

ПЕНЯ ОЛЕКСАНДР

Національний технічний університет України «Київський політехнічний інститут імені Ігоря Сікорського»

e-mail: [penya.s@ukr.net](mailto:penya.s@ukr.net)

СУЛЕМА ЄВГЕНІЯ

Національний технічний університет України «Київський політехнічний інститут імені Ігоря Сікорського»

<https://orcid.org/0000-0001-7871-9806>e-mail: [sulema@pzks.fpm.kpi.ua](mailto:sulema@pzks.fpm.kpi.ua)

## ПРОЄКТУВАННЯ АРХІТЕКТУРИ ПРОГРАМНОЇ СИСТЕМИ ЦИФРОВОГО ДВІЙНИКА З НЕВИЗНАЧЕНІСТЮ ДАНИХ

*В роботі запропоновано універсальну архітектуру програмної системи цифрових двійників з різними джерелами невизначеності в даних. Запропонована архітектура використовує сервіс-орієнтований підхід для забезпечення незалежності модулів та можливості масштабування. Розглянуто компоненти системи, їх особливості та доцільні технології реалізації.*

*Ключові слова: архітектура програмної системи, цифрові двійники, невизначеність.*

PENIA OLEKSANDR, SULEMA YEVGENIYA

National technical university of Ukraine "Ihor Sikorsky Kyiv polytechnic institute"

### ARCHITECTURE DESIGN OF DIGITAL TWIN SOFTWARE SYSTEM WITH DATA UNCERTAINTY

*Modern technologies have greatly increased data availability through higher data transferring capacity (mainly speed and accessibility of computer networks) as well as the possible sources of data like relatively inexpensive sensors and controllers, wearable electronics, smart appliances and machines, both household and industrial. This in turn leads to greater interest and feasibility of digital twin technologies. However, in a complex distributed and mixed-computing environment where such systems operate, data is not guaranteed to be certain. Both technical difficulties and vague inputs by design, such as operators' subjective estimates, are sources of uncertainty in data and digital twin software systems should account for that in order to provide more accurate representation of and insights into real object or process. Current publications focus more on the general concepts and classification of uncertainty in digital twins, while research into practical aspects of software design is lacking. As such, this paper presents a general architecture of a digital twin software system with data uncertainty. System's inputs are comprised of manual user inputs through the UI, direct sensor data about the object under consideration and intermediary results from on-edge computing. System's outputs are modeling, projection and visualization data, provided to users via the UI and control signals onto actuators that influence the real objects. The system itself consists of six modules that are loosely coupled in a service-oriented architecture design to allow for independent change, utilization of the most appropriate technologies and scalability. These are graphical user interface (GUI), authorization and authentication service, preprocessing module, uncertain data processing module, modeling and data analysis module and persistent data storage. The paper provides an overview for each of the modules, their responsibilities and considerations on implementation, such as applicable technologies and design patterns.*

*Keyword: software system architecture, digital twin, uncertainty.*

### Постановка проблеми

Сучасні технології отримання та передачі даних дозволяють отримати як ніколи багато інформації про досліджувані об'єкти. Смартфони, IP-камери, спеціалізовані сенсори, розумні пристрої є доступними і поширеними засобами отримання великої кількості гетерогенних мультимодальних даних, а майже повсюдна наявність комп'ютерних мереж, в тому числі широкосмугового кабельного інтернету та супутникового і мобільного підключення дозволяє ще і отримувати такі дані у великих обсягах у режимі реального часу або близькому до нього. Завдяки цьому як методи створення, так і області застосування технології цифрових двійників значно розвинулись. Втім, навіть із сучасними засобами, отримання даних неперервно і точно не є гарантованим. Через можливі збої в передачі сигналів або отриманню даних, які можуть містити умисну неточність, такі як суб'єктивні оцінки користувачів, постає необхідність розширення досліджень методів створення цифрових двійників, і програмного забезпечення що їх реалізує, які можуть обробляти такі випадки.

