

Dezvoltarea de metode de analiza automata a sistemelor software

Conducător științific : prof.dr.ing. Vladimir Crețu

Computer Science and Engineering Department
"Politehnica" University of Timisoara

Iulie, 2018

Direcțiile și intențiile de studiu avute în vedere pentru programul doctoral

În cadrul temei de doctorat voi investiga metode de rafinare/filtrare a dependențelor logice extrase din modul de evoluție a sistemelor software, pentru a le putea folosi în aplicații de reconstrucție arhitecturală. Voi investiga în ce măsură utilizarea informațiilor aduse de dependențele logice poate aduce îmbunătățiri metodelor de reconstrucție arhitecturală, atât pentru cele de reclusterizare cât și pentru cele de identificare a modulelor arhitecturale importante.

Pregătirea anterioară

Studiile absolvite până în prezent sunt:

- **Licența** la facultatea de Automatică și Calculatoare, UPT, în domeniul CTI, specializarea Calculatoare, cu opționale pe pachetul de Software Engineering, media de absolvire 8,87.

Tema lucrării de licența a fost *“Sistem distribuit pentru gestionarea interacțiunii clienților cu sistemul de versionare.”* obținând nota 9,66 în sesiunea Iunie 2016.

- **Master** la facultatea de Automatică și Calculatoare, UPT, în domeniul CTI, specializarea Information Technology, media de absolvire 9,75.

Tema lucrării de dizertație a fost *“An analysis of the relationship between structural and logical dependencies in software systems.”* obținând nota 10 în sesiunea Iunie 2018.

Gradul de inițiere în activități de cercetare

În cadrul departamentului de Calculatoare cu ocazia elaborării lucrării de dizertație.

Lucrarea de disertație a avut ca obiectiv principal studiul relațiilor dintre diferitele tipuri de dependențe software: dependențe structurale, extrase din analiză statică a codului sursă, și dependențe logice, extrase din modul de evoluție al sistemului regăsit în sistemul de versionare.

Tool pentru extragerea dependențelor

S-au studiat 17 sisteme open-source cpp și java. Pentru aceasta, am construit un tool de analiză pentru extragerea ambelor tipuri de dependențe software.

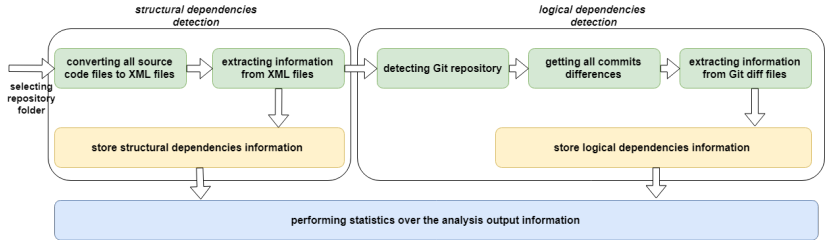


Figure 1: Tool workflow

Metode de filtrare ale dependențelor logice

Dependentele logice au fost extrase în funcție de următoarele filtre :

- ▶ Numărul de fișiere existente într-un commit. Valorile pentru acest filtru: 5, 10, 20 și fără limită de fișiere.
- ▶ Numărul de apariții a unei dependente logice. Valorile pentru acest filtru: 1, 2, 3 și 4 apariții.
- ▶ Cu/fara luarea în considerare a comentariilor ca schimbări valide.

Rezultate

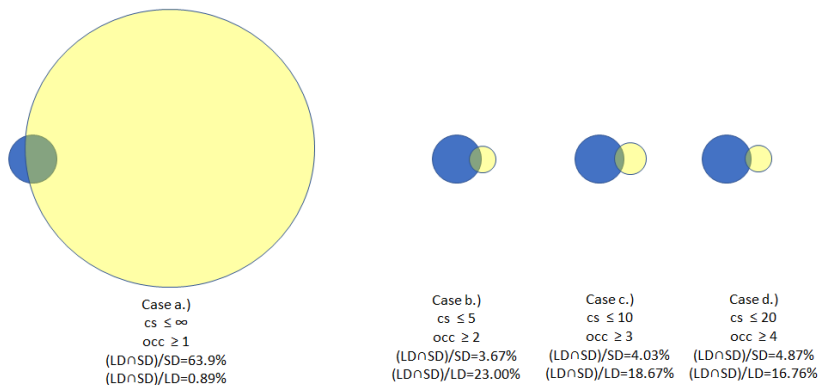


Figure 2: Venn Diagrams

Concluzii preliminare

- ▶ Cea mai mare influență asupra rezultatelor o are filtrarea după numărul de fișiere și filtrarea după numărul de apariții a dependențelor logice
- ▶ Filtrarea comentariilor are o influență mică asupra rezultatelor finale.

Direcții de continuare a cercetării

În urma măsurătorilor aproximativ 80% din dependentele logice nu sunt și structurale. În viitor dorim să studiem cauzele acestei diferențe mari între cele două categorii prin:

- ▶ Extragerea dependentelor structurale și din versiuni mai vechi ale codului, nu doar din cea mai recentă versiune.
- ▶ Filtrarea pe baza trendului crescator sau descrescator in timp al numarului de aparitii al dependentelor logice.
- ▶ Utilizarea informațiilor aduse de dependențele logice în îmbunătățirea metodelor de reconstrucție arhitecturală.

Gradul de inițiere în activități practice

La locul de munca de la Continental(2014-prezent) ca software developer. Munca mea implică dezvoltare de produse software folosind diferite limbaje de programare: Python, C++, C# în medii de dezvoltare ca: Visual Studio, PyCharm, folosind diferite sisteme de versionare : SVN, Git.

Lucrări științifice

[1] Adelina Stana, Ioana Șora, „*An analysis of the relationship between structural and logical dependencies in software systems*” trimisă la Sesiunea de Comunicări Științifice Studentești UPT, comunicată.

[2] Adelina Stana, Ioana Șora, Vladimir Crețu, „*Logical dependencies between classes: how to find them and how to use them ?*” trimisă la The 34th IEEE International Conference on Software Maintenance and Evolution (ICSME) 2018 - Doctoral Symposium, Madrid, Spania, Sept 2018, lucrare acceptată.

[3] Ioana Șora, Adelina Stana, „*Identifying logical dependencies from co-changing classes*” trimisă la The 7th International Workshop on Mining Software Repositories (SOFTWAREMINING-2018) - colocated with The 33rd IEEE/ACM International Conference on Automated Software Engineering (ASE 2018), Montpellier, Franta, Sept 2018, în așteptarea deciziei de review.