

УДК 004.94

## СТАН І ПРОБЛЕМИ ОЦІНКИ ЯКОСТІ ІНФОРМАЦІЙНИХ СИСТЕМ

І. В. Гілета, Ю. Ю. Білас

Українська академія друкарства,  
вул. Під Голоском, 19, Львів, 79020, Україна

Здійснено аналіз поняття якості інформаційної системи. Встановлено, що сприйняття та інтерпретація якості залежить від суб'єкта розгляду. Оцінка якості базується на моделі якості. Проаналізовано модель якості МакКола та Боема. У стандарті ISO/IEC 9126 для оцінки якості програмного забезпечення інформаційних систем визначають цілі, атрибути та метрики. Характеристики та атрибути якості інформаційних систем розділені на три групи: категорійні, кількісні та якісні. Проблеми агрегації характеристик якості на основі схеми взаємозв'язку. У стандарті IEEE 1061 розглянуто методологію за ієрархічною схемою. Важливим складником у процесі забезпечення якості програмного забезпечення інформаційних систем є управління якістю під час проєктування, зокрема під час об'єктно-орієнтованого проєктування. Істотний вплив на якість об'єктно-орієнтованих систем мають такі характеристики: способи абстрагування об'єктів, локалізація, інкапсуляція, наслідування та поліморфізм. Відомими метриками для об'єктно-орієнтованих систем є метрики Чайдембера та Кемерера. Розглядається підхід до оцінки якості програмного забезпечення з використанням експертних оцінок. Інтегральний показник якості інформаційних систем формується з урахуванням часткових показників та з врахуванням ваг. Вкрай актуально мати єдиний критерій, який однозначно визначатиме якість інформаційних систем щодо забезпечення потреб бізнесу.

**Ключові слова:** інформаційні технології, програмний продукт, якість програмного коду, показники якості, стандарти якості.

**Постановка проблеми.** Сьогодні інформація та інформаційні технології розглядаються як один із стратегічних ресурсів підприємства. Це характеризує підходи до проєктування, експлуатації та модернізації інформаційних систем з погляду реалізації інтересів замовників. Процеси розробки та впровадження інформаційних систем (ІС) перебувають під строгим управлінським контролем.

Інвестиції, що спрямовуються на інформаційні технології, призначені для успішного розвитку бізнесу, отримання конкурентних переваг. Ступінь, в якому ІС задовольняють потреби бізнесу, залежить від споживчої якості системи. Згідно з міжнародним стандартом ISO 9000:2005 поняття якості продукції визначається як набір характеристик здатності об'єкта (діяльності чи процесу, продукту, послуги тощо) задовольняти встановлені або очікувані потреби. Це визначення можна застосувати до ІС. Водночас потрібно зазначити, що ІС належать до

категорії складних систем, і під час формування методичного підходу до оцінки якості ІС необхідно враховувати різні аспекти: якість системної інфраструктури, програмно-технічного забезпечення, якість даних, якість адміністрування та послуг. Крім того, сприйняття та тлумачення поняття якості залежить від погляду, з якого воно розглядається. Поняття «якість ІС» має різне значення для кінцевих користувачів, клієнтів, менеджерів і розробників.

**Аналіз останніх досліджень та публікацій.** Аналіз літературних публікацій, розгляд стандартів оцінки якості програмного забезпечення, застосування відомих моделей якості ІС виявив, що наявні підходи не дають повного кількісного опису рівня якості, не враховують семантики предметної галузі, операційного та інструментального оточення, а також методології розробки. Традиційні метрики не враховують якість даних та взаємодію даних з потоком обчислень. Отже, потреба визначення інтегрального критерію якості інформаційної системи є актуальним завданням для ухвалення стратегічних рішень щодо забезпечення потреб бізнесу.

**Мета статті** — аналіз методологічних підходів щодо оцінки якості інформаційних систем та вироблення обґрунтованого рішення для обчислення показника якості.

**Виклад основного матеріалу дослідження.** Оцінка споживчої якості інформаційної системи та її програмного забезпечення базується на певній моделі якості.

