

Огляд інструментів для побудови UML-діаграм класів

Кодола Г. М., Волинець Н. С.
ДВНЗ «УДХТУ», Gkodola@gmail.com

In this work the overview of modern software with open source and online versions for building UML-diagrams considered. An example of the construction of the class diagram shown.

В основі об'єктно-орієнтованого підходу (ООП) покладена об'єктна декомпозиція, при цьому статична структура програмного забезпечення (ПЗ) описується в термінах об'єктів і зв'язків між ними, а динамічний об'єкт ПЗ описується в термінах обміну повідомленнями між об'єктами. Кожний об'єкт системи має свою власну поведінку, яка моделює поведінку об'єкта реального світу.

Об'єктна модель є природним засобом представлення реального світу. Вона становить концептуальну основу ООП. Основними принципами її побудови є: абстрагування, інкапсуляція, модульність, ієрархія.

До основних понять ООП (елементів об'єктної моделі) відносяться: об'єкт, клас, атрибут, операція, компонент, зв'язок, поліморфізм, спадкування.

Об'єкт – це сутність предметної області або програмної системи, який має чітко визначену поведінку. Будь-який об'єкт має стан, поведінку та індивідуальність. Стан об'єкта визначається значенням його властивостей (атрибутів) і зв'язками з іншими об'єктами, воно може змінюватися з часом. Поведінка визначає дію об'єкта і його реакцію на запити від інших об'єктів. Поведінка представляється за допомогою набору повідомлень, які сприймаються об'єктом (операції, які може виконати об'єкт).

Структура і поведінка складних об'єктів визначають загальний для них клас. Клас – це множина об'єктів, які зв'язані спільністю властивостей, поведінки, зв'язку і семантики. Будь-який об'єкт є екземпляром класу. Визначення класів і об'єктів – одна з самих складних задач об'єктно-орієнтованого проектування.

Між елементами об'єктної моделі існують різні типи зв'язків:

- асоціація – це семантичний зв'язок між класами;

- агрегація – більш сильний тип зв'язку між цілим і його частинами;

- залежність – зв'язок між двома елементами моделі, при якому зміни в специфікації одного елементу можуть привести до змін в іншому елементі;

- узагальнення – зв'язок «тип-підтип».

Для відображення об'єктних моделей використовуються діаграми класів.

Діаграма класів – це набір статичних, декларативних елементів моделі. Діаграми класів можуть застосовуватися при прямому проектуванні, тобто в процесі розробки нової системи, так і при зворотному проектуванні – опису існуючих систем і систем, що використовуються. Таким чином, діаграма класів – це остаточний результат проектування і відправна точка процесу розробки.

Архітектор ПЗ в першу чергу звертає увагу на об'єкти предметної області. А програміст концентрує увагу на поведінці цих об'єктів, використовуючи класи, до яких вони належать. Тому діаграма класів і є однією з найважливіших діаграм UML.

UML (Unified Modeling Language – уніфікована мова моделювання) – мова графічного опису для об'єктного моделювання в області розробки ПЗ. UML є мовою широкого профілю, це відкритий стандарт, що використовує графічні позначення для створення абстрактної моделі системи, яка називається UML моделлю.

UML був створений для визначення, візуалізації, проектування і документування в основному програмних систем. UML не є мовою програмування, але в засобах виконання UML-моделей, як коду, що інтерпретується, можлива кодогенерація.

Діаграма UML – це графічне зображення елементів системи у формі зв'язаного графа з вершинами (сутностями) і ребрами (відношеннями).

Класи – головні сутності при моделюванні програмних систем за допомогою UML. Моделювання класів дає змогу побудувати та представити систему в термінах об'єктно-орієнтованої концепції. Таке представлення системи має важливі переваги порівняно з іншими моделями. По-перше, модель класів відображає предметну область задачі, а, отже, є зрозумілою спеціалісту в цій предметній області, що дає змогу залучати до проектування системи вузькопрофесійних спеціалістів. По-друге, модель класів тривіально перетворюється в реалізацію на об'єктно-орієнтованих мовах з повним збереженням семантики моделі.

Моделі класів в UML можуть відображатися різними типами діаграм, кожен з яких відображає ті чи інші аспекти моделі і, відповідно, організації системи. Найчастіше моделі класів зображають за допомогою діаграм класів (Class Diagram) [1-2].

На даний час на ринку присутня велика кількість і повноцінних засобів для UML-моделювання, і програм для побудови діаграм, в тому числі і UML.

Такі продукти, як Acceleo, ArgoUML, Astade, Dia, Gaphor, NetBeans, Umbrello UML

Modeller, Software Ideas Modeler, StarUML можуть бути завантажені з сайту виробника безкоштовно.

З розглянутих можна виділити StarUML 2008 року оновлення, та Software Ideas Modeler, 2013 року оновлення, які виглядають найбільш функціональними з безкоштовних продуктів і можуть виконувати роль повноцінної заміни комерційних програм для UML-моделювання.

Серед онлайн-ових середовищ UML-проектування можна виділити наступні: Gliffy, Creately, Lucidchart, Cacco, які від вищеозначених програмних засобів мають наступні переваги:

– для роботи з цими програмами потрібен браузер і доступ до Інтернету;

– можливість сумісної співпраці онлайн для команди розробників.

Дані програми мають trial версії, які дозволяють безкоштовно оцінити можливості і переваги цих програм.

