliżonych problemów pozwoliło mu to na opracowanie w warstwie koncepcyjnej i formalnej środków modelowania decyzji architektonicznych oraz zaproponowanie procesu opracowywania ewolucji architektury składających się na metodykę dedykowaną dla rozwoju systemów o architekturze usługowej. Autor starał się zapewnić środki wyrazu bogatsze od tych dostępnych w istniejących odpowiednikach, a z drugiej strony uniknąć nadmiernej eskalacji zakresu w obszarach, gdzie istniejące uniwersalne środki (jak np. język modelowania BPMN) są satysfakcjonujące.

5. Czy autor wykazał umiejętność poprawnego i przekonywującego przedstawienia uzyskanych przez siebie wyników /zwięzłość, jasność, poprawność redakcyjna rozprawy/?

Struktura pracy jest klarowna, a argumentacja generalnie kompletna i spójna. Drobne niedociągnięcia redakcyjne nie utrudniają (poza ograniczeniami czytelności kilku rysunków) odbioru.

6. Jakie są słabe strony rozprawy i jej główne wady?

Nie wyczerpano natomiast niektórych zagadnień związanych z modelowaniem wymagań oraz z rozważeniem szczegółów implementacyjnych ewentualnego narzędzia które wspierałoby opracowaną metodykę. Decyzja o sporządzeniu pracy w języku polskim (pomimo w większości anglojęzycznych obiektów odniesienia) ogranicza zasięg oddziaływania opracowanych rozwiązań.

7. Jaka jest przydatność rozprawy dla nauk technicznych?

Opracowane środki modelowania i proces mogące potencjalnie usprawnić projektowanie i ewolucję oprogramowania opartego na architekturze usługowej staną się faktycznie efektywne w przypadku opracowania narzędzia informatycznego implementującego tak sformułowaną metodykę. Stąd też pracę należy scharakteryzować jako wkład w inżynierię oprogramowania w dziedzinie narzędzi dla zarządzania wymaganiami, zarządzania zmianą oraz specyfikacji architektury systemów usługowych.

Szczegółowe uwagi do pracy zamieszczono w dołączonej do niniejszego formularza sekcji pt. Ocena szczegółowa.

8. Do której z następujących kategorii Recenzent zalicza rozprawę:

a/ nie spełniająca wymagań stawianych rozprawom doktorskim przez obowiązujące przepisy
b/ wymagająca wprowadzenia poprawek i ponownego recenzowania
<u>c/ spełniająca wymagania</u>
d/ spełniająca wymagania z wyraźnym nadmiarem
e/ wybitnie dobra, zasługująca na wyróżnienie

Powyższe pytania mają charakter pomocniczy. Wskazane jest takie formułowanie treści recenzji, by można ją było odczytywać bez przeczytania pytań.

Ocena szczegółowa

Tematyka pracy i jej istotność

Rozprawa stanowi wkład w obszar rozwiązań metodycznych inżynierii oprogramowania w zakresie projektowania oprogramowania w architekturze usługowej (SOA) i integracji usług. Podjęty problem został umotywowany doświadczeniami Autora z udziału w konstrukcji rzeczywistego systemu opartego na takiej architekturze oraz osadzony w kontekście istniejącego stanu sztuki w tym standardów przemysłowych (m.in. standardy ISO dotyczące architektury oprogramowania, systemów usługowych, inżynierii wymagań, czy też języka BPMN).

