

УДК (UDC) 004.051

**Гамзаєв Рустам
Олександрович**

к.т.н., доцент; доцент кафедри інтелектуальних програмних систем і технологій, ННІ комп'ютерних наук та штучного інтелекту, Харківський національний університет імені В. Н. Каразіна, майдан Свободи 4, м. Харків, Україна, 61022
e-mail: rustam.gamzayev@karazin.ua
<https://orcid.org/0000-0002-2713-5664>

Інтелектуальна інформаційна технологія підтримки мінливості процесів життєвого циклу програмного забезпечення кіберфізичних систем

Актуальність. Розробка програмного забезпечення (ПЗ) кіберфізичних систем (КФС) має враховувати специфічні особливості їх побудови та функціонування, що передбачає можливість підтримки мінливості проектних ресурсів та системних рішень на всіх основних етапах життєвого циклу (ЖЦ) КФС. Вирішення цих проблем неможливо без використання інтелектуальних методів та засобів і тому тематика цього дослідження є актуальною науково-технічною задачею.

Мета. Метою роботи є розробка інтелектуальної інформаційної технології (ІТ), яка забезпечує наскрізну підтримку мінливості проектних активів на всіх основних фазах ЖЦ ПЗ КФС, що, у кінцевому рахунку, має підвищити показники якості критично важливих процесів розробки та супроводу таких систем.

Методи дослідження. На основі критичного аналізу та методологічного узагальнення деяких вже отриманих раніше наукових та практичних результатів, розроблена структурно-функціональна схема ІТ, яка інтегрує знання-орієнтовані модельно-технологічні засоби, що дозволяє забезпечувати підтримку властивостей варіабельності, адаптивності, конфігурування та настроюваності проектних рішень та програмних компонентів КФС на етапах доменного інжинірингу, архітектурного проектування, конструювання коду та супроводу компонентів її ПЗ.

Результати. На прикладах систем «Розумний будинок» та мобільних систем доповненої реальності досліджені деякі суттєві особливості побудови та функціонування КФС, сформовано методологічний базис для знання-орієнтованої розробки ПЗ таких систем. Запропонована узагальнена схема ІТ в нотатції IDEF0, визначені її основні функціональні блоки, проведені програмні експерименти та обчислені кількісні метрики, які показали сумарне зростання показників якості процесів розробки та супроводу ПЗ приблизно на 22,4%.

Висновки. Представлені дослідження підтвердили доцільність застосування знання-орієнтованих моделей, методів та інструментальних засобів для розробки та супроводу ПЗ КФС, і можливість створення наскрізної інтелектуальної інформаційної технології, яка підтримує властивості змінності проектних ресурсів та системних рішень на всіх основних фазах ЖЦ КФС і що, в свою чергу, дозволяє суттєво підвищити рівень якості процесів створення таких систем.

Ключові слова: інтелектуальні моделі та методи, кіберфізична система, програмне забезпечення, життєвий цикл, інформаційна технологія, мінливість, варіабельність, адаптивність, конфігурування, налаштування, метрика, якість

Як цитувати: Гамзаєв Р.О. Інтелектуальна інформаційна технологія підтримки мінливості процесів життєвого циклу програмного забезпечення кіберфізичних систем. *Вісник Харківського національного університету імені В.Н. Каразіна, серія Математичне моделювання. Інформаційні технології. Автоматизовані системи управління.* 2025. вип.. 66. С.6-18. <https://doi.org/10.26565/2304-6201-2025-66-01>

How to quote: Gamzayev R. O. "Intelligent information technology to support changeability in software life cycle processes of cyber-physical systems", *Bulletin of V.N. Karazin Kharkiv National University, series Mathematical modelling. Information technology. Automated control systems*, vol. 66, pp. 6-18, 2025. <https://doi.org/10.26565/2304-6201-2025-66-01>

1 Вступ. Мета та актуальність дослідження

Сучасні кіберфізичні системи (КФС) представляють собою складні апаратно-програмні комплекси, що безпосередньо поєднані з реальними фізичними процесами та пристроями у зовнішньому середовищі (яке, як правило, пов'язано з Internet), і які забезпечують при цьому

функціональність, необхідну для різних груп їх кінцевих користувачів. Структурно-функціональна складність, розподілена системна архітектура, гетерогенність ресурсів даних та деякі інші особливості КФС (див. більш детально нижче) зумовлюють необхідність використання нових підходів до розробки їх програмного забезпечення (ПЗ), а останнім часом беззаперечним трендом в цьому домені стає з широке застосування моделей, методів і технологій штучного інтелекту (artificial intelligence - AI). Зокрема, у деяких новітніх роботах з цієї тематики [1–3] розглядається використання технологій AI для підтримки ключових процесів життєвого циклу (ЖЦ) КФС, таких як інженерія вимог, моделювання даних, конструювання коду та автоматизоване тестування компонентів ПЗ. Окрім того, засоби AI дедалі частіше інтегруються в етапи супроводу й експлуатації ПЗ у рамках практик культури DevOps для реалізації автоматичного масштабування, самовідновлення (self-healing), інтелектуального моніторингу та оптимізації розгортання програмних мікросервісів, як це реалізується, зокрема, в операційних середовищах для їх оркестровці, на кшталт Kubernetes [4]. В той же час слід зазначити, що існує досить обмежена кількість публікацій щодо застосування методів доменного інжинірингу, знання-орієнтованих моделей підтримки варіабельності та адаптивності компонентів ПЗ КФС на етапах їх проектування та супроводу, а також недостатньо висвітлюються питання побудови лінійок програмних продуктів для КФС [5,6] як ефективного шляху у розробці таких систем для забезпечення можливостей їх адаптації, конфігурування та налаштування з урахуванням постійних змін у вимогах їх замовників, фахівців з супроводу та кінцевих користувачів.

Саме тому метою цього дослідження є розробка інтелектуальної інформаційної технології (ІТ), яка забезпечує наскрізну та керовану підтримку мінливості проектних активів на всіх основних фазах ЖЦ ПЗ КФС, і що, у кінцевому рахунку, має підвищити показники якості критично важливих процесів розробки та супроводу таких систем.

2 Методологічний базис побудови інтелектуальної інформаційної технології для підтримки мінливості у процесах життєвого циклу ПЗ КФС

2.1 Особливості побудови та функціонування програмного забезпечення КФС

Беручи до уваги загальне визначення властивостей сучасних КФС [1] і з метою дослідження саме особливостей розробки та супроводу ПЗ таких систем, у цій роботі в подальшому розглядаються застосунки класу «Інтернет речей (Internet of Things -IoT / «Розумний будинок (Smart Home - SH)» [7], і мобільні системи доповненої реальності (mobile augmented reality system - MARS) [8].

Для IoT / SH систем, які підтримують взаємодію різних типів пристроїв та програмних компонентів для автоматизації процесів життєзабезпечення та їх керування користувачами у віддаленому режимі, можна зазначити такі особливості їх побудови та функціонування як [7]:

- 1) розподілена багаторівнева системна архітектура;
- 2) велика кількість взаємопов'язаних апаратних та програмних компонентів,
- 3) необхідність функціонування більшості з них в режимі «7/24».

Слід особливо зазначити, що останнім часом системи класу IoT виявляють новий тренд у своєму розвитку, який отримав назву «Internet of Military Things (IoMT)» - «Інтернет військових речей», або «Internet of Military Defense Things (IoMDT)» - «Інтернет речей військового захисту» [9], і що має виняткову важливість для сучасних потреб національної безпеки нашої держави.

