

УДК 004.415.2:519.876.5

О.П. Доренський

Кіровоградський національний технічний університет, Кіровоград

МЕТОД СИНТЕЗУ ТЕСТОВИХ МОДЕЛЕЙ ПОВЕДІНКИ ПРОГРАМНИХ ОБ'ЄКТІВ ІНФОРМАЦІЙНО-ТЕЛЕКОМУНІКАЦІЙНОЇ СИСТЕМИ СПЕЦІАЛЬНОГО ПРИЗНАЧЕННЯ

Під час розроблення складних програмних систем, якою є інтегрована інформаційно-телекомунікаційна система спеціального призначення, що розробляється, ще на етапі проектування виникають до двох третин помилок. З метою їх виявлення у роботі запропоновано метод синтезу тестових моделей поведінки програмних об'єктів, який полягає в поетапному виконанні побудови моделей послідовностей повідомлень програмних об'єктів, синтезу логічних і тестових моделей станів та діяльностей класів, перетворення моделей послідовностей у моделі кооперації програмних об'єктів. Застосування синтезованих логічних та тестових структур дає можливість повно тестувати динамічну компоненту комплексної моделі об'єктно-орієнтованого програмного забезпечення та, як результат, виявляти алгоритмічні і системні помилки на етапі проектування інформаційно-телекомунікаційної системи спеціального призначення.

Ключові слова: програмне забезпечення, інформаційно-телекомунікаційна система, проектування, програмний об'єкт, поведінка, тестова модель.

Вступ

В рамках реформування системи управління Державної прикордонної служби України створюються електронна система управління, автоматизовані інформаційні системи та здійснюється їх інтегрування в єдину інформаційно-телекомунікаційну систему (ІТС) Держприкордонслужби, інших правоохоронних органів України [1]. Реалії сьогодення [2] гостро поставили необхідність створення ІТС спеціального призначення в інтересах Державної прикордонної служби України. Її основними функціями є збирання, оброблення й передавання інформації. У той же час, специфіка діяльності Держприкордонслужби визначає потребу не закупівлі або модернізації існуючого програмного забезпечення (ПЗ), а розроблення й впровадження унікального ПЗ на основі сучасних технологій створення та супроводу складних програмних систем.

Є очевидним, що єдина ІТС Державної прикордонної служби України – багатокомпонентна складна програмна система. Особливостями її складових (підсистем) є обмін надвеликими об'ємами даних, забезпечення високої надійності, функціонування у реальному масштабі часу, забезпечення оперативності оброблення, приймання (збирання) та передачі інформації каналами зв'язку, циклічність функціонування тощо. Тож ПЗ повинне відповідати ряду жорстких вимог щодо високопродуктивності, гнучкості, надійності, а також забезпечення можливості вдосконалення, масштабування, нарощування функціональності й швидкості розроблення і впровадження за якнайменших витрат коштів та часу.

Комплексного врахування і реалізація зазначених особливостей, вимог та характеристик програмного забезпечення, яке створюється, можливо досяг-

ти застосуванням об'єктно-орієнтованої технології (ООТ) проектування й розроблення ПЗ [3].

Істотна надлишковість об'єктно-орієнтованого ПЗ (ООПЗ) ІТС спеціального призначення в інтересах Держприкордонслужби України, великі об'єми даних, які циркулюють між елементами ПЗ, визначають суттєве ускладнення структур програмних об'єктів (ПО) та їх взаємодії у програмній системі. Це може призвести до істотного збільшення кількості внесених і залишкових помилок у ООПЗ, першочергово: системних й алгоритмічних. Внаслідок цього істотно знижується надійність ПЗ, збільшується час тестових перевірок, що призводить до збільшення тривалості розроблення, впровадження та супроводу ІТС спеціального призначення.

Одним з напрямів скорочення кількості помилок в ООПЗ є забезпечення належного значення повноти його тестування. Проте здійснити це з погляду виявлення алгоритмічних та системних помилок вручну або з застосуванням традиційних методів не вдається [4]. Тож, задля їх виявлення слід здійснювати тестування проектних моделей ПЗ і, головне, моделей поведінки програмних об'єктів об'єктно-орієнтованого програмного забезпечення ІТС спеціального призначення. Вони описують взаємодію програмних об'єктів з погляду функціонування програмної системи, яка розробляється на основі ООТ. При цьому, трасування тестів доцільно проводити не на проектних моделях, які по суті є статичними, а на відповідних їм тестових моделях.