### Аналіз попередніх досліджень

Нещодавні дослідження також приділяють увагу проблемам неточності в контексті цифрових двійників. Так, наприклад, в [1] наведено детальний огляд самого поняття невизначеності в концепції цифрових двійників, її джерел, впливу, методів оцінки та оптимізації і т.п. Автори [2] також розглядають категорії і вплив невизначеності даних в моделях цифрових двійників і наводять огляд методів і підходів до врахування невизначеності в побудові та візуалізації цифрових двійників. Робота [3] зосереджена на дослідженнях в науково-практичній області оптимізації міцності конструкцій для аерокосмічних місій з невизначеністю.

Розглянуті дослідження враховують проблему невизначеності в цифрових двійниках, але не пропонують універсальних підходів проектування програмного забезпечення цифрових двійників з врахуванням невизначеності. Відповідно, існує задача створення такої архітектури для розроблення програмного забезпечення, яке враховує та зменшує вплив невизначеності під час конструювання та роботи з цифровим двійником досліджуваного об'єкта.

### Виклад матеріалу дослідження

Проектування архітектури слід розпочати з визначення особливостей та вимог до системи програмного забезпечення цифрових двійників із невизначеністю. Такі системи приймають темпоральні

мультимодальні дані як із автоматичних джерел, так і з введення користувачів. Отримані дані мають бути синхронізовані, агреговані та поєднані для використання у моделі цифрового двійника. Також, необхідно забезпечити довготривале збереження отриманих даних і ефективний доступ до них. Виведення таких систем потребує засобів візуалізації досліджуваного об'єкта і доцільних засобів аналітики. Також, системи цифрових двійників можуть працювати у різних режимах:

– Відокремлене моделювання для оптимізації вхідних параметрів. В такому режимі вхідні дані із допустимого інтервалу задаються аналітиком, а програмна модель передбачає перебіг процесів та результати з метою вибору кращого режиму роботи та кінцевого результату.

– Моніторингова система у реальному часі. Режим подібної програмної системи за замовчанням: вхідні дані отримуються з вимірів реального об'єкта/процесу, програмне забезпечення ж відображає поточну ситуацію у зручній і зрозумілій формі і надає зведену аналітичну інформацію і, можливо, проекцію на короткочасне майбутнє з попередженням про можливі аномальні стани.

– Система контролю. Через цифровий двійник також можливе управління реальним об'єктом через актуатори, що супроводжують систему цифрового двійника. Такий режим здебільшого актуальний для автоматичних або автоматизованих виробничих і промислових систем, оскільки для інших застосувань безпосереднє управління об'єктом може бути неможливо, наприклад, для досліджуваних медико-біологічних об'єктів, або потребувати спеціального режиму доступу. [4]

Таким чином, архітектура розробленої програмної системи цифрових двійників повинна володіти такими характеристиками:

Виокремлення та ізоляція компонентів. Система повинна складатись із слабо пов'язаних між собою елементів щоб забезпечити можливість їх незалежної модифікації, конфігурації та оновлення. Також, це допоможе забезпечити масштабованість системи, оскільки навантаження на неї можуть зростати завдяки доступності величезної кількості вхідних даних і врахувати її розподілену природу, яка може передбачати різні методи введення даних, різні платформи їх обробки (on-edge, централізована, розподілена) та їх комбінації.

Довготривале зберігання темпоральних мультимодальних даних. Структура даних, що надходять до системи характеризується тим, що всі знаходження асоціюються з відповідною часовою міткою і можуть мати несталу структуру. Для ефективної роботи з такими даними необхідно передбачити відповідне сховище і методи доступу до таких даних, що можуть швидко доступатись до довільних елементів інформації за часовим проміжком. Зберігання ж має забезпечувати узгодженість і тривалість отриманих даних протягом довільного часового проміжку.

Універсальність архітектури. Розроблювана програмна система має бути застосовна до широкого ряду різноманітних об'єктів і процесів, тому архітектура повинна забезпечити високий рівень абстракції від конкретних деталей реалізації, але при цьому не обмежувати розробника у впровадженні таких деталей для специфічної прикладної задачі.

Врахування невизначеності. Отримувані дані можуть містити певну міру невизначеності в силу перешкод в отриманні сигналів, суттєво різній частоті отримання вимірів або запланованій суб'єктивності отримуваних даних. Система повинна реалізовувати методи обробки таких випадків, наприклад, методами інтерполяції чи авторегресії.