Для систем класу MARS, які забезпечують візуалізацію навколишнього фізичного середовища та його доповнення іншими віртуальними об'єктами із використанням мультимедійної інформації (текст, графіка, відео і т.п.), характерними є наступні властивості (або вимоги) [10]:

- 1) функціонування з урахуванням обмежень на такі ресурси мобільних пристроїв (МП) як продуктивність процесора, обсяг оперативної пам'яті, ємність акумулятора МП, та розміри екрану;
- 2) велика кількість та різноманітність конфігурацій програмного та апаратного забезпечення МП у різних груп користувачів;
- 3) інтенсивне використання мережевого трафіку для обміну даними між окремими МП та серверами системи.

Вищезазначені особливості цих 2-х класів КФС узагальнено можуть бути сформульовані наступним чином: (I) велика структурно-функціональна складність та гетерогенність їх компонентних конфігурацій та інформаційних ресурсів; (II) необхідність використання цих

систем за умов обмежених апаратно-програмних ресурсів та змінного обчислювального навантаження. Саме вони й визначають основні вимоги до розробки та супроводу ПЗ систем КФС.

2.2 Концептуальне узагальнення поняття мінливості у процесах ЖЦ ПЗ КФС

Зважаючи на постійного зростання складності функціональних задач, які мають бути вирішені шляхом використання КФС, на теперішній час чітко визначилася тенденція до розробки вже не окремих, хоча й досить складних програмних продуктів, а до створення та подальшого ефективного використання сукупностей взаємопов'язаних програмних компонентів, які отримали назву лінійок програмних продуктів (ЛПП) [4-6]. Головними ознаками цих нових підходів слід вважати їх спрямованість на забезпечення варіативності (variability), адаптації (adaptivity), конфігурування (configurability) та налаштування (customizability) у процесах опрацювання вимог, розробки архітектурних рішень, програмних компонентів, функціоналу та користувацьких властивостей. Для концептуального узагальнення цих споріднених властивостей ПЗ на різних етапах його ЖЦ пропонується використовувати поняття «мінливості (changeability)», яке визначає здатність ПЗ зберігати (або керовано змінювати) свої основні показники якості за умов будь-яких передбачуваних (або запланованих) змін у процесах розробки та експлуатації КФС. Слід зазначити, що поняття мінливості в цілому у системній інженерії і, зокрема, його логічний зв'язок з поняттям надійності (robustness) вперше стало предметом публікації фахівців з *Massachusetts Institute of Technology (MIT)* [11], а у цьому дослідженні воно графічно подано у вигляді UML діаграми класів, яка наведена на рисунку 1.

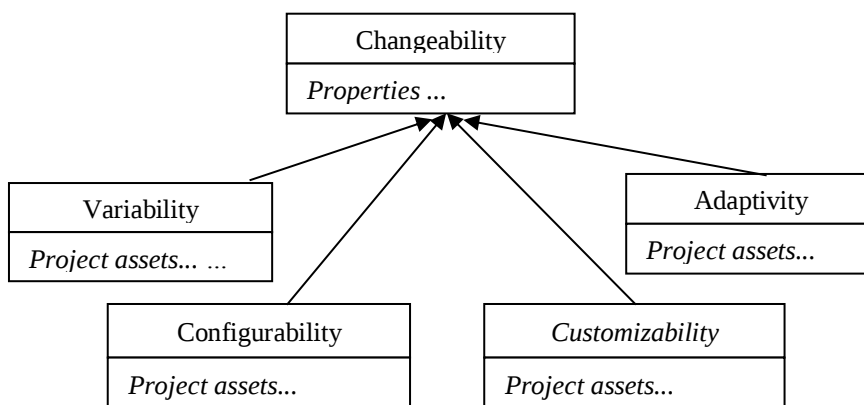


Рис. 1. - Концептуальна схема узагальнення різних типів мінливості у процесах ЖЦ ПЗ КФС

Fig. 1- Conceptual diagram of generalization of different types of variability in the life cycle processes of CPS software.

Властивість мінливості ПЗ може бути забезпечено шляхом розробки та застосування відповідних моделей, методів і інструментальних засобів на всіх основних етапах ЖЦ КФС, які позначені як *Project assets* (проектні активи) в UML класах на рисунку 1, і для структурованого представлення яких запропонована фреймова модель [12], фрагмент якої наведено на рисунку 2.

Frame_OMC: “Operation Metamodel for Changeability in SPL lifecycle (LC)”

{Slot_1: “SPL LC Phases” = (Domain Engineering, Architectural Design, Code Construction, Software Deployment / Maintenance)

Slot_2: “Changeability subtypes” = (Variability, Adaptivity, Configurability, Customizability)

Slot_3: “Changeability Models” = (Feature models, Algorithmic models, DSL-based models, Recommendation models));

Frame_DM: “Domain Engineering”

{ Slot_1: “Methods” = (FODA, ODM, ...)

Slot_2: “CASE tools” = (FeatureIDE, Eclipse EMF, ...)

..... }

Рис. 2. - Фрагмент фреймової моделі мінливості процесів та артефактів у ЖЦ ПЗ КФС

Fig. 2 - Fragment of the frame model of variability of processes and artifacts in the CPS software life cycle.

Особливості побудови та переваги застосування цієї моделі обговорюються в [12], де також наведені більш детальні відомості про задіяні в ній конкретні методи доменного інжинірингу (фреймовий слот “*Methods*”): FODA (Feature Oriented Domain Analysis), ODM (Organization Domain Modeling) та відповідні інструментальні CASE-засоби для їх практичного застосування (фреймовий слот “*CASE tools*”): *FeatureIDE*, *Eclipse EMF* та деякі інші.

2.3 Основні методологічні принципи розробки ПЗ КФС

На основі проведеного аналізу особливостей КФС (див. 2.1) та концептуального узагальнення поняття мінливості на різних фазах ЖЦ ПЗ (див. 2.2) можливо сформулювати основні методологічні принципи розробки ПЗ КФС у наступний спосіб [13]:

- (1) доменне моделювання (domain modeling) властивостей ПЗ КФС із використанням багатовимірних методів опрацювання експертних знань (handling of expert knowledge);
- (2) архітектурне проектування ПЗ (software architecting) з можливістю адаптації програмно-апаратних ресурсів КФС, зокрема, із використанням методів логічного виводу на основі аналізу прецедентів (case-based reasoning);
- (3) розробка лінгвістичних моделей і спеціальних мовних засобів (domain-specific language) для автоматизації процесів конструювання коду програмних компонентів КФС;
- (4) використання методів та технологій штучного інтелекту, зокрема, шляхом побудови рекомендаційних систем (recommender system) для динамічного конфігурування апаратно-програмних компонентів КФС.

Для практичного застосування цих підходів у процесах розробки ПЗ на різних фазах ЖЦ КФС потрібна інтелектуальна інформаційна технологія, основні проектні та програмні рішення якої раніше вже були отримані в реальних застосунках та представлені в [7,10,14-20].

3 Узагальнена схема інтелектуальної інформаційної технології (ІІТ) та підхід до сумарної оцінки підвищення показників якості процесів ЖЦ ПЗ КФС

3.1 Функціональна схема запропонованої ІІТ в нотації IDEF0

На рисунку 3 наведена схема запропонованої ІІТ з використанням нотації IDEF0.