У публікації [1] модель якості програмного забезпечення, яка відома як модель якості МакКола, визначається трьома групами характеристик: фактори, які формує кінцевий користувач інформаційної системи; критерії, що визначає розробник програмного забезпечення; метрики, які використовуються для кількісного опису та вимірювання якості. Фактори якості згруповані у три групи: функціонування; використання; ревізії.

До групи «Функціонування» входять такі показники: коректність, надійність, ефективність, цілісність та практичність. До групи «Впровадження» належать показники переносимості, повторного використання та здатності до взаємодії (інтеграція компонентів).

Критерії якості в моделі МакКола є кількісними рівнями факторів, які використовуються як цілі під час розробки інформаційної системи.

Метриками якості в моделі визначено такі показники: зручність перевірки на відповідність стандартам; точність управління та обчислень; ступінь стандартності інтерфейсів; функціональна повнота; однорідність використовуваних правил проектування та документації; ступінь стандартності форматів даних; стійкість до помилок; ефективність роботи; розширюваність; широта галузі потенційного використання; незалежність від апаратної платформи; повнота протоколювання помилок та інших подій; модульність; зручність роботи; захищеність; самодокументованість; простота роботи; незалежність від програмної платформи; можливість співвідношення проекту з вимогами; зручність навчання.

Метрика МакКола пропонує оцінювати показники якості за 10-бальною шкалою. Введені метрики можуть впливати на кілька факторів якості, а фактори оцінюються на основі згортки деяких метрик. При цьому коефіцієнти входження

метрик у згортки здебільшого є евристичними і мають конкретизуватися для конкретної організації, видів програмного забезпечення, команд розробки тощо.

Для моделі МакКола одним із суттєвих недоліків є труднощі при оцінюванні метрик за 10-бальною шкалою. В експерта може виникати складність застосування подібної шкали оцінювання до таких характеристик, як розширюваність, самодокументованість і простота роботи. Крім того, процедури згортки метрик обмежують їхнє застосування для різних об'єктів.

У моделі якості за Боемом [2, 3] здійснено розвиток моделі МакКола. Розширений список доповнено такими факторами: ясність; зручність внесення змін; документованість; здатність до відновлення функцій; зрозумілість; адекватність; функціональність; універсальність; економічна ефективність. Для моделі якості за Боемом зберігаються ті ж недоліки, які властиві моделі МакКола.

Міжнародна організація зі стандартизації (International Organization for Standardization, ISO) прийняла стандарт для моделі якості програмного забезпечення ISO/IEC 9126 [4–7]. У цьому стандарті вводяться поняття внутрішньої та зовнішньої якості, а також якості програмного забезпечення під час використання у різних контекстах. Внутрішня якість визначається характеристиками програмного забезпечення без урахування його поведінки. Зовнішня якість характеризує ПЗ зі сторони його поведінки. Якість ПЗ при використанні в різних контекстах — це споживча якість. Потрібно зазначити, що у створенні якісного ПЗ істотною є якість технологічних процесів його розроблення.

У стандарті ISO/IEC 9126 для оцінки якості програмного забезпечення інформаційних систем визначають цілі, атрибути та метрики. Як цілі досягнення заданої якості інформаційної системи розглядаються функціональність, надійність, практичність або зручність використання, ефективність, підтримка, переносимість або мобільність. Цілі визначаються атрибутами якості.

Відповідно до стандарту ISO/IEC 9126 у моделі якості програмного забезпечення використовуються такі цілі та атрибути:

- функціональність: придатність до певної роботи, точність, правильність, здатність до взаємодії, відповідність стандартам та правилам, захищеність;
- надійність: зрілість, завершеність (зворотна до частоти відмов), стійкість до відмов, здатність до відновлення працездатності при відмові, відповідність стандартам надійності;
- практичність, зручність використання: зрозумілість, зручність навчання, працездатність, привабливість, відповідність стандартам практичності;
- ефективність: часові характеристики, використання ресурсів, відповідність стандартам ефективності;
- підтримка: аналізованість, змінність, зручність для внесення змін, ризик виникнення несподіваних ефектів під час внесення змін, контрольованість, зручність перевірки, відповідність стандартам підтримки;
- перенесення, мобільність: адаптованість, зручність установки, здатність до співіснування з іншим ПЗ, зручність заміни іншого ПЗ даних, відповідність стандартам переносимості.

