

УДК 004.4:629.735.072.8:004 (043.5)

Рябокін Ю.М., к.т.н.

МЕТОД КОНСТРУЮВАННЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ПУЛЬТІВ ІНСТРУКТОРА АВІАЦІЙНИХ ТРЕНАЖЕРІВ

Національний авіаційний університет

yulia_r@rambler.ru

Розглянуто проблеми авіаційного тренажеробудування. Запропоновано доменно-орієнтований метод конструювання програмного забезпечення пультів інструктора для різних типів авіаційних тренажерів

Ключові слова: успадкований авіаційний тренажер, пульт інструктора, конструювання програмного забезпечення, багаторазово використовуваний програмний компонент.

Вступ

Починаючи з 90-х років ХХ століття авіаційний тренажер, разом з літальним апаратом, став основним навчальним засобом для підготовки льотного складу. Дозволяючи максимально приблизити умови навчання до умов реальної діяльності авіаційних спеціалістів, тренажери найбільш повно забезпечують формування умінь та навиків при індивідуальній та груповій підготовці авіаційних спеціалістів.

Основними перевагами використання тренажерів є [1]: висока економічність, невеликі витрати часу на тренування, можливість всебічного контролю процесу тренування, широкі варіації умов і ситуацій при тренуваннях, можливість «заморожування» умов, повторення та зміна часового масштабу (прискорення чи уповільнення) тренувальної вправи, можливість виконання «польоту» на літаку з гіпотетичними параметрами, незалежність від метеорологічних умов та насиченості повітряним транспортом зони аеропорту, безпека.

В Україні значна частина використовуваних авіаційних тренажерів – успадковані. Існуючі пульти інструкторів більшості успадкованих авіаційних тренажерів реалізовані апаратно, морально застарілі та фізично зношені в процесі тривалої експлуатації і, найчастіше не підлягають ремонту із-за припинення виробництва їх компонентів і запасних частин. Це є причиною замінити апаратні частини пульта

інструктора на нові апаратно-програмні засоби, на основі сучасних комп'ютерних технологій [2-3]. При цьому, в якості апаратного забезпечення використовуються персональні або промислові комп'ютери загального призначення. Тому, програмне забезпечення (ПЗ) пульта включає системне ПЗ (операційні системи, протоколи обміну, драйвери) і прикладне ПЗ, яке реалізує функціональність пульта [3].

Внаслідок заміни морально застарілого та фізично зношеного апаратного забезпечення успадковане ПЗ вимагає відповідної утилізації. Для забезпечення відновлення/подовження застосування успадкованого ПЗ використовується інженерне направлення – утилізація ПЗ [4]. Основна мета утилізації – продовження застосування ПЗ і його компонент, базуючись на відновленні, переробці та повторному використанні.

Аналіз останніх досліджень та публікацій

У нинішній час у світовій практиці тренажерна індустрія охоплює майже всі сфери взаємодії людини і техніки – космонавтику, авіацію, атомну енергетику. У галузі авіаційного тренажеробудування значних успіхів досягли закордонні фірми такі як *Thomson Training & Simulation* (Великобританія, США, Франція), *Raytheon Training, Lockheed Martin* (США), *Wicat Europe* (Франція), *Drake Electronics Limited* (Великобританія), ЗАТ «Транзас», ЕРА, ЦНТП «Динаміка» (Ро-

сія) [1-3, 5]. В цих державах велика увага приділяється створенню і використанню тренажерів різних типів і різноманітного призначення, що продиктована постійним збільшенням вартості проведення навчання та більшими можливостями обчислювальної техніки.

В Україні за останні 20 років тренажеробудування занепало [1, 5-6]. Старі підприємства в галузі авіаційного тренажеробудування втратили свої найкращі кадри, а також передові технології того часу, стали неспроможними до створення нових, більш вдосконалених технологій, які використовує все світове суспільство. Нові підприємства, що виникли за останні десятиліття, не мають того фундаментального досвіду розробки і виробництва авіаційних тренажерів, які для цього необхідні і властиві підприємствам старої тренажеробудівної промисловості. Таким чином, підприємства, що працюють в області тренажеробудування, спираються на безнадійно застарілий досвід минулого століття і роблять в невеликих кількостях авіаційні тренажери, морально застарілими вже на момент їх виготовлення. А це неприпустимо з точки зору забезпечення безпеки польотів, тієї мети, заради якої тренажери створюються.