Ролі користувачів. Система передбачає роботу з різними типами користувачів, які мають різні потреби та, відповідно, різні можливості для взаємодії з програмним забезпеченням. Тож слід забезпечити підсистему авторизації та управління ролями користувачів, які визначають їх права та доступну функціональність.

Адаптивний аналіз даних. Досліджувані за допомогою технології цифрових двійників рідко залишаються незмінними з часом. Відповідно, програмна система повинна надавати механізми адаптації до змін поведінки реального об'єкта. Окрім цього, важливою функцією повноцінного цифрового двійника є виявлення закономірностей і зв'язків в отриманих даних, які можуть проявлятись протягом значного періоду часу (довготривала сезонність).

З урахуванням окреслених вимог, розглянемо високорівневе представлення архітектури програмної системи цифрового двійника та особливості її компонентів. Основні компоненти наведено на рис. 1.

Для забезпечення модульності і масштабованості система має використовувати сервіс-орієнтований підхід, який дозволяє підтримувати розподілену систему сенсорів, актуаторів, проміжних вузлів контролю та обробки. [5] Також, можливо розгортання надлишкових вузлів для розподілення навантаження та використання IaaS-платформ, таких як хмарні сервіси. Відповідно, комунікація між сервісами повинна відбуватись з використанням API-викликів і відповідних засобів, серед яких можливе використання:

– REST API - простий в реалізації і підтримці механізм комунікації сервісів по протоколу HTTP/HTTPS.

– WebSocket - розширення протоколу HTTP для підтримання постійного і швидкісного з'єднання у мережі, більш вимогливий до мережі та потребує більше інфраструктурної підтримки в програмних системах.

– RPC - віддалений виклик процедур, найбільш ефективний з точки зору швидкодії, але потребує значних зусиль в реалізації, підтримці та відлагодженні в програмних системах. [7]

Вибір методу комунікації між сервісами залежить від вимог до системи, очікуваного навантаження та процесі розробки. Значною перевагою сервіс-орієнтованої архітектури є можливість використання різних підходящих засобів в окремих компонентах, в тому числі і методів комунікації. Наприклад, зовнішні API введення-виведення даних доцільно використати REST через простоту підключення зовнішніх вузлів комунікації, в той час як в середині системи використати gRPC для підвищення швидкодії.

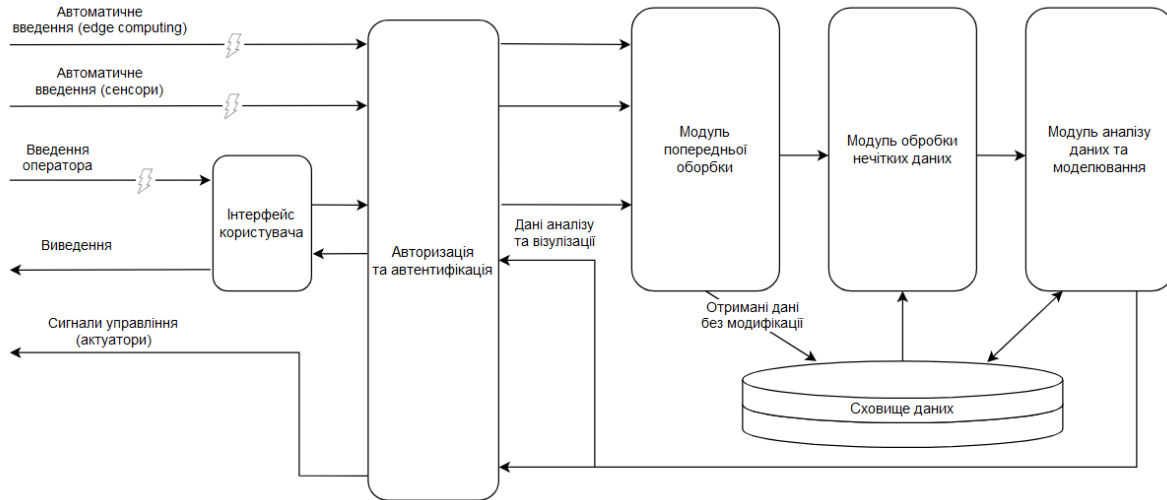


Рис. 1. Схема архітектури програмної системи цифрового двійника з невизначеністю

В дизайні окремих сервісів так само присутня свобода вибору технологій і алгоритмів. Інкапсуляція через відкритий (в контексті системи, не обов'язково відкритий для зовнішнього доступу) API зводить функціонування сервісів до “чорного ящика”, внутрішні особливості роботи якого недоступні і не потрібні їх користувачам.