Функціональні вимоги формулюються у вигляді тверджень, що описують поведінку системи, які можуть бути формально перевірені. Надійність характеризує поведінку системи при виході межі допустимих значень параметрів функціонування через збій у системі. Оцінюючи атрибути надійності, застосовуються методи теорії ймовірностей і математичної статистики. Зручність використання досить важко оцінюється, здебільшого використовуються експертні методи. Ефективність належить до найважливіших кількісних показників якості програмних систем. Наявні програмно-апаратні засоби та методики дають змогу прогнозувати інтегральні значення показників ефективності системи. Супроводжуваність характеризує необхідні ресурси на супровід та модернізацію системи, що витрачаються експлуатаційним персоналом. У цьому використовуються методики прогнозування витрат за виконання типових процедур супроводу. Перенесення системи характеризує можливість вибору компонентів системного оточення. Оцінка переносимості може залежати від погляду зацікавлених осіб: переносимість на різні апаратні платформи, переносимість на різні програмні платформи, переносимість на різні апаратні і програмні платформи.

Модель якості, яка створюється в рамках стандарту ISO/IEC 9126, визначається загальними характеристиками програмного продукту. Характеристики залежать від підхарактеристик якості та атрибутів програмного забезпечення, які, на думку авторів стандарту, мають піддаватися точному опису та виміру. Вимоги якості видаються як обмеження на модель якості. Під час оцінки якості продукту спочатку оцінюються атрибути програмного виробу за допомогою метрик і формується шкала оцінки залежно від можливих ступенів відповідності атрибута обмеженням, що накладаються. Для кожної окремої оцінки атрибута градація зазвичай вибирається заново і залежить від вимог якості, що накладаються на нього. Набір отриманих атрибутів є критерієм для оцінки підхарактеристик і характеристик, і як результат якості продукту загалом.

Коментуючи стандарт ISO 9126-4, наголошується, що характеристики, підхарактеристики та атрибути якості програмних систем можна розділити на три групи: категорійні, кількісні та якісні.

Категорійні показники якості програмного забезпечення інформаційних систем характеризують широкий спектр класів, призначень та функцій інформаційних систем. До цієї групи належать характеристики, що визначають функціональність, захищеність та важливість.

Кількісні показники якості програмного забезпечення, такі як надійність та ефективність, можуть бути виміряні та наведені чисельними характеристиками.

Якісні характеристики програмного забезпечення мають описовий, якісний вигляд. Такі характеристики оцінюють експертними способами, при цьому застосовують бальну оцінку або використовують лінгвістичні оцінки рівня якості: відмінний, хороший, задовільний чи незадовільний.

Водночас концепція планування та управління процесами аналізу якості програмного забезпечення, що розглядається, не вирішує питання спільного використання категоріальних, кількісних і якісних показників, і, крім того, проблеми

агрегації характеристик якості ПЗ на основі схеми взаємозв'язку залишаються відкритими. Це зумовлює необхідність розвитку методології інтегральної оцінки якості програмного забезпечення інформаційних систем.

Основним недоліком методології, викладеної у стандартах ISO/IEC 9126, ISO/IEC 25010:2011, є невизначеність у спільному використанні кількісних та якісних характеристик програмного забезпечення, а також не опрацьованість питань агрегації значень характеристик на основі значень атрибутів.