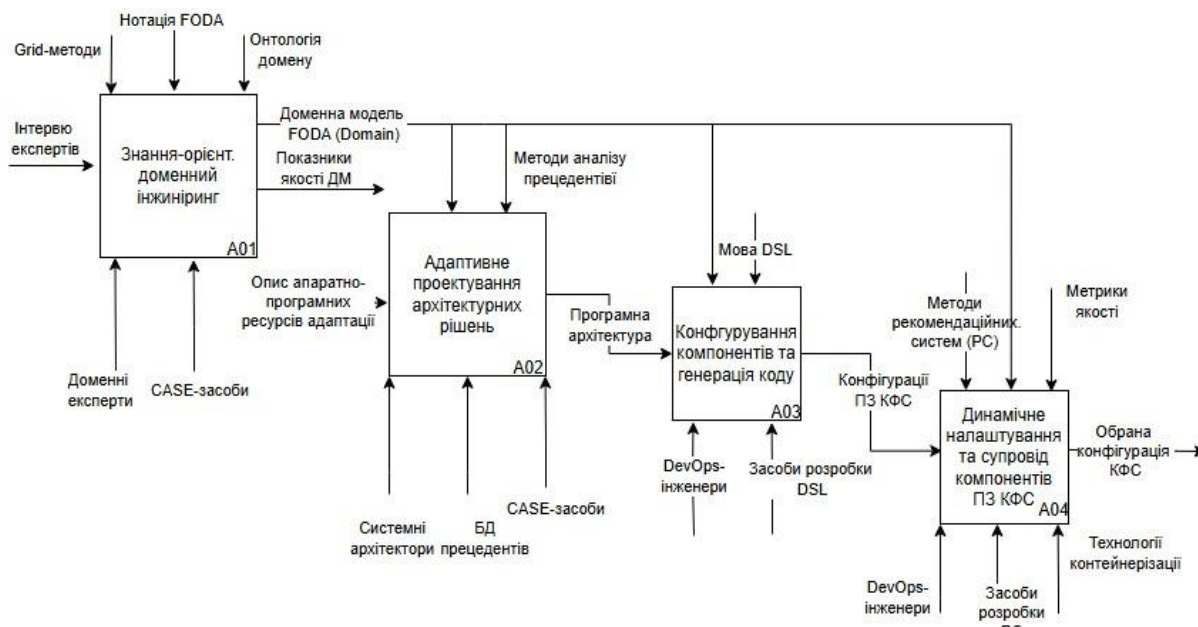


Рис. 3 - Узагальнена схема запропонованої ІІТ в нотації IDEF0

Fig. 3 - Generalized diagram of the proposed IIT in the IDEF0 notation.

У її складі міститься 4 основних функціональних блоків (ФБ), які технологічно забезпечують послідовну реалізацію основних методологічних принципів (1)-(4) розробки ПЗ КФС (див. п. 2.3) із застосуванням конкретних моделей, методів та CASE-засобів (див. п. 2.2). Нижче кожен з цих

4-х ФБ розглянуто більш детально, а також запропоновано підхід до отримання сумарної оцінки підвищення показників якості процесів ЖЦ ПЗ КФС в результаті послідовного застосування цієї ІТ. Для опису роботи кожного ФБ використовуються стандартні терміни визначення його інтерфейсів: *Input*, *Control*, *Mechanism*, *Output* в нотації IDEF0.

3.2 Функціональний блок (ФБ) “Знання-орієнтований доменний інжиніринг”

Цей ФБ, який позначено на рисунку 3 як А01, забезпечує технологічний процес побудови доменної моделі предметної області (ПрО) застосування відповідної КФС із використання додаткових знань експертів, розробників та майбутніх користувачів її функціональних можливостей. Цей процес побудови розширеної, знання-орієнтованої ДМ для систем класу IoT/SH (див. вище у п. 2.1) детально розглянуто в [14,15], а його основні функції виглядають наступним чином (див. рисунок 3):

- *вхідними даними (Input)* для ФБ А01 є текстовий опис інтерв'ю, отриманих від групи експертів у цій ПрО, що у загальному випадку є неструктурованим гетерогенним інформаційним ресурсом, для ефективної обробки якого потрібно застосувати додаткові методи опрацювання експертних знань;
- *алгоритм управління (Control)* роботою ФБ А01 використовує метод побудови репертуарних сіток (repertory grid method) [14], що дозволяє отримати багатовимірну модель експертних знань відповідної ПрО, або доменна модель (ДМ), яка враховує технічні, соціальні (користувацькі) та економічні вимоги до розробки майбутньої КФС, і для формалізованого представлення цієї моделі мотивовано обрано застосування нотації FODA (Feature-oriented Design Analysis), також додатково може бути використана одна з вже існуючих онтологій для цього домену [15];
- *механізм реалізації (Mechanism)* ФБ А01 передбачає участь у цьому процесі інженерів з обробки знань та доменних експертів, які застосовують відповідні CASE-засоби: системи Protégé, GridSuite, SOVA [15];
- *результатом роботи (Output)* ФБ А01 є знання-орієнтована ДМ, яка позначена як FODA (Domain), що слугує інформаційним ресурсом для управління роботою всіх інших ФБ з метою забезпечення властивостей мінливості на інших фазах ЖЦ ПЗ КФС, крім того, для подальшого використання в системі розраховуються показники якості цієї ДМ (див. рисунок 3).

Приклад фрагменту побудованої у такий спосіб варіабельної FODA – моделі для Про «Розумний будинок» [15] представлено на рисунку 4.

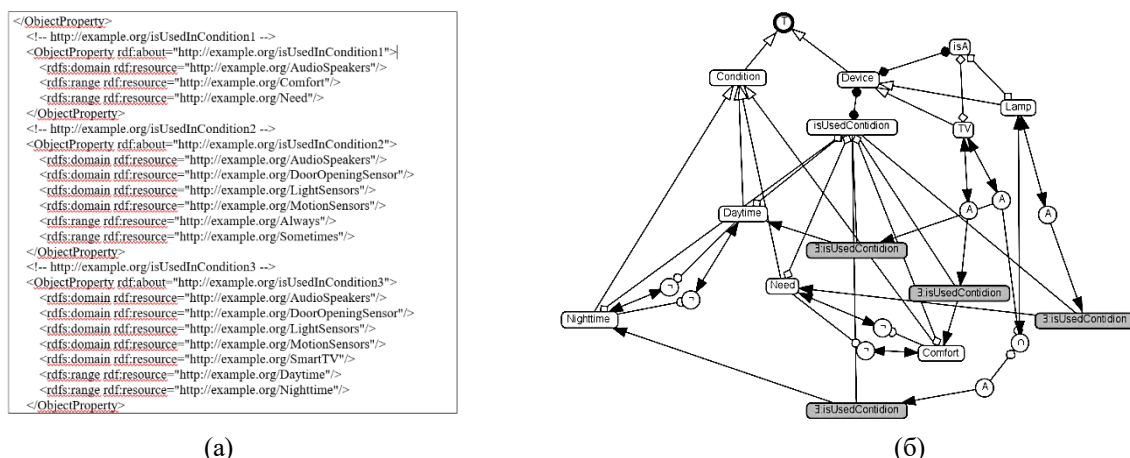


Рис. 4 - Результати роботи ФБ А01: (а) – опис варіабельної FODA-моделі в форматі XML / OWL, (б) - семантичний онтограф, який згенеровано на основі опису FODA - моделі

Fig. 4 - Results of the A01 FB operation: (a) – description of the variable FODA model in XML / OWL format, (b) – semantic ontograph generated based on the description of the FODA mode

Результати експериментального дослідження показників структурно-функціональної складності ДМ [16], побудованих із використанням запропонованого підходу, у порівнянні зі

складністю ДМ, отриманих із застосуванням стандартних методів доменного моделювання, таких, як ODM (Organizational Domain Modeling) та JODA (Joint integrated avionics Object oriented Domain Analysis), показали зменшення їх складності приблизно на 21,1%.