Постановка задачі

Наразі в Україні та за її межами використовуються літаки 70-90-х років минулого століття. Члени екіпажу таких літаків готуються за допомогою успадкованих авіаційних тренажерів [1, 5-6]. За процесом підготовки членів екіпажу спостерігає інструктор, який контролює якість пілотування, імітацію взаємодії пілотів із диспетчером, управління імітацією відмов літака, аналізує помилки пілотів та особливості їх пілотування. Пульт інструктора є основним інструментом, що використовується при виконанні вказаних завдань та одним із основних компонентів успадкованих авіаційних тренажерів [1, 5-6]. Загалом, пульт – це наукоємна, технічно складна та досить дорога складова авіаційного тренажера. А відновлення пультів інструкторів успадкованих авіаційних

тренажерів – це актуальне, науково-практичне та складне завдання.

При реінженерії авіаційні тренажери переводяться на сучасну обчислювальну базу, при цьому важливу роль відіграє ПЗ. У зв'язку із зростанням обсягів успадкованого ПЗ реінженерія набуває особливої актуальності. Чималий вклад у вирішення завдань реінженерії успадкованого ПЗ внесли багато вчених: *T.Biggerstaff, J.Bergey, F.Brooks, E.Chicofsky, M.Fowler, M.Freeman, V.Rajlish, Ф.Андон, Л.Бабенко, К.Лавріщева, В.Мейтус, О.Перевозчикова, Б.Позін, М.Сидоров.*

Створення умов для ефективного відновлення та реінженерії успадкованих авіаційних тренажерів та пультів інструктора, і зокрема ПЗ, шляхом прямої та зворотної інженерії є наразі актуальним науково-практичним завданням.

Виклад основного матеріалу дослідження

Відновлення ПЗ – це направлення, що поєднує методи і засоби, які застосовуються для відновлення працездатності ПЗ і його компонентів [4]. Відновленню може підлягати вихідний код, проектні рішення, документація по експлуатації і функціонуванню та інші дані, тому основними процесами, що виконуються при цьому є аналіз вихідного коду або інших формалізованих текстів і виконання перетворень.

Відновлення ПЗ може здійснюватися, як методами прямої інженерії так і зворотної.

Переробка (реінженерія) – направлення, що поєднує методи і засоби, які застосовуються в тих випадках, коли потрібно змінити ПЗ, що неефективно працює або є непридатним як по формі, так і по змісту [4]. Переробка дозволяє не тільки відновити працездатність успадкованого ПЗ, але й поліпшити його шляхом застосування методів, технологій і засобів, які були недоступні раніше.

Завдання відновлення та переробки ПЗ успадкованих авіаційних тренажерів та пультів інструктора не нове. В своїх

роботах автори [7-8] вирішували це завдання шляхом відновлення працездатності існуючого ПЗ; застосуванням реінженерії ПЗ авіаційних тренажерів шляхом створення нового пульта інструктора на новій елементній базі з використанням інформації про успадкований тренажер.

ПЗ пульта інструктора авіаційного тренажера є зрілим та економічно життєздатним доменом, що характеризується об'ємністю та складністю. При розробці такого виду ПЗ виникає потреба у витраті значних ресурсів (часу, фінансів), що є неприйнятним в умовах світової економічної кризи та недостатнього фінансування розробок в авіаційній галузі. Тому питання зниження витрат на розробку такого виду ПЗ є актуальним.

Одним із способів вирішення цієї проблеми в роботі пропонується – повторне використання, шляхом доменної інженерії раніше отриманих рішень-результатів: ідей, знань, вимог, результатів аналізу, проектування, тестування, документування та коду під час розробки нового ПЗ.

Повторне використання ПЗ – це використання для створення нового ПЗ знань, здобутих із завершених розробок програмних систем [4, 9]. Ці знання фіксуються в різноманітних формах – від специфікації вимог до текстів і документації.

Одна із основних задач утилізації заключається в побудові багаторазово використовуваних компонент.

