

Dezvoltarea de metode de analiză automată a sistemelor software

Student : Stana Adelina Diana
Conducător științific : prof.dr.ing. Vladimir Crețu

Departamentul Calculatoare și Tehnologia Informației
Universitatea Politehnica Timișoara

Iulie, 2018

Cuprins

- ▶ Direcțiile și intențiile de studiu avute în vedere pentru programul doctoral.
- ▶ Concordanța dintre domeniul de doctorat și pregătirea anterioară.
- ▶ Gradul de inițiere:
 - ▶ în activități practice
 - ▶ în activități de cercetare
- ▶ Lucrări științifice publicate sau comunicate cu tematică asociată domeniului de doctorat.

Direcțiile și intențiile de studiu avute în vedere pentru programul doctoral

- ▶ investigarea a noi metode de rafinare/filtrare a dependențelor logice extrase din modul de evoluție a sistemelor software.
- ▶ investigarea în ce măsură informațiile aduse de dependențele logice aduc îmbunătățiri ale metodelor de reconstrucție arhitecturală:
 - ▶ reclusterizare
 - ▶ 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 practice

La locul de muncă de la Continental ca software developer (4 ani).

- ▶ limbaje de programare: Python, C++, C#
- ▶ medii de dezvoltare: Visual Studio, PyCharm
- ▶ sisteme de versionare : SVN, Git

Gradul de inițiere în activități de cercetare

Lucrarea de disertație: “An analysis of the relationship between structural and logical dependencies in software systems.”



Figure 1: Dependente logice și structurale

Tool pentru extragerea dependențelor

Am construit un tool de analiză pentru extragerea ambelor tipuri de dependențe software.

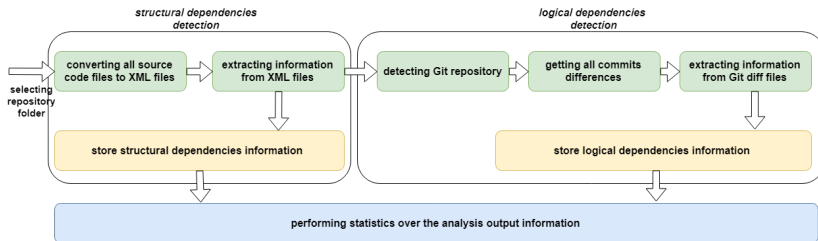


Figure 2: Tool workflow

Metode de filtrare ale dependențelor logice

Dependențele 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/fără luarea în considerare a comentariilor ca schimbări valide.

Rezultate

Am studiat 17 sisteme open-source cpp si java.

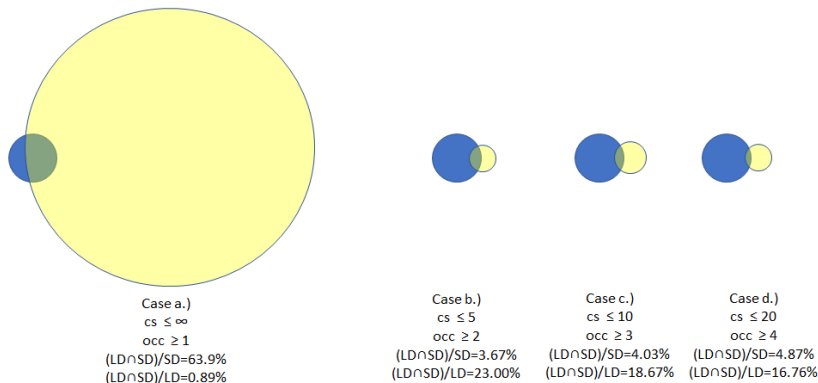


Figure 3: Relațiile între dependențele logice și structurale pentru diferite variante de filtre.

Concluzii

- ▶ Cea mai mare influență asupra rezultatelor o are filtrarea după numărul de fișiere și filtrarea după numărul de apariții a dependențelor logice.
- ▶ Filtrele aplicate individual nu duc la filtrarea dependentelor cu adevărat logice, cea mai bună soluție pentru filtrare fiind combinarea filtrelor propuse.
- ▶ Filtrarea comentariilor are o influență mică asupra rezultatelor finale ($\pm 3\%$).

Direcții de cercetare abordate la doctorat

- ▶ În urma măsurătorilor aproximativ 80% din dependențele logice nu sunt și structurale. În viitor dorim să studiem cauzele acestei diferențe mari între cele două categorii prin:
 - ▶ Extragerea dependențelor structurale și din versiuni mai vechi ale codului, nu doar din cea mai recentă versiune.
 - ▶ Filtrarea pe baza trendului crescător sau descrescător în timp al numărului de apariții al dependențelor logice.
- ▶ Utilizarea informațiilor aduse de dependențele logice în îmbunătățirea metodelor de reconstrucție arhitecturală prin combinarea lor cu dependențele structurale.

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] 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.

Lucrări științifice

[3] 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.