3.3 Функціональний блок “Адаптивне проектування архітектурних рішень”

Після проведення доменного інжинірингу як першого із основних етапів ЖЦ ПЗ КФС, наступним важливим кроком з точки зору забезпечення можливостей мінливості її проектних активів (див. п. 2.3) є етап адаптивного проектування архітектурних рішень, які мають враховувати загальні особливості КФС, що сформульовані у п. 2.1. Для забезпечення цих процесів у запропонованій ПТ передбачений ФБ А02 (див. схему на рисунку 3), а загальний алгоритм його роботи може бути представлений у наступний спосіб (більш детально вони представлені та досліджені у [10,17]:

- *вхідними даними (Input)* для ФБ А02 є опис апаратно-програмних ресурсів КФС, які мають бути використані для розробки адаптивних архітектурних рішень;
- *алгоритм управління (Control)* базується на використанні вже побудованої доменної моделі нотації *FODA (Domain)* (див. п. 3.2), а для пошуку проектних рішень, враховуючи високу складність та слабоформалізований характер процесів архітектурного проектування систем КФС, використовуються методи логічного висновку на основі аналізу прецедентів (case-based reasoning - CBR) і набір кількісних метрик для оцінки поточного стану ресурсів адаптації та показників якості функціонування КФС (конкретні приклади таких метрик для мобільних систем доповненої реальності наведені в [9]);
- *механізм реалізації (Mechanism)* ФБ А02 передбачає участь у цьому процесі системних архітекторів, які застосовують базу даних прецедентів та інструментальне ПЗ (див. більш детально в [16]);
- *результатом роботи (Output)* ФБ А02 є системна програмна архітектура для визначеного типу КФС, яка має враховувати можливості її адаптації до змін в оточуючому середовищі її функціонування, обмеження на наявні обчислювальні ресурси та ін.

В [9] розроблена еталонна 3-х рівнева програмна архітектура для систем MARS із вбудованим блоком адаптивного управління, яка у вигляді UML діаграми розміщення компонентів представлена на рисунку 5.

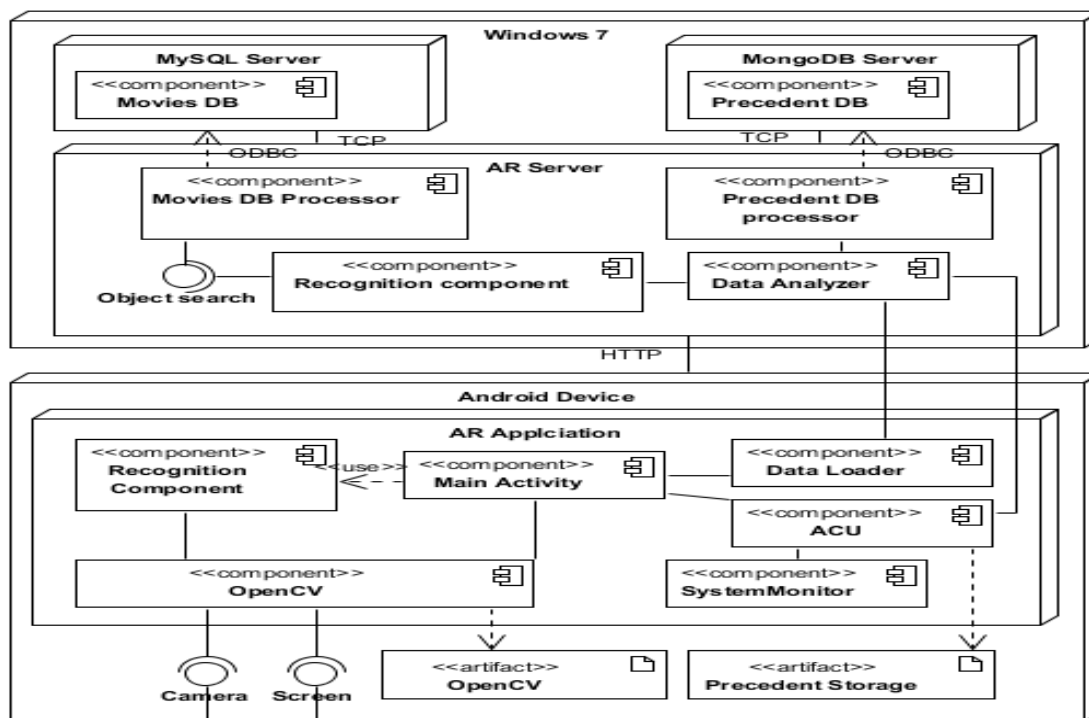


Рис. 5 - Архітектура систем MARS із вбудованим блоком адаптивного управління

Fig. 5 - Architecture of MARS systems with an embedded adaptive control unit.

Основні функціональні компоненти МСДР, які забезпечують можливість адаптації системи, є наступні [16]

1) на рівні клієнтських застосунків (AR Application): ACU (Adaptive Control Unit) – це блок адаптивного управління (БАУ) на основі CBR-методів, Precedent Storage – локальне сховище даних для поточних прецедентів (Berkeley DB engine), System Monitor – компонент для моніторингу стану ресурсів адаптації наявних МП (процесор, пам'ять, відео-камера та ін.);

2) на рівні серверу бізнес-логіки (AR Server): Data Analyzer – це компонент для аналізу даних про стан клієнтських застосунків; Precedent DB Processor – процесор БД прецедентів;

3) на рівні серверів БД (DB Server): Precedent DB – це БД прецедентів (MongoDB).

В [16] наведені результати експериментального дослідження впливу роботи БАУ на можливості коригування такого важливого показника якості функціонування системи MARS як роздільна здатність екрану відеокамери на клієнтському застосунку. В середньому, в залежності від рівня завантаженості обчислювальних ресурсів відповідного МП, використання запропонованого підходу дозволило підвищити якість зображень приблизно на 28,2%.

3.4 Функціональний блок “Конфігурування компонентів та генерація коду”

Після проведення етапу архітектурного проектування, як другого з їх послідовності у ЖЦ ПЗ КФС, запропонована ІТ передбачає можливість застосування інтелектуальних засобів для автоматизації процесів конфігурування програмних компонентів та генерування вихідного коду цільової системи. У роботі [17], на основі аналізу можливих підходів до вирішення цих задач для забезпечення варіативності отриманих програмних рішень із урахуванням вимог до якості ПЗ різних груп користувачів та розробників КФС, мотивовано обрано застосування концепції та технології створення предметно-орієнтованих мов програмування (domain-specific language – DSL). Для забезпечення цих процесів у запропонованій ІТ передбачений ФБ A03 (див. схему на рисунку 3), який забезпечує вирішення наступних основних задач (більш детально вони представлені та досліджені у [6,17]):

- *вхідними даними (Input)* для ФБ A03 є компонентна програмна архітектура із відповідними функціональними можливостями її адаптації, яка створюється в результаті роботи ФБ A02 (приклад такої архітектури для систем MARS наведено на рисунку 5);
- *алгоритм управління (Control)* роботою ФБ A03, як і ФБ A02, використовує отриману раніше у ФБ A01 варіабельну FODA (Domain) модель визначеної Про (див. п. 3.2), а також передбачає розробку та застосування відповідної мови DSL;
- *механізм реалізації (Mechanism)* ФБ A03 потребує участі у цьому процесі DevOps фахівців, які застосовують засоби розробки та використання DSL (напр., на мові Python), а також інструментарій для роботи з мовами програмування загального призначення, напр., C++ /C#, Java та ін. (див. більш детально в [17]);
- *результатом роботи (Output)* ФБ A03 є опис альтернативних конфігурацій програмних компонентів КФС на мові DSL, на основі яких генерується їх вихідний код.

