



УДК 004.4:004.8:65.012.45

<https://doi.org/10.31713/vf220261>

Єрошкін Ю. М. [1; ORCID ID: 0009-0009-1018-5487],
к.т.н., асистент,
Волощук В. А. [1; ORCID ID: 0000-0003-0687-8968],
д.т.н., професор
Єрошкін О. Ю. [2; ORCID ID: 0009-0006-3976-7526],
магістр, асистент,

¹Національний технічний університет України «Київський політехнічний університет імені Ігоря Сікорського», м. Київ

²Вроцлавський університет науки і техніки, м. Вроцлав, Польща

АРХІТЕКТУРНИЙ ПІДХІД ДО МІНІМІЗАЦІЇ МОДИФІКАЦІЙ ЯДРА КОРПОРАТИВНИХ ІНФОРМАЦІЙНИХ СИСТЕМ

У статті досліджено архітектурний підхід до мінімізації модифікацій ядра корпоративних інформаційних систем, що експлуатуються протягом тривалого часу та містять значну кількість користувацьких розширень. Запропонований підхід ґрунтується на винесенні бізнес-логіки за межі ядра системи з використанням хмарних платформ і сервісно-орієнтованої інтеграції. Апробацію архітектурного рішення виконано на прикладі корпоративних систем управління виробничими та бізнес-процесами, які широко застосовуються в енергетиці та на підприємствах критичної інфраструктури.

Показано можливість адаптації стратегії “чистого ядра” (Clean Core) до систем попереднього покоління, що дозволяє зберегти стабільність функціонування та водночас підготувати інформаційну систему до подальшої модернізації.

Ключові слова: корпоративні інформаційні системи; архітектура програмного забезпечення; хмарні платформи; сервісно-орієнтована інтеграція; SAP Business Technology Platform; Clean Core.

Постановка проблеми. Корпоративні інформаційні системи, що тривалий час експлуатуються на підприємствах енергетичної галузі та інших об'єктах критичної інфраструктури, характеризуються значною кількістю модифікацій ядра, накопичених у процесі

адаптації до специфічних бізнес-процесів. Такі модифікації, хоча й забезпечують функціональну гнучкість на ранніх етапах експлуатації, з часом призводять до зростання технічного боргу, ускладнення супроводу системи та підвищення ризиків під час її оновлення або модернізації.

Перехід до корпоративних інформаційних систем нового покоління вимагає перегляду підходів до розширення функціональності та інтеграції. Як показують сучасні дослідження архітектури ERP-систем, стратегічний тренд полягає у переході до хмарних архітектур із використанням сервісних моделей, що підвищує стійкість та гнучкість систем [1; 2]. Традиційні методи розвитку систем шляхом прямого втручання в стандартний код ядра виявляються малоефективними, оскільки суттєво ускладнюють процеси міграції, тестування та подальшого розвитку. Водночас повна заміна існуючої системи або її радикальна перебудова є економічно затратною та організаційно ризикованою, що особливо критично для підприємств із безперервними виробничими процесами.

Практика експлуатації корпоративних ERP-систем із тривалим життєвим циклом у критично важливих галузях демонструє, що кумулятивні модифікації ядра можуть призводити до порушень цілісності системи, зростання тривалості релізів і неможливості своєчасного встановлення пакетів підтримки. Тому завдання мінімізації втручань у стандартне ядро є ключовим для забезпечення надійності функціонування та стабільного виконання бізнес-процесів.

У зв'язку з цим актуальною є проблема пошуку архітектурних рішень, які дозволяють мінімізувати модифікації ядра корпоративних інформаційних систем без втрати необхідної функціональності та зберегти стабільність їх експлуатації. Особливої уваги потребує адаптація сучасних архітектурних стратегій, орієнтованих на системи нового покоління, до умов експлуатації систем, що вже перебувають у продуктивному використанні. Вирішення цієї проблеми є ключовим чинником забезпечення поступової, керованої та безболісної еволюції корпоративних інформаційних систем у напрямі сучасних технологічних платформ.