Розглянемо ролі і особливості компонентів системи більш детально. Інтерфейс користувача надає зручний спосіб взаємодії із системою. Функціональність і можливості інтерфейсу залежить від ролі користувача, отриманої після автентифікації. Враховуючи розподілену природу систем цифрових двійників, інтерфейс має бути доступний на різних платформах. Веб-інтерфейс із використанням UI-фреймворків (наприклад, React) дозволяє створювати зручні і функціональні веб-застосунки, добре інтегрується із сервісним підходом, а також може бути запакований у вигляді мобільного або десктопного застосунку. Сервіс-орієнтована архітектура дозволяє за необхідністю розробку спеціалізованих інтерфейсів з використанням інших технологій, але це накладає додаткові витрати ресурсів на розроблення та підтримку, тому їх використання виправдане лише у випадку крайньої необхідності в контексті проекту.

Модуль авторизації і автентифікації необхідний для підтвердження прав доступу до ресурсів системи. Будучи розподіленою мережевою системою, вона повинна надавати відкритий доступ до точок доступу API, але взаємодія має бути доступна лише авторизованим джерелам. Слід зауважити, що не лише вхідні запити мають пройти авторизацію, а й вихідні, оскільки сервіси обробки мають підтвердити свою ідентичність для отримання доступу до підсистем управління реальним об'єктом і надання результатів аналізу для уникнення атак типу man-in-the-middle. Забезпечити механізми автентифікації і підтвердження ідентичності можна за рахунок комбінації сервісу авторизації, який видає тимчасові токени доступу і засобів асиметричної криптографії, які за принципом дії схожі на електронний цифровий підпис.

Наступні компоненти складають безпосереднє ядро системи цифрового двійника. Сховище даних - сервіс, який надає доступ до даних цифрового двійника. Досі найбільш популярний вибір для забезпечення довготривалого збереження даних є реляційні бази даних. Втім, враховуючи сформульовані вимоги до сховища, для даної системи більш актуальним буде використання NoSQL бази даних, наприклад, можна використати Amazon DynamoDB з її можливостями автоматичного масштабування та майже постійного часу доступу до даних, незалежно від їх обсягу (звісно, з адаптованим під запити дизайном схеми ключів та індексів). Також, можливе застосування спеціалізованих СУБД для роботи з часовими рядами (TSDB), таких як TimescaleDB або InfluxDB. [6] Можливості таких сервісів мають бути достатніми для більшості випадків завдяки можливостям шардінгу та довільного горизонтального масштабування. У рідкому випадку коли це не так і потім даних в системі перевищує навіть можливості спеціалізованих NoSQL сервісів необхідне спеціалізоване рішення, яке використовує технології Big Data, наприклад, Data Lake.

Модуль попередньої обробки даних є вхідною точкою ядра цифрового двійника і забезпечує, в першу чергу, збереження отриманих даних в сховищі. Інші процедури обробки залежать від конкретного контексту, але можуть бути реалізовані за шаблоном Filters and Pipes для побудови складеної багатоетапної процедури попередньої обробки. Додатковою можливістю є створення параметризованих правил попередньої обробки і динамічне створення конвеєра з цих правил (рис. 2), наприклад, реалізуючи шаблон будівельник. Так можна забезпечити гнучкість системи та універсальність, а отже і повторне використання, блоку попередньої обробки.

