

## Oponentský posudok

habilitačnej práce Ing. Ivana Poláška, PhD.

„*Contribution to Development and Refactoring of Information Systems*“

Predkladaný oponentský posudok byl vypracovaný na základe menovania dekanou FIIT STU v Bratislave prof. Ing. Máriou Bielikovou, PhD. listom z 12. 10. 2016 (č. 6279/183/2016).

Predložená habilitačná práca je napísaná na 85 číslovaných stránkach, je rozdelená na úvod, na tri 3 kapitoly podľa tématického zamerania na podporu vývoja rozsiahlych informačných systémov, na súhrn obsahujúci aj prípadné ďalšie témy či smery ďalšieho výskumu problematiky a končí zoznamom použitej literatúry. Členenie práce je prehľadné a logicky správne. Kapitoly sú vhodne delené na podkapitoly, čo umožňuje pomerne pohodlnú orientáciu v texte habilitačnej práce. Jazyková úroveň – práca je napísaná v anglickom jazyku – je podľa môjho názoru vyhovujúca. Ilustrácie sú výstižné a vhodne dopĺňujú text. Celková grafická úprava práce je výborná.

V úvode práce (s. 1-5) je ozrejmenej motivácia hľadania stále efektívnejších postupov návrhu a vytvárania informačných systémov, napr. cestou postupného a neustáleho zlepšovania softvéru pomocou refaktorovania a podpory znovupoužiteľnosti a pod., čo predstavuje aj reakciu na výzvy praxe a čoho výsledky by napomohli nepochybne aj ďalšiemu výskumu problematiky.

Druhá kapitola (*3D UML methods for analysis and design of the large systems*, s.7-24) analyzuje rozšírenie klasickej notácie jazyka UML na modelovanie systémov z dvojdimenzionálnej podoby jednotlivých diagramových techník do možností trojdimenzionálneho zobrazenia. Za najzreteľnejší praktický prínos tejto kapitoly považujem vnorenie nového *xD UML* do návrhu postupu a dynamického rámca vývoja umožňujúceho (vy)tvoriť prostredie obsahujúce niekoľko pracovných plôch, s ktorými možno pracovať paralelne s niekoľkými modelmi a typmi UML diagramov umiestnených v simulovanom trojdimenzionálnom prostredí. Nejde pritom iba o ambíciu využiť atraktívnosť 3D zobrazenia a vytvoriť len vyššie pohodlie a plasticnosť vzhľadu 3D, ale ponúknuť aj širšie chápanú multidimenzionalitu (preto sa v práci objavuje aj označenie *xD UML*), ktorá by využila jednotlivé rezy v priestore na zobrazenie samostatných komponentov a scenárov použitia, časové, autorské a tímové verzie modelov, jednotlivé druhy elementov vo vrstvách a pod., na podporu tvorby odľahčenej architektúry (*lean architecture*), rezy alternatívnych a paralelných tokov a ďalšie alternatívy, ktoré naznačujú presah z 3D priestoru do prirodzenej multidimenzionality spomínaných možností (modulov, procesov, alternatív a podobne), ako prípravy prostredia pre dosiahnutie efektívnej architektúry systému a pre optimalizácie procesu analýzy a návrhu.

V tretej kapitole (*Anti-pattern detection and refactoring of the large system*, s. 25-47) je jedným z najdôležitejších prínosov vytvorenie modelu pozitívnych a negatívnych závislostí anti-vzorov alebo tzv. „pachov“ (autor práce to vníma ako synonymum), kde negatívna závislosť znamená vznik nových pachov v kóde pri odstránení konkrétneho vyhládaného pachu a pozitívna závislosť mapuje zánik aj ďalších pachov odstránením zvoleného pachu.



Tieto modely môžu slúžiť ako dáta pre rozhodovacie algoritmy na zlepšenie výsledkov vytváraného expertného systému, kedy postupnosť odstraňovania pachov v kóde alebo v modeloch môže postupovať efektívnejšie pri poznaní, ktoré pachy na seba viažu maximum pozitívnych a minimum negatívnych závislostí. Ako ďalšie pozitívum možno hodnotiť autorovu snahu refaktorovať nielen výsledný zdrojový kód systému (v podkapitolách 3.6 *Rule-based approach*, 3.7 *Identification Using Syntax Tree*, 3.9 *Semi-automatic refactoring to aspect-oriented platform*), ale aj počítačový model (podkapitoly 3.4 *Bit-vector algorithm extension*, 3.5 *The Similarity scoring algorithm*, 3.8 *Identification using OCL query*).