Метою дослідження є обґрунтування та опис архітектурного підходу до мінімізації модифікацій ядра корпоративних інформаційних систем, що тривалий час перебувають в експлуатації, шляхом винесення прикладної бізнес-логіки за межі ядра з



використанням хмарних платформ і сервісно-орієнтованої інтеграції. Реалізація запропонованого підходу спрямована на забезпечення стабільної роботи існуючих систем і створення передумов для їх поступової та керованої модернізації з орієнтацією на інформаційні системи нового покоління.

Виклад основного матеріалу. Методологічною основою дослідження є стратегія “чистого ядра” (Clean Core), запропонована компанією SAP SE для сучасних ERP-систем, зокрема SAP S/4HANA [3; 4]. Основна ідея цієї стратегії полягає у збереженні стандартного ядра системи незмінним та реалізації прикладної бізнес-логіки у вигляді зовнішніх розширень, що дозволяє зменшити технічний борг і спростити подальші оновлення.

Водночас значна кількість підприємств, у тому числі в енергетичній галузі та на об'єктах критичної інфраструктури, продовжують експлуатувати ERP-системи попереднього покоління, зокрема SAP ECC 6.0. Хоча стратегія Clean Core формально не була розроблена для таких систем, її базові архітектурні принципи можуть бути адаптовані до гібридних сценаріїв за умови використання хмарних платформ розширення. Така адаптація відповідає загальним тенденціям модернізації ERP-систем у напрямі гібридних архітектур і міграції бізнес-функціональності в хмарні сервіси [1; 2].

Архітектурний дизайн застосування цього підходу для SAP ECC 6.0 наведено на рис. 1.

У запропонованій архітектурі ядро системи залишається у середовищі SAP ECC, де зберігаються бізнес-дані та виконуються стандартні транзакційні процеси, тоді як прикладна бізнес-логіка реалізується у хмарному середовищі SAP Business Technology Platform (BTP), зокрема в ABAP Environment [5]. Взаємодія між хмарними застосунками та ERP-системою здійснюється через сервісно-орієнтовану інтеграцію з використанням протоколу OData версії 2 [6].

Ключовим елементом реалізації цього патерну є Service Consumption Model (SRVC), яка використовується для формалізованого споживання сервісів у хмарному ABAP-середовищі. SRVC забезпечує генерацію проксі-класів, типів даних і шаблонів виклику сервісів, що є необхідним для побудови застосунків на основі моделі RESTful Application Programming (RAP). Передача запитів між SAP BTP та SAP ECC відбувається по захищених каналах із використанням сервісів Destination, Connectivity та Cloud Connector, що унеможливорює пряме втручання в ядро системи.

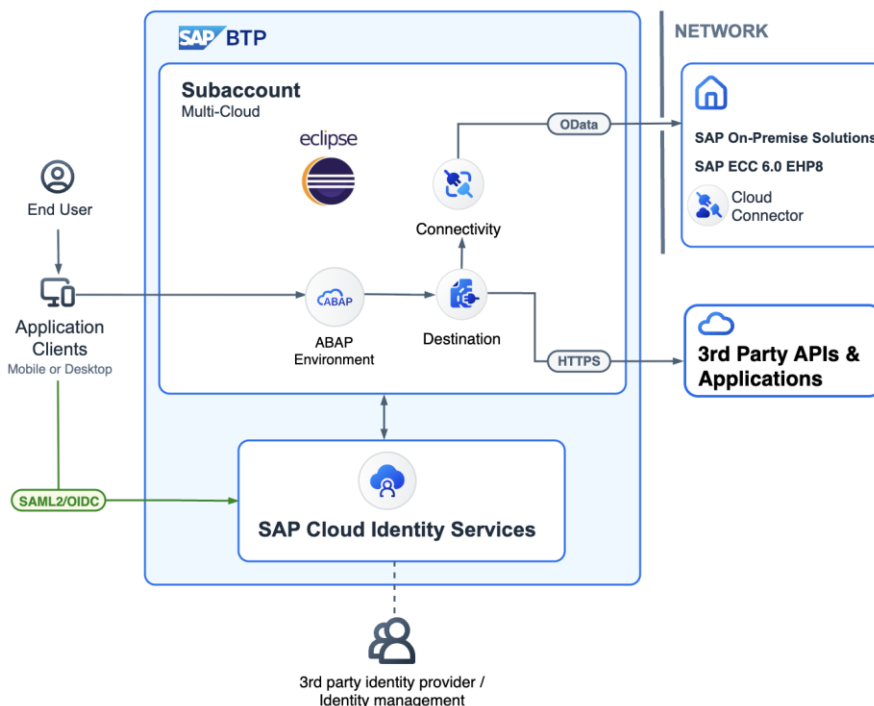


Рис. 1. Архітектура реалізації стратегії Clean Core для SAP ECC 6.0 через SAP BTP

