

## Hodnocení vedoucího diplomové práce

Autor práce: **Rudolf Augusta**

Název práce: **Porovnání nástrojů ověřujících zpětnou kompatibilitu Java knihoven**

Autor práce se zabýval zpětnou kompatibilitou modulárních Java aplikací. Jako součást práce vytvořil velké množství testovacích dat, které simulují inkrementální vývoj softwarové komponenty z pohledu různých syntaktických změn v API. Následně autor dohledal nástroje, které se zaměřují na detekci chyb v takto změněných API a otestoval jak dobře je dovedou detekovat.

Praktická část práce v podobě tvorby testovacích dat a ověření nástrojů splnila bez výhrad očekávání. Autor pracoval aktivně, účastnil se pravidelných schůzek a ochotně reagoval na vnesené dotazy a požadavky. Několik nedostatků vidím pouze v textu práce.

V kapitole 2 je poněkud nejasné tvrzení, že pro zaručení zpětné kompatibility je třeba dodržet zdrojovu, binární a behaviorální kompatibilitu. Tvrzení, že splnění těchto tří oblastí zajistí zpětnou kompatibilitu není v textu ničím podloženo. Dále pak tvrzení zní jako by se jednalo o tři disjunktní charakteristiky. Triviálně lze však chápat např. binární nekompatibilitu také jako změnu chování - dojde k pádu programu, ke které předtím nedocházelo. Dále potom autor tvrdí, že neexistují nástroje na ověření chování systému. Souhlasil bych, že neexistuje jednotný komplexní nástroj, ale s tvrzením že neexistují vůbec žádné nástroje se souhlasit nedá. Co třeba JUnit, Selenium, ... ? V kapitole 3 se opakují některé informace o zpětné kompatibilitě. Není zcela pravda, že zvýšení viditelnosti je vždy kompatibilní změna jak je tvrzeno. Problém totiž může nastat v případě dědičnosti, kdy zvýšení viditelnosti v předku bude kolidovat s viditelností potomka. Potom je poněkud zavádějící mluvit o inliningu jako o kompatibilní změně. Pokud je totiž konstanta včleněna do kódu klienta, vazba se tím ztratila a není co hodnotit z pohledu kompatibility. Celkově je tedy text napsán tak, že čtenář většinou rozumí, co autor chtěl říct, avšak mělo být lépe argumentováno.

Další má výhrada směřuje k poněkud chaoticky popsané metodice. V kapitole 4 autor uvádí, že testování prováděl manuálně měněním parametru "package", ovšem jaksi už není zmíněno, že je řeč o ANT scriptu na CD, který má tento parametr. Informace je doplněna až v kapitole 5.2. Myslím, že někdo jiný by těžko dovedl reprodukovat autorovi výsledky. Navíc tahle část by se dala jednoduše automatizovat a potom by se v popisu dalo napsat, který skript je třeba spustit.

Kapitola 4 přehledně popisuje testovací data, několikrát se však opakuje chybné tvrzení, že linkování provádí ClassLoader. Správně má být samozřejmě Linker.

**SOUHLASÍ  
S ORIGINÁLEM**



**Západočeská univerzita v Plzni**  
Fakulta aplikovaných věd  
katedra informatiky a výpočetní techniky

Kapitola 5.2.1 dle názvu hovoří o metrikách, ale ve skutečnosti žádné metriky ve smyslu číselných hodnot neuvádí. Respektive pouze říká, že každá vlastnost nástroje bude hodnocena jedním bodem. Součet bodů je potom přičten k celkovému počtu správně nalezených nekompatibilit. Toto se mi nejví jako stanovení vzájemně porovnatelných metrik. Pravděpodobně je pro hodnocení nástrojů klíčová schopnost nalezení nekompatibilit. Nastavená metodika však může určit jako lepší nástroj ten který sice chyboval v testech, ale má například větší množství pluginů. Dle mého názoru by stačilo místo těchto, poněkud umělých, metrik pouze popsat na co jsou konkrétní nástroje vhodné více a na co méně z pohledu nasazení do vývojového cyklu a případů užití.

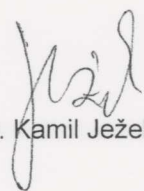
I přes nedostatky lze práci považovat za splněnou a vytvořená data s výsledky budou v budoucnu velice užitečná a to buď pro budoucí rozšiřování nástroje JaCC, nebo pro benchmark nástrojů třetích stran. Dále věřím, že po úpravě popisu metodiky je možno na základě práce vytvořit publikaci na některou z praktičtější orientovaných konferencí.


Navrhuji hodnocení známkou **výborně** a práci doporučuji k obhajobě.

Otázky:

Z popisu testovacích dat je patrné, že jste nerealizoval testy z pohledu dědičnosti a i vypadá jako byste na dopady nad kompatibilitou vůbec neuvažoval. Je dědičnost uvažována? A pokud ne, tak proč? Nemá význam pro zpětnou kompatibilitu?

V Plzni 27.5.2016

  
Ing. Kamil Ježek, Ph.D.

  
SOUHLASÍ  
S ORIGINÁLEM

Západočeská univerzita v Plzni  
Fakulta aplikovaných věd  
katedra informatiky a výpočetní techniky