УДК 004.891.3: 004.3

АНАЛІЗ ІНЦИДЕНТІВ, СПРИЧИНЕНИХ ПОМИЛКАМИ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

Т.О.ГОВОРУЩЕНКО, Р.А.МАЛЯРЧУК, І.В.ВОВК
Хмельницький національний університет

У даній статті автори проаналізували причини та наслідки катастроф та інцидентів, пов'язаних із помилками та збоями програмного забезпечення на різних етапах життєвого циклу, як результатів прояву рівня якості програмного забезпечення.

In this article, the authors analyzed the causes and consequences of disasters and accidents, caused by the software errors and failures at the different stages of software lifecycle, as the results of software quality level.

Ключові слова: помилки програмного забезпечення (ПЗ), етапи життєвого циклу (ЖЦ) ПЗ, збої ПЗ, катастрофи через помилки та збої у ПЗ, якість ПЗ.

Вступ. Найважливішою характеристикою ПЗ з точки зору розробника є складність ПЗ, а з точки зору користувача - його якість. Криза у галузі забезпечення якості ПЗ проявилась ще 50 років тому, але при наявності ряду методів та засобів, залученні фахівців для розроблення технологій та стандартів забезпечення якості, якість ПЗ, як і раніше, залежить від знань та досвіду розробників.

Наразі в більшості проектів основна частина зусиль з виправлення дефектів все ще виконується на етапі тестування, але політика раннього виявлення та виправлення дефектів може в декілька разів знизити фінансові та часові витрати на розроблення ПЗ.

Постійне зростання складності функцій ПЗ неминуче призводить до збільшення його обсягу та трудомісткості створення. Сучасне ПЗ обсягом в мільйони рядків коду в принципі не може бути безпомилковим. Проблема полягає в тому, щоб виявити рівень якості проекту та спрогнозуватри якість розроблюваного ПЗ з врахуванням того, що деяка невідома кількість помилок та дефектів завжди залишається в складних програмних комплексах. Отже, *актуальною* є задача прогнозування рівня якості розроблюваного ПЗ на ранніх етапах життєвого циклу (ЖЦ).

Серед причин можливих невдач програмних проектів - нечітке й неповне формулювання вимог до ПЗ; невірне розуміння або недостатній аналіз специфікації проекту; незадовільне проектування; недостатнє залучення користувачів до роботи над проектом; відхилення від специфікації під час проектування або програмування; новизна використовуваних технологій; неповнота або відсутність тестування; ігнорування неочікуваних результатів під час експлуатації ПЗ. Отже, очевидно, що причини невдач, інцидентів та катастроф - помилки у ПЗ - вносяться на різних етапах життєвого циклу ПЗ.

Постановка задачі. Для подальшого вирішення задачі прогнозування якості програмного забезпечення на етапі проектування слід проаналізувати причини та наслідки відомих випадків прояву низького рівня якості програмного забезпечення різних типів, обумовлених помилками на різних етапах життєвого циклу ПЗ.

1. Причини і наслідки інцидентів, викликаних помилками у системному програмному забезпеченні. Аналіз помилок програмного забезпечення, внесених на різних етапах ЖЦ ПЗ, які стали причинами інцидентів у системному ПЗ, відображено на рис.1.

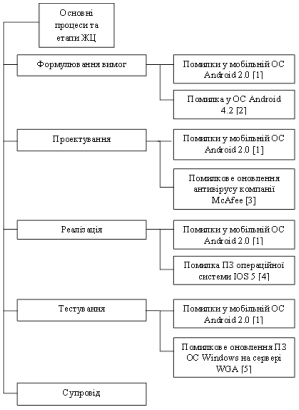

Рис.1. Помилки у системному ПЗ на різних етапах ЖЦ

Наслідками інцидентів, спричинених вищенаведеними помилками, стали:

1) фінансові збитки та втрата репутації компаніями Apple [4], Google [2], McAfee [3], Microsoft [5];
2) серйозна загроза для безпеки мобільної ОС Android 2.0 через 359 програмних помилок різного ступеня ризику [1].

2. Причини і наслідки катастроф та інцидентів, викликаних помилками у вбудованому програмному забезпеченні. Проаналізуємо, на яких етапах ЖЦ були внесені помилки у вбудоване ПЗ, які призвели до катастроф та інцидентів, а також які наслідки мали ці катастрофи та інциденти. Такий аналіз наведено у таблиці 1.