Для формалізації запропонованого архітектурного підходу мінімізацію модифікацій ядра доцільно розглядати як задачу оптимізації. Повна функціональність корпоративної інформаційної системи F подається у вигляді об'єднання стандартної функціональності ядра F_{core} та функціональності, реалізованої у зовнішніх архітектурних розширеннях F_{ext} .

$$F = F_{core} \cup F_{ext}, F_{core} \cap F_{ext} = \emptyset. \quad (1)$$

Модифікації ядра розглядаються як сукупність умовних одиниць, кожна з яких відповідає окремому втручання у стандартний код. Сукупний обсяг модифікацій ядра описується цільовою функцією:

$$M_{core} = \sum_{i=1}^n \omega_i \cdot m_i, \quad (2)$$



де m_i – умовна одиниця i -ї модифікації ядра; ω_i – ваговий коефіцієнт критичності модифікації; n – загальна кількість модифікацій.

Модифікації ядра мають різний рівень складності та впливу на стабільність системи, тому кожна з них оцінюється умовною кількістю одиниць m_i , що відображають обсяг втручання у

стандартний код. Для об'єктивного порівняння пропонується використовувати експертну шкалу оцінювання, у якій значення m_i є

узагальненою величиною та не вимірюється у фізичних або часових одиницях. Подібний підхід відповідає методології SAP Clean Core Assessment, що передбачає інвентаризацію та класифікацію модифікацій ядра [4].

Для урахування критичності кожної модифікації вводиться ваговий коефіцієнт ω_i . Значення цього коефіцієнта визначається експертним шляхом та відображає ризик негативного впливу модифікації на цілісність і стабільність системи. У результаті величина внеску окремої модифікації до цільової функції M_{core} визначається добутком $\omega_i \cdot m_i$, що дає змогу формалізувати технічний борг системи.

Узагальнена класифікація рівнів складності та критичності модифікацій подана в таблиці 1.

Таблиця 1

Рівні складності модифікацій

Рівень модифікації	Характеристика втручання	m_i	ω_i
Низький	Просте розширення, UI-exit, кастомізація без зміни логіки	1–2	1
Середній	Зміни логіки обробки даних, додавання функцій у стандартні процеси	3–5	1,5
Високий	Глибше втручання у транзакційну логіку критичних модулів	6–8	2–2,5
Дуже високий	Модифікація стандартних API/BAPI, що впливають на інтеграції	9–10	3

Уведені параметри m_i і ω_i не є функціонально пов'язаними величинами. У реальних сценаріях між цими параметрами може спостерігатися певна емпірична кореляція, проте в моделі вони

розглядаються як незалежні оцінки, що дозволяє уникнути змішування трудомісткості та ризиків втручання.

Метою архітектурного підходу є мінімізація значення M_{core} за умови збереження повної функціональності системи, що досягається шляхом перенесення частини бізнес-логіки з множини F_{core} до множини F_{ext} .

Для оцінювання ефективності застосування підходу вводиться відносний показник:

$$\Delta M = \frac{M_{core} - M_{core}^{new}}{M_{core}}, \quad (3)$$

де M_{core}^{new} – значення цільової функції після реалізації архітектурних змін.

У контексті розглядуваної архітектури множина F_{core} відповідає стандартній функціональності SAP ECC 6.0, тоді як множина F_{ext} реалізується у хмарному середовищі SAP BTP. Застосування ABAP Environment і моделі RAP дозволяє створювати прикладні сервіси та бізнес-логіку поза межами ядра ERP-системи, використовуючи протокол OData як основний канал доступу до бізнес-даних.

