

ПОРІВНЯЛЬНИЙ АНАЛІЗ МОНОЛІТНОЇ ТА РОЗПОДІЛЕНОЇ МУЛЬТИАГЕНТНИХ СИСТЕМ У КОНТЕКСТІ ІНФОРМАЦІЙНОЇ ТЕХНОЛОГІЇ ПЕРЕВІРКИ СХОЖОСТІ ЦИФРОВИХ РОБІТ СТУДЕНТІВ

Вінницький національний технічний університет;

Анотація

У роботі представлено аналіз сучасних архітектурних підходів до розробки мультиагентних систем – монолітної та розподіленої. Особливу увагу приділено перевагам і недолікам кожної архітектури за такими критеріями, як масштабованість, надійність, продуктивність і складність розробки.

Ключові слова: мультиагентна система, монолітна архітектура, розподілена система, масштабованість, продуктивність, відмовостійкість.

Abstract

This paper analyzes contemporary architectural approaches to developing multi-agent systems: monolithic and distributed. Particular attention is given to the advantages and disadvantages of each architecture based on criteria such as scalability, reliability, performance, and development complexity.

Key words: multi-agent system, monolithic architecture, distributed system, scalability, performance, fault tolerance.

Вступ

Швидкий розвиток інформаційних технологій висуває нові вимоги до якості, гнучкості та ефективності програмних систем. У контексті зростаючих запитів на адаптивність та інтелектуальність, мультиагентні системи (МАС) здобувають дедалі більшу популярність, оскільки їх архітектура визначає ключові аспекти взаємодії агентів, рівень їхньої розподіленості, масштабованості та здатності до самоорганізації.

Враховуючи стрімкий розвиток обчислювальних технологій, зростаючу складність завдань і потребу у високій доступності рішень, вибір оптимальної архітектури є вирішальним фактором успішного впровадження МАС. Саме тому в цій роботі проаналізовано сучасні архітектурні підходи до побудови МАС – монолітну та розподілену моделі.

Метою дослідження є аналіз архітектурних стратегій проектування МАС із фокусом на продуктивність, масштабованість, надійність та складність реалізації. Особлива увага приділяється порівнянню основних переваг та недоліків монолітних та розподілених моделей на основі аналізу сучасних методів і результатів прикладних досліджень.

Результати дослідження

МАС являють собою складні програмні архітектури, у яких автономні агенти координують свої дії для досягнення спільної мети [1, 2]. Архітектурна організація МАС може варіюватися – від монолітних моделей, де всі агенти функціонують у межах одного процесу, до розподілених рішень, що базуються на взаємодії агентів, розміщених на окремих вузлах мережі. Вибір архітектури істотно впливає на такі важливі характеристики системи, як продуктивність, масштабованість, надійність та складність супроводу.

Монолітна архітектура передбачає функціонування всіх агентів у межах одного обчислювального середовища (процесу або сервера), що забезпечує спільний доступ до пам'яті та дозволяє реалізувати взаємодію з мінімальними витратами на комунікацію [3]. Приклад монолітної архітектури в контексті

інформаційної технології перевірки схожості текстових цифрових робіт студентів для освітньої платформи наведено на рис. 1.