Аналіз останніх досліджень і публікацій показав, що у дослідженнях [5 – 7] започатковано розв'язання означеної проблеми. Вказані праці є підґрунтям для створення тестових моделей поведінки програмних об'єктів (ПО) складних систем, які

розробляються на основі ООТ. У роботі [5] запропоновано метод синтезу моделей станів ПО об'єктно-орієнтованого ПЗ. Процес побудови цієї моделі, виходячи з її скінченно-автоматного представлення, розглянуто з погляду абстрактного синтезу скінченного автомата. У дослідженні викладено й обґрунтовано особливості синтезу скінченно-автоматної моделі станів ПО ООПЗ, побудову відображення, яке визначає множину каналів керування класу об'єктів, порядок приведення його до автоматного виду, а також побудову канонічної множини подій та їх регулярних виразів для відображення, яке визначає множину каналів керування класу програмних об'єктів. На основі отриманих результатів запропоновано метод синтезу моделей станів ПО ООПЗ. Запропонований метод забезпечує формалізацію процесу визначення станів та їх взаємозв'язків у життєвому циклі екземпляра класу програмної системи, а також дозволяє зменшити трудомісткість процесу розроблення динамічної компоненти комплексної моделі об'єктно-орієнтованого ПЗ під час проектування на логічному рівні.

Розробленню метода синтезу тестових структур взаємодії програмних об'єктів ООПЗ присвячено дослідження [6]. На основі положень об'єктно-орієнтованої методології, імітаційного моделювання та теорії мереж Петрі запропоновано процедуру формалізованого представлення моделі кооперації ПО у термінах формального апарату Е-мереж, алгоритм функціонування переходу тестової моделі взаємодії програмних об'єктів та процедуру реєстрації результатів функціонування тестової моделі.

Постановка задачі. Синтез і трасування тестових моделей поведінки програмних об'єктів програмного забезпечення ІТС спеціального призначення на стадії його проектування надає можливість моделювання процесу взаємодії ПО в динаміці їх використання. Як показав аналіз останніх досліджень і публікацій, методи синтезу моделей станів програмних об'єктів та синтезу тестових структур взаємодії ПО об'єктно-орієнтованого програмного забезпечення є основою для тестування взаємодії програмних об'єктів з метою виявлення алгоритмічних й системних помилок, а також отримання показників якості, які характеризують такі властивості складних програмних систем як безвідмовність, функціональна повнота тощо.

Таким чином, **метою роботи** є розроблення метода синтезу тестових моделей поведінки програмних об'єктів складних програмних систем, які проектуються й розробляються на основі об'єктно-орієнтованої технології. Розв'язок поставленої задачі вбачається у поетапному виконанні: формалізації концептуальних одиниць об'єктно-орієнтованої мови моделювання задля подальшого використання математичних методів з метою автоматизації процесів побудови й перетво-

рення моделей поведінки ПО ООПЗ; послідовного перетворення об'єктно-орієнтованих моделей поведінки програмних об'єктів задля отримання тестових структур взаємодії ПО; комплексного об'єднання можливостей моделювання й інструментальних засобів підтримки розроблення і тестування моделей поведінки ПО об'єктно-орієнтованого ПЗ на єдиній технологічній основі.

Основна частина

Задля представлення первинних проектних рішень щодо формалізації поведінки ПО в ООТ проектування використовують моделі послідовностей повідомлень програмних об'єктів (МППО). Результат їх побудови сформулюємо як бієктивне відображення $F_{MmSsu} : MM \rightarrow \sigma(SSU)$, де MM (від англ. model of messages) є множиною всіх моделей послідовностей повідомлень ПО, SSU (від англ. set of scripts of use cases) – множина сценаріїв, $\sigma(SSU)$ є множиною всіх підмножин SSU і кожна підмножина $SSU_{su} \in \sigma(SSU) (su \in SU, F_{Su\sigma(SSU)}(su) = SSU_{su})$. При цьому, $\forall MM_{su} \in MM \exists su \in SU (F_{Su\sigma(SSU)}(su) = SSU_{su}, F_{MmSsu}(MM_{su}) = SSU_{su})$. Цю модель задаватимемо:

– множиною об'єктів $SO_{su} = \{so_1, so_2, \dots, so_n\}$, де $SO_{su} \subset SO$; n – потужність множини SO_{su} ,

$\forall so_i \in SO_{su} \exists sc_j = F_{SoSc}(so_i) (sc_j \in SC_{su}, SC_{su} \subset SC$,

$SC_{su} = \{sc_1, sc_2, \dots, sc_m\}$ визначає класи, які реалізують прецедент $su \in SU$, m – потужність множини SC_{su} ; $i = 1, \dots, n$; $j = 1, \dots, m$;

– множиною акторів $ST_{su} \subseteq ST$, де множина $ST_{su} = ST_{su}^D \cup ST_{su}^V$; $ST_{su}^D \subseteq ST^D$, $ST_{su}^V \subseteq ST^V$; ST^D – множина операторів; ST^V – множина фізичних об'єктів;

– множиною екземплярів подій SL_{su} в рамках прецедента su , де $SL_{su} \subseteq SL_{sol} \cup \dots \cup SL_{son}$;

– множиною діяльностей SH_{su} в рамках прецедента su , де множина $SH_{su} \subseteq SH_{scl} \cup \dots \cup SH_{scm}$;

– функцією породження $F_{SotSl} : (SO_{su} \cup ST_{su}) \times SL_{su} \rightarrow \{0, 1\}$; якщо $F_{SotSl}(sot, sl) = 1$, то екземпляр події $sl \in SL_{su}$ породжується ПО або зовнішньою сутністю sot , $sot \in (SO_{su} \cup ST_{su})$.

Візуалізація, специфікування та документування моделі послідовностей повідомлень ПО ООПЗ можливо здійснити за допомогою відповідної діаграми (множини діаграм) [4, 8]. Графічно їх представляють таблицею, об'єкти в яких розташовуються

уздовж осі X, а повідомлення (екземпляри подій у порядку зростання часу) – вздовж осі Y.

Процес побудови МППО ООПЗ із множини MM, де $\forall MM_{su} \in MM \exists su \in SU (F_{SU\sigma(Ssu)}(su) = SSU_{su}, F_{MmSsu}(MM_{su}) = SSU_{su})$, розглядається як наступна сукупність дій:

– у процесі декомпозиції предметної області ООПЗ з множини $SO_{su} = \{so_1, so_2, \dots, so_n\}$, де $SO_{su} \subset SO$, $F_{SoSe} : SO \rightarrow SES$, для кожного сценарію визначається відповідна підмножина об'єктів;

– з множини $ST_{su} \subseteq ST$, де $ST_{su} = ST_{su}^D \cup ST_{su}^V$, здійснюється визначення акторів, які беруть участь у сценарії;

– з множини SL_{su} , де $SL_{su} \subseteq SL_{so_1} \cup SL_{so_2} \cup \dots \cup SL_{so_n}$, $F_{SISI} : SL \rightarrow SI$ ідентифікується множина повідомлень у вигляді відповідних екземплярів подій (для кожного екземпляра події визначається ПО, який відповідає за його обробку);

– з множини SH_{su} , де $SH_{su} \subseteq SH_{sc_1} \cup SH_{sc_2} \cup \dots \cup SH_{sc_m}$, визначається відповідна діяльність ПО у відповідь на екземпляр події (кожне отримане повідомлення);

– утворюється підмножина класів прецедентів множини SC (словник класів) шляхом ідентифікації класів за допомогою одного з основних методів класифікації відповідно до формул (1), (2), (3) [9]:

$$F_{Sc\sigma(So)} : SC \rightarrow \sigma(SO), \quad (1)$$

де $\sigma(SO)$ є множиною всіх підмножин CO, SC (від англ. set of classes) – множина класів ООПЗ;

$$F_{SoSc} : SO \rightarrow SC; \quad (2)$$

$$\begin{aligned} [F_{SoSc}(so) = sc] \Leftrightarrow & \left[\left(\exists \zeta \subseteq SP_E \left(\forall sp \in \zeta (F_{SpSap}(sp) = sap, \right. \right. \right. \\ & \left. \left. \left. F_{SesSp}(ses, sp) = 1, F_{SoSes}(so) = ses \right) sap \in \gamma (\gamma \subset SAP) \right) \right] \& \\ & \left(\forall so_i, so_j \in SO (F_{SoSc}(so_i) = sc, F_{SoSc}(so_j) = sc) \exists sfo \in SFO, \right. \\ & \left. (F_{SoSfo}(so_i) = sfo \ F_{SoSfo}(so_j) = sfo) \right). \quad (3) \end{aligned}$$

З використанням отриманих результатів здійснюється наступний крок: будуються логічні моделі індивідуальної поведінки об'єктів ООПЗ та відповідні їм тестові моделі за допомогою метода синтезу станів ПО об'єктно-орієнтованого ПЗ [5]. Після цього слід розробити модель діяльності класу ПО (МДКПО) з метою представлення операцій класів програмних об'єктів ООПЗ, які виконуються у відповідних станах моделей станів екземплярів класу ПО. Ця процедура здійснюється в рамках заданої множини дій для кожного класу об'єктів

$SHA_{sc}^{sh} \in \sigma(SHA_{sc})$, де $F_{Sh\sigma(Sha)}(sh) = SHA_{sc}^{sh}$. При цьому, виходячи з скінченно-автоматного представлення МДКПО, порядок її перетворення у тестову модель є ідентичним утворенню тестової МСПО в термінах автоматних мереж Петрі [5].

Побудова результуючих об'єктно-орієнтованих моделей поведінки ПО-моделей кооперації ПО, які є ізоморфними МППО, здійснюються шляхом перетворення зв'язків між програмними об'єктами, які обмінюються повідомленнями в рамках моделі послідовностей повідомлень ПО. Оскільки модель кооперації та МППО ізоморфні одна до одної, тож процедура отримання першої моделі з другої є достатньо тривіальною.

Завершальним кроком є побудова тестових моделей кооперації ПО ООПЗ, застосовуючи метод синтезу [6]. В результаті цього отримується E-мережна модель, в якій алгоритм функціонування мережного переходу використовує функції станів моделі станів екземплярів класу ПО та функції переходів моделі діяльності класу програмних об'єктів ООПЗ, чим визначаються вхідні й вихідні позиції мережного переходу.

Узагальнивши запропонований метод, алгоритм метода синтезу тестових моделей поведінки ПО програмного забезпечення ІТС спеціального призначення в інтересах Держприкордонслужби України на стадії проектування за допомогою ООТ складається з п'яти базових кроків:

- 1) побудова моделей послідовності повідомлень програмних об'єктів;
- 2) синтез логічних і тестових моделей станів програмних об'єктів;
- 3) розроблення логічних та тестових моделей діяльності класів програмних об'єктів;
- 4) перетворення моделей послідовностей у моделі кооперації програмних об'єктів;
- 5) синтез тестових структур взаємодії програмних об'єктів.

Використання розроблених тестових моделей ООПЗ забезпечує сукупне тестування динамічної компоненти комплексної моделі об'єктно-орієнтованого програмного забезпечення на стадії його проектування і, як результат, виявлення алгоритмічних та системних помилок у ПЗ ІТС спецпризначення на ранніх стадіях життєвого циклу.

Висновки

За результатами дослідження запропоновано метод та алгоритм синтезу тестових моделей поведінки об'єктів програмного забезпечення ІТС спеціального призначення. Інтегрувавши в собі об'єктно-орієнтоване й імітаційне моделювання та інструментальні засоби підтримки побудови і тестування програмних об'єктів, вони дозволяють автоматизувати процес тестування ПЗ на стадії його проектування.

Розв'язок поставленої задачі досягнуто шляхом формалізації концептуальних одиниць об'єктно-орієнтованої мови моделювання задля використання математичних методів, чим забезпечено автоматизацію побудови й перетворення моделей поведінки ПО ООПЗ, а також реалізації перетворення об'єктно-орієнтованих моделей поведінки програмних об'єктів у тестові структури їх взаємодії.

Застосування логічних та тестових моделей поведінки програмних об'єктів тестування динамічної компоненти комплексної моделі об'єктно-орієнтованого програмного забезпечення дозволяє на стадії проектування ПЗ ІТС спеціального призначення виявляти алгоритмічні й системні помилки, істотно скоротити трудомісткість і тривалість розроблення та провадження програмної системи.

Перспективою подальших розвідок є узагальнення технологічної схеми проектування об'єктно-орієнтованого ПЗ, проведення техніко-економічного аналізу й оцінювання ефективності застосування запропонованого метода на стадії проектування програмного забезпечення складних систем з використанням об'єктно-орієнтованої технології.

Список літератури

1. Концепція розвитку Державної прикордонної служби України на період до 2015 року // Указ Президента України від 19 червня 2006 року № 546/2006 [Електрон. ресурс]. – Режим доступу: <http://zakon2.rada.gov.ua/laws/show/546/2006>.

2. Астахов С. Позиції оборони / Сергій Астахов // Кордон: Центральний друкований орган Державної прикордонної служби України. – 2014. – № 3 (94). – С. 4-5.

3. Буч Г. Объектно-ориентированный анализ и проектирование с примерами приложений на C++: пер. с англ. – М.: "Бином", 2009. – 560 с.

4. Інженерія програмного забезпечення: навч. посіб. / [О.А. Смирнов, О.В. Коваленко, Є.В. Мелешко та ін.]. – К.: ПВЛ КНТУ, 2013. – 409 с.

5. Dorensky O. Method of the models' synthesis for software automated system objects' states in digital images processing / Oleksandr Dorensky // 3. наук. пр. Кіровоградського національного технічного університету: Техніка в сільськогосподарському виробництві, галузеве машинобудування, автоматизація. – 2014. – Вип. 27. – С. 283-292.

6. Доренський О.П. Метод синтезу тестових структур взаємодії програмних об'єктів під час проектування ПЗ на основі об'єктно-орієнтованої технології / О.П. Доренський // Системи управління, навігації і зв'язку. – 2014. – Вип. 3 (27). – (Препринт / ЦНДІ НІУ).

7. Dorensky O.P. The structure of a dynamic component of software systems' comprehensive model within the individual behavior of conceptual units / O.P. Dorensky // Advanced Information Systems and Technologies (AIST 2014): Proceedings of the 3rd International Conference, May 14-16, 2014, Sumy, Ukraine. – Sumy: Publishing and Production Enterprise "Mriya-1", 2014. – P. 99-100.

8. Дудзяний І.М. Об'єктно-орієнтоване моделювання програмних систем: Навч. посібн. / І.М. Дудзяний. – Львів: Видавничий центр ЛНУ ім. І. Франка, 2007. – 108 с.

9. Смирнов А.А. Математическая формализация процесса проектирования объектно-ориентированного программного обеспечения информационных систем / А.А.Смирнов, А.П. Доренский // Информационные технологии и системы в управлении, образовании, науке: моногр.; под ред. В.С. Пономаренко. – Х.: Цифрова друкарня № 1, 2014. – С. 22-36.

Надійшла до редколегії 25.07.2014

Рецензент: д-р техн. наук, проф. І.В. Рубан, Харківський університет Повітряних Сил ім. І. Кожедуба, Харків.

МЕТОД СИНТЕЗА ТЕСТОВЫХ МОДЕЛЕЙ ПОВЕДЕНИЯ ПРОГРАММНЫХ ОБЪЕКТОВ ИНФОРМАЦИОННО-ТЕЛЕКОММУНИКАЦИОННОЙ СИСТЕМЫ СПЕЦИАЛЬНОГО НАЗНАЧЕНИЯ

А.П. Доренский

При разработке сложных программных систем, которой является разрабатываемая интегрированная информационно-телекоммуникационная система специального назначения, на этапе проектирования возникают до двух третей ошибок. С целью их выявления в работе предложен метод синтеза тестовых моделей поведения программных объектов, заключающийся в поэтапном осуществлении построения моделей последовательностей сообщений программных объектов, синтеза логических и тестовых моделей состояний программных объектов и деятельности классов программных объектов, преобразовании моделей последовательностей в модели кооперации программных объектов. Использование синтезированных логических и тестовых структур обеспечивает полное тестирование динамической компоненты комплексной модели объектно-ориентированного программного обеспечения, выявление алгоритмических и системных ошибок на этапе проектирования системы.

Ключевые слова: программное обеспечение, информационно-телекоммуникационная система, проектирование, программный объект, поведение, тестовая модель

METHOD OF SYNTHESIS TEST MODELS OF SOFTWARE OBJECTS BEHAVIOUR OF INFORMATION AND TELECOMMUNICATION SYSTEMS OF SPECIAL PURPOSE

O.P. Dorensky

Integrated information and telecommunications system of special purpose is a complex software system and there are more than half of all errors appear at the design stage. In order to detect them in the proposed method for the synthesis of test behaviours of software objects, which is the percentage of completion of construction of models of message sequence software objects, logic synthesis and test models of state and activity classes transformation sequences of models in a model of cooperation at program objects. Usage of synthesized logic and text structure enables complete testing of complex dynamic component model of object-oriented software. This makes possible to identify algorithmic and system errors in the design phase of information and telecommunication systems and the devices.

Keywords: software, information and telecommunication system, design, program object, behavior, test model.