ДОДАТОК А

Графічний матеріал кваліфікаційної роботи

Міністерство та науки України
Харківський національний університет радіоелектроніки

Кафедра ЕОМ

КВАЛІФІКАЦІЙНА РОБОТА
«Методи аналізу текстів програм на основі семантичних моделей»

Виконала:
ст. гр. СПзм-20-1
Бочарова О.О.

Керівник:
зав.каф. Коваленко А.А.

Мета та завдання

Метою кваліфікаційної роботи є досліження існуючих методів аналізу текстів програм на основі семантичних моделей.

Об'єкт дослідження: вихідні тексти та семантичні моделі програм, які написані з використанням декількох мов програмування.

Завдання:

- аналіз способів внутрішнього подання та обробки редагованого тексту програм в інтегрованих середовищах розробки з метою виявлення недоліків існуючих засобів підтримки мультимовних програмних проектів;
- розробка методу предметно-орієнтованого аналізу вихідних текстів програм у процесі редагування на основі семантичних моделей;
- розробка та реалізація програмного засобу аналізу вихідних текстів програм.

Життєвий цикл програмного продукту

3

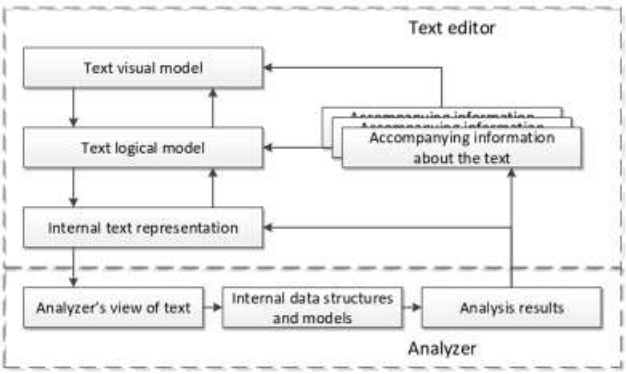
4

Популярні IDE

	Назва IDE	Сайт проекту
1	Visual Studio	https://visualstudio.microsoft.com/
2	Eclipse	https://www.eclipse.org/
3	Visual Studio Code	https://code.visualstudio.com/
4	IDEA	https://www.jetbrains.com/idea/
5	NetBeans	https://netbeans.org/
6	Xcode	https://developer.apple.com/xcode/

5

Схема процесу передачі інформації в текстових редакторах IDE



6

Приклад некоректного аналізу тексту програми

```
    SmtpUseSsl = false,
    SmtpUseDefaultCredentials = true,
    SmtpD                                                    pDirectory,
    SmtpP
};

private static readonly DisposableList _disposables = new DisposableList();

[BeforeTestRun]
public static void Setup()
{
    _disposables.Add(new MicroLearningService(TestServiceConfiguration));
}
```

bool LearningServiceConfiguration.SmtpUseDefaultCredentials { get; set; }

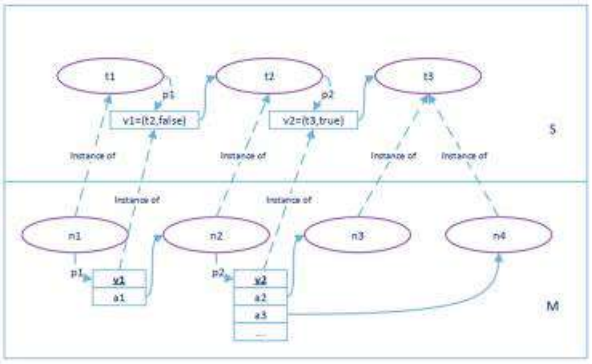
The field 'ServiceContext._disposables' is never used

7

Типова схема розширюваності інтегрованих середовищ розробки щодо семантичного аналізу текстів програм

8

Ілюстрація формального уявлення семантичної моделі



9

Метод аналізу вихідних текстів програм

Крок 1. Нехай безліч Ψ – тексти т програмної системи, написані на мовами з синтаксичними специфікаціями $S_т$:

$(т_n, S_{т_n}) \in \Psi, 1 \le n \le |\Psi|$,

де $S_{т_n}$ – синтаксична специфікація тексту програми $т_n$, n – номер тексту т програмної системи.
Тоді результатом синтаксичного аналізу цих текстів буде безліч синтаксичних моделей Υ:

$\Upsilon = \{\upsilon_{S_т} \forall (т, S_т) \in \Psi\}, |\Psi| = |\Upsilon|$,

де безліч Ψ – тексти $т$ програмної системи.