У стандарті IEEE 1061 [10] розглядається методологія за ієрархічною схемою. На верхньому рівні ієрархічної моделі, що визначає погляд менеджерів, користувачів та замовників, задаються нетехнічні характеристики (чинники): вимоги щодо якості; властивості (атрибути), встановлення пріоритетів властивостей та їх зв'язку з вимогами; присвоєння атрибутів факторам якості, що відображають уявлення замовника на якість; встановлення вимірювань для факторів якості та визначення допустимих варіацій для величин якості. На середньому рівні ієрархії, що визначає погляд аналітиків, конструкторів та розробників, задаються характеристики технічного рівня. На цьому рівні здійснюється декомпозиція факторів якості у вимірювані характеристики програмного забезпечення, що визначаються як підфактори. На нижньому рівні моделі виробляють декомпозицію підфакторів метрики, які можуть бути застосовані безпосередньо до програмного продукту або процесу розробки. Цей рівень визначає правила та норми, яким має задовольняти продукт чи процес для забезпечення факторів якості. Метрики слугують як непрямі заходи прямих вимірів чинників якості на верхніх рівнях ієрархії.

На якість інформаційної системи, крім параметрів, що враховуються у моделі якості, впливає рівень зрілості організацій, які здійснюють проектування та розробку ІС. Рівні зрілості організацій визначають стандарти CMM/CMMI (Capability Maturity Model/ Capability Maturity Model Integration) та ISO /IEC 15504 (SPICE).

Стандарт CMM/CMMI визначає п'ять рівнів зрілості організації, які визначають можливості організації із забезпечення певного рівня якості продуктів, що випускаються. Стандарт SPICE (Software Process Improvement and Capability Determination) відображає міжнародний досвід у галузі розробки програмного забезпечення та спрямований на постійне покращення процесу розробки програмного забезпечення. Потрібно зазначити, що вищезазначені стандарти є найбільш розвиненими моделями якості програмних систем, що пройшли випробування на практиці.

Важливим складником у процесі забезпечення якості програмного забезпечення інформаційної системи є управління якістю під час проектування. Особливого значення це набуває при об'єктно-орієнтованому (ОО) проектуванні. Істотний вплив на якість ОО інформаційних систем мають такі характеристики: способи абстрагування об'єктів, локалізація, інкапсуляція, наслідування та поліморфізм.

Способи абстрагування дають змогу виділити суттєві характеристики інформаційної системи без урахування другорядних деталей. Це дає змогу чітко визначити концептуальні межі системи з погляду потенційного користувача. Базовою

абстракцією об'єктно-орієнтованого підходу є клас, який може бути представлений на різних рівнях деталізації у різний спосіб. Об'єктно-орієнтовані метрики абстрагування мають представляти абстракції у термінах вимірів класів.

Локалізація фіксує спосіб групування інформації у програмі та системі. В об'єктно-орієнтованих системах інформація групується усередині класів чи об'єктів. Клас в ОО-системах є сукупністю об'єктів, що мають загальні властивості та методи. Атрибути та методи класу розглядають як його властивості та обов'язки відповідно. Атрибути представляють обов'язок знання чогось про клас, а способи — обов'язки дії, тобто виконання певних функцій. Оскільки базовим елементом ОО-систем є клас, тому метрики мають застосовуватися до класу чи об'єкта.

Інкапсуляція характеризує процес поділу конструкції об'єкта та його поведінки. Водночас структури даних та елементи реалізації методів об'єктів є невидимими для інших об'єктів у системі. Доступ до стану об'єкта здійснюється через його інтерфейсні методи чи інтерфейси. У цьому інтерфейси сприймаються як загальнодоступні зобов'язання. Деякі методи об'єкта можуть бути приховані за інтерфейсом — закриті методи. Метрики обліку інкапсуляції мають враховувати складність класу, з одного боку, здатність його взаємодії з іншими класами.

Наслідування — окремий випадок структурного взаємозв'язку між класами, в якому реалізуються ідеї спеціалізації та абстракції. Дочірній клас успадковує властивості та зобов'язання батьківського класу та передбачає його спеціалізацію шляхом додавання специфічних атрибутів та методів. Наслідування поширюється через усі рівні ієрархії класів. Метрики наслідування характеризують ефективності програмної системи щодо повторного використання класів і адаптації їх до умов застосування.