Bolo by zaujímavé dozvedieť sa, ktoré metódy pre identifikáciu pachov sú prakticky používané z tých, čo sú v práci analyzované?

Štvrtá kapitola (*Information and knowledge retrieval and their graphical representation for collaborative software development*, s. 49-69) analyzuje metódy rozoznania tém (4.5 *Extracting, identifying and visualization of the content in software project using swarm intelligence*, 4.6 *Software visualization using clustering inspired by ant colony*) a autorov týchto tém v zdrojovom kóde (4.2 *Source Code Authors*) a to na znovupoužitie alebo aspoň využitie znalostí (*knowledge sources*), uložených v zdrojovom kóde, resp. v samotných autoroch tohto zdrojového kódu. Ide nielen o technologické vedomosti (napríklad pri tvorbe používateľského rozhrania, logickej vrstvy, databázových služieb) ale aj o tzv. „doménové znalosti“, ktoré môžu byť rozoznané a využité pri tvorbe nových systémov. Navyše autor disponuje potrebnými znalosťami z oblasti rozoznávania vzorov a anti-vzorov, ktoré môžu byť využité pre evaluáciu kvality a „vzretosti“ príslušného znalostného zdroja.

Záver práce vyznieva ako zaujímavý náčrt novej metódy pre softvérové inžinierstvo, kde sa využijú metódy, analyzované a vytvárané v kapitolách o multidimenzionálnom UML, refaktorovaní ale aj rozoznávaní znalostných zdrojov v kóde. Autor navrhuje metódu a dynamický rámec pre paralelný zápis prípadov použitia, odvodenia aaj priamu tvorbu modelov a kódu aj s okamžitou evaluáciou vygenerovaného systému. Túto metódu nazýva *agilná modelom riadená evolúcia softvéru (AMDSE)* a prvý základ tejto metódy prezentuje vo svojich publikáciách a aj v tejto práci.

Zaujímalo by ma, čo je ešte potrebné prioritne dokončiť z navrhovanej metódy AMDSE?

Práca samotná obsahuje všetky formálne náležitosti, je vhodne štrukturovaná a čitateľná. Čiastočne som to už konštatoval. Tu však dodávam, že si cením aj zoznam vybratých publikácií autora, viažucich sa k jednotlivým témam habilitačnej práce (*Table 2 Selected publications of the author*, s. 5), z ktorých by som vyzdvihol najmä tri príspevky: *Polášek & Uhlár: Extracting, Identifying and Visualisation of the Content, Users and Authors in Software Projects (LNCS Transactions on Computational Science)*, *Záhoranský & Polášek: Text Search of Surnames in Some Slavic and Other Morphologically Rich Languages Using Rule Based Phonetic Algorithms (IEEE/ACM Transactions on Audio, Speech & Language Processing, 2015)* a *Gregorovič & Polášek: Analysis and Design of Object-oriented Software using Multidimensional UML (iKnow, Graz, ACM 2015)*.

Z celej habilitačnej štúdie je zrejma skúsenosť autora z praxe vývoja a výskumu softvéru, jeho kompetentnosť a erudovanosť z oblasti vývoja informačných systémov a aj jeho snaha analyzovať potrebné metódy do hĺbky (dokladajú to napr. spomínané rozsiahlejšie články z *LNCS Transactions on Computational Science* a *IEEE/ACM Transactions on Audio, Speech & Language Processing*). Zreteľne je viditeľná aj jeho silná motivácia venovať sa

optimalizácii softvérových architektúr a zrýchľovania procesov ich vzniku, a to nielen na úrovni výskumu a vývoja, ale aj v oblasti autorovho úspešného pedagogického pôsobenia na STU.

Doporučujem predloženú prácu k ohajobe a po splnení všetkých ostávajúcich podmienok habilitácie jej autora Ing. Ivana Poláška, PhD. na udelenie pedagogicko-vedeckého titulu docent.

V Opave, 17. 1. 2017

Prof. RNDr. Jozef Kelemen, DrSc.