

МУЛЬТИАГЕНТНИЙ ПІДХІД ДО ПРОЕКТУВАННЯ ІНФОРМАЦІЙНО-АНАЛІТИЧНИХ СИСТЕМ

© 2017 ПРИМОСТКА А. О.

УДК 65.011.56

Примостка А. О. Мультиагентний підхід до проектування інформаційно-аналітичних систем

Досліджено мультиагентний підхід до проектування інформаційно-аналітичних систем. Проаналізовано існуючі підходи до проектування мультиагентних систем та обґрунтовано додаткові завдання, що виникають у процесі розробки архітектури системи. Проаналізовано високорівневу архітектуру мультиагентної системи. Кожна підсистема, що входить у загальну систему, розглянута загалом. Особливо представлена деталізація архітектури підсистеми збору та зберігання даних до рівня окремих агентів. Розроблена архітектура підсистеми дозволяє її використання у будь-якій мультиагентній системі, що має на меті збір та аналіз великої кількості даних. Також представлена деталізація підсистеми аналізу даних. Архітектура даної підсистеми представляє собою узагальнену модель роботи підприємства в контексті мультиагентного підходу. Для кожного модуля та агента, які є структурними елементами підсистем, наведено опис функцій та їх призначення.

Ключові слова: інформаційно-аналітична система, мультиагентна система, підсистема збору та зберігання даних, підсистема аналізу даних, модуль, агент.

Рис.: 3. **Табл.:** 3. **Бібл.:** 10.

Примостка Андрій Олександрович – кандидат економічних наук (Київ)

E-mail: aprimostka@irpin.com

УДК 65.011.56

UDC 65.011.56

Примостка А. А. Мультиагентный подход к проектированию информационно-аналитических систем

Исследован мультиагентный подход к проектированию информационно-аналитических систем. Проанализированы существующие подходы к проектированию мультиагентных систем и обоснованы дополнительные задачи, которые возникают в процессе разработки архитектуры системы. Проанализирована высокоуровневая архитектура мультиагентной системы. Каждая подсистема, которая входит в общую систему, рассмотрена в целом. Отдельно представлена детализация архитектуры подсистемы сбора и сохранения данных до уровня отдельных агентов. Разработанная архитектура подсистемы позволяет ее использование в любой мультиагентной системе, которая своей целью имеет сбор и анализ большого количества данных. Также представлена детализация подсистемы анализа данных. Архитектура данной подсистемы представляет собой обобщенную модель работы предприятия в контексте мультиагентного подхода. Для каждого модуля и агента, которые являются структурными элементами подсистем, представлены описание функций и их назначение.

Ключевые слова: информационно-аналитическая система, мультиагентная система, подсистема сбора и хранения данных, подсистема анализа данных, модуль, агент.

Рис.: 3. **Табл.:** 3. **Библ.:** 10.

Примостка Андрей Александрович – кандидат экономических наук (Київ)

E-mail: aprimostka@irpin.com

Prymostka A. O. The Multiagent Approach to Projecting the Information-Analytical Systems

The multiagent approach to projecting the information-analytical systems was researched. The existing approaches to projecting the multiagent systems were analyzed, the additional tasks that arise in the process of developing the system architecture were substantiated. The high-level architecture of a multiagent system was analyzed. Each subsystem, which is part of the overall system, is taken into consideration in general. A detailization of architecture of the subsystems for collection and storage of data down to the level of individual agents is presented separately. The developed subsystem architecture provides its use in any multiagent system that is aimed at collection and analysis of large amounts of data. Also a detailization of the subsystem of data analysis is presented. The architecture of this subsystem is a generalized model of the enterprise's performance in the context of the multiagent approach. For each module and each agent as structural elements of subsystems the article provides a description of functions and their purpose.

Keywords: information-analytical system, multiagent system, subsystem for collection and storage of data, subsystem of data analysis module, agent.

Fig.: 3. **Tbl.:** 3. **Bibl.:** 10.

Prymostka Andrii O. – PhD (Economics) (Kyiv)

E-mail: aprimostka@irpin.com

У сучасному динамічному середовищі успішність бізнесу вирішальною мірою залежить від швидкості та обґрунтованості прийнятих управлінських рішень, що вимагає постійного вдосконалення інформаційно-аналітичних систем. Перспективним напрямом розробки та проектування інформаційно-аналітичних систем є мультиагентний підхід як один з напрямів розвитку парадигми об'єктно-орієнтованого програмування. Хоча мультиагентний підхід виник у сфері комп'ютерних наук, зараз він застосовується у багатьох галузях, таких як економіка, біологія, екологія, соціологія, система оборони, транспорт, логістика та ін.