SRVC виступає зв'язувальною ланкою між математичною моделлю та практичною реалізацією архітектурного підходу, оскільки саме вона забезпечує формалізований і типізований доступ до даних ядра без використання низькорівневих інтеграційних механізмів. Таким чином, перенесення функціональності з F_{core} до F_{ext} безпосередньо зменшує значення M_{core} , не порушуючи стабільності ERP-системи.

Для ілюстрації застосування показника ΔM розглянемо умовний приклад, що відповідає шкалі оцінювання, наведеної в табл. 1. Нехай у початковому стані ядро системи містить чотири модифікації, класифіковані за рівнем складності та критичності:



- модифікація 1 – середній рівень складності ($m_1=4$), середня критичність ($\omega_1=1,5$);
- модифікація 2 – високий рівень складності ($m_2=7$), підвищена критичність ($\omega_2=2,0$);
- модифікація 3 – низький рівень складності ($m_3=2$), низька критичність ($\omega_3=1,0$);
- модифікація 4 – дуже високий рівень складності ($m_4=9$), висока критичність ($\omega_4=3,0$);

Тоді зважений показник модифікацій ядра до впровадження архітектурного підходу за формулою (2) становить:

$$M_{core} = 1,5 \cdot 4 + 2,0 \cdot 7 + 1,0 \cdot 2 + 3,0 \cdot 9 = 49.$$

Припустимо, що в результаті перенесення частини бізнес-логіки у хмарне середовище:

- модифікація 4 (найбільш критична та трудомістка) повністю усувається з ядра;
- обсяг модифікації 2 зменшується (наприклад, її частину винесено у зовнішній сервіс), унаслідок чого умовне значення m_2 знижується з 7 до 3, тоді як рівень критичності ω_2

залишається незмінним.

Новий зважений показник модифікацій ядра після застосування архітектурного підходу матиме вигляд:

$$M_{core} = 1,5 \cdot 4 + 2,0 \cdot 3 + 1,0 \cdot 2 = 14.$$

Відносний ефект від застосування архітектурного підходу визначається за формулою (3):

$$\Delta M = \frac{M_{core} - M_{core}^{new}}{M_{core}} = \frac{49 - 14}{49} \approx 0,71,$$

що відповідає скороченню зваженого обсягу модифікацій ядра приблизно на 71%. Такий результат демонструє, що навіть часткове винесення найбільш складних і критичних модифікацій за межі ядра дає суттєвий ефект у контексті мінімізації технічного боргу та полегшення подальшого оновлення системи.

Подібні концептуальні підходи до прогнозування динаміки технічного боргу в ERP-системах описано в сучасних роботах з модернізації ІТ-інфраструктур підприємств [2].

Порівняльну динаміку зміни обсягу модифікацій ядра з використанням запропонованого архітектурного підходу та без нього наведено на рис. 2.