Розглянуті пакети – дуже мала частина всього доступного в Інтернеті програмного забезпечення для візуального моделювання за допомогою UML. Перелік іншого ПЗ для створення UML-діаграм можна знайти, наприклад, на [3] Для ознайомлення з переліком і основними характеристиками онлайн ПЗ можна, наприклад, в роботі [4]. На сайті [5] розташовано перелік посилань на інші каталоги подібних програм.

Розглянемо приклад побудови UML-діаграми для наступної задачі.

На деякому підприємстві маєтсья декілька типів співробітників з різною формою оплати праці. Нехай є наступні види оплати:

1. Фіксована зарплатня на рік.
2. Фіксована зарплатня на рік плюс відсоток від продажу, виконаних робітником.
3. Погодинна зарплатня.
4. Погодинна зарплатня плюс оплата за підвищеним тарифом за час роботи поза нормою.

Для побудови діаграми класів була обрана середовище Software Ideas Modeler

В UML класи зображують прямокутником, розділеним на три секції, в яких послідовно записують назву класу, атрибути та операції. Назву абстрактного класу позначають курсивом. Атрибути та

операції мають чітко визначені формати запису, які відображають їхні найважливіші характеристики (назви, типи тощо). За необхідності секції атрибутів і/або операцій можна опустити. Назва класу має бути унікальною в межах пакета, в якому розташовано клас.

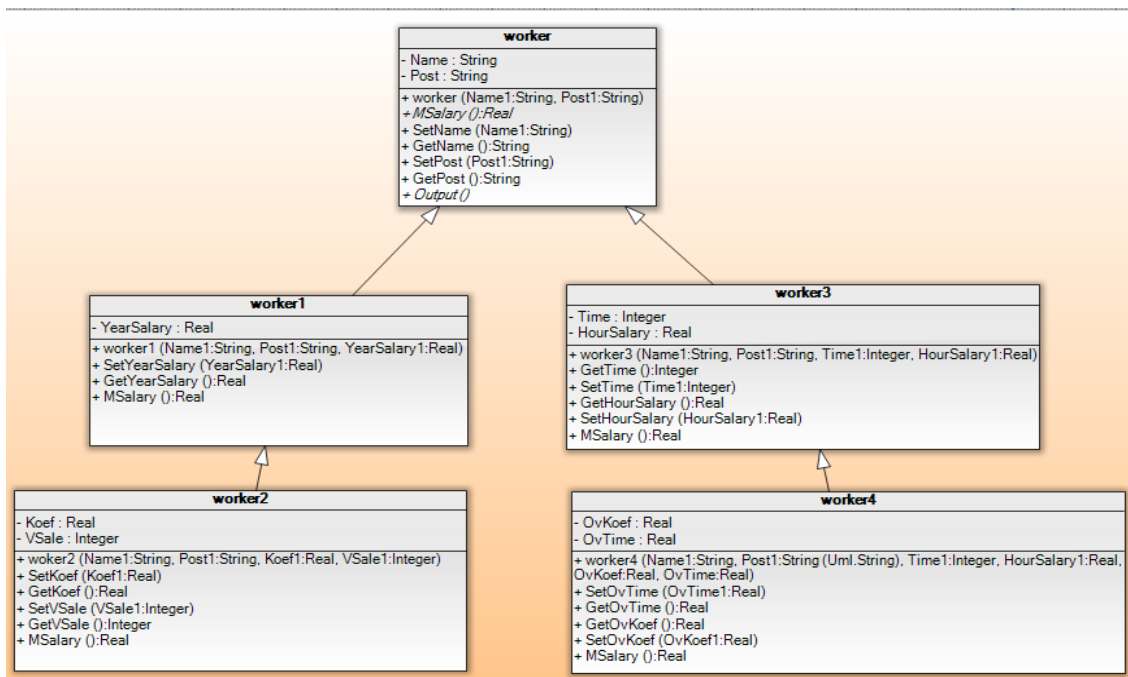


Рисунок 1 – Діаграма класів

На рис. 1 в верхній частині кожного прямокутника знаходиться найменування класу, а в нижній – його поля і методи. Базовим є клас *worker*, який має наступні поля: *name* – ПІБ робітника, *post* – посада робітника. *Salary* – віртуальна абстрактна функція для розрахунку місячної зарплатні. *Output* – абстрактна функція для виводу інформації про робітника на екран. Інші поля мають таке значення: *YearSalary* – зарплата за рік, *Time* – відпрацьовані години, *HourSalary* – оплата за годину, *Koef* – коефіцієнт від продажу, *VSale* – об'єм продаж, *OvKoef* – коефіцієнт переробки, *OvTime* – час переробки.

Вибір середовища UML-проекування – питання складне та неоднозначне, і вирішити

його кожний повинен для себе сам, виходячи з своїх потреб, рівня знань і т. і.

ЛІТЕРАТУРА

- [1] Гради Буч, Роберт А Максимчук. Объектно-ориентированный анализ и проектирование с примерами приложений (3-е издание). – М.: Вильямс, 2008. – 721 с.
- [2] Дудзяний І. М. Об'єктно-орієнтоване моделювання програмних систем. http://ami.lnu.edu.ua/books/ami_OOM.pdf
- [3] http://www.objectsbydesign.com/tools/umltools_byCompany.html.
- [4] Morgan Masters. How the BA Can Take Advantage of Free Online Diagramming Tools <http://www.modernanalyst.com/Resources/Articles/tabid/115/articleType/ArticleView/articleId/1556/How-the-BA-Can-Take-Advantage-of-Free-Online-Diagramming-Tools.aspx>.
- [5] <http://www.uml.org/#Links-UML2Tools>