Формалізовано мова DSL визначається як кортеж наступного вигляду

$$DSL = \langle FODA (Domain), Grammar (Lexer; Parser), Code_Generator \rangle, \quad (1)$$

де: FODA (Domain) – це модель варіабельності властивостей ПЗ для цільової КФС у певній Про (Domain); Grammar (Lexer, Parser) - це набір моделей, алгоритмів, ресурсів даних і програмних компонентів, які використовуються для реалізації всіх граматичних правил для цього DSL, включаючи лексичний аналіз (Lexer) і синтаксичний аналіз (Parser) для вхідного коду; Code_Generator - це механізм генерації вихідного коду, який перетворює синтаксично та семантично перевірений скрипт на DSL у послідовність інструкцій мови програмування загального призначення.

Важливо підкреслити, що визначення DSL за виразом (1) забезпечує модельно-керований підхід до його розробки, тому що задіяна у ньому модель FODA (Domain) відображає всі структурні та семантичні особливості відповідної Про. У [17] це проілюстровано на прикладі

розробки DSL для ПрО «Розумний будинок», де наведені конкретні фрагменти скриптів на мові DSL та блоки згенерованого вихідного коду на мові C++.

Результати експериментального дослідження [6,17] використання мови DSL для автоматизації конфігурування програмних компонентів системної архітектури та генерації їх вихідного коду у застосунках для ПрО «Розумний будинок» показали, що таким шляхом забезпечується зростання ефективності цих процесів приблизно на 16,8%.

3.5 Функціональний блок “Динамічне налаштування та супровід компонентів ПЗ КФС”

Заключним етапом застосування запропонованої ІТ, який є необхідним у ЖЦ ПЗ КФС, враховуючи зміни у вимогах їх користувачів, є задача динамічного налаштування конфігурацій її програмних компонентів у процесі супроводу відповідної КФС. Для її ефективного вирішення також доцільно застосування інтелектуальних методів і технологій, зокрема, одним з можливих варіантів є використання рекомендаційних систем (recommender system - RS) [18], які забезпечують накопичення та подальше опрацювання консолідованої інформації про стан КФС, що уможливорює динамічне налаштування її функціоналу для різних груп користувачів у відповідності зі змінами в навколишньому середовищі. У контексті цього дослідження формальне визначення RS може бути подано у наступний спосіб [18]:

1) нехай $\underline{U}$ – це є множина користувачів КФС, $\underline{C}$ – це є множина всіх її програмних компонентів, які мають такі функціональні властивості, що відповідають вимогам множини її користувачів $u \subseteq \underline{U}$, а множина $\underline{R} \subseteq \underline{C}$ – це є ранжирований список підмножини таких компонентів, $r \in \underline{R}$, - це є певний компонент у списку $\underline{R}$;

2) результат застосування RS полягає в тому, щоб забезпечити вибір $r \in \underline{R}$ таким чином, аби це максимально повно відповідало вимогам певної групи користувачів: $u \subseteq \underline{U}$;

3) тоді якщо E – це деяка метрика оцінки задоволеності певного користувача ПЗ КФС, $af(r,u)$ це функція оцінки важливості рекомендації окремого компоненту $r \in \underline{R}$ для визначених користувачів $u \subseteq \underline{U}$, то проблема генерації рекомендації в RS може бути подана як вирішення задачі пошуку максимуму цільової функції $f(r,u) = E \rightarrow \max$.

Всі ці завдання у загальній схемі ІТ (див. рисунок 3) вирішує ФБ А04, алгоритм роботи якого спрощено може бути пояснений у наступний спосіб:

- *вхідними даними (Input)* для ФБ А04 є опис альтернативних (мінливих) конфігурацій програмних компонентів КФС на мові DSL, а також масиви контекстних даних, які відображають стан середовища функціонування цієї системи та наявні потреби (або майбутні вподобання) її користувачів;
- *алгоритм управління (Control)* роботою ФБ А04, як і ФБ А02 та ФБ А03, використовує отриману раніше у ФБ А01 варіабельну FODA (Domain) модель визначеної ПрО (див. п. 3.2), а також передбачає застосування методів і технологій RS, таких як, наприклад, спільна фільтрація (collaborative filtering), кластеризація, N-мірна тензорна факторизація та деякі ін. [19];
- *механізм реалізації (Mechanism)* ФБ А04 передбачає участь у цьому процесі фахівців з практик DevOps, які застосовують інструментальні засоби розробки RS, зокрема, проекти з відкритим кодом, такі як *Racoon*, *LensKit for Python*, *CARSKit* (див. більш детально в [18]), крім того, для реалізації розміщення програмних компонентів обраної у такий спосіб конфігурації мають бути застосовані різні технології контейнеризації [4];
- *результатом роботи (Output)* ФБ А04 є обрана конфігурація програмних компонентів

КФС, які відповідають певним критеріям якості функціонування цільової системи.

На рисунку 6 наведена UML діаграма варіантів використання тестової RS, яка була розроблена на основі проекту з відкритим кодом *CARSKit* для забезпечення динамічного налаштування системних конфігурацій у застосунках для ПрО «Розумний будинок».

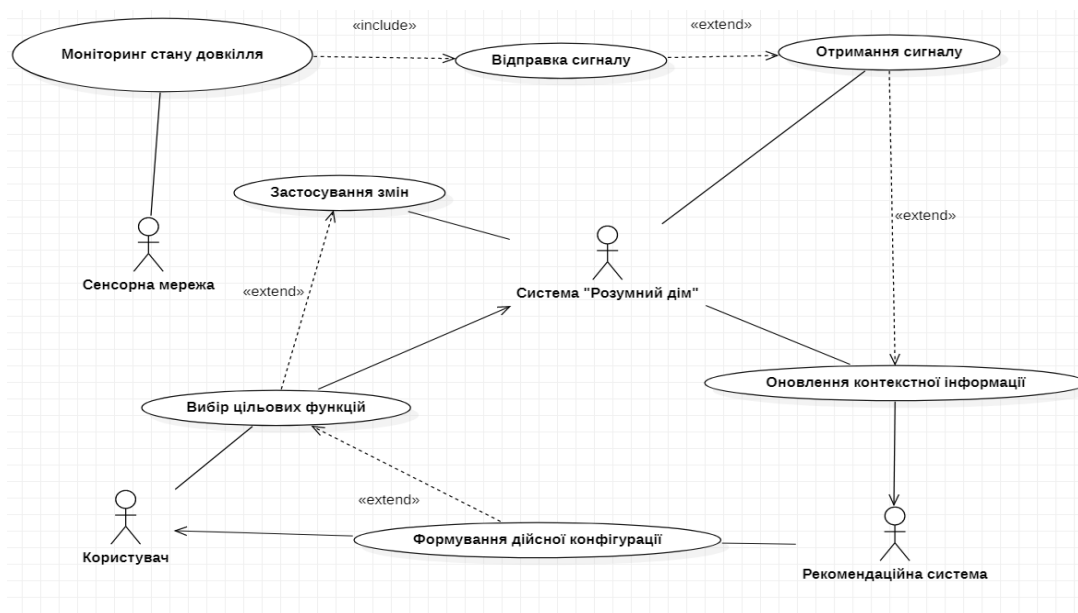


Рис. 6 - Сценарії використання RS для налаштування конфігурацій системи «Розумний будинок»
Fig. 6 - RS usage scenarios for configuring the "Smart Home" system.