Таблиця 1. Аналіз часу внесення помилок у вбудоване ПЗ та їхніх наслідків

№ п/п	Подія	Причина	Наслідки
Етап формулювання вимог			
1.	У травні 1971 року радянський "Космос-419" не виконав запланований старт до Марсу [6]	Невірність специфікації	Втрата апарату
2.	У травні 1971 року бортова система радянської станції "Марс 2" не змогла відстикуватись від головного корабля [6]	Невірність специфікації	Невиконання завдання
3.	Аварія англійського військового гелікоптера Chinook у 1994 році [6]	Невірність специфікації	Загинуло 29 чоловік
4.	"Смертельні" сеанси радіаційної терапії із застосуванням Therac-25 у 1985-1987 роках в США та Канаді [6-8]	Неповнота специфікації	6 пацієнтів одержали смертельну дозу опромінення
5.	Вибух ракети-носія Ariane 5 у1996 році [6-8]	Протиріччя вимог щодо необхідності забезпечення надійності та максимально допустимого навантаження	Вартість наукового обладнання та витрати на розроблення - 7,5 мільярдів доларів, "упущена вигода" – 2 мільярди доларів

Етап проектування			
6.	"Смертельні" сеанси радіаційної терапії із застосуванням Therac-25 [6-8]	Помилки у розробленні та постановці проекту; некоректні процедури оцінки та прогнозування ризиків	6 пацієнтів одержали смертельну дозу опромінення
7.	Помилка процесора Intel Pentium у 1994 році [7]	Відхилення від специфікації	Втрати компанії – 475 мільйонів доларів
8.	Аварія автоматичної міжпланетної станції Mars Climate Orbiter та зникнення зв'язку із Mars Polar Lander у 1999 році [6, 8]	Помилка в проекті через використання різних одиниць вимірювання	327,6 мільйонів доларів - апарати та 91,7 мільйонів доларів - запуски
9.	Аварійне падіння другого ступеня ракетоносія "Рокот" у 2005 році [9]	Логічна помилка в алгоритмі системи керування	Втрата європейського наукового супутнику CryoSat
10.	Аварія аеробуса A330 у жовтні 2008 року [10]	Помилка у системному алгоритмі обробки даних	110 пасажирів та 9 членів екіпажу були поранені
11.	Падіння російської ракети-носія "Зенит-3SL" у 2013 році [11]	Невірний проект системи керування	Втрата супутника "Intelsat-27"
Етап реалізації			
12.	Невдалий запуск першого американського супутника до Венери “Mariner1” у липні 1962 р. [7]	Помилка у програмі – у операторі циклу замість коми програміст поставив крапку	Втрата 135 мільйонів доларів
13.	Вибух ракети Ariane 5 у1996 році [6-8]	Помилка "Operand Error"	Втрата 9,5 мільярдів доларів

14.	"Смертельні" сеанси радіаційної терапії із застосуванням Therac-25 [6-8]	Помилки типу "race condition"; невірна реалізація інтерфейсу	6 пацієнтів одержали смертельну дозу опромінення
15.	Катастрофа літака "Боїнг-757" у Колумбії в грудні 1995 року [12]	Помилка в одному символі програмної системи керування польотом	Загинуло 159 людей
16.	Порушення польоту ракетоносія Titan IV у 1999 році [13]	Помилка в константі ПЗ системи керування двигуном	Втрата супутника Milstar
17.	Аварія ракети-носія "Зеніт" [6]	Програмна помилка	Невиведення на орбіту супутника для системи стільникового телефонного зв'язку
18.	2 аварії конверто-планів Bell V-22 у 2000 році [14]	Програмні помилки	Загинуло 23 чоловіки
19.	Припинення функціонування апарату Mars Global Surveyor у грудні 2006 року [6]	Помилка адресації бортового ПЗ	Втрата апарату та результатів його роботи
20.	Відмова бортових комп'ютерів на 12 винищувачах-"невидимках" F-22 у 2007 році [6]	Програмна помилка	Повернення винищувачів на військову базу у США
21.	Падіння трьох супутників ГЛОНАСС у грудні 2010 року [6]	Помилка у програмі	Втрата 138 мільйонів доларів