$\upsilon_{S_т}$ – синтаксична модель за специфікацією $S_т$ для тексту програми $т$.

Крок 2. Нехай Θ – предметні галузі, у межах яких виконується аналіз текстів програмної системи, задані набором специфікацій семантичних трансляцій:

$\Theta = \{R_n = (S_{1_n}, S_{2_n}, \_, \_, \_)\}, 1 \le n \le |\Theta|$,

де Rn – специфікація семантичної трансляції,

S_{1_n} – специфікація вихідної семантичної моделі,

S_{2_n} – специфікація цільової семантичної моделі.

Крок 3. Предметно-орієнтований аналіз на основі семантичних моделей для тексту т дає набір семантичних моделей Ωт, що є набором всіх семантичних моделей, отриманих у результаті трансляцій $f(m, r)$, починаючи з синтаксичної моделі прямо чи побічно для даної семантичної моделі $m = (S, \_, \_, \_, \_, \_)$ при наявності відповідної специфікації семантичної трансляції $r = (S, \_, \_, \_, \_, \_) \in \Theta$ для специфікації семантичної моделі S:

10

Метод аналізу вихідних текстів програм

$$\Omega_т = \bigcup_{\substack{m_0 = \upsilon_{S_т} \\ m_n = (S, \_, \_, \_, \_, \_) \\ \exists r_{n \to n+1} = (S, \_, \_, \_, \_, \_) \in \Theta}} \{m_n\}, m_{n+1} = f(m_n, r_{n \to n+1})$$

де m_n – семантична модель (2.5), специфікація семантичної трансляції $r_{n \to n+1}$, $f(m_n, r_{n \to n+1})$ – функція, що здійснює трансляцію:

$$f(m_{S_a}, r_{a \to b}) = \varsigma(S_b, \{m' = \imath(m_{S_a}) \forall \imath \in I,\ m_{S_a} = (S_a, \_, \_, \_, \_, \_)$$
$$r_{a \to b} = (S_a, S_b, Q, P, I)\}),$$

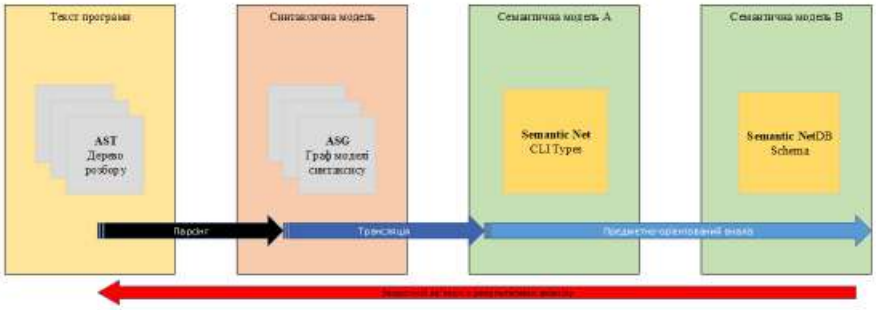
де $r_{a \to b}$ – специфікація семантичної трансляції з семантичної моделі m_{S_a} за специфікацією S_a у семантичну модель за специфікацією S_b, m' – часткова цільова модель, що отримується в результаті застосування трансляції $\imath \in I$ з специфікації $r_{a \to b}$ до моделі m_{S_a}, ς – функція, що поєднує набір семантичних моделей m'_Σ із загальною специфікацією S в одну загальну модель:

$$\varsigma = (S, m'_\Sigma) \to (S,$$
$$N_b = \varsigma'(m'_\Sigma, (S, N_{m'_\Sigma}, \_, \_, \_, \_) \to N_{m'_\Sigma}),$$
$$A_b = \varsigma'(m'_\Sigma, (S, \_, A_{m'_\Sigma}, \_, \_, \_) \to A_{m'_\Sigma}),$$
$$V_b = \varsigma'(m'_\Sigma, (S, \_, \_, V_{m'_\Sigma}, \_, \_) \to V_{m'_\Sigma}),$$
$$E_b = \varsigma'(m'_\Sigma, (S, \_, \_, \_, E_{m'_\Sigma}, \_) \to E_{m'_\Sigma}),$$
$$F_b = \varsigma'(m'_\Sigma, (S, \_, \_, \_, \_, F_{m'_\Sigma}) \to F_{m'_\Sigma})$$
$$),$$
$$\varsigma' = (Y, z) \to \cup z(y) \forall y \in Y$$
$$M_\Gamma = \cup \Omega_т \forall \upsilon_{S_т} \in \Upsilon,$$

$$\Gamma = \{\gamma_S = \varsigma(s, X) : \forall x = (s, \_, \_, \_, \_, \_) \in X \subseteq M_\Gamma\},$$

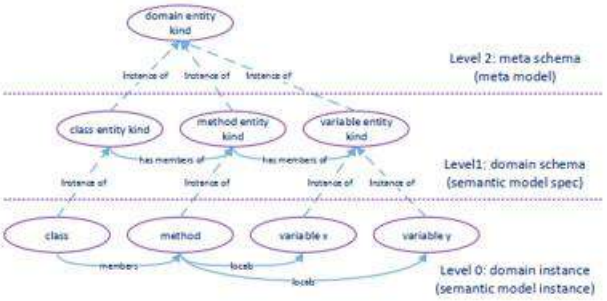
11

Приклад ходу аналізу для моделі БД у програмі C#



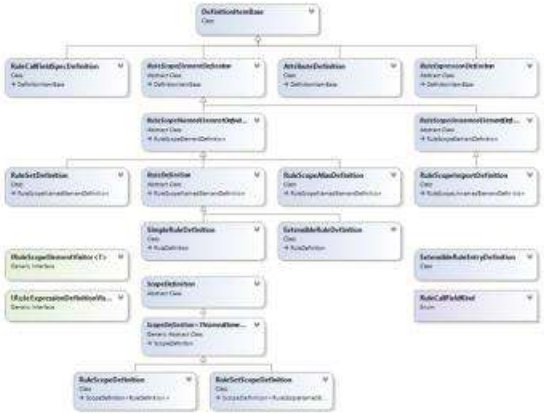
12

Приклад семантичної метамоделі (level 1) та моделі (level 0) програмного коду для мови C#