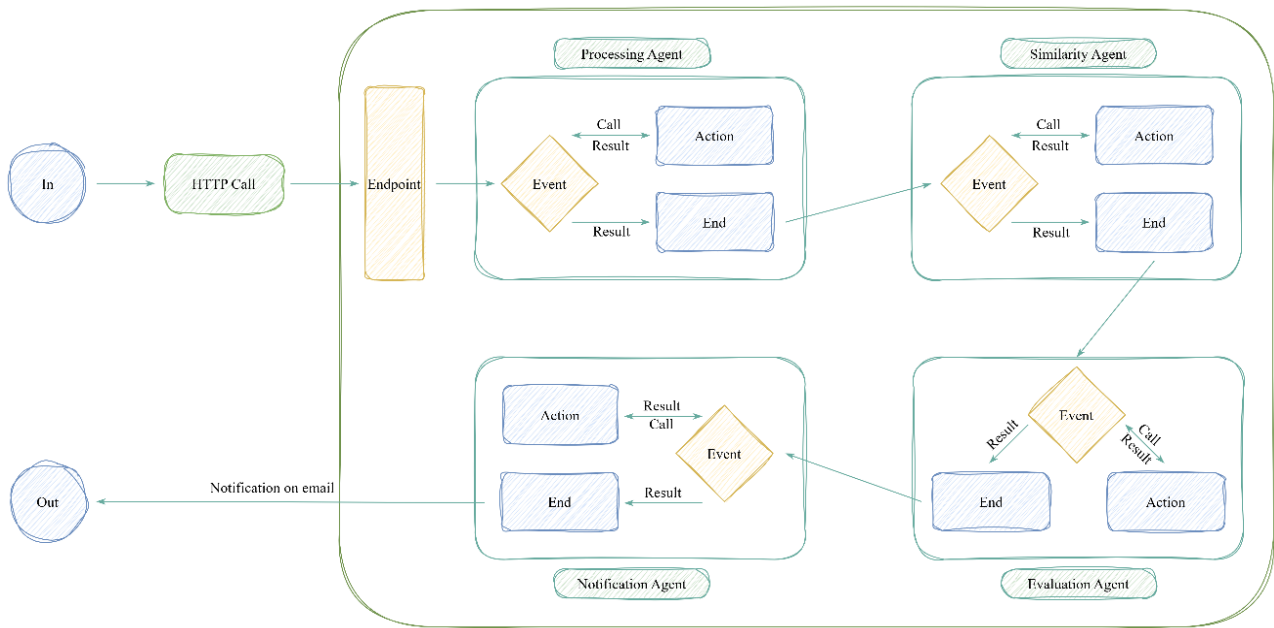


Рис. 1. Монолітна архітектура MAC у контексті інформаційної технології перевірки схожості цифрових робіт студентів

Основні переваги системи такі:

- обмін інформацією між агентами через спільну пам'ять значно знижує затримки, порівняно з мережевими комунікаціями, що сприяє високій продуктивності системи;
- відсутність потреби у мережевій взаємодії полегшує реалізацію логіки взаємодії між агентами та скорочує загальний час розробки;
- єдине обчислювальне середовище полегшує керування ресурсами, синхронізацію дій агентів та моніторинг виконання.

Основні недоліки системи:

- ресурсні обмеження одного вузла обмежують розширення системи – як у кількості агентів, так і в обробці великих обсягів даних;
- збій одного агента або помилка в загальному середовищі виконання може спричинити порушення роботи всієї системи;
- оновлення одного компонента часто потребує повторної збірки всієї системи, що ускладнює підтримку та розгортання оновлень.

У розподіленій архітектурі кожен агент є незалежним процесом, який функціонує на окремому вузлі мережі. Взаємодія між агентами реалізується за допомогою мережових протоколів, що забезпечує високу гнучкість і розширюваність системи [4]. Приклад розподіленої архітектури в контексті інформаційної технології перевірки схожості текстових цифрових робіт студентів для освітньої платформи наведено на рис. 2.

Основні переваги системи:

- система легко розширюється шляхом додавання нових агентів або вузлів без необхідності кардинальної перебудови архітектури;
- завдяки ізоляції агентів, відмова одного компонента не призводить до зупинки всієї системи;
- агенти можуть бути розміщені в різних регіонах, що дозволяє адаптувати систему до розподілених сценаріїв обробки даних та підвищити її доступність.

Основні недоліки:

- необхідність у мережевій взаємодії між компонентами системи ускладнює розробку та може призводити до додаткових затримок через особливості мережевого середовища;
- забезпечення консистентності у розподіленому середовищі вимагає додаткових механізмів реплікації, узгодження станів та обробки конфліктів;

- реалізація механізмів безпеки, синхронізації, маршрутизації повідомлень і розподілу ресурсів потребує додаткових обчислювальних витрат та ретельного проектування архітектури, що ускладнює масштабування системи та підвищує вимоги до її технічного обслуговування.