Поліморфізм дає змогу забезпечити абстрагування загальних властивостей інформаційної системи. Це реалізується шляхом перевизначення функціональності методів, що забезпечує спеціалізацію класів в ієрархії спадкування.

Широко відомими метриками для об'єктно-орієнтованих систем є метрики Чайдембера та Кемерера, які орієнтовані на класи [11]: вага методів класів; глибина дерева успадкування; кількість нащадків; зв'язування між об'єктами; відгук класу; низьке зчеплення методів.

Потрібно зазначити, що метрики Чайдембера та Кемерера, зорієнтовані на оцінку якості інструментальних програмних засобів та їх застосування до програмного забезпечення корпоративних інформаційних систем, потребують доопрацювання. Це пояснюється тим, що програмне забезпечення інформаційних систем розробляється на базі конкретних програмних платформ з використанням CASE-засобів швидкої розробки програмних систем, бібліотек класів і максимально використовується механізм повторного використання кодів та програмних рішень, що значною мірою впливає на якість програм, що розробляються для корпоративних інформаційних систем і не знайшло відображення у метриках Чайдембера та Кемерера.

Однією з основних переваг об'єктно-орієнтованого підходу до аналізу та проектування інформаційних систем є властивість повторного використання

рішень на різних фазах життєвого циклу інформаційної системи. Забезпечення якості ІС у цьому випадку реалізується у вигляді методик використання шаблонів. В об'єктно-орієнтованій технології проектування шаблоном називають опис проблеми та її вирішення, які можна застосувати для розробки інших систем. Шаплони можуть застосовуватися під час проектування та аналізу, розробки і на рівні коду програм.

Інтегральна оцінка якості інформаційної системи здійснюється на основі аналізу взаємовпливу різних параметрів якості продукту з використанням експертних оцінок. Інтегральний показник якості програмного забезпечення формується з використанням часткових показників з урахуванням його «ваги». «Вагу» показника визначають експерти. Для оцінювання якості характерна така послідовність дій.

1. Замовник разом з групою експертів призначають вагові коефіцієнти важливості:  $p_{ki}$  —  $i$ -го показника  $k$ -ої властивості і  $P_k$  —  $k$ -ої властивості.

Причому  $\forall k, i, p_{kj} \in [0, 1], \sum_{i=1}^{N_k} r_{ki} = 1; \forall k, P_k \in [0, 1], \sum_{k=1}^N P_k$ , де  $P_k$  — число показників  $k$ -ої властивості;  $N$  — число властивостей.

2. Група експертів визначає величину  $R_{ki} \in [0, 1]$  — експертну оцінку  $i$ -го показника  $k$ -ої властивості.

3. Обчислюється  $R_{qual\ k}$  — комплексна оцінка  $k$ -ої властивості за формулою:  $R_{qual\ k} = \sum_{i=0}^{N_k} R_{ki} P_{ki}$ , де  $R_{ki}$  — експертна оцінка  $i$ -го показника  $k$ -ої властивості;  $p_{ki}$  — експертний ваговий коефіцієнт важливості  $i$ -го показника  $k$ -ої властивості;  $N_k$  — число показників  $k$ -ої властивості.

4. Обчислюється  $R_{total}$  — інтегральний показник якості програмного об'єкта за такою формулою  $R_{total} = \sum_{k=1}^N R_{qual\ k} P_k$ , де  $P_k$  — експертний ваговий коефіцієнт важливості  $k$ -ої властивості;  $N$  — число властивостей.

Одним із суттєвих недоліків такого підходу є те, що експертам пропонується оцінити низку показників (коефіцієнти важливості та значення показника якості) у діапазоні  $[0, 1]$ , що є не простим завданням, що містить істотну невизначеність. Потрібно зазначити, що низка показників, які пропонують оцінити експерту, є за своєю природою та інтерпретація їх кількісної оцінки є вкрай невизначеною. За цих умов вдаються до лінгвістичної оцінки властивостей, що є природнішим для менеджерів та кінцевих користувачів з погляду споживчої якості програмних продуктів.