Мультиагентна система (МАС) є сукупністю кількох програмних агентів (автономних модулів), які взаємодіють між собою. Розробка МАС надає можливість суттєво спростити представлення складних систем, до яких належать й інформаційно-аналітичні, та врахувати в них можливі дії користувача. Головною перевагою

МАС є здатність до виконання значного обсягу робіт високої складності, які жоден агент автономно зробити не спроможний. Застосування ієрархічного підходу для проектування дозволяє розробляти агентів МАС паралельно та незалежно. Також мультиагентний підхід до проектування систем передбачає просту інтеграцію з уже існуючими інформаційно-аналітичними системами, взаємодію та оперативне оновлення аналітичної інформації, використання тих джерел даних, які були доступні в межах існуючої в користувача системи. Мультиагентний підхід до побудови інформаційно-аналітичних систем суттєво підвищує ефективність управлінського процесу, а отже, є актуальним напрямом досліджень.

Вивчення наукових праць, присвячених розвитку інформаційних технологій в економіці, показало, що на цей час сформувалися різні концепції та підходи до побудови інформаційно-аналітичних систем, які відрізняються можливостями доступу до сховищ і баз даних,

логією представлення, рівнем автоматизації, кількістю користувачів [1–10]. Перспективним напрямом є наукові дослідження мультиагентних систем, активізація яких була зумовлена зростанням потужності комп'ютерної техніки. Зокрема, виокремилися напрями застосування мультиагентних систем до обробки інформації та добування даних у розподілених системах і сховищах даних [2–5]. Для забезпечення захисту даних у таких системах було запропоновано також використовувати мультиагентні системи [6]. Отже, застосування мультиагентного підходу до побудови інформаційно-аналітичних систем потребує подальших досліджень.

Мultiагентна система – це сукупність агентів (автономних модулів), здатних до взаємодії. Під програмним агентом розуміють апаратну або програмну сутність (модуль), здатну діяти в інтересах досягнення цілей, поставлених перед нею іншим учасником системи: як користувачем, так і розробником, а також іншим агентом. Агента можна розглядати як повністю незалежну та автономну одиницю мультиагентної системи. Із зростанням рівня деталізації такий підхід значно спрощує проектування мультиагентної системи в цілому. Multiагентна система також може бути представлена як сукупність окремих підсистем, тоді на кожному рівні деталізації потрібно окремо визначити функціональність тих підсистем та агентів, які є елементами цього рівня, та визначити онтологію, яка буде використовуватися всередині системи.

Для проектування МАС використовуються різні підходи, зокрема такі, як Tropos, KADS, Prometheus, ADELFE [7–10] та інші. Ці підходи базуються на класичних методиках розробки програмного забезпечення. Для складних мультиагентних систем розробляються спеціальні методики проектування та реалізації на базі вже розроблених підходів або поєднання кількох підходів в одній методиці. У класичних алгоритмах розробки програмного забезпечення (зокрема, у процесі безпосередньої розробки без врахування організаційних етапів, таких, наприклад, як підтримка чи тестування програмних модулів) виділяють три основні етапи розробки нових програмних комплексів: 1) визначення вимог до системи; 2) проектування, або розробка архітектури; 3) реалізація.

З урахуванням специфіки мультиагентних систем у методиках проектування передбачено деталізоване та специфічне розбиття класичних етапів на окремі фази, в кожній з яких увага акцентується на конкретних цілях та завданнях, які вважаються найбільш важливими на поточний момент процесу розробки. Наприклад, етап проектування може розбиватися на фази архітектурного та деталізованого дизайну, що фокусуються на специфікації системи згідно з вимогами, виділеними на попередніх фазах. Архітектурний дизайн визначає глобальну архітектуру системи в термінах підсистем, пов'язаних через потоки даних та управління. У фазі детального дизайну метою є визначення можливостей агентів та їх взаємодії.

Отже, у процесі проектування вимоги до функціонування кожного агента визначаються в межах вимог конкретної мультиагентної системи. Таким чином, компонентний підхід – з одного боку, і обмеження кожної

конкретної системи – з іншого, визначають необхідність проектувати МАС у такий спосіб, щоб агенти могли бути використані й в інших МАС. Тобто, слід обмежувати специфіку всієї системи лише в рамках одного або декількох агентів. Це призводить до того, що існуючі методи проектування МАС є занадто складними та мають досить тривалий період узгодження вимог агентів. Перевагою мультиагентного підходу є його потенціал для горизонтального масштабування. Це означає, що за можливості створення у процесі роботи таких етапів, які можуть виконуватися агентами паралельно, кількість таких агентів можна змінювати для зміни швидкості обробки даних. Для впровадження техніки розбиття даних на розділи використовуються різні методи поділу зон відповідальності агентів, наприклад, методи самоорганізації агентів з використанням протоколів для реалізації розподіленої хеш-таблиці. Усі ці переваги й зумовляють вибір мультиагентного підходу до побудови інформаційно-аналітичних систем.

На найвищому рівні деталізації мультиагентна інформаційно