22.	Згоряння російської автоматичної міжпланетної станції "Фобос-Грунт" у січні 2011 року [15]	Помилка програмування	Втрати: 1,2 мільярда рублів - вартість станції та 4 мільярди рублів - роботи по проекту
23.	Неробототздатність чотириядерних процесори AMD Phenom і Opteron у 2007 році [12]	Помилка реалізації буферу перетворення адреси кеш-пам'яті	Постійні зависання системи, втрата репутації компанії
24.	У 2005 році на автомобілях Toyota невірно працювали бензинові двигуни та освітлення [7]	Помилка програмування	Відзив 160000 автомобілей Prius і втрата 3 мільярдів доларів
Етап тестування			
25.	Помилка процесора Intel Pentium у 1994 році [7]	Неповнота тестування	Втрати компанії – 475 мільйонів доларів
26.	"Космос-419" не виконав старт до Марсу [6]	Неповнота тестування	Втрата апарату
27.	Аварія Mars Climate Orbiter та зникнення зв'язку із Mars Polar Lander [6, 8]	Неповнота тестування	Втрата 419,3 мільйонів доларів
28.	Вибух ракети Ariane 5 у1996 році [6-8]	Недостатність тестування	Втрата 9,5 мільярдів доларів
Етап супроводу та експлуатації			
29.	"Смертельні" сеанси радіаційної терапії із застосуванням Therac-25 [6-8]	Ігнорування неочікуваних результатів та нерозуміння підказок системи	6 пацієнтів одержали смертельну дозу опромінення
30.	Аварія Mars Climate Orbiter та зникнення зв'язку із Mars Polar Lander [6, 8]	Ігнорування неочікуваних результатів	Втрата 419,3 мільйонів доларів

До помилок ПЗ із катастрофічними наслідками також належать наступні помилки у *вбудованому ПЗ*: 1) відмова системи енергопостачання супутника "Коронас-Фотон" внаслідок помилкового проекту та неповноти тестування, яка призвела до втрати зв'язку із ним (2009 рік); 2) помилкова реалізація ПЗ гальмівної системи японського зонду "Акацукі", що призвела до неможливості виводу його на орбіту Венери (2010 рік); 3) помилковий фрагмент коду мережевої карти мережі прикордонної та митної служби аеропорту Лос-Анджелесу, що призвів до паралічу прикордонної та митної служби (2007 рік).

Отже, з результатів таблиці 1 очевидно, що помилки ПЗ, які спричинили різноманітні катастрофи та інциденти, були внесені на різних етапах ЖЦ, але велика кількість помилок була внесена на ранніх його етапах. Забезпечення можливості раннього виявлення таких помилок, прогнозування їх впливу на виникнення помилок на наступних етапах ЖЦ, а також оцінювання якості проекту та прогнозування рівня якості розроблюваного за проектом ПЗ в кінці етапу проектування дали б можливість уникнути ряду катастроф та інцидентів (Космос-419, Марс 2, Chinook, Therac-25, Ariane 5, Intel Pentium, Mars Climate Orbiter, A330, Рокот, Зенит 3SL).

3. Причини і наслідки катастроф та інцидентів, викликаних помилками у прикладному ПЗ. Проаналізуємо тепер, на яких етапах життєвого циклу виникали помилки, які призвели до катастроф та інцидентів, пов'язаних із використанням прикладного ПЗ, та які наслідки мали ці катастрофи та інциденти таблиця 2.

Таблиця 2. Аналіз часу внесення помилок у прикладне ПЗ та їхніх наслідків

№ п/п	Подія	Причина	Наслідки
Етап формулювання вимог			
1.	Збій у системі Нью-Йоркського банку [16]	Недостатність пам'яті через невірні вимоги	32 мільярди доларів
2.	У січні 1990 року в телефонній компанії AT&T відбулась 9-годинна аварія [7, 8]	Проблеми з граничними умовами у специфікації	75 млн. нездійснених дзвінків, втрата 60 мільйонів доларів