Результати експериментального дослідження оцінки точності алгоритмів прогнозування потрібних конфігурацій КФС із застосуванням технологій RS показали [19,20], що таким шляхом забезпечується зростання точності прогнозування приблизно на 24,2%.

3.6 Емпіричний підхід до визначення сумарної оцінки підвищення якості процесів ЖЦ ПЗ КФС у результаті застосування запропонованої ПТ

Для отримання сумарної оцінки підвищення якості процесів ЖЦ КФС, що може бути отримана з використанням розробленої ПТ (див. п. п. 3.1-3.5), пропонується враховувати деякі наявні емпіричні дані про середні питомі проектні витрати на основних етапах ЦЖ ПЗ.

Зокрема, у відомій роботі [21] наведено узагальнені статистичні дані з цих питань, на основі яких можливо визначити відповідні коефіцієнти питомої ваги (важливості) для значень локальних оцінок показників якості процесів на основних етапах ЖЦ ПЗ КФС, а саме:

- для фази специфікації вимог (20% витрат): $K_1 = 0,20$ (2),
- для фази архітектурного проектування (15% витрат): $K_2 = 0,15$ (3),
- для фази розробки (25% витрат): $K_3 = 0,25$ (4),
- для тестування та розгортання (40% витрат): $K_4 = 0,40$ (5),

де сума значень K_{Sum} цих коефіцієнтів (K_1-K_4) дорівнює одиниці: $K_{Sum} = 0,20+0,15+0,25+0,40 = 1,0$.

Тоді, враховуючи значення локальних показників якості процесів окремих етапів ЖЦ ПЗ, які наведені у п. п. 3.2-3.5 (у %), та використовуючи коефіцієнти з виразів (2) – (5), сумарну середню оцінку підвищення якості процесів розробки ПЗ КФС Q_{Sum} можна розрахувати наступним чином:

$$Q_{Sum} = K_1 * 21.1\% + K_2 * 28.3\% + K_3 * 16.8\% + K_4 * 24.2\% = \\ = (0.20 * 21.1 + 0.15 * 28.3 + 0.25 * 16.8 + 0.4 * 24.2) \% = 22.4\% \quad (6)$$

Запропонований підхід є, безперечно, емпіричним за своєю природою, і отримана оцінка (6) суттєво залежить від наявних статистичних даних щодо проектних витрат на всіх основних етапах ЖЦ ПЗ. У якості більш точного механізму для її визначення може бути застосований один з існуючих структурованих методів прийняття багатокритеріальних експертних рішень, наприклад, метод аналізу ієрархій (Analytical Hierarchy Process) [22].

4 Висновки та напрямки подальших досліджень

У цьому дослідженні розглянута актуальна науково-технічна задача розробки інтелектуальної інформаційної технології (ІТ), яка забезпечує наскрізну підтримку властивостей мінливості проектних активів на всіх основних фазах ЖЦ КФС і що, у кінцевому рахунку, дозволяє

підвищити показники якості критично важливих процесів розробки та супроводу таких систем. На основі методологічного узагальнення вже отриманих раніше наукових та практичних результатів, розроблена структурно-функціональна схема ІТ в нотації IDEF0, яка інтегрує окремі знання-орієнтовані моделі, методи та інструментальні засоби для забезпечення властивостей варіабельності, адаптивності, конфігурування та настроюваності проектних рішень та програмних компонентів КФС відповідно на етапах доменного інжинірингу, архітектурного проектування, конструювання коду та супроводу компонентів її ПЗ. Проведені програмні експерименти у реальних проектах з розробки та супроводу застосунків «Розумний будинок» та прототипу мобільної системи доповненої реальності показали можливість та довели доцільність створення наскрізної ІТ, яка підтримує властивості мінливості системних рішень на різних фазах ЖЦ ПЗ КФС і що, в свою чергу, дозволяє підвищити рівень якості таких систем у середньому на 22,4 %.

У якості подальших напрямків досліджень по цій проблематиці можливо запропонувати розробку та експериментальну перевірку можливостей застосування альтернативних модельно-технологічних комплексів для підвищення інтелектуального рівня процесів на всіх основних етапах ЖЦ ПЗ КФС, з метою отримання керованого синергетичного ефекту [1,2] для підвищення якості функціонування таких систем.

СПИСОК ЛІТЕРАТУРИ

1. Lu Chengjie, Pablo Valle, Jiahui Wu et al., “Foundation Models for Software Engineering of Cyber-Physical Systems: The Road Ahead” in: arXiv:2504.04630 [cs.SE], Apr 2025. <https://doi.org/10.48550/arXiv.2504.04630>
2. Amar Banerjee, Venkatesh Choppella, “Control Software Engineering Approaches for Cyber-Physical Systems: A Systematic Mapping Study” *ACM Trans. Cyber-Phys. Syst.*, Vol. 9, Issue 1, Article No.: 6, pp. 1 – 33, 2025. <https://doi.org/10.1145/3704737>
3. Sanghoon Lee, Jiyeong Chae, Haewon Jeon et al., “Cyber-Physical AI: Systematic Research Domain for Integrating AI and Cyber-Physical Systems”, *ACM Trans. Cyber-Phys. Syst.* Vol. 9, Issue 2, Article No.: 19, pp. 1 – 33, 2025. <https://doi.org/10.1145/3721437>
4. Р. О. Гамзаєв, В.Х. Мурадова, М.В. Ткачук, “Дослідження альтернативних технологій контейнерів для віртуалізації процесів розміщення компонентів лінійок програмних продуктів”, *Вісник Харківського національного університету імені В.Н. Каразіна, Серія “Математичне моделювання. Інформаційні технології. Автоматизовані системи управління”*, № 53, 2022, с. 12-20. <https://doi.org/10.26565/2304-6201-2022-53>
5. Kristof Meixner, Kevin Feichtinger, Sayyid Fadhlillah et al., “Variability Modeling of Products, Processes, and Resources in Cyber-Physical Production Systems Engineering”, *Journal of Systems and Software* Vol. 211, 2024. <https://doi.org/10.1016/j.jss.2024.112007>
6. Jacob Krüger, Sebastian Nielebock, Sebastian Krieter et al., “Beyond Software Product Lines: Variability Modeling in Cyber-Physical Systems”, in *Proc. of the 21st International Systems and Software Product Line Conference* – Vol. A, 2018, pp. 237 – 241, <https://doi.org/10.1145/3106195.3106217>
7. Гамзаєв О. Р., Ткачук М. В., “Розробка проблемно-орієнтованої мови моделювання для підтримки варіабельності програмного забезпечення в системах "Розумний будинок". Сучасний стан наукових досліджень та технологій в промисловості. 2023. № 1 (23). С. 45–56. <https://doi.org/10.30837/ITSSI.2023.23.045>
8. Jacky Cao, Kit-Yung Lam, Lik-Hang Lee et al., “Mobile Augmented Reality: User Interfaces, Frameworks, and Intelligence”, *ACM Computing Surveys*, Vol. 55, Issue 9, 2018, pp. 1 – 36, 2023 <https://doi.org/10.1145/3557999>
9. Maurice Khabbaz, Abdellah Chehri, Holger Claussen et al., “Guest Editorial: The Internet of Military Defense Things: State-of-the-Art Challenges, Future Evolution, and Revolutionary Applications”, *IEEE Internet of Things Magazine*, Vol. 8, Issue 2, pp. 14-16, 2025. <https://doi.org/10.1109/MIOT.2025.10907814>
10. М. Tkachuk, A. Vekshin, and R. Gamzayev, “A Model-Based Framework for Adaptive Resource Management in Mobile Augmented Reality System”, in *Proc. of the ICTERI-2016: 12th International Conference on ICT in Education, Research and Industrial Applications: Integration, Harmonization and Knowledge Transfer*, Kyiv, Ukraine, June 21-24, 2016, CEUR-WS.org/Vol-1614, pp.41-56. https://doi.org/10.1007/978-3-319-69965-3_2