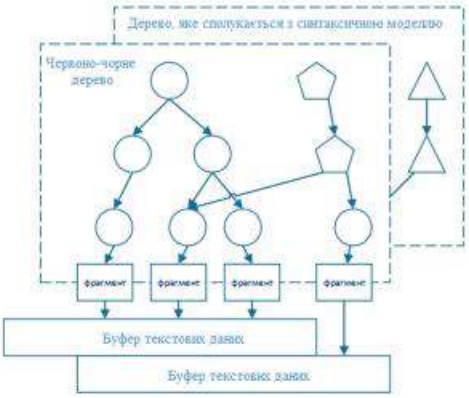
13

Діаграма класів об'єктної моделі специфікацій синтаксичних моделей



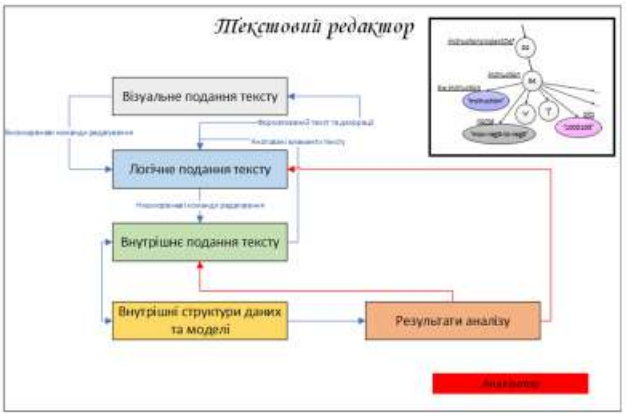
14

Структура даних текстового документа усередині редактора



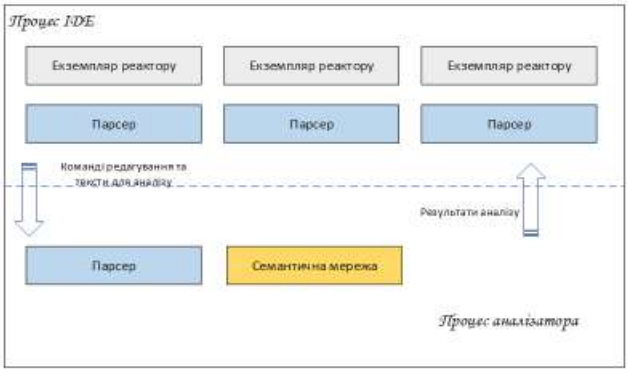
15

Запропонована архітектура текстового редактору



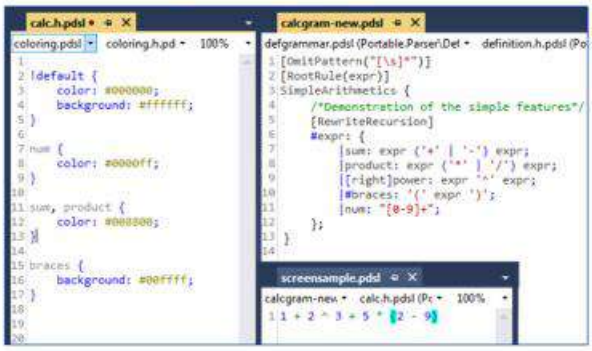
16

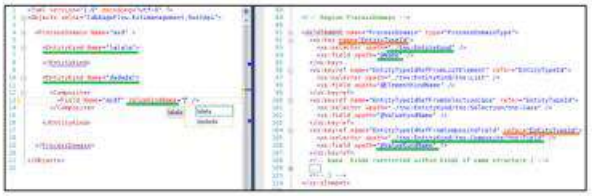
Ілюстрація програмної архітектури



17

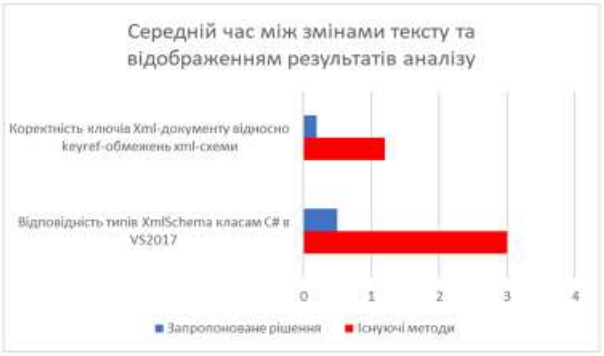
Створений текстовий редактор у середовищі Visual Studio 2017





18

Результати експериментів



ВИСНОВКИ

19

Розроблено метод предметно-орієнтованого аналізу на основі ітеративних перетворень семантичних моделей програм, що дозволяє здійснювати семантичний аналіз текстів програм, що розробляються із застосуванням кількох мов програмування. Розроблено програмну архітектуру та структури даних для подання тексту та семантичних, що динамічно змінюються моделей програм, що дозволяє інтегрувати реалізацію розробленого методу предметно-орієнтованого аналізу в різні середовища розробки.