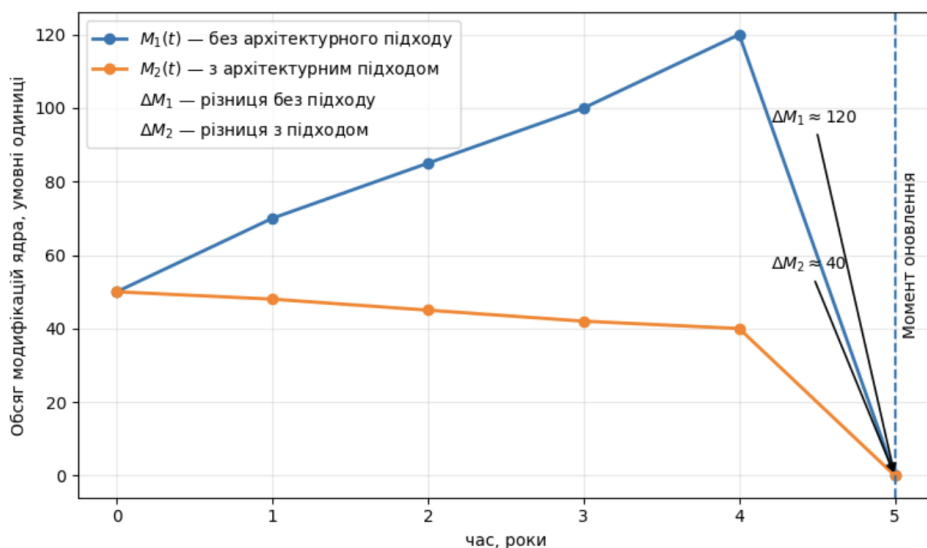


Рис.2. Динаміка модифікацій ядра до оновлення системи

У сценарії без винесення бізнес-логіки кількість модифікацій поступово зростає протягом усього періоду експлуатації, що призводить до різкого скорочення їх обсягу в момент переходу системи на нову версію та супроводжується підвищеними ризиками цього процесу. Натомість застосування архітектурного підходу забезпечує поступове зменшення обсягу модифікацій упродовж декількох років, унаслідок чого дельта змін у момент оновлення є незначною. Це наочно демонструє переваги запропонованого підходу як інструменту керованої підготовки корпоративних інформаційних систем до подальшої модернізації.

Висновки. У статті обґрунтовано архітектурний підхід до поступового зменшення модифікацій ядра корпоративних

інформаційних систем, що вже тривалий час експлуатуються, шляхом перенесення частини бізнес-логіки за межі ERP у хмарні середовища та застосування сервісно-орієнтованих механізмів інтеграції (OData, Service Consumption Model). Запропонована математична модель дала змогу формалізувати процес скорочення втручань у ядро та визначити відносний показник ефективності архітектурних змін.

Порівняльний аналіз динаміки модифікацій упродовж життєвого циклу системи засвідчив, що застосування описаної архітектури забезпечує поступове зменшення технічного боргу та знижує імовірність виникнення проблем під час оновлення програмної платформи. Це узгоджується з висновками міжнародних досліджень, що демонструють переваги поступового впровадження хмарних технологій у корпоративні ІТ-системи [1].

Результати дослідження свідчать, що запропонована архітектура може стати основою для стандартизації процесів модернізації ERP-систем та формування методичних рекомендацій для підприємств, які прагнуть забезпечити стабільність критичних бізнес-процесів у процесі технологічного розвитку.

1. Lee C., Kim H.F., Lee B.G. A Systematic Literature Review on the Strategic Shift to Cloud ERP: Leveraging Microservice Architecture and MSPs for Resilience and Agility [Електронний ресурс] // *Electronics*. – 2024. – Vol. 13, No. 14, Article 2885. – DOI: <https://doi.org/10.3390/electronics13142885> (дата звернення: 21.12.2025).
2. Yadav S. Hybrid Cloud Strategies for ERP Modernization [Електронний ресурс] // *International Journal of Applied Management (IJAM)*. – 2025. – Vol. 38, Issue S3, Article 207. – DOI: <https://doi.org/10.12732/ijam.v38i3s.207> (дата звернення: 21.12.2025).
3. Noel D'Costa. SAP Clean Core Strategy: What it means for your business [Електронний ресурс]. – Режим доступу: <https://noelcosta.com/sap-clean-core-strategy-what-it-means-for-your-business/> (дата звернення: 21.12.2025).
4. SAP Community. Clean Core Strategy: a practical approach to get a cleaner on-premise system [Електронний ресурс]. – Режим доступу: <https://community.sap.com/t5/technology-blog-posts-by-members/clean-core-strategy-a-practical-approach-to-get-a-cleaner-on-premise-system/ba-p/14180311> (дата звернення: 21.12.2025).
5. SAP SE. ABAP Extensibility and ABAP Environment Overview [Електронний ресурс]. – Режим доступу: <https://help.sap.com/docs/btp/abap-environment> (дата звернення: 21.12.2025).
6. OPC Router. SAP Interfaces: RFC, SOAP and OData Integration

[Електронний ресурс]. – Режим доступу: <https://www.opc-router.com/sap-interfaces> (дата звернення: 21.12.2025).