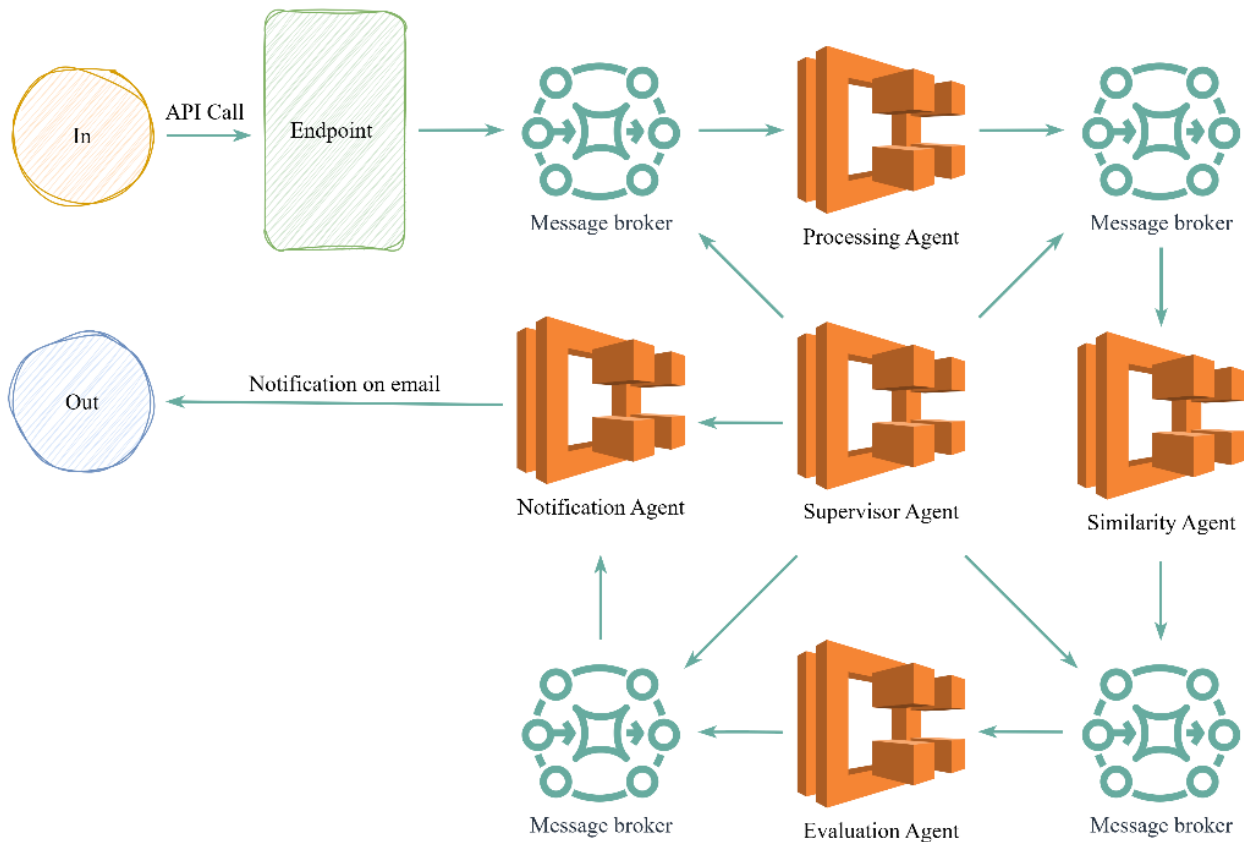


Рис. 2. Розподілена архітектура MAS у контексті інформаційної технології перевірки схожості цифрових робіт студентів

Висновки

Архітектурний вибір між монолітною та розподіленою моделлю MAS є дуже важливим етапом у процесі проектування, оскільки він безпосередньо впливає на подальшу масштабованість системи, її підтримку, продуктивність і стійкість до збоїв.