3.	У квітні 1998 року на AT&T відбулась 26-годинна аварія в мережі ретрансляції кадрів [7, 8]	Проблеми з прихованими граничними умовами у специфікації	Непрацездатність служб, пов'язаних з передачею даних
4.	Помилка Y2K - помилка двоцифрового збереження року дати (1999 рік) [7, 8]	Невірність або неповнота специфікації	Втрати 500 мільярдів доларів
Етап проектування			
5.	У вересні 1983 року на станції раннього виявлення "Серпухов-15" спрацювала система виявлення [8]	Невірно спроектована система розпізнавання	Світ був на межі ядерної війни
6.	У лютому 1991 року мобільна система протиракетної оборони Patriot не перехопила іракську ракету Scud [6-8]	Помилка заокруглення - некоректний розрахунок місцезнаходження ракети, що наближалась	Загинули 28 американських солдат та близько 100 чоловік одержали поранення
7.	Помилка Y2K - помилка двоцифрового збереження року дати (1999 рік) [7, 8]	Невірний проект	Втрати 500 мільярдів доларів
8.	"Смертельна" терапія у Національному онкологічному інституті у Панама-Сіті в 2001 році [7]	Некоректне обчисення доз радіації у ПЗ компанії Multidata Systems International	28 онкохворих пацієнтів зазнали надмірного опромінення, декілька хворих померли
9.	Аварія на заводі по переробці урану у Західній Австралії в грудні 2001 року [6]	Логічна помилка у алгоритмі	Викид радіоактивної речовини

Етап реалізації			
10.	У листопаді 1988 року 10% Інтернету було інфіковано worm-вірусом Морріса [7, 8]	Помилки у програмі worm-віруса	Вихід з ладу близько 6000 комп'ютерів
11.	Відсутність електропостачання в частині США та Канади у серпні 2003 року [17]	Помилка "race condition" у системі управління аварійною сигналізацією	Збиток склав від 7 до 10 мільярдів доларів
12.	Некоректна робота Skype у листопаді 2012 року [18]	Дефект у функції скидання паролю	Несанкціонований доступ до чужого аккаунта
13.	Невірна робота біржевих "роботів" компаній ММВБ-РТС і Knight Capital Group у 2012 році [19, 20]	Помилка у ПЗ "робота"	Збитки 4,3 мільйона доларів + 440 мільйонів доларів
14.	Пожежа на космічному шаттлі Columbia у 2003 році [6-8]	Невірно складений звіт про пошкодження покриття крил	Загибель 7 астронавтів та втрата шаттлу
Етап тестування			
15.	Аварія в мережі ретрансляції кадрів компанії AT&T [7, 8]	Недостатність тестування	Непрацездатність служб, пов’язаних з передачею даних
Етап супроводу та експлуатації			
16.	Вразливість ПЗ соціальної мережі Facebook у листопаді 2012 року [21]	Парсінг пошукових запитів з боку серверного ПЗ соцмережі	Доступ до 1,32 мільйонів аккаунтів користувача без введення паролю
17.	Обвал даху Хардфордської спортивної арени у січні 1978 року [8]	Помилка використання CAD-програми	70 мільйонів доларів та 20 мільйонів доларів упущеної вигоди

Варто розглянути також інші помилки у *прикладному ПЗ*, які призвели до катастрофічних наслідків: 1) невірний розрахунок навантаження на ПЗ на етапі формулювання вимог призвів до падіння рейтингу та втрати 500 мільярдів доларів компанією Dow Jones Industrial Average на біржевих торгах (1987 р.) [8]; 2) невірно спроектована система розпізнавання Patriot призвела до помилкової ідентифікації британського бомбардувальника Tornado як ворожої ракети (2003 р.) - загинули 2 пілоти [7, 8]; 3) 500 незворотніх помилок програмування у системі для Агентства підтримки дітей Великобританії призвели до втрати більше 1 мільярда доларів (2004 р.); 4) впровадження системи ЕГАИС з помилками програмування та недостатнім тестуванням призвело до паралічу гуртових поставок алкоголю в Росії (2006 р.); 5) некоректна експлуатація системи SCADA призвела до відсутності повідомлення про прорив трубопровода (1999 р.) - 3 людських життя та 45 мільйонів доларів [7, 8].

Аналіз таблиці 2 показує, що помилки прикладного ПЗ, які спричинили катастрофи та інциденти, були внесені на різних етапах життєвого циклу, але більшість помилок була внесена на ранніх етапах. Тобто, як і у випадку з вбудованим ПЗ, забезпечення можливості раннього виявлення таких помилок та прогнозування рівня якості прикладного ПЗ на етапі проектування також дали б можливість уникнути ряду катастроф та інцидентів, причини яких були внесені на етапах формулювання вимог та проектування.

Висновки. Виконаний аналіз показує, що, якщо в індустрії розроблення та оцінювання якості програм не відбудеться кардинальних змін, то світ і надалі очікуватимуть техногенні катастрофи та інциденти, викликані помилками в коді або збоями при управлінні складними програмними системами.

