

ІНТЕЛЕКТУАЛЬНИЙ СЕРВІС ДЛЯ ПРОГРАМУВАННЯ МОВОЮ FORTRAN У MICROSOFT VISUAL STUDIO

Штогрин М.Ю.¹⁾, Малашко І.О.²⁾, Гуменюк А.М.³⁾, Микитюк В.А.⁴⁾

Західноукраїнський національний університет
^{1)магістрант;} ^{2)магістрант;} ^{3)аспірант;} ^{4)аспірант;}

I. Постановка проблеми

Ріст значення ефективності в будь-якому секторі економіки є одним із ключових факторів, що впливають на оцінку ефективності підприємства, яка дає можливість реалізувати ефективну і прибуткову стратегію розподілу ресурсів[1-3]. Оцінка ефективності необхідна не тільки власникам і менеджерам, але і кредиторам, враховуючи щорічний ріст збиткових компаній. Тому розуміння важливості ефективності і її оцінки можуть допомогти підприємствам уникнути банкрутства і приймати більше обґрунтовані рішення при розподілі ресурсів.

II. Мета роботи

Метою дослідження є розробка сервісу для програмування мовою Fortran у Microsoft VisualStudio.

III. Розробка моделей мовного пакету

Перерахуємо завдання, які має вирішити мовний сервіс FRIS для врахування специфіки написання сучасних складноструктурованих програм:

- надавати контекстно-залежну допомогу, в якій, крім визначення елемента мови програмування, має бути присутнім його смисловий опис;
- мати вбудовану підтримку зовнішніх загально використовуваних бібліотек;
- мати вбудовану підтримку сучасних засобів паралельного програмування: MPI, OpenMP та SIMD-операцій;
- забезпечувати візуальне виділення елементів зазначених бібліотек та засобів розпаралелювання;
- працювати у процесі написання текстів програм.

Розглянемо розроблені моделі, які забезпечують реалізацію зазначених вимог.

Спочатку визначимо місце, яке займає мовний сервіс у процесі створення програмного забезпечення. Класичною моделлю такого процесу є модель водоспаду чи каскадна модель [1-3]. Вона складається з наступних процесів, або компонент, що йдуть один за одним (див.рис.1).

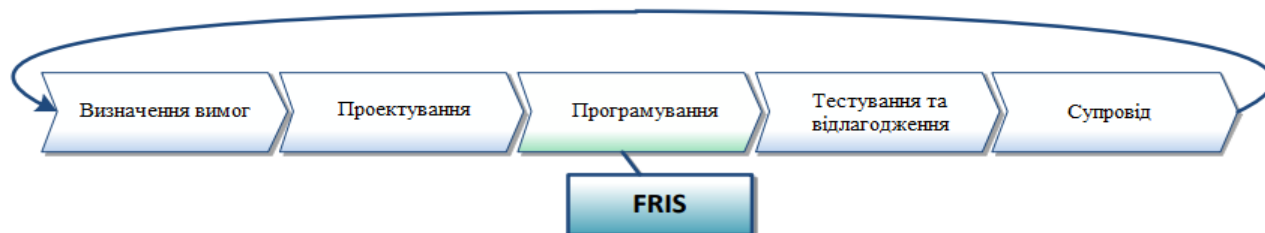


Рисунок 1 –Місце FRIS у процесі створення програмного забезпечення

Мовний сервіс використовується у процесі «програмування» життєвого циклу програмного забезпечення

IV. Абстрактна модель мовного сервісу для розширеної підтримки мови програмування

Реалізувати зазначені вимоги можливо з використанням абстрактної (загальної) моделі мовного сервісу [3], що забезпечує розширену підтримку мови програмування. Модель не прив'язана до будь-якої конкретної мови програмування та інтегрованого середовища розробки. Мовний сервіс може бути представлений у вигляді 5 основних блоків (див.рис.2). Стрілками позначені обміни даними між блоками.

Блок інтеграції з IDE містить реалізацію програмних інтерфейсів, необхідних для взаємодії з IDE. Він відповідає за підписку мовного сервісу на події редагування тексту користувачем у редакторі та за відповідні відгуки,наприклад, за підсвічування синтаксису та надання інформації для роботи можливостей технології IntelliSense.

Блок аналізу відповідає за проведення лексичного, синтаксичного, семантичного аналізу. Отримуючи від блоку інтеграції з IDE ті чи інші події, він виконує відповідні дії.

Блок зберігання розпізнаних елементів є центральним сховищем даних про всі елементи, необхідні роботи мовного сервісу. У загальному випадку він є різновидом таблиці символів. Наповнення блоку зберігання може складатися з двох джерел: з блоку аналізу як результат аналізу файлів з текстами програм і з блоку серіалізації/десеріалізації елементів у разі використання опису API для сторонніх бібліотек у вигляді XML-подання.



Рисунок 2–Структурна схема комплексу програм оцінки ефективності діяльності та управління виручкою підприємств

Блок серіалізації/десеріалізації елементів виконує дві функції. По-перше, він дозволяє зберігати вміст програмних проектів у вигляді XML файлів опису API та коментарів документування до них. По-друге, він дозволяє відновлювати вміст програмних проектів з їх XML моделей.

Блок моделі представлення елементів є сполучною ланкою, своєрідним адаптером елементів блоку зберігання до того виду, який необхідний для використання в блоці інтеграції з IDE. Так, розпізнані елементи можуть містити деяку інформацію, яка не потрібна функціям технології IntelliSense, або навпаки, не містити потрібної інформації. У моделі представлення елементів містяться типи даних – адаптери для елементів блоку зберігання, які відповідають вимогам блоку інтеграції з IDE. Також тут реалізуються всілякі функції вибірки та пошуку необхідної інформації.

При реалізації мовного сервісу слід враховувати, що аналіз текстів програм необхідно проводити у режимі безпосереднього його редагування користувачем. Це означає, що в більшості випадків аналізований текст буде перебувати в лексично, синтаксично або семантично некоректному стані з погляду специфікації мови програмування. Цю особливість необхідно враховувати під час побудови відповідних аналізаторів.

Друга особливість полягає в тому, що аналіз для підсвічування синтаксису здійснюється у VisualStudio у строковому режимі. Аналізатору, в термінах VS колорайзеру, для аналізу передається рядок і стан, в якому він знаходився в кінці аналізу попереднього рядка. Це означає, що відповідний аналізатор необхідно проектувати з урахуванням можливості збереження свого стану на довільний момент часу та відновлення своєї роботи з будь-якого такого стану. Подібний підхід дозволяє проводити інкрементальний аналіз, що особливо актуально для великих файлів із текстами програм (понад 10000 рядків). Тоді при зміні частини рядків необхідно провести аналіз цих рядків, а не всього файлу в цілому.

Висновок

Розроблено абстрактну модель мовного сервісу, що забезпечує розширену підтримку мови програмування та призначена для побудови мовних сервісів для різних мов програмування та інтегрованих середовищ розробки. Конкретизація даної моделі для мови програмування Fortran та середовища розробки Microsoft VisualStudio дозволяє вирішити завдання щодо врахування особливостей написання сучасних складно структурованих програм.

Список використаних джерел

1. Hurd, Walter J., and Charles M. Anderson. Fortran Programming and Numerical Methods for Scientists and Engineers. CRC Press, 2018.
2. Metcalf, Michael, John Reid, and Malcolm Cohen. Modern Fortran Explained. Oxford University Press, 2011.
3. Chapman, Stephen J. Fortran for Scientists & Engineers. McGraw-Hill Education, 2018