Dotknięto tym samym zagadnień rozciągających się w procesie tworzenia oprogramowania na obszarze pomiędzy różnorodnymi (funkcjonalnymi i, zwłaszcza, pozafunkcjonalnymi) wymaganiami wynikającymi ze sformułowanej potrzeby biznesowej a rozwiązaniami szczegółowymi związanymi z implementacją. O ile zaś działania implementacyjne są przez współczesne środowiska programistyczne i narzędzia integracji oprogramowania efektywnie wspierane a ich integralność weryfikowalna, o tyle te poprzedzające kroki, wiodące od wymagań, poprzez decyzje architektoniczne do implementacji, mogą być narażone na powstanie niespójności lub rozwiązań nieoptymalnych, jeśli w procesie wytwórczym zabraknie odpowiednich prawideł funkcjonowania albo gdy okażą się one zbyt złożone w użyciu, nieefektywne, bądź jeśli z innych względów nie uda się wyegzekwować ich stosowania. Pierwszoplanowym zadaniem dla języków modelowania i metodyk będzie w tym przypadku opanowanie złożoności pochodzącej z dziedziny problemowej oraz tej związanej z techniczną warstwą rozwiązania. Warstwa abstrakcji zapewniana w architekturze usługowej przez interfejsy usług nie stanowi w tym przypadku znaczącego ułatwienia, choć naturalnie narzuca pewne prawidła projektowania i daje możliwość wypracowania specjalizowanych rozwiązań dla projektowania architektury. Kolejną komplikację stanowi konieczność uwzględnienia dynamicznej niekiedy ewolucji wymagań i związanych z nimi zmian rozwiązań. Trafnie odzwierciedla to nota występująca w wielu szablonach dokumentów metodyk wytwarzania oprogramowania czy zarządzania projektami głosząca "*This document is only valid on the day it was printed*".

Problemami tymi zajmuje się szereg ogólnych metodyk rozwoju oprogramowania (zarówno tych zaliczanych do kategorii tradycyjnych jak i tzw. zwinnych (*agile*)), wzbogacanych o specyficzne dla obszaru problemowego rozwiązania modelowania, narzędzia oraz procedury postępowania. Przykładem tego rodzaju wkładu jest też niniejsza rozprawa, która rozwija metodyczne aspekty dokonywania i dokumentowania ewolucji oprogramowania w odniesieniu do wyspecjalizowanej dziedziny systemów usługowych.

Ponieważ, w przeciwieństwie do samego kodu oprogramowania, aktualność i spójność dokumentów dotyczących tych kroków nie jest wymuszana przez narzędzia, wszechobecna w projektach informatycznych presja czasu może łatwo doprowadzić do zaniedbania tych kroków, co w dalszej perspektywie wieść może do nawarstwiania się nieoptymalnych rozwiązań oraz skutecznie utrudnić projektowanie kolejnych zmian systemu wymuszonych przez zmieniające się zewnętrzne uwarunkowania. Można więc argumentować, że pożądane jest zapewnienie w procesie budowy i utrzymania systemów następujących elementów:

- dyscyplina aktualizacji dokumentów architektonicznych,

- zapewnienie zdolności prześledzenia (*traceability*) wymagań w sensie ich odwzorowania na takie elementy projektu oprogramowania jak procesy biznesowe, problemy architektoniczne, rozwiązania architektoniczne, implementacje oraz testy,
- zabezpieczenie powstającej w toku projektowania systemu wiedzy – w szczególności w postaci uzasadnienia przyjętych rozwiązań oraz wskazania ich wad lub zalet,
- zapewnienie takich rozwiązań organizacyjnych i wsparcia narzędziowego, aby powyższe właściwości mogły być osiągnięte bez nadmiernego narzutu pracochłonności.

Powyższe cele wydają się jasne, jednakże skuteczność ich realizacji, to jest potencjał opracowanych rozwiązań metodycznych, bywa trudny do obiektywnej weryfikacji bez sprawdzenia na szerszą skalę w rzeczywistych projektach. To ostatnie zaś trudno osiągnąć, jeśli rozwiązanie złożone z języka modelowania oraz procesów składających się na projekt metodyki nie zostaną wcześniej zmaterializowane w postaci dojrzałego narzędzia. W tym więc należy upatrywać trudności w ocenie prac o charakterze i zakresie takim jak niniejsza rozprawa.

Sytuację poprawia natomiast osadzenie tego typu rozwiązania w kontekście innych, powszechniej znanych i zweryfikowanych praktycznie rozwiązań, poprzez opracowanie ich udoskonalenia czy specjalizacji oraz skonfrontowanie ich z rzeczywistym przykładem zastosowania. Obie te cechy zostały w ocenianej pracy zapewnione, dowodząc pragmatycznego podejścia Autora do postawionego problemu. Można zatem stwierdzić, że problem będący przedmiotem rozprawy został postawiony trafnie oraz że przyjęto adekwatny kierunek dla jego rozwiązania.