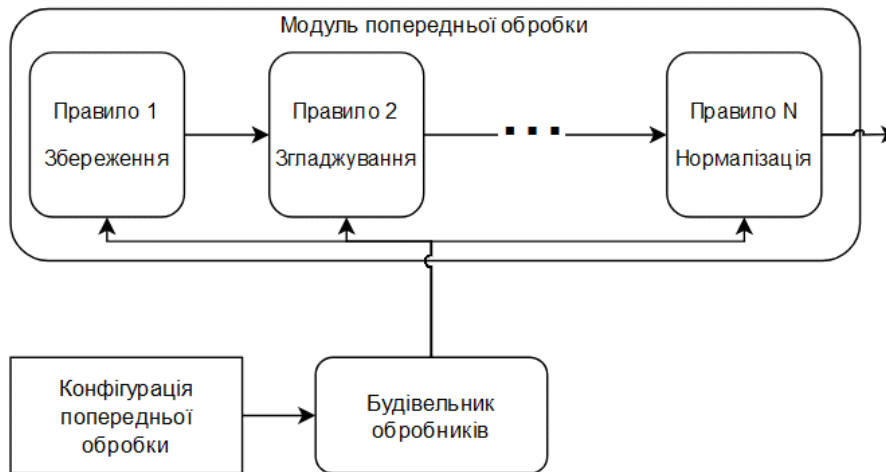


Рис. 2. Конструювання модуля попередньої обробки за конфігурацією

Сам же модуль попередньої обробки є заступником (Proxy) потоку даних реального об'єкта для подальших сервісів системи.

Модуль обробки нечітких даних відповідає за оброблення невизначеності в даних та поєднання інформації. Методи реалізації даної відповідальності визначаються задачею та контекстом побудови відповідної системи цифрового двійника. Алгоритми, які використовуються для усунення нестачі даних можуть включати інтерполяцію, методи групи авторегресії і плаваючого середнього (ARIMA і модифікації), обчислення розподілу, тощо [9]. Для обробки невизначеності можливе використання методів кодування або відновлення нечітких значень. [8]

Дизайн та алгоритмічне забезпечення модуля аналізу та моделювання так само визначається специфікою задачі. Він і забезпечує основну функціональність цифрового двійника - інтелектуальний аналіз даних - і моделювання поведінки реального об'єкта. Різноманітні режими роботи системи цифрового двійника, які обговорювались раніше, забезпечуються перенаправленням потоків введення даних та незначною зміною конфігурації сервісів.

### Висновки

В даній статті запропоновано універсальну архітектуру програмної системи цифрових двійників з невизначеністю. Вона слідує сервіс-орієнтованому підходу для забезпечення масштабованості та модульності системи і складається із інтерфейсу користувача, модулів авторизації та автентифікації, попередньої обробки, узгодження та поєднання даних, аналізу даних і моделювання та сховища темпоральних даних. Розглянуто особливості проектування і реалізації цих модулів, можливі проблеми в їх використанні та доцільні технології для їх вирішення, можливості алгоритмічного забезпечення модулів аналізу даних системи цифрових двійників.

### Література

1. Thelen A. et al. A comprehensive review of digital twin. part 2: roles of uncertainty quantification and optimization, a battery digital twin, and perspectives //Structural and multidisciplinary optimization. 2023. Т. 66. №. 1. P. 1.
2. Ríos J. et al. Uncertainty of data and the digital twin: a review. International Journal of Product Lifecycle Management. 2020. Т. 12. №. 4. P. 329-358.
3. Karve P. M. et al. Digital twin approach for damage-tolerant mission planning under uncertainty. Engineering Fracture Mechanics. 2020. Т. 225. P. 106766.
4. Kritzinger W. et al. Digital Twin in manufacturing: A categorical literature review and classification. Ifac-PapersOnline. 2018. Т. 51. №. 11. P. 1016-1022.
5. Laskey K. B., Laskey K. Service oriented architecture. Wiley Interdisciplinary Reviews: Computational Statistics. 2009. Т. 1. №. 1. P. 101-105.
6. Naqvi S. N. Z., Yfantidou S., Zimányi E. Time series databases and influxdb. Studienarbeit, Université Libre de Bruxelles. 2017. Т. 12. P. 1-44.
7. Feng X., Shen J., Fan Y. REST: An alternative to RPC for Web services architecture. 2009 First International Conference on future information networks. IEEE, 2009. P. 7-10.
8. Saade J. J., Diab H. B. Defuzzification methods and new techniques for fuzzy controllers. 2004.
9. Makridakis S. Time series prediction: Forecasting the future and understanding the past. International Journal of Forecasting. 1994. Т. 10. №. 3. P. 463-466.