11. Adam M. Ross, Donna H. Rhodes, Daniel E. Hastings, "Defining changeability: Reconciling flexibility, adaptability, scalability, modifiability, and robustness for maintaining system lifecycle value", *Systems Engineering*, Vol. 11 (3), pp. 246-262, 2008. <https://doi.org/10.1002/sys.20098>
12. R. O. Gamzayev, "Frame-based Operation Metamodel to Chageability Support in Life Cycle of Software Product Lines", на *XXIII Всеукраїнська науково-технічна конференція молодих вчених, аспірантів та студентів*, Одеса, 20-21 квітня 2023 р. - Одеса, ОНТУ, 2023 р. – с. 215-217.
13. Р. О. Гамзаєв, "Знання-орієнтована інформаційна технологія забезпечення мінливості процесів і компонентів у життєвому циклі кіберфізичних систем", на «*Проблеми інформатики та моделювання (ПІМ-2023)*», *Тези XXIII Міжнародної науково-технічної конференції*, Харків: НТУ "ХПІ", 2023, с. 39.
14. R.O. Gamzayev, M.V. Tkachuk, D.O. Shevkoopias, "Handling of Expert Knowledge in Software Product Lines Development with Usage of Repertory Grids Method // Вісник Харківського національного університету імені В.Н. Каразіна, Серія «Математичне моделювання. Інформаційні технології. Автоматизовані системи управління». - № 47, 2020. – С. 13-24. <https://doi.org/10.26565/2304-6201-2020-47-02>
15. R.O. Gamzayev, M.V. Tkachuk, D.O. Shevkoopias, "Knowledge-oriented Information Technology to Variability Management on Domain Analysis Stage in Software Development", «*Сучасні інформаційні системи*», *наук.-техн. ж-л, НТУ «ХПІ»*, том 2, № 4, 2020. с. 39-47. <https://doi.org/10.20998/2522-9052.2020.4.06>
16. М.В. Ткачук, І.О. Мартінкус, К.А. Нагорний, Р. О. Гамзаєв, "Про один підхід до оцінки ефективності застосування методів доменного моделювання при розробці сімейств програмних систем", *Збірник наукових праць Харківського національного університету Повітряних Сил*, № 5(54), 2017. – С. 127-134. http://nbuv.gov.ua/UJRN/ZKhUPS_2017_5_25
17. Mykola Tkachuk, Oleksii Vekshin, Rustam Gamzayev, "Architecting for Adaptive Resource Management in Mobile Augmented Reality Systems: Models, Metrics and Prototype Software Solutions", in A. Genige et al. (Eds.): *ICTERI 2016: Revised Selected Papers, Series title: Communications in Computer and Information Science*, Vol. 783: Springer-Verlag Berlin Heidelberg, 2017. – pp. 17-35. https://doi.org/10.1007/978-3-319-69965-3_2
18. Rustam Gamzayev, Mykola Tkachuk and Oleksandr Nelipa. "Domain-Specific Language for Adaptive Development of "Smart-Home" Applications", in *Proc. of the 1st International Workshop on Information Technologies: Theoretical and Applied Problems 2021 (ITTAP-2021)*, Ternopil, Ukraine, November 16-18, 2021, CEUR-WS.org/Vol-3039, pp.154-165.
19. Rustam Gamzayev, Mykola Tkachuk, "Dynamic Configuration of Software Products Lines for Smart-Home Applications based on Recommender Systems Framework", на *Інформаційні системи та технології: праці 10-ї Міжнародної науково-технічної конференції*, Харків - Одеса, 13-19 вересня 2021 року / *наук. ред. А. Д. Тевяшев, Л. Б. Петришин, В.В. Бескоровайний, В.Г. Кобзєв.* – ХНУРЕ, 2021, с. 85-89.
20. Р. О. Гамзаєв, М. В. Ткачук, "Застосування методів і технологій рекомендаційних систем для конфігурування динамічних лінійок програмних продуктів", *Вісник Національного технічного університету "ХПІ". Серія: Системний аналіз, управління та інформаційні технології: зб. наук. пр.*, Харків: НТУ "ХПІ", № 1 (5), 2021, с. 91-97. <https://doi.org/10.20998/2079-0023.2021.01.15>
21. Ian Sommerville, *Software engineering* (10th edition, Global Edition), Pearson Education, 2016.
22. Thomas L. Saaty, Luis G. Vargas, *Models, Methods, Concepts & Applications of the Analytic Hierarchy Process*, 2nd Edition, Springer Science+Business Media, New York, 2012.

REFERENCES

1. Chengjie Lu, Pablo Valle, JiahuiWu et al., "Foundation Models for Software Engineering of Cyber-Physical Systems: The Road Ahead" In: arXiv:2504.04630 [cs.SE], Apr 2025. <https://doi.org/10.48550/arXiv.2504.04630>
2. Amar Banerjee, Venkatesh Choppella, "Control Software Engineering Approaches for Cyber-Physical Systems: A Systematic Mapping Study", *ACM Trans. Cyber-Phys. Syst.*, Vol. 9, Issue 1 Article No.: 6, pp. 1 – 33, Jan 2025. <https://doi.org/10.1145/3704737>
3. Sanghoon Lee, Jiyeong Chae, Haewon Jeon et al., "Cyber-Physical AI: Systematic Research Domain for Integrating AI and Cyber-Physical Systems", *ACM Trans. Cyber-Phys. Syst.* Vol. 9, Issue 2, (April 19 2025), pp. 1 – 33. <https://doi.org/10.1145/3721437>