Багаторазово використовуваний програмний компонент (БВК) – елементи знань про минулий досвід розробки програмних систем, якими можуть користуватися не лише їх розробники і їх можна адаптувати для створення нових систем [4, 6, 9].

БВК можна представити моделлю, яка має наступний вигляд [9]:

$$БВК = (T, I, F, R),$$

де T – тип компонента; I – інтерфейс компонента (відкрита частина); F – функціональність компонента; R – реалізація (прихована частина) – програмний код.

Таким чином, програмний БВК є модулем, яким може бути підпрограма, клас, модуль.

Нове ПЗ із застосуванням повторного використання створюється двома способами: синхронним та асинхронним. У роботі акцент зроблено на асинхронному способі створення БВК, оскільки цей спосіб направлений на планове наповнення репозитарію широким набором компонентів, а ця умова є критичною для успішного застосування повторного використання.

За способом реалізації процесів розробки БВК повторне використання є систематичним та несистематичним [9]. У роботі акцентується увага на систематичному повторному використанні. Основу систематичного повторного використання становить доменний аналіз. Він є процесом доменної інженерії та являє собою діяльність зі знаходження та формального опису подібностей та відмінностей у середині домену [9]. Використання доменної інженерії при створенні БВК дає можливість максимізувати

прибуток, скоротити час виходу нового продукту на ринок та зменшити витрати на розробку ПЗ [6, 9].

З метою забезпечення вирішення поставлених задач в роботі пропонується створити метод конструювання ПЗ пультів інструкторів авіаційних тренажерів.

Конструювання ПЗ – це етап життєвого циклу (ЖЦ) створення ПЗ, наступний після проектування, що націлений на побудову працюючого коду (рис. 1) [9-10]. Тобто, конструювання є процесом перетворення артефактів, створених людиною на попередніх етапах ЖЦ, в артефакти зрозумілі комп'ютеру за допомогою інструментальних засобів та мов програмування.

Результатом конструювання є програмний код.

Конструювання є важливою частиною процесу розробки ПЗ. В залежності від розміру проекту на конструювання зазвичай йде 30-80% загального часу роботи [4, 9-10].

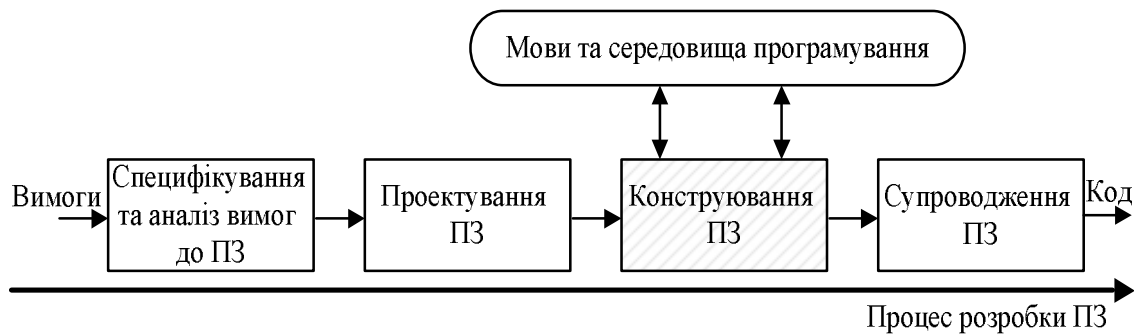


Рис. 1. Розширений ЖЦ розробки ПЗ

Інколи конструювання називають кодуванням або програмуванням [9-10]. В роботі, в термін «конструювання ПЗ» вкладено зміст поняття «програмування».

В 90-х роках ЖЦ створення ПЗ був розширений завдяки застосуванню нових напрямлень в інженерії ПЗ, що вплинули на складові циклу. Вплив відображався в нових задачах, продуктах, процесах та вимогах до ресурсів. До звичайного ЖЦ розробки ПЗ було додано два процеси – доменний аналіз (ДА) та утилізація ПЗ [4, 9].

Застосування ДА під час розробки ПЗ давало можливість виділяти та створювати БВК шляхом повторного використання.

Конструювання ПЗ відбувається засобами, якими є мови та середовища програмування.