Важливим аспектом керування якістю програмного забезпечення є дотримання належних показників на всіх фазах життєвого циклу програмної системи. Отже, для забезпечення якості програмних продуктів необхідне застосування спеціальних методів та засобів підготовки високоякісних специфікацій, постійний контроль відповідності вимогам, повне тестування системи, впровадження стандартів та формальних процедур, контроль змін, вимірювання, аудит та реєстрація характеристик якості. Розглядаючи якість ПЗ як багатофакторний показник, який залежить від прикладної галузі та категорії користувачів, фактори, що визначають

якість ПЗ, поділяють на три групи. До першої групи належать робочі характеристики: коректність, надійність, ефективність, зручність та простота використання, цілісність. До другої групи — характеристики супроводу ПЗ: модифікованість, гнучкість, зручність та простота тестування. Третя група характеристик — можливості розвитку та інтеграції програмного забезпечення: переносимість, придатність до повторного використання та сумісність. Необхідно виявляти та систематично відстежувати зв'язок характеристик якості ПЗ з архітектурою системи, що дає змогу ухвалювати обґрунтовані рішення та прогнозувати якість системи.

Якщо розглядати питання оцінки якості інформаційної системи з погляду менеджменту підприємства, то для такого рівня ухвалення стратегічних рішень вкрай актуально мати єдиний критерій, який однозначно визначатиме допустимо чи неприпустимо якість інформаційної системи щодо забезпечення потреб бізнесу. Наявні методики аналізу якості інформаційних систем не дають повного кількісного опису рівня якості, крім того, вони не враховують семантики предметної галузі, операційного та інструментального оточення, а також методології розробки. Традиційні метрики не враховують якість даних та взаємодію даних з потоком обчислень.

**Висновки.** Аналіз наявних підходів та методологій забезпечення якості інформаційних систем показує, що для різних фаз життєвого циклу ІС необхідно використовувати моделі, побудовані відповідно до вимог основних зацікавлених осіб кожної фази розроблення з урахуванням технологічних особливостей реалізацій. Так під час проектування та розробки програмного забезпечення ІС необхідно враховувати погляди менеджерів, користувачів, розробників та замовників на якість програмного продукту, а також наявні технології проектування. Створюючи ІС на базі програмних систем, що тиражуються, на перший план виходить споживач системи і першорядне значення набуває споживча якість програмних продуктів. Вищевикладене визначає актуальність розвитку методології оцінки якості інформаційних систем для кожної фази її життєвого циклу.

#### СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. McCabe J. A complexity measure. *IEEE Trans. Software Engineering*. 1976. 2 (4). Pp. 308–320.
2. Characteristics of Software Quality / Boehm B. W., Brown J. R., Kaspar H., Lipow M., MacLeod G., Merritt M. J. North Holland, 1978.
3. Boehm B. W. Software Risk Management. *IEEE Computer Society Press*, CA, 1989.
4. ISO/IEC 9126-1: 2001. Software engineering – Software product quality - Part 1: Quality model. Geneva, Switzerland : International Organization for Standardization.
5. ISO/IEC DTR 9126-2. 2001. Software engineering – Software product quality - Part 2: External metrics. Geneva, Switzerland : International Organization for Standardization.
6. ISO/IEC DTR 9126-3. 2000. Software engineering – Software product quality - Part 3: Internal metric. Geneva, Switzerland : International Organization for Standardization.
7. ISO/IEC TR 9126-4: 2004. Software engineering – Product quality - Part 4: Quality in use metrics. Geneva, Switzerland : International Organization for Standardization.



8. ISO/IEC 25010:2011 Systems and software engineering — Systems and software Quality Requirements and Evaluation (SQuaRE) — System and software quality models.
9. ISO 8402:1994 Quality management and quality assurance - Vocabulary. ISO 9000:2000 Quality management systems — Fundamentals and vocabulary.
10. PMI Standards Committee, William R. Duncan, Director of Standards. A Guide to the Project Management Body of Knowledge. Nettoun Square, PA : Project Management Institute, 1996.
11. Chidamber S. R., Kemerer C. F. A metrics suite for object-oriented design. *IEEE Trans. Software Engineering*. 1994. 20. Pp. 476–493.

