

Białystok, 5 grudnia 2009

Wiktor DANKO, dr hab.,
Wydział Informatyki
Politechnika Białostocka

Recenzja rozprawy doktorskiej
pana mgr Marka WARPECHOWSKIEGO
nt.: *O wyznaczaniu klas rozszerzanych
i innych problemach semantyki Javy*

Pozwolę sobie rozpocząć recenzję od przytoczenia kilku pierwszych zdań autoreferatu pana Marka Warpechowskiego:

„Powstanie Javy i jej rozwój zwróciły uwagę na konieczność opisanie semantyki tego języka. Powstało wiele różnych opracowań z których wymienimy tu [5], [8] i książkę [6]. Obecnie Java ma kilkanaście kompilatorów. Zbadaliśmy ich zachowanie na różnych przykładach i stwierdziliśmy, że nie są one zgodne: kompilacja pewnych programów daje różne wyniki dla różnych kompilatorów. ...”.

W moim odczuciu, stanowią one potwierdzenie faktu, że rozprawa pana Marka Warpechowskiego dotyczy problematyki niezwykle ważnej, z punktu widzenia praktyki programowania w jednym z najczęściej używanych obecnie języków programowania.

Jedną z zasadniczych potrzeb, które odczuwa osoba posługująca się ustalonym produktem informatycznym, jest możliwość dotarcia do pewnych i precyzyjnych informacji na jego temat. W przypadku języka programowania jest to tym bardziej istotne, ponieważ jest on narzędziem do wytwarzania innych, zazwyczaj powszechnie użytkowanych produktów informatycznych.

W mojej opinii, rozprawa pana Marka Warpechowskiego stanowi ważny element w dyskusji nad wypracowaniem jednolitej semantyki nie tylko dla języka *Java*, ale również dla uniformizacji semantyki powszechnie używanych konstrukcji języków programowania obiektowego.

W tym momencie chciałbym zwrócić uwagę na znaczące doświadczenie Autora w analizie porównawczej semantyki różnych języków programowania obiektowego. Bibliografia zawiera szereg pozycji informujących o aktywności Autora w dyskusjach na temat semantyki programowania obiektowego już w latach osiemdziesiątych. Potwierdza to bardzo dobre przygotowanie Autora do podjęcia się przedstawienia propozycji doprecyzowania opisu semantyki języka *Java*; tak właśnie rozumiem główny cel rozprawy.

Dbłość Autora o zgodność propozycji definicji semantyki *Javy* z innymi publikowanymi, mniej formalnymi opisami semantyki, odnajdujemy w autoreferacie,

we fragmencie dotyczącym jednej z dwu głównych funkcji (*inh*, *bind*), przy pomocy których Autor definiuje semantykę:

” ... Autor pozwala sobie wyrazić przekonanie że podana powyżej (z użyciem abstrakcyjnej struktury danych) definicja *bind* odpowiada możliwie dokładnie rozwlekłemu, rozproszonemu i nieformalnemu opisowi znaczenia nazw typów w Javie. ...”.

Jednym z elementów decydujących o jednoznacznym rozumieniu semantyki Javy jest jednoznaczne rozwiązywanie kwestii wyznaczenia, dla każdej klasy *K* w danym programie klasy, którą klasa *K* rozszerza, (tzn. klasy *M*, z której dana klasa *K* bezpośrednio dziedziczy)

Propozycję doprecyzowania semantyki języka *Java* Autor rozpoczyna od zdefiniowania uproszczonego modelu grafowego zależności pomiędzy klasami programu. Proponowana przez Autora, dla ustalonego programu *P*, struktura S_P ,

$S_P = \langle \text{Classes}, \text{Id}, \text{Types}, \text{decl}, \text{name}, \text{ext}, \text{Root}, \text{Object} \rangle$

zawiera nośniki reprezentujące:

klasy zadeklarowane w programie *P* (*Classes*),

identyfikatory programu *P* (wraz z dodatkowym identyfikatorem *Object*),

zbiory nazw typów (*Types*), występujące po słowie kluczowym *extends*,

Funkcja *decl* przyporządkowuje danej klasie klasę, która ją bezpośrednio otacza (lub *Root*), gdy nie ma takiej klasy.

Funkcja *name* wyznacza identyfikator klasy zaś funkcja *ext* podaje nazwę typu występującego po słowie *extends* w deklaracji klasy *K*.

Autor poddaje analizie strukturę grafu o zbiorze wierzchołków *Classes* i gdzie krawędzie definiuje funkcja *decl*.

Funkcja *inh* przyporządkowuje danej klasie jej klasę rozszerzaną. Wyznacza ona tzw. relację dependencji (dependency relation). Żąda się by funkcja *inh* nie tworzyła cykli klas.

Definicja funkcji *bind* korzysta z definicji funkcji *inh*; formalna (łączna) definicja funkcji *inh* oraz *bind* może być prosto przedstawiona w postaci uwikłanej. Intuicyjnie, $bind(\text{type in } K)$ oznacza ”klasę stanowiącą znaczenie wystąpienia nazwy *type* w klasie *K*”.