При створенні ПЗ використовують класичні методи програмування, що орієнтовані на звичайну розробку ПЗ. Серед них виділяють [4, 9, 11]:

- структурні;
- підпрограмні;
- модульні;
- об'єктно-орієнтовані;
- мегамодульні.

Крім того, застосовують різні підходи до створення ПЗ (аспектно-орієнтований, компонентний та інші). Останнім часом широкої популярності набув підхід на основі повторного використання – компонентний [12-13].

Підхід ґрунтується на побудові ПЗ із існуючих програмних компонентів – фізично окремих частин ПЗ, які взаємоді-

ють між собою через стандартизовані інтерфейси. При цьому, програмний компонент є самодостатньою частиною програмного коду [12-13].

Програмні компоненти, створені в реалізованих програмних проектах, зберігаються в репозитаріях (бібліотеках). В новому програмному проекті, виходячи із вимог замовника, виявляються вимоги до компонентів, далі перевіряється наявність цих компонентів в бібліотеці. Якщо вони знайдені, то компоненти виймаються із репозитарію і використовуються повторно. В іншому випадку створюються нові компоненти, вони застосовуються в проекті і включаються в репозитарій.

Застосування компонентного підходу до програмування сприяє можливості багаторазового використання, покращенню якості ПЗ та продуктивності розробників ПЗ, зменшенню вартості розробки до 70% та зменшенню на 30% часу розробки програмних продуктів [9, 12-13].

Оскільки, цей підхід орієнтований на багаторазове використання компонентів, то створюється програмне середовище конструювання.

У роботі пропонується метод конструювання ПЗ, сутність якого полягає у використанні компонентного підходу до програмування, шляхом застосування середовища БВК, що створюється на основі систематичного асинхронного повторного використання за методом доменній інженерії (рис. 2). Таким чином, середовище програмування та ПЗ є доменно-орієнтованими.

Відповідно до запропонованого методу, компоненти середовища створюються виходячи з досвіду, який накопичено в домені та мають формалізоване зображення, яке складається із онтологій

– графічні та аналітичні нотації, математичні моделі доменних сутностей, шаблони програмного коду. Компоненти зберігаються в репозитарії середовища.

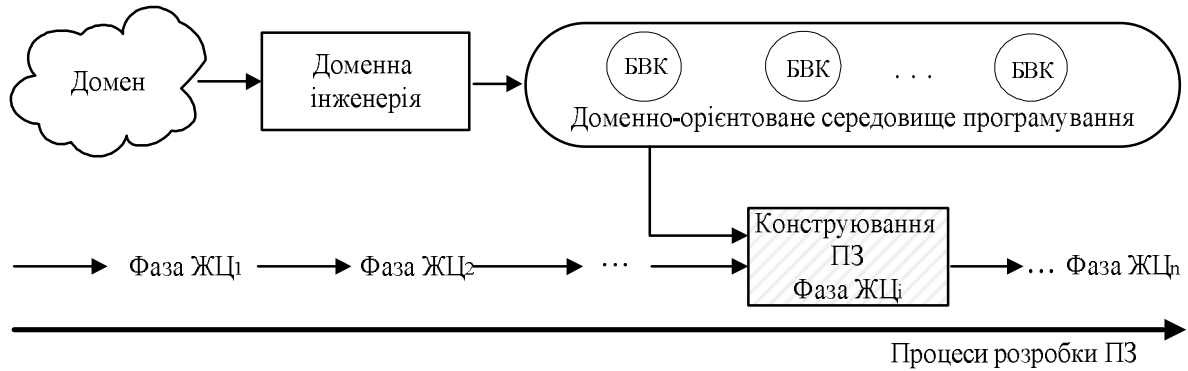


Рис. 2. Конструювання ПЗ за доменно-орієнтованим методом

Доменний аналітик разом з доменним експертом аналізують досвід накопичений в домені, виділяють та описують БВК (рис. 3). Доменний інженер використовує формалізований опис БВК, здійснює створення БВК, середовища БВК та зберігає БВК до репозитарію.

Як відомо, процес визначення та створення БВК здійснюється відповідно до обраного методу з проведення доменної інженерії [9, 13-14]. Процес вибору методу доменної інженерії залежить від зрілості та стабільності домену і є важливою складовою доменно-орієнтованого методу, що впливає на

процес доменної інженерії та її вихідні результати.

