

УДК 004.773

М.Р. Петрик, докт. фіз.-мат. наук, проф., П.П. Теслюк

Тернопільський національний технічний університет імені Івана Пулюя, Україна

МЕТОДИ МОДЕЛЮВАННЯ СИСТЕМИ ПЛАНУВАННЯ РЕСУРСІВ ПІДПРИЄМСТВА

Petryk, Ph. D., P.P. Tesliuk

METHODS OF MODELING THE ENTERPRISE RESOURCE PLANNING SYSTEM

Система планування ресурсів підприємства (ERP-система) (англ. Enterprise Resource Planning System) — корпоративна інформаційна система (КІС), призначена для автоматизації обліку й управління. Як правило, ERP-системи будуються за модульним принципом і в тому або іншому ступені охоплюють всі ключові процеси діяльності компанії.

Системи планування ресурсів підприємства створюються в процесі колективної праці фахівців різних предметних областей. Діяльність, спрямована на створення інформаційної системи (ІС), називається розробкою ІС. Вона зазвичай здійснюється поетапно з урахуванням можливого життєвого циклу інформаційних продуктів і послуг.

Для розробки ІС використовують різні моделі і методи. Моделювання ІС може здійснюватися з використанням унікальних і універсальних методів. Так, наприклад, послідовність розробки ІС здійснюється з використанням таких самих методів моделювання, що і використовуються при визначенні життєвого циклу ІС. Розглянемо методи моделювання, що застосовуються при розробці інформаційних систем.

Концептуальне моделювання являє собою структурований процес створення систем, що складається з наступних етапів: аналіз, проектування, програмування, тестування, впровадження.

Важливою формою системного аналізу складних систем є імітаційне моделювання, що описує процеси функціонування систем у вигляді алгоритмів. Його застосовують у випадках, коли необхідно врахувати велику різноманітність вихідних даних, вивчити протікання процесів в різних умовах. Процес імітації на будь-якому етапі може бути припинений для проведення наукового експерименту на описовому рівні, результати якого після оцінки та обробки можуть бути використані на наступних етапах імітації.

Каскадна модель ІС складається з послідовно виконаних етапів. Кожен етап повністю закінчується до того, як почнеться наступний. Етапи не перекриваються в часі: наступний етап не починається до тих пір, поки не завершиться попередній. Повернення до попередніх етапів не передбачено або всіляко обмежено. виправлення помилок відбувається лише на стадії тестування. Результат з'являється тільки в кінці розробки ІС. Критерієм появи результату є відсутність помилок і точну відповідність отриманої ІС первісної її специфікації. Для цієї моделі характерна автоматизація окремих непов'язаних завдань, що не вимагає виконання інформаційної інтеграції та сумісності, програмного, технічного та організаційного сполучення. В рамках вирішення окремих завдань каскадна модель за термінами розробки та надійності виправдовувала себе. Застосування каскадної моделі до великих і складних проектів, внаслідок великої тривалості процесу проектування і мінливості вимог, за цей час призводить до її практичної не реалізованості.

У ітераційній моделі з проміжним контролем, так само, як і в моделі “водоспад” використовується послідовність розташування етапів створення ІС. Але кожен

наступний етап має зворотний зв'язок з попередніми етапами. виправлення помилок відбувається на кожному з етапів, відразу при виявленні проблеми – проміжний контроль. Наступний етап не починається, поки не завершиться попередній. При першому проході по моделі зверху вниз, як тільки виявлена помилка, здійснюється повернення до попередніх етапів (від низу до верху), що викликав помилку. Етапи виявляються розтягнутими в часі. Результат з'являється тільки в кінці розробки ІС, як і в моделі “водоспад”.

Спіральна модель (spiral model) була розроблена у середині 1980-х років Барі Боемом. Вона ґрунтується на класичному циклі Демінга PDCA (plan-do-check-act). При використанні цієї моделі ІС створюється в кілька ітерацій (витків спіралі) методом прототипування.

Нині ця модель досить поширена. Найвідоміші приклади її реалізації - це RUP (Rational Unified Process) фірми Rational і MSF (Microsoft Solution Framework). Створення ІС за такої моделі має ітераційний характер і рухається по спіралі, проходячи стадії, де на кожному витку уточнюються характеристики майбутнього інформаційного продукту.

Суттєва особливість спіральної моделі полягає в тому, що прикладне ПЗ створюється не відразу, а частково, з використанням методу прототипування. Прототип - це програмний компонент, що реалізує окремі функції і зовнішні інтерфейси ПЗ. Створення прототипів здійснюється кількома ітераціями. Кожна ітерація відповідає створенню фрагмента або версії ПЗ, уточнюються цілі і характеристики проекту, оцінюється якість отриманих результатів і плануються роботи наступної ітерації. На кожній ітерації виробляється ретельна оцінка ризику перевищення термінів і вартості проекту, щоб визначити необхідність виконання ще однієї ітерації, ступінь повноти і точності розуміння вимог до системи, а також доцільність припинення проекту. Спіральна модель позбавляє користувачів і розробників ПЗ від необхідності повного й точного формулювання вимог до системи на початковій стадії, оскільки вони уточнюються на кожній ітерації. У такий спосіб уточнюються і послідовно конкретизуються деталі проекту і зрештою вибирається обґрунтований варіант, який і реалізується.

Об'єктно-орієнтована модель, методологія припускає конструювання програмного рішення з готових об'єктів, для яких визначаються правила їх взаємодії, що переводять об'єкти з одного стану в інший. Така модель, що передбачає повну відповідність процесу розробки положенням об'єктно-орієнтованої методології (об'єктно-орієнтований аналіз, проектування, програмування), ефективна у великих проектах, а також там, де застосовуються так звані засоби швидкої розробки (RAD, Rapid Application Development), засновані на цих технологіях і містять готові бібліотеки класів.

Література

1. М. Петрик, О. Петрик, Моделювання програмного забезпечення, Тернопіль, 2015
2. Рамбо Дж. UML 2.0. Объектно-ориентированное моделирование и разработка. –2-е издание; пер. с англ. Дж. Рамбо, М. Блаха – Питер, 2007. Рамбо Дж. UML 2.0. Объектно-ориентированное моделирование и разработка. – 2-е издание; пер. с англ. Дж. Рамбо, М. Блаха – Питер, 2007.
3. Якобсон, А., Уніфікований процес розробки програмного забезпечення. А. Якобсон, Г. Буч, Дж. Рамбо – СПб.: Питер, 2002. – 496 с.