Treść pracy

Praca składa się ośmiu (wliczając Wprowadzenie i Podsumowanie) rozdziałów oraz bibliografii liczącej 211 pozycji. Ze względu na tematykę pracy oraz pochodzenie większości źródeł nieco rozczarowuje decyzja Autora o przedstawieniu rozprawy w języku polskim. Wydaje się, że sporządzenie jej, jak szeregu artykułów w dorobku Autora, w języku angielskim stworzyłoby szanse na skuteczniejszą weryfikację proponowanych rozwiązań czy też ich zaadaptowanie w tworzonych poza naszym krajem rozwiązaniach.

Rozdział 1 („Wprowadzenie") i rozdział 2 („Cel i zakres rozprawy") mają podobny charakter, stąd mogą być scharakteryzowane łącznie. Uzasadniają wybór tematu i precyzują kontekst prac. Określono w nich m.in. tezę rozprawy oraz wskazano, z dokładnością do podrozdziałów, fragmenty pracy stanowiące oryginalny wkład Autora w dziedzinę architektury oprogramowania.

Rozdział 3 pełni rolę wprowadzenia w problematykę rozprawy. Przedstawiono pojęcie systemów usługowych oraz notacje i modele służące opisowi ich architektury. Zaprezentowano również własny przykład systemu usługowego, który posłużył w dalszych rozdziałach pracy do zilustrowania omawianych tam zagadnień.

Rozdział 4 wprowadza w szczegółowy przedmiot rozprawy, to jest modelowanie architektury w postaci decyzji architektonicznych. Wprowadzono kluczowe pojęcia tej dziedziny oraz omówiono istniejące modele i notacje służące reprezentowaniu decyzji architektonicznych. W szczególności zaprezentowano notację i narzędzie MAD (*Maps for Architectural Decisions*) opracowane przy udziale Autora, które stały się podstawą rozwiązania przedstawionego w dalszej części pracy, to jest metodyki MAD4SOA.

Rozdział 5 stanowi centralną część pracy i zawiera, po wprowadzeniu przedstawiającym istniejące metodyki rozwoju systemów usługowych, najbardziej eksponowane elementy oryginalne pracy. Składają się nań:

- propozycja procesu wprowadzania zmiany do systemu usługowego, stanowiąca rozwinięcie procesu z ISO/IEC 20000;
- model problemów i decyzji architektonicznych oraz model procesu ewolucji, łączący decyzje architektoniczne i ich kontekstem biznesowym;
- formalizacja modelu.

Pojęcia składające się na metodykę przedstawiono w postaci opisu słownego oraz diagramów klas w języku UML, zaś składające się nań procesy – w postaci opisu słownego oraz diagramów BPMN.

Formalizacji modelu obejmującego pojęcia metodyki oraz relacje pomiędzy nimi dokonano z wykorzystaniem języka Alloy oraz narzędzia Alloy Analyzer. W tekście pracy przytoczono komplet formuł. Jedną z końcowych sekcji rozdziału zatytułowano „Weryfikacja modelu", choć pojęcie to nie wydaje się w tym wypadku w pełni uprawnione. Przedstawiono tam mianowicie powoływanie przykładowych instancji formalnego modelu, wskazując, że przykłady te pozostają spójne ze sformułowanymi w specyfikacji ograniczeniami i z praktyczną intuicją dotyczącą spójnych stanów modelu. Jest to więc raczej ilustracja poszczególnych cech modelu wykorzystująca narzędzie Alloy Analyzer, niemniej jednak użyteczna dla objaśnienia szczegółów metodyki. Kilka dalszych uwag dotyczących tej części pracy znajduje się pośród uwag szczegółowych poniżej.