Вибір між монолітною та розподіленою архітектурою повинен ґрунтуватися на ретельному аналізі функціональних та нефункціональних вимог: цільового навантаження, вимог до відмовостійкості, мережних умов, обмежень щодо часу відгуку, доступності обчислювальних ресурсів та перспектив еволюції системи.

Монолітна архітектура демонструє переваги в умовах обмежених обчислювальних ресурсів або під час розробки невеликих систем, де важлива щільна інтеграція компонентів, мінімізація затримок між агентами та спрощена координація. Такі системи, як правило, простіші в розгортанні та тестуванні, що знижує витрати на початкову розробку. Однак вони мають обмежену гнучкість під час масштабування та є вразливими до збоїв, що можуть вивести з ладу всю систему.

Натомість розподілена архітектура MAS краще підходить для складних, динамічних або територіально розподілених систем, які повинні обробляти великі обсяги даних або працювати в режимі високої доступності. Завдяки ізоляції агентів, можливості горизонтального масштабування та кращій відмовостійкості, такі системи здатні адаптуватися до змін навантаження та інфраструктури. Разом із тим, розподілені MAS вимагають більш складного підходу до синхронізації, безпеки, моніторингу та обробки помилок, що ускладнює їхнє тестування та супровід.

Гібридні підходи, що поєднують переваги обох архітектур, також можуть розглядатись як перспективний компроміс для реалізації складних мультиагентних рішень.

СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. Russell S. Artificial Intelligence: A Modern Approach (Pearson series in artificial intelligence). 4-th US ed. / S. Russell, P. Norvig. – New York: Pearson, 2020. – 1136 p.
2. Методичні вказівки з лабораторного практикуму з дисципліни “Мультиагентні системи та технології” для студентів спеціальності 122 – “Комп’ютерні науки” / І. Р. Арсенюк, О. В. Сілагін, В. С. Озеранський – Вінниця, ВНТУ. – 2018. – 58 с.
3. Muscan G. Moving from monolithic to microservices architecture for multi-agent systems / World Journal of Advanced Engineering Technology & Sciences. – 2025. – № 0480 / G. Muscan, B. Pranav [Електронний ресурс] Режим доступу: <https://doi.org/10.30574/wjaets.2025.15.1.0480>
4. Shrivastav A. Multi-agent systems: Architectural patterns for high-throughput processing / Adarsh Shrivastav. – Сідней: Medium, 24 Oct. 2024. – [Електронний ресурс] Режим доступу: <https://medium.com/@adarshshrivastav/multi-agent-systems-architectural-patterns-for-high-throughput-processing-f971c451d698>

Соболев Данило Олександрович – студент групи 2КН-24м, факультет інтелектуальних інформаційних технологій та автоматизації, Вінницький національний технічний університет, Вінниця, e-mail: 01-24-174.stud@vntu.edu.ua

Федишен Богдан Вікторович – студент групи 2КН-24м, факультет інтелектуальних інформаційних технологій та автоматизації, Вінницький національний технічний університет, Вінниця, e-mail: 01-24-176.stud@vntu.edu.ua

Арсенюк Ігор Ростиславович – кандидат технічних наук, доцент, доцент кафедри комп’ютерних наук, Вінницький національний технічний університет, м. Вінниця, e-mail: air@vntu.edu.ua

Sobolev Danylo O. – Student of the 2CS-24M group, Department of Intellectual Information Technologies and Automation, Vinnytsia National Technical University, Vinnytsia, e-mail: 01-24-174.stud@vntu.edu.ua

Fedyshen Bohdan V. – Student of the 2CS-24M group, Department of Intellectual Information Technologies and Automation, Vinnytsia National Technical University, Vinnytsia, e-mail: 01-24-176.stud@vntu.edu.ua

Arseniuk Igor R. – Associate Professor of the Computer Sciences Department, Vinnytsia National Technical University, Vinnytsia, e-mail: air@vntu.edu.ua