

ПОРІВНЯЛЬНИЙ АНАЛІЗ СИСТЕМ ПРОЕКТУВАННЯ
ПРОБЛЕМНО-ОРІЄНТОВАНИХ МОВ ПРОГРАМУВАННЯ

Очеретяний О.К., к.т.н. Баклан І.В.

НТУУ «Київський політехнічний інститут імені Ігоря Сікорського»

Київ, Україна

Вступ. Доменні мови приваблюють все більше і більше користувачів, оскільки вони спеціалізовані та оптимізовані для певної проблемної област. В даний час кількість нових мов програмування є значною [1], але мови загального призначення часто не відповідають конкретним потребам кінцевих користувачів. Доменні мови - один із способів вирішити цю проблему. Доменні мови можуть бути розділеними на різні незалежні виміри: внутрішній проти зовнішнього, чи текстовий проти графічного чи табличного. Основний фокус у порівнянні віддається текстовим синтаксисам, оскільки вони мають кілька додаткових переваг, інтеграції в існуючі засоби розробки, доповнення коду, маркери помилок та швидкі виправлення. Тоді як Фаулер описав початкове бачення систем підтримки проектування мов програмування [2], кілька зрілих текстових систем з'явилися в останні роки.

Мета роботи. Таким чином, виникає необхідність дослідження ефективності існуючих систем підтримки проектування мов програмування з отриманням критеріїв якості процесу проектування проблемно-орієнтованих мов програмування. Також, необхідно з'ясувати, які системи підтримки проектування найкраще забезпечують ефективну розробку проблемно-орієнтованих мов

Дослідження. Там, де проблемно-орієнтовані мови перейшли у виробниче використання, результати були дуже обнадійливими [3]. Однак багато проектів, які перейшли на використання проблемно-орієнтованих мов, зупинилися до стадії виробництва або навіть пілотної стадії. Окрім звичайних причин, які впливають на всі проекти, одні з них найпоширеніших скарги були на обраний інструмент – доменну мову або його інструмент її моделювання. З широким різноманіттям способів проектування для проблемно-орієнтованих мов, існує напрочуд невелика кількість емпіричних досліджень, що порівнюють їх. На даний час існує велика різниця в кількості часу що необхідні для досягнення

однакової ефективності у використанні доменної мови при використанні різних інструментів проектування. Деякі системи підтримки проектування проблемно-орієнтованих мов вимагають роботи в 50 разів більше, ніж інші [4]. Розробники мов, які також є програмістами, можуть віддати перевагу роботі з фреймворком, а не з системою проектування мов, але це може бути в 2000 разів повільніше [5].

Проведення емпіричного порівняння систем підтримки проектування мов програмування (СППМП) є важким, особливо з огляду на широкий спектр зусиль, необхідних для досягнення однакових результатів. У більшості випадків лише малі мови можуть бути побудовані виключно для експерименту: інакше використання менш ефективних інструментів займе занадто багато часу експерименту. Більш реалістичні дані доступні з повнорозмірних промислових використань завжди, за визначення проблемно-орієнтованості, базуватися на побудові різних мов, а отже не підійдуть для дослідження.

Таким чином, емпіричні дослідження в промисловості, як правило, концентруються на порівняння продуктивності проектувальних систем з проблемно-орієнтованих мов до продуктивності до використання даних мов. Ці цифри мають пряму цінність для компанії при оцінці того, який підхід використовувати для побудови їхньої продукції, а також при оцінці початкової вартості та віддачі щодо інвестування у використання проблемно-орієнтованих мов в іншому домені. Інша компанія, дивлячись на результати, може побачити, що використання засобу проектування були успішними, а отже, чи гідні вони їх розгляд. Однак дане порівняння ніяк не покаже різницю у використанні різних засобів проектування.

Таким чином, можливо кілька різних видів порівняння, наприклад:

- порівняння систем проектування як різних способів створення проблемно-орієнтованих мов для тієї самої мови;
- порівняння зусиль щодо оновлення отриманої проблемно-орієнтованої мови коли змінюється засіб проектування, проблема або доменне рішення;
- порівняння продуктивності отриманої проблемно-орієнтованої мови проти рукописного написання того самого коду;
- порівняння продуктивності різних зроблених мов для одного домену з різними робочими середовищами.

Висновок. У дослідженні наведено порівняння систем підтримки проектування мов програмування, що орієнтовані на роботу із проблемно-орієнтованими мовами. Перевагами даних систем є орієнтація на роботу із конкретними областями для опису нотаций мов програмування. Системи дозволяють створювати необхідні засоби для роботи із спроектованими мовами програмування та забезпечують базові покращення в процесі проектування. Недоліками даних систем є відсутність схожості в роботі систем та продуктах які отримує розробник на виході.

Література

1. S. Dmitriev. Language oriented programming: The next programming paradigm. JetBrains onBoard, 1(2), 2004
2. Martin Fowler. 2010. Domain Specific Languages (1st. ed.). Addison-Wesley Professional.
3. Kelly, S., and Tolvanen, J.-P. 2008. Domain-Specific Modeling: Enabling Full Code Generation. Wiley.
4. El Kouhen, A., Dumoulin, C., Gérard, S., and Boulet, P. 2012. Evaluation of Modeling Tools Adaptation. CNRS HAL.
5. Kelly, S. 2004. Comparison of Eclipse EMF/GEF and MetaEdit+ for проблемно-орієнтованих мов. In Proceedings of the OOPSLA & GPCE Workshop on Best Practices for Model Driven Software Development.