Rozdział 6 zawiera studium przypadku opracowane przez Autora na podstawie jego doświadczeń z udziału w projekcie budowy i rozbudowy systemu dla sektora finansowego. Studium prezentuje cztery kolejne wydania systemu charakteryzujące się kolejnymi przyrostami zakresu wymaganej funkcjonalności oraz implikowanymi przez nie nowymi problemami architektonicznymi. Opisowi kolejnych wydań wraz z formułowanymi dlań wariantami rozwiązań architektonicznych, ich ewaluacją i wyborami towarzyszą diagramy komponentów UML, diagramy procesów BPMN oraz diagramy problemów architektonicznych oraz decyzji architektonicznych w notacji MAD (diagramy ADRD oraz ADPM). Taka konfrontacja notacji z rzeczywistym przykładem potwierdza przydatność wprowadzonych przez nią konstrukcji, choć naturalnie nie jest w stanie potwierdzić produktywności zakładanego rozwiązania, gdyż to wymagałoby zastosowania kompletnego narzędzia implementującego zakładaną metodykę. Realizacja takiego narzędzia zapewne odsłoniłaby kolejne szczegóły wymagające rozstrzygnięcia, a być może wręcz pewne pomysły na modyfikację przyjętych przez notację rozwiązań. Niemniej jednak studium przypadku stanowi bardzo wartościowy element rozprawy.

Rozdział 7 ponownie nawiązuje do stanu sztuki w zakresie modeli decyzji architektonicznych przytoczonego w rozdziale 4, konfrontując opracowaną metodykę, to jest MAD4SOA, z modelem Archium, modelem Krutchena oraz z modelem RADM. Wszystkie te porównania wskazują na bogatszy zakres informacyjny modeli stosowanych w MAD4SOA.

Rozdział 8 stanowi krótkie podsumowanie. Istotną jego częścią jest wskazanie czterech kierunków rozwinięcia nakreślonego rozwiązania. Dwa pierwsze z nich, to jest rozwinięcie środków dla opisu i śledzenia zmian wymagań oraz opracowanie narzędzia realizującego zaproponowaną metodykę wydają się szczególnie mocno uzasadnione gdyż stanowią właśnie

te zagadnienia, których w nakreślonym tu rozwiązaniu wydaje się brakować dla zapewnienia mu pragmatycznej kompletności.

Uwagi szczegółowe

Struktura pracy jest przejrzysta i kompletna. Część wniosków bądź wątpliwości nasuwających się czytelnikowi w toku lektury jest adresowana w dalszych sekcjach tekstu, kilka natomiast szczegółów można uznać za przedyskutowane w sposób nie w pełni wyczerpujący.

Autor wykazuje dobre zrozumienie uwarunkowań biznesowych kształtujących wymagania nakładane na projektowany system i uwzględnia ten kontekst w swoim modelu. Mimo to środki wyrazu dostępne dla samego dokumentowania wymagań, ich źródeł czy też ich zmian budzą na tle reszty modelu pewien niedosyt. Wskazana wydaję się np. możliwość włączenia np. daty sformułowania czy określenia stanu wariantu / problemu, jak też precyzyjne odwołanie do źródła wymagań w rozumieniu decyzji biznesowej.

Nieco dezorientująca jest przyjęta konwencja w której wszystkie klasy składające się na zaproponowany metamodel oraz ich atrybuty noszą nazwy w języku polskim, zaś asocjacje – nazwy w języku angielskim. Jak się wydaje nie wszędzie jest to wymuszone zapożyczeniami pojęć (np. nazw rodzajów relacji) z języka UML. W niektórych przypadkach brak liczności przy końcach asocjacji (np. rys. 4.9 czy niektóre elementy rys. 5.8) zubaża zawartość informacyjną metamodelu. Być może lepiej sprawdziłaby się tutaj konwencja stosowana np. w specyfikacji metamodelu UML, gdzie zarówno liczności jak i nazwy ról są explicite umieszczane przy końcach wszystkich praktycznie asocjacji.