4. R.A. Gamzayev, V.Kh. Muradova, M.V. Tkachuk. "A study on alternative container-based technologies for virtualization of components deployment in software product lines", *Bulletin of KhNU by V.N. Karazin, ser. "Mathematical modeling. Information Technology. Automated control systems"*, No. 53, 2022, pp. 12-20. [in Ukrainian] <https://doi.org/10.26565/2304-6201-2022-53>
5. Kristof Meixner, Kevin Feichtinger, Sayyid Fadhllillah et al., "Variability Modeling of Products, Processes, and Resources in Cyber-Physical Production Systems Engineering", *Journal of Systems and Software*, Vol. 211, 2024. <https://doi.org/10.1016/j.jss.2024.112007>
6. Jacob Krüger, Sebastian Nielebock, Sebastian Krieter et al., "Beyond Software Product Lines: Variability Modeling in Cyber-Physical Systems", in *Proc. of the 21st International Systems and Software Product Line Conference* – Vol. A, 2018, pp. 237 – 241. <https://doi.org/10.1145/3106195.3106217>
7. Rustam Gamzayev, Mykola Tkachuk, "Development of problem-specific modeling language to support software variability in "Smart home" systems", *Innovative Technologies and Scientific Solutions for Industries*, 2023, No. 1 (23), P. 45-56. <https://doi.org/10.30837/ITSSI.2023.23.045>
8. Jacky Cao, Kit-Yung Lam, Lik-Hang Lee et al., "Mobile Augmented Reality: User Interfaces, Frameworks, and Intelligence", *ACM Computing Surveys*, Vol. 55, Issue 9, 2018, pp. 1 – 36. <https://doi.org/10.1145/3557999>
9. Maurice Khabbaz, Abdellah Chehri, Holger Claussen et al., "Guest Editorial: The Internet of Military Defense Things: State-of-the-Art Challenges, Future Evolution, and Revolutionary Applications", *IEEE Internet of Things Magazine*, Vol. 8, Issue 2, 2025, pp. 14-16. <https://doi.org/10.1109/MIOT.2025.10907814>
10. M. Tkachuk, A. Vekshin, and R. Gamzayev, "A Model-Based Framework for Adaptive Resource Management in Mobile Augmented Reality System", in *Proc. of the ICTERI-2016: 12th International Conference on ICT in Education, Research and Industrial Applications: Integration, Harmonization and Knowledge Transfer*, Kyiv, Ukraine, June 21-24, 2016, CEUR-WS.org/Vol-1614, pp.41-56. https://doi.org/10.1007/978-3-319-69965-3_2
11. Adam M. Ross, Donna H. Rhodes, Daniel E. Hastings, "Defining changeability: Reconciling flexibility, adaptability, scalability, modifiability, and robustness for maintaining system lifecycle value", *Systems Engineering*, Vol. 11 (3), pp. 246-262, 2008. <https://doi.org/10.1002/sys.20098>
12. R. O. Gamzayev, "Frame-based Operation Metamodel to Chageability Support in Life Cycle of Software Product Lines", in *Abstracts of the XXIII All-Ukrainian Scientific and Technical Conference of Young Scientists, Postgraduates and Students*, Odessa, 20-21 April 2023, ONTU, pp. 215-217.
13. R. O. Gamzayev, "Knowledge-oriented information technology to ensure a changeability on processes and components in life cycle of cyber-physical systems", *Problems of Informatics and Modeling (PIM-2023). Abstracts of the XXIII International Scientific and Technical Conference*, Kharkiv: NTU "KhPI", 2023. – P. 129.
14. R.O. Gamzayev, M.V. Tkachuk, D.O. Shevkoplias, "Handling of Expert Knowledge in Software Product Lines Development with Usage of Repertory Grids Method", *Bulletin of KhNU by V.N. Karazin, ser. "Mathematical modeling. Information Technology. Automated control systems"*, - No. 57, 2020. – pp. 13-24. <https://doi.org/10.26565/2304-6201-2020-47-02>
15. R.O. Gamzayev, M.V. Tkachuk, D.O. Shevkoplias, "Knowledge-oriented Information Technology to Variability Management on Domain Analysis Stage in Software Development", *Advanced Information Systems*, 2020. Vol. 4, No. 4, pp. 39-47. <https://doi.org/10.20998/2522-9052.2020.4.06>
16. M.V. Tkachuk, I.O. Martinkus, K.A. Nagornyi, R.A. Gamzayev, "Towards the effectiveness assessment approach to domain modeling methods application in software product family development", *Collection of scientific papers of the Kharkiv National University of the Air Force*, No. 5(54), 2017. – P. 127-134. [in Ukrainian] http://nbuv.gov.ua/UJRN/ZKhUPS_2017_5_25
17. Mykola Tkachuk, Oleksii Vekshin, Rustam Gamzayev, "Architecting for Adaptive Resource Management in Mobile Augmented Reality Systems: Models, Metrics and Prototype Software Solutions", in A. Genige et al. (Eds.): *ICTERI 2016: Revised Selected Papers, Series title: Communications in Computer and Information Science*, Vol. 783: Springer-Verlag Berlin Heidelberg, 2017. – pp. 17-35. https://doi.org/10.1007/978-3-319-69965-3_2
18. Rustam Gamzayev, Mykola Tkachuk and Oleksandr Nelipa, "Domain-Specific Language for Adaptive Development of "Smart-Home" Applications", in *Proc. of the 1st International Workshop*

on Information Technologies: Theoretical and Applied Problems 2021 (ITTAP-2021), Ternopil, Ukraine, November 16-18, 2021, CEUR-WS.org/Vol-3039, pp.154-165.

19. Rustam Gamzayev, Mykola Tkachuk. "Dynamic Configuration of Software Products Lines for Smart-Home Applications based on Recommender Systems Framework", in *Information systems and technologies: Proceedings of the 10-th International Scientific and Technical Conference*, September 13-19, 2021 Kharkiv - Odesa, Ukraine, scientific editors: A.D. Tevyashev, L.B. Petryshyn, V.V. Bezkorovainy, V.G. Kobzev. – KhNURE, 2021. - pp. 85-89.
20. Rustam Gamzayev, Mykola Tkachuk. "Using Methods and Technologies of Recommendation Systems for Dynamic Software Product Lines Configuration", *Bulletin of National Technical University "KhPI". Series: System Analysis, Control, and Information Technologies*, No. 1 (5), 2021, pp. 91-97. [in Ukrainian] <https://doi.org/10.20998/2079-0023.2021.01.15>
21. Ian Sommerville, Software engineering (10th edition, Global Edition), Pearson Education, 2016.
22. Thomas L. Saaty, Luis G. Vargas, Models, Methods, Concepts & Applications of the Analytic Hierarchy Process, 2nd Edition, Springer Science+Business Media, New York, 2012.

Gamzayev Rustam PhD, Associate Professor; Associate Professor of the Department of Intelligent Software Systems and Technologies, Education and Research Institute of Computer Sciences and Artificial Intelligence, V. N. Karazin Kharkiv National University, 4 Svobody Sq., Kharkiv, 61022, Ukraine

Intelligent information technology to support changeability in software life cycle processes of cyber-physical systems

Actuality. The development of software for cyber-physical systems (CPS) should take into account the specific features of their construction and operation, which supposes to support a changeability of project assets and system solutions at all main stages of their life cycle (LC). Solving these problems is impossible without the usage of intellectual methods and tools, and therefore the topic of this study is the actual scientific and technical task.

Goal. The aim of this work is to develop the intelligent information technology (IIT) that provides end-to-end support for the changeability of project assets in all major phases of the LC CFS, which finally has to improve the quality indicators of critically important development and maintenance processes of such systems.

Research methods. Based on a critical analysis and methodological generalization of some previously obtained scientific and practical results, the structural and functional scheme of IIT has been developed, which integrates some knowledge-oriented model-technological tools, which allows to ensure support for the properties of variability, adaptability, configuration and adjustability of design solutions and software components of the CFS at their LC stages such as domain engineering, architectural design, code construction, and maintenance of their software components.

Results. Using the examples of Smart home systems and mobile augmented reality systems development, some essential features of the CFS's construction and operation have been studied, and the methodological basis for a knowledge-oriented software development of such systems has been formed. The generalized IIT scheme in IDEF0 notation has been proposed, its main functional blocks have been defined, software experiments have been conducted, and quantitative metrics are calculated, that shows the total increase in the quality indicators of software development and maintenance processes by approx. 22,4%.

Conclusions. The presented studies confirmed the feasibility of using knowledge-oriented models, methods and tools for the development and maintenance of CFS software, and the possibility to design the end-to-end intelligent information technology, which supports the properties of changeability of project assets and system solutions at the main phases of the LC CFS, that, in turn, allows to significantly processes quality improving in creation of such systems.

Keywords: intelligent models and methods, cyber-physical system, software, life cycle, information technology, changeability, variability, adaptability, configuration, customization, metric, quality