Аналіз відомих випадків прояву низького рівня якості, спричинених помилками у різнотипному ПЗ, дозволив авторам виявити ряд спільних причин виникнення помилок на різних етапах ЖЦ ПЗ: 1) невірність або неповнота специфікації (етап формулювання вимог); 2) невірно розроблений проект та відхилення від специфікації (етап проектування); 3) недостатнє розуміння інструментів та конструкцій мови - помилки програмування (етап реалізації); 4) неповнота тестування (етап тестування); 5) ігнорування неочікуваних результатів та нерозуміння підказок системи (етап експлуатації).

Аналіз наслідків катастроф та інцидентів, спричинених помилками у ПЗ різних типів показав, що найбільш критичними та катастрофічними є помилки у вбудованому ПЗ.

У статті проілюстровано, що деяких помилок можна було уникнути при ранньому оцінюванні проекту та прогнозуванні якості розроблюваного програмного забезпечення. Отже, оцінювати якість проекту, а також прогнозувати якість розроблюваного за проектом ПЗ необхідно на ранніх етапах життєвого циклу програмного забезпечення, бажано в кінці етапу проектування. Тоді перспективним напрямком подальших досліджень є розроблення методу прогнозування якості програмного забезпечення на основі опрацювання результатів статичного та метричного аналізу.

Використані джерела:

1. Study: 359 Android code flaws pose security risks // http://news.cnet.com/8301-30685_3-20021437-264.html?part=rss&subj=news&tag=2547-1_3-0-20
2. Google promises fix for Android 4.2 December birthday bug // http://www.androidcentral.com/google-s-promises-fix-android-42-december-birthday-bug
3. McAfee DAT 5958 Update Issues // http://isc.sans.edu/diary/McAfee+DAT+5958+Update+Issues/8656
4. How to fix battery life issues with iOS 6 or iPhone 5 // http://www.imore.com/how-fix-battery-life-problems-ios-6-or-iphone-5
5. MS server error marks PCs as ‘nongenuine’ // https://windowssecrets.com/patch-watch/ms-server-error-marks-pcs-as-nongenuine/
6. Цена программной ошибки // http://www.cusoft.ru/error.php
7. Software goes wrong, we all know that, but just how wrong can it go? // http://www.datareservoir.co.uk/bugs/
8. 20 Famous Software Disasters // http://sandipsandilya.wordpress.com/2011/01/17/20-famous-software-disasters/
9. Software glitch blamed for CryoSat loss // http://www.theregister.co.uk/2005/10/27/cryosat_downed_by_software/
10. Qantas terror blamed on computer // http://www.stuff.co.nz/travel/australia/6163633/Qantas-terror-blamed-on-computer

11. Падение ракеты-носитель "Зенит-3SL" // http://ukrday.com/politika/novosti.php?id=57865
12. Ю.Карпов. MODEL CHECKING: Верификация параллельных и распределенных программных систем – СПб: БХВ-Петербург, 2010 – 560 с.
13. Nancy G. Leveson. Systemic Factors in Software-Related Spacecraft Accidents // http://sunnyday.mit.edu/accidents/space2001.pdf
14. 19 Killed in Marine Plane Crash // http://www.helis.com/news/2000/v22crash.htm
15. Системный сбой российского космоса // http://ria.ru/analytics/20120201/554086364.html
16. Explaining Settlement Fails // http://www.ny.frb.org/research/current_issues/ci11-9/ci11-9.html
17. US–Canada Power System Outage Task Force, Final Report on the August 14, 2003 Blackout in the United States and Canada: Causes and Recommendations, tech. report, US Dept. of Energy, Apr. 2004
18. В связи с обнаруженной уязвимостью, позволяющей взломать чужой аккаунт, Skype отключил функцию восстановления паролей // http://zn.ua/TECHNOLOGIES/v_svyazi_s_obnaruzhennoy_uyazvimostyu,_pozvolyayuschey_vzlomat_chuzhoy_akkaunt,_skype_otklyuchil_fun.html
19. Торговый робот на бирже FORTS обокрал хозяина на $4 000 000 // http://deipara.com/torgovyj-robot-na-forts-obokral-xozyaina-na-4-000-000
20. Robot stock traders lose $440,000,000 in 45 minutes, need someone to spell it out // http://www.engadget.com/2012/08/03/rogue-automatic-trading/
21. В коде Facebook обнаружена уязвимость, раскрывающая данные пользователей // http://zn.ua/TECHNOLOGIES/v_kode_facebook_obnaruzhena_uyazvimost,_raskryvayuschaya_dannye_polzovateley.html