

ОСОБЛИВОСТІ СТВОРЕННЯ СУЧАСНИХ ІНФОРМАЦІЙНИХ СИСТЕМ

Булашенко А.В., *викладач*
Шосткінський інститут СумДУ

Клас програмованих засобів (часто інтегрованих із CASE-системами) складають програмні системи мов програмування четвертого покоління (4GL). Такі мови являють користувачу більш зручні засоби для формування інтерфейса із кінцевим користувачем (наприклад, у вигляді меню чи форм), забезпечують порівняно прості можливості для взаємодії із системою управління базами даних, а також являють собою (звичайно, досить примітивні) засоби програмування. Основною перевагою мов четвертого покоління є те, що вони забезпечують можливість так званого "швидкого прототипування додатків (rapid prototyping)".

Це означає, що при використанні 4GL можна дійсно швидко створити працюючий прототип майбутньої системи, що забезпечує необхідний інтерфейс із кінцевим користувачем та взаємодіючий із макетом бази даних (а можливо, і з реальною базою даних, якщо вона до цього часу підготовлена). Доводиться говорити про "прототип", оскільки більшість сучасних систем 4GL не забезпечують тієї ефективності прикладних систем, яку дають звичайні мови програмування (як тепер прийнято називати, 2GL чи 3GL). Разом з тим, слід помітити, що вже існує маса реально використовуваних інформаційних систем, розроблених виключно на тому або іншому 4GL. Кто знає, що буде завтра, але тенденція до збільшення використання 4GL очевидна.

В кінці кінців, деяка кількість зауважень по поводу методології проектування та розробці інформаційних систем. Для професійних програмістів постійною проблемою є розрив між даними та програмою. При наявності складно структурованої інформації проектування схеми бази даних є не менш складною задачею, ніж написання власної прикладної системи.

Очевидно, що завжди гірше мати дві складні задачі, ніж одна, навіть якщо у сукупності її складність перевищує складність кожної їх перших задач окремо. Розв'язання цієї проблеми пропонує об'єктно-орієнтований підхід. Якщо говорити зовсім коротко, суть цього підходу складається у тому, що проектуються не данні та програми окремо, а об'єкти, що поєднують в собі і данні, та програми, інформаційно та функціонально характеризуючі відповідні суттєвості предметної області. Підхід корисний як з методологічної точки зору (зникають дві різні характеристики предметної області - данні та програми поєднуються у об'єкти), так і з точки зору техніки проектування та розробці програмних систем (замість двох технічно не пов'язаних, але логічно переплєтених гілок утворюється один надійний стовбур).

Зазначимо, що останнім часом подавляюча більшість CASE-систем та 4GL, якщо не орієнтуються, то звертають увагу на об'єктно-орієнтований підхід. Більш того, почали з'являтися методики по об'єктно-орієнтованому використанню засобів автоматизованого проектування та розробці інформаційних систем, які (засоби) початково для цього не призначалися. На сьогодні розробнику інформаційної системи, як мінімум, потрібно мати базові знання о сучасних СУБД, що орієнтуються у світі інструментальних засобів розробці програмних систем та мати уявлення про об'єктно-орієнтований підхід до проектування розробці програм. Очевидно, що при цьому непогано ще вміти програмувати.