Autor zaproponował dwa algorytmy wyznaczania funkcji *inh*. Obydwa algorytmy wyznaczania funkcji *inh*, niedeterministyczny i deterministyczny, mogą być traktowane jako stosowalne w praktyce narzędzia testowania poprawności użytych w programie deklaracji klas.

Algorytm niedeterministyczny generuje, dla danego programu, jedną z możliwych propozycji zdefiniowania funkcji *inh*.

Algorytm deterministyczny wyznacza propozycję definicji funkcji *inh*, o ile nie występuje cykl w relacji dependencji.

Algorytmy te zostały zaimplementowane i wykorzystane.

Na podkreślenie zasługuje fakt, że obydwa algorytmy, stanowiące praktyczną realizację proponowanej przez Autora idei oparcia definicji semantyki dziedziczenia i zagnieżdżenia o formalnie zdefiniowane *inh* i *bind* zostały poddane w pracy pełnej analizie z punktu widzenia ich poprawności i złożoności.

Przytoczony w pracy nietrywialny przykład dotyczący niejednoznaczności definiowania semantyki (funkcje *bind* i *Bind*) wydaje się wskazywać, że wykorzystywana wielokrotnie w definiowaniu semantyki języków programowania metoda oparta o twierdzenia typu "fix-point", może zawodzić w przypadku obiektowych języków programowania. Ten pojawiający się tu w sposób naturalny problem, jest w moim odczuciu, ważny dla informatyki teoretycznej i z pewnością wart wyjaśnienia.

Z całą pewnością pracę pana Marka Warpechowskiego należy zaliczyć do nurtu badawczego w informatyce teoretycznej związanego z wypracowaniem jednolitej, zunifikowanej semantyki języków programowania. Aczkolwiek nie można twierdzić, że finalnym rezultatem dotychczasowych prac tego nurtu jest jakakolwiek unifikacja, to za główny efekt tych prac uznałbym ugruntowanie przekonania, że przy oferowaniu języka programowania, jako narzędzia tworzenia użytkowego oprogramowania, niezbędnym jest dążenie do spełnienia aczkolwiek minimalistycznego, to jednak oczywistego żądania: precyzyjnego opisu semantyki. W tym kontekście rozprawa pana Marka Warpechowskiego

- zwraca uwagę na fakt braku dostatecznie precyzyjnego i jednoznacznego opisu semantyki języka programowania *Java*, co Autor ilustruje szeregiem przykładów w początkowych fragmentach rozprawy; przykłady te pozostają w ścisłym związku z praktyką swobodnego wykorzystywania różnorodnych możliwości języka *Java*,
- zawiera propozycje rozwiązania napotkanych problemów poprzez budowę precyzyjnej semantyki.

Przedstawiona w pracy propozycja doprecyzowania opisu semantyki języka *Java* a także innych języków programowania obiektowego jest, w moim odczuciu, najważniejszym fragmentem świadczącym o kompetencjach Autora.

Dotychczasowy rozwój narzędzi programowania pokazuje, że nie należy oczekiwać, że wraz z pojawieniem się języka programowania nowego typu, równocześnie pojawi się precyzyjny i pełen opis jego semantyki. Ważne jest jednak, by czasokres wyjaśniania wszelkich wątpliwości był możliwie najkrótszy.

W tym sensie praca pana Marka Warpechowskiego jest ważnym elementem skracającym czas osiągnięcia pełnego opisu semantyki tak powszechnie używanego języka programowania obiektowego, jakim jest *Java*.

Nie mam najmniejszej wątpliwości, że praca pana Warpechowskiego jest ważnym głosem, przestrzegającym przed niefrasobliwością w prezentowaniu semantyki konstrukcji programowania obiektowego oraz popełnianiem brzemiennej w skutki niedopowiedzeń.

Nie zauważyłem w pracy istotnych usterek redakcyjnych uniemożliwiających zrozumienie tekstu rozprawy. Podobnie jest z usterkami edytorskimi. O dostrzeżonych mankamentach poinformowałem Autora.

Podsumowując uważam, że problematyka rozprawy pana Marka Warpechowskiego jest bardzo ważna z punktu widzenia zarówno teorii opisu semantyki języków programowania obiektowego, jak i wyjaśniania mechanizmów definiowania złożonych pojęciowo elementów tak powszechnie używanego języka programowania obiektowego, jakim jest Java.

Nie mam wątpliwości, że rozprawa stanowi istotny, oryginalny wkład w rozwiązywanie problemu, jakim jest kwestia jednoznaczności definiowania semantyki języków programowania obiektowego.

Oznacza to, zgodnie z art. 13.1. Ustawy z dnia 14.03.2003 (o stopniach naukowych i tytule naukowym), że przedstawiona praca pana Marka Warpechowskiego spełnia wszelkie wymogi ustawy, stawiane rozprawom doktorskim. W mojej opinii, jest to naprawdę bardzo dobra praca doktorska.

Proszę zatem Wysoką Radę Wydziału Matematyki, Informatyki i Mechaniki Uniwersytetu Warszawskiego o przyjęcie pracy pana Marka Warpechowskiego jako rozprawy doktorskiej, i dopuszczenie jej autora do dalszych etapów przewodu doktorskiego.

Wilhelm Dąbko