В роботі, в якості методу доменної інженерії, пропонується метод організаційного доменного моделювання (ОДМ) [14-16]. Основні процеси, які реалізуються згідно методу ОДМ є наступними [14]:

- визначення домену;
- збір та накопичення інформації щодо домену;
- доменний аналіз;
- формалізоване зображення компонент;
- розробка середовища БВК.

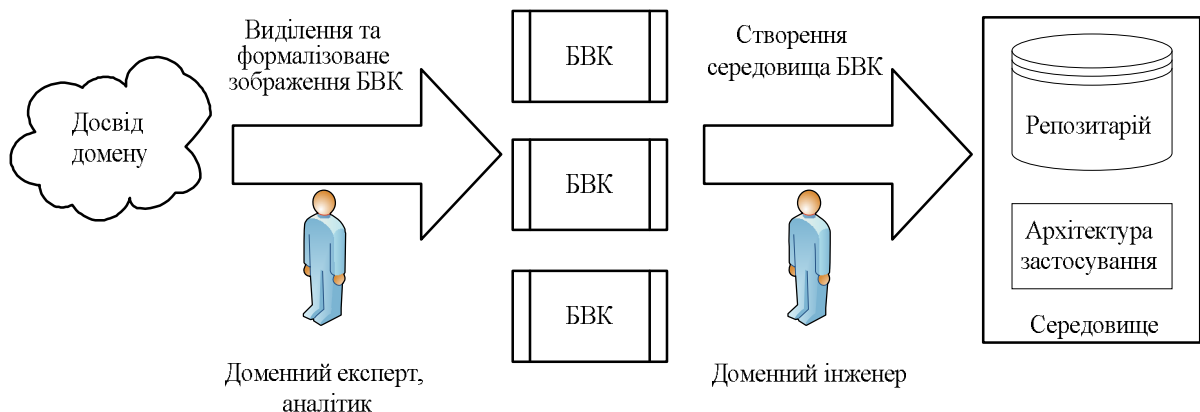


Рис. 3. Схема створення та збереження БВК

Перший процес доменно-орієнтованого методу – визначення домену, визначає загальні риси та межі домену, відносини з іншими доменами.

Під час виконання процесів збору та накопичення даних здійснюється визначення методів збору та джерел інформації щодо домену.

ДА представляє собою один із найважливіших процесів доменної інженерії за будь-яким її методом, оскільки відображає властивості, ролі та зв'язок основних сутностей домену, які поєднуються в моделі домену. На основі моделі здійснюється визначення та формування БВК [15-17].

Процес ідентифікації та формалізованого зображення компонент полягає в створенні та формалізованому представленні компонент у вигляді онтологій [18-19], які в роботі пропонується зображувати: графічними та аналітичними нотаціями, математичними моделями, та шаблонами програмного коду. Отримані представлення зберігаються у середовищі БВК.

Для організації середовища існує два підходи [4, 9, 11]: композиційний та адаптивний. В роботі застосовується адаптивний підхід, сутність якого полягає у використанні наперед визначеної багаторазово використовуваної архітектури застосування і множини БВК для заповнення вільних місць в архітектурі ними.

Тому, шляхом доменної інженерії створюється не тільки множина БВК, але і багаторазово використовувана архітектура доменно-орієнтованого застосування:

$$S = (R, A),$$

де S – середовище; R – репозитарій БВК; A – багаторазово використовува-

на архітектура доменно-орієнтованого застосування.

Адаптивний підхід при організації багаторазово використовуваної архітектури доменно-орієнтованого застосування дозволяє виділити постійну архітектуру і обмежену множину точок в ній для внесення змін.

Запропонований в роботі доменно-орієнтований метод досліджується на прикладі створення середовища, направлено на конструювання ПЗ пультів інструктора різних типів авіаційних тренажерів. Але метод може застосовуватися для будь-якого іншого зрілого домену.

Відповідно, при цьому, репозитарій становить собою сховище БВК, зокрема інтерфейсних елементів, параметрів, журналу та програм польоту, яким властиве формалізоване зображення у вигляді онтологій.