REFERENCES:

1. Lee C., Kim H.F., Lee B.G. A Systematic Literature Review on the Strategic Shift to Cloud ERP: Leveraging Microservice Architecture and MSPs for Resilience and Agility [Electronic resource] // *Electronics*. – 2024. – Vol. 13, No. 14, Article 2885. – DOI: <https://doi.org/10.3390/electronics13142885> (accessed December 21, 2025).
2. Yadav S. Hybrid Cloud Strategies for ERP Modernization [Electronic resource] // *International Journal of Applied Management (IJAM)*. – 2025. – Vol. 38, Issue S3, Article 207. – DOI: <https://doi.org/10.12732/ijam.v38i3s.207> (accessed December 21, 2025).
3. Noel D'Costa. SAP Clean Core Strategy: What it means for your business [Electronic resource]. Available at: <https://noelcosta.com/sap-clean-core-strategy-what-it-means-for-your-business/> (accessed December 21, 2025).
4. SAP Community. Clean Core Strategy: a practical approach to get a cleaner on-premise system [Electronic resource]. Available at: <https://community.sap.com/t5/technology-blog-posts-by-members/clean-core-strategy-a-practical-approach-to-get-a-cleaner-on-premise-system/bap/14180311> (accessed December 21, 2025).
5. SAP SE. ABAP Extensibility and ABAP Environment Overview [Electronic resource]. Available at: <https://help.sap.com/docs/btp/abap-environment> (accessed December 21, 2025).
6. OPC Router. SAP Interfaces: RFC, SOAP and OData Integration [Electronic resource]. Available at: <https://www.opc-router.com/sap-interfaces> (accessed December 21, 2025).

Yeroshkin Y. M. ^[1; ORCID ID: 0009-0009-1018-5487],

Candidate of Engineering (Ph.D.), Assistant,

Voloshchuk V. A. ^[1; ORCID ID: 0000-0003-0687-8968],

Doctor of Engineering, Professor

Yeroshkin O. Y. ^[1; ORCID ID: 0009-0006-3976-7526],

Master of Science in Engineering, Assistant,

¹National Technical University of Ukraine “Igor Sikorsky Kyiv Polytechnic Institute”, Kyiv

²Wroclaw University of Science and Technology, Wroclaw, Poland

ARCHITECTURAL APPROACH TO MINIMIZING CORE MODIFICATIONS IN CORPORATE INFORMATION SYSTEMS

Enterprise resource planning (ERP) systems that have been in operation for many years typically accumulate a significant number of custom extensions embedded directly into the system core. While such modifications provide short-term business flexibility, they accumulate technical debt, impede upgrade processes, and increase risks associated with transitioning to modern ERP architectures. In response, SAP SE introduced the Clean Core strategy for SAP S/4HANA, which promotes preservation of the standard ERP core and recommends implementing application-specific logic outside the core using cloud extensibility. Although originally intended for new-generation platforms, the underlying architectural principles of Clean Core can also be adapted to legacy environments.

This paper proposes an architectural approach that enables gradual reduction in the volume of core modifications in existing ERP systems by separating standard functionality and externalized extensions. The approach is demonstrated on an SAP ECC 6.0 landscape integrated with SAP Business Technology Platform (SAP BTP). Application-specific logic is relocated to the ABAP Environment on SAP BTP and interacts with the ERP system through service-oriented interfaces implemented via the OData protocol. The Service Consumption Model serves as the key mechanism for consistent and type-safe consumption of remote services and integration with the RAP programming model.

A mathematical model for quantifying the volume of core modifications is introduced to formalize the minimization objective. The model enables calculation of a relative effectiveness indicator ΔM , which measures the reduction of core modifications following the transition to the proposed architecture. A numerical example and a comparative graph illustrate the dynamics of modification volumes over several years, both with and without the architectural approach.

The obtained results indicate that the proposed approach contributes to controlled modernization of legacy ERP systems by mitigating upgrade risks, reducing technical debt, and preparing the system landscape for migration to contemporary ERP platforms. The research results can be used by organizations operating mission-critical ERP environments, particularly in energy and infrastructure sectors, to plan their transition roadmap toward Clean Core principles without disruptive system replacement.

Keywords: corporate information systems; software architecture; cloud platforms; service-oriented integration; SAP Business Technology Platform; Clean Core.

Отримано: 29 березня 2026 року
Прорецензовано: 3 травня 2026 року
Прийнято до друку: 29 травня 2026 року



© 2026 [Yeroshkin Y. M., Voloshchuk V. A., Yeroshkin O. Y.]. Licensee [NUWEE]. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution-NonCommercial (CC BY-NC) license (creativecommons.org).