W diagramach na rys. 4.9 i potem na rys. 5.8 pojęcie Wymaganie występuje w trzech różnych rolach na dwa różne sposoby: jako połączone asocjacją z pojęciem Problem oraz jako atrybuty w pojęciach Zaleta i Wada. Nie jest jasne, czy takie zróżnicowanie sposobu zamodelowania (atrybut vs. rola w asocjacji) ma zaakcentować odmienność traktowania Wymagania w tych dwóch kontekstach, czy jest jedynie przełożeniem sposobu wizualizacji tych powiązań w składni konkretnej. Wymaganie występuje na diagramach w dwóch kontekstach: jako samodzielne węzły grafu powiązane z RFC oraz, niejako w innej warstwie, jako element zagnieżdżone w symbolach Zalet i Wad. Ten drugi kontekst nie jest reprezentowany w formalizacji. Ponieważ atrybuty na diagramach metamodelu te nie specyfikują typów, nie mamy w pełni jednoznacznego potwierdzenia, że w obu przypadkach (wymaganie wynikłe z RFC vs. wymaganie powiązane z zaletą lub wadą) mamy do czynienia z tym samym bytem. Zakładając, że tak – przy pełnym rozwinięciu diagramu ADPM będziemy mieć w przypadku wymagania powiązanego z zaletą lub z wadą do czynienia z dwiema jego manifestacjami na diagramie. Na komentarz zasługiwałaby tutaj czytelność oraz intuicyjność takiej składni, jak też ogólnie użyteczność i częstość występowania konstrukcji łączącej explicite wymaganie z zaletą / wadą (Na kilka zamieszczonych w pracy obszernych diagramów prezentujących rzeczywiste przykłady modeli MAD4SOA znajdziemy tylko jeden przykład wymagania powiązanego z zaletą lub wadą (wymaganie "Zgodne z SOA" na rys. 6.15)).

Sekcja 5.3 skłania m.in. do rozważenia, czy nie byłoby zasadne wbudować w proces jakiegoś mechanizmu agregowania RFC tak, by oszczędzić nakłady i zakłócenia związane z ewolucją oraz zoptymalizować decyzje w odniesieniu do sumy postulowanych zmian a nie pojedynczych zmian (zwłaszcza, że str. 63 wspomina rubrykę „równolegle realizowane zmiany").

Zamieszczona w podrozdziale 5.5 formalizacja wydaje się w kilku miejscach obłożona nadmiernymi ograniczeniami. Np.:

- dekompozycja Problemu na podproblemy (relacja *decomposesInto*) nie może mieć przy tak sformułowanych ograniczeniach efektywnie więcej niż jednego poziomu;
- ograniczenia nałożone na relację *forces* czynią przykład z rysunku 5.14 (dwukierunkowa zależność) niepoprawnym;
- jeśli w modelu choć jeden Problem ma zdefiniowane warianty, model będzie mógł osiągnąć poprawność tylko w przypadku nadania wszystkim Problemom stanu ProblemRozwiazany.

Część rysunków okazała się w drukowanej formie pracy nieczytelna. Dotyczy to zwłaszcza rysunków 5.20 i 5.21.

Wreszcie, sekwencja modeli zaprezentowanych dla poszczególnych wydań systemu w rozdziale 6 skłania do sugestii, że skoro jednym z celów jest dokumentowanie ewolucji, to warto nakreślić, w jaki sposób możnaby wizualizować wersjonowanie problemów i rozwiązań w docelowym narzędziu.

Ocena pracy i wniosek

Autor w zadowalający sposób potwierdził tezę pracy, że „możliwe i celowe jest stworzenie takiej metodyki ewolucji i rozwoju systemów usługowych, która zachowuje zasób wiedzy architektonicznej obejmujący kontekst zmian wprowadzonych we wcześniejszych krokach ewolucji, dylematy rozważane przez architektów oraz racje stojące za dokonanymi przez nich wyborami". Podjął on ważny problem z obszaru inżynierii oprogramowania i na podstawie adekwatnie rozpoznanego stanu sztuki poczynił wkład w udoskonalenie jego rozwiązań.

Formułując zręby metodyki projektowania i ewolucji systemów usługowych Autor określił środki specyfikowania problemów i wariantów rozwiązań architektonicznych, środki dokumentowania oceny wariantów rozwiązań, ich współzależności oraz motywów podjętych decyzji, a także powiązania ich w podstawowym zakresie z kontekstem wymagań płynących z warstwy biznesowej projektu. Nakreślił również proces ewolucji określający sposób wykorzystania ww. modeli oraz metody ich formalizacji, potencjalnie pomocne w doprecyzowaniu części ograniczeń na potrzeby ewentualnej implementacji opartego na nich narzędzia.

Wskazane w ocenie niedoskonałości nie wpływają na ogólnie pozytywną ocenę dokonanego wkładu, wobec czego stwierdzam, że oceniana praca spełnia wymagania stawiane rozprawom doktorskim przez obowiązujące przepisy i wnoszę o dopuszczenie p. mgr. Szymona Kijasa do dalszych etapów przewodu doktorskiego.

Piotr Habela