#### REFERENCES

1. McCabe, J. (1976). A complexity measure. *IEEE Trans. Software Engineering*, 2 (4), 308–320 (in English).
2. Boehm, B. W., Brown, J. R., Kaspar, H., Lipow, M., MacLeod, G., & Merritt, M. J. (1978). Characteristics of Software Quality. North Holland (in English).
3. Boehm, B. W. (1989). Software Risk Management. *IEEE Computer Society Press*, CA (in English).
4. ISO/IEC 9126-1: 2001. Software engineering – Software product quality - Part 1: Quality model. Geneva, Switzerland : International Organization for Standardization (in English).
5. ISO/IEC DTR 9126-2. 2001. Software engineering – Software product quality - Part 2: External metrics. Geneva, Switzerland : International Organization for Standardization (in English).
6. ISO/IEC DTR 9126-3. 2000. Software engineering – Software product quality - Part 3: Internal metric. Geneva, Switzerland : International Organization for Standardization (in English).
7. ISO/IEC TR 9126-4: 2004. Software engineering – Product quality - Part 4: Quality in use metrics. Geneva, Switzerland : International Organization for Standardization (in English).
8. ISO/IEC 25010:2011 Systems and software engineering — Systems and software Quality Requirements and Evaluation (SQuaRE) — System and software quality models (in English).
9. ISO 8402:1994 Quality management and quality assurance - Vocabulary. ISO 9000:2000 Quality management systems — Fundamentals and vocabulary (in English).
10. PMI Standards Committee, William R. Duncan, Director of Standards. A Guide to the Project Management Body of Knowledge. Nettoun Square, PA : Project Management Institute, 1996 (in English).
11. Chidamber, S. R., & Kemerer, C. F. (1994). A metrics suite for object-oriented design. *IEEE Trans. Software Engineering*, 20, 476–493 (in English).

doi: 10.32403/1998-6912-2022-2-65-22-31

### STATUS AND PROBLEMS OF QUALITY ASSESSMENT OF INFORMATION SYSTEMS

I. V. Hileta, Yu. Yu. Bilas

Ukrainian Academy of Printing,  
19, Pid Holoskom St., Lviv, 79020, Ukraine  
hileta@gmail.com

*The concept of quality of an information system is analyzed. It is established that the perception and interpretation of quality depends on the subject of consideration. The quality assessment is based on the quality model. The quality model of McCall and Boehm is analyzed. The ISO/IEC 9126 standard defines goals, attributes, and metrics for assessing the quality of information system software. The quality model, which is created within the framework of the ISO/IEC 9126 standard, is determined by general characteristics. The information system quality characteristics and attributes are divided into three groups: categorical, quantitative and qualitative. Problems of aggregation of quality characteristics based on the interconnection scheme. This necessitates the development of a methodology for integrated quality assessment of information system software. The IEEE 1061 standard describes a hierarchical methodology. An important component in the process of ensuring the quality of information systems software is quality management in design, in particular, in object-oriented design. The following characteristics have a significant impact on the quality of object-oriented information systems: methods of object abstraction, localization, encapsulation, inheritance, and polymorphism. The well-known metrics for object-oriented information systems are the Chaidembert and Kemerer metrics. An approach to assessing software quality using expert opinions is considered. An integral indicator of the quality of information systems is formed taking into account partial indicators and taking into account weights. An important aspect of software quality management is its compliance with proper indicators at all phases of the software system life cycle. The issue of assessing the quality of information systems from a management perspective is considered. It is extremely important to have a single criterion that will unambiguously determine the quality of information systems to ensure business performance.*

**Keywords:** *information technology, software product, program code quality, quality indicators, quality standards.*

*Стаття надійшла до редакції 26.05.2022.*

*Received 26.05.2022.*