Архітектура доменно-орієнтованого застосування – структура, яка містить відношення між частинами застосування, що заповнюються БВК при конструюванні ПЗ (рис. 4). Панель пульта інструктора є головним композиційним класом (*InstructorPanel*), що включає такі класи: *InterfaceElement* (інтерфейсний елемент), *Parameter* (параметр), *FlightJournal* (журнал польоту), *FlightProgram* (програма польоту).

Крім того, до складу панелі входить елемент – компонент обміну параметрами (*ComponentInterchange*), призначений для організації двостороннього обміну параметрами між пультом інструктора і авіаційним тренажером за UDP-протоколом. Він представлений класами *UDPClient* – клієнт, який передає повідомлення та *UDPServer* – сервер, що приймає повідомлення.

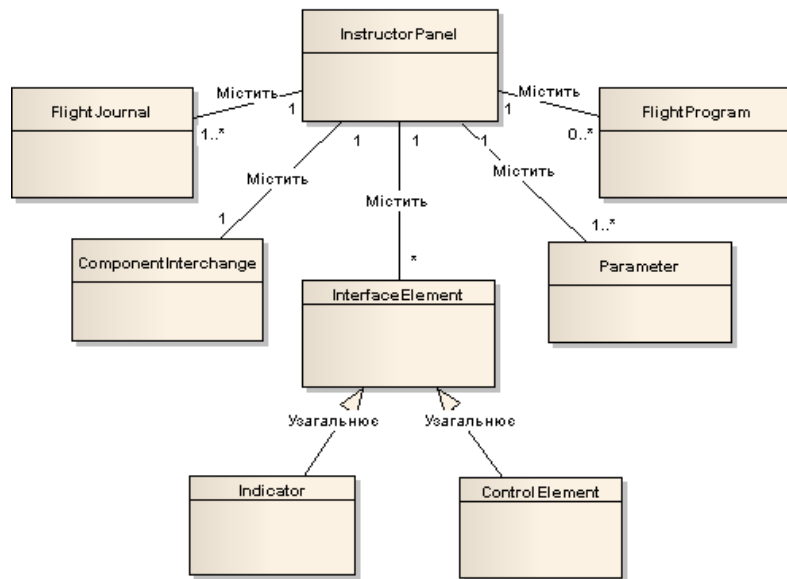


Рис. 4. Архітектура доменно-орієнтованого застосування

Висновки

Повторне використання, як метод утилізації, залишається одним з найбільш дієвіших шляхів забезпечення підвищення якості та зниження витрат на конструювання ПЗ. Сутність його полягає в застосуванні накопиченого досвіду в домені і компонентів існуючого ПЗ при створенні нового ПЗ.

Основу систематичного повторного використання складає доменна інженерія. Використання доменної інженерії при створенні БВК дає можливість максимізувати прибуток, скоротити час виходу нового продукту на ринок та зменшити видатки на розробку ПЗ.

Оскільки стан галузі авіаційних тренажерів дозволяє створювати високо рівневі, багаторазово використовувані рішення, то використання доменної інженерії при створенні таких рішень дає можливість виділити відповідну інформацію домену, джерела інформації, термінологію домену, таксономію і на їх основі побудувати модель домену, онтологію.

Запропонований в роботі доменно-орієнтований метод конструювання ПЗ пультів інструктора авіаційних тренажерів шляхом застосування середовища багаторазово використовуваних програмних компонентів, яке створюється методами

доменної інженерії та систематичним асинхронним повторним використанням, дає можливість конструювати ПЗ пультів інструктора різних типів авіаційних тренажерів шляхом використання програмного середовища компонентів багаторазового використання для забезпечення багаторазової розробки та супроводу ПЗ пультів інструктора авіаційних тренажерів.

Список літератури

1. Серегин Г.Н. Авиационные тренажеры – реальный путь к повышению безопасности полетов / Г.Н. Серегин // «Право и безопасность». – 2006. – №3-4. С. 20-21.
2. Сидоров Н.А. Реинженерия программного обеспечения информационно-моделирующих тренажерных комплексов / Н.А. Сидоров, В.Т. Недоводеев, В.А. Хоменко, И.П. Сердюк, Е.Н. Сидоров // Управляющие системы и машины. – 2008. – № 4. – С. 68-74.
3. Сидоров М.О. Методологічні принципи реінженерії програмного забезпечення успадкованих авіаційних тренажерів / М.О. Сидоров, Л.М. Иванова, В.А. Хоменко // Матеріали VIII Міжн.наук-техн. конф. АВІА-2007.– 2007. – Т. 1. – С. 13.119-13.122.

4. Сидоров Н.А. Восстановление, повторное использование и переработка программного обеспечения. I / Н.А. Сидоров // УСим. – К. – 1998. – № 3. – С. 74-83.
5. Луцький М.Г. Метод створення програмного забезпечення пульта інструктора авіаційного тренажеру / М.Г. Луцький, Ю.М. Рябокінь // Вісник НТУУ «КПІ». Інформатика, управління та обчислювальна техніка: зб. наук пр. – К.: Век+, – 2009. – № 51. – С. 78–83.
6. Рябокінь Ю.М. Формування повторно використовуваних рішень при створенні програмного забезпечення пульта інструктора авіаційного тренажеру методом доменного аналізу / Ю.М. Рябокінь // Інженерія програмного забезпечення, – 2010. – № 2. – С. 44-53.
7. Сидоров Е.Н. Архитектура программного обеспечения пульта инструктора авиационного тренажера / Е.Н. Сидоров // Міжнародна науково-практична конференція аспірантів і студентів «Інженерія програмного забезпечення 2008». – К. – 2009. – С. 45–49.
8. Хоменко В.А. Шаблон программного обеспечения устройств святы с объектом авиационных тренажеров / В.А. Хоменко, Е.Н. Сидоров, И.Б. Мендзевровский // Проблемы программирования. – К. – 2008. – С. 30–40.
9. Лаврищева Е.М. Методы и средства инженерии программного обеспечения. Учебник / Е.М. Лаврищева, В.А. Петрухин // Московский физико-технический институт. – 2006. – 304с.
10. Конструирование программного обеспечения (Software Construction по SWEBOK). [Електронний ресурс]. – Режим доступу: [http:// http://swebok.sorlik.ru/](http://swebok.sorlik.ru/)
11. Лаврищева К.М. Методи програмування. Теорія, інженерія, практика / К.М. Лаврищева. – К. – Науково-виробниче підприємство «Видавництво Наукова думка», 2006. – 452с.
12. Грищенко В.Н. Методы и средства компонентного программирования / В.Н. Грищенко, Е.М. Лаврищева // Кибернетика и системный анализ, 2003. – № 1 – С. 39–55.
13. Crnkovic I. Component-Based Software Engineering: building systems from Components / I. Crnkovic, S. Larsson, J. Stafford // At 9th Conference and Workshops on Engineering of Computer Based Systems. – 2002. – vol. 27. – N 3. – P. 47–50.
14. Simos M., Creps D., Klinger C., Levine L Lockheed Martin Tactical Defense Systems. Organization Domain Modeling (ODM) Guidebook: Version 2.0 Manassas: STARS-VC-A025/001/00. – 1996. – 509 p.
15. Alana E. Domain engineering methodologies survey / E. Alana, A. Rodriguez // GMV AEROSPACE AND DEFENCE S.A. – 2007. – 38 p.
16. Dudnyk V. Method of domain analysis to information support of aircraft / V. Dudnyk, U. Ryabokin, I. Mendzhevovskiy // The Fifth World Congress «AVIATION IN THE XXI-st CENTURY». Safety in Aviation and Space Technologies. –К.: NAU. – 2012. – P.1.10.1-1.10.19.
17. Рябокінь Ю.М. Доменний аналіз програмного забезпечення інтерфейсу пульта інструктора авіаційного тренажеру / Ю. М. Рябокінь, І. Б. Мендзевровський // Проблеми інформатизації та управління: – зб. наук. пр. – К.: НАУ, – 2012. Вип. 4 (40) – С. 77–82.
18. Kleshchev A.A. An ontology-oriented approach to user interface development / A.A. Kleshchev, V.V. Gribova //International Journal «Information Theories & Applications» . – 2003. – Vol. 10(1) – P.87-94.
19. Палагин А.В. Методика проектирования онтологии предметной области / А.В. Палагин, Петренко Н.Г., К.С. Малахов // Комп'ютерні засоби, мережі та системи . – 2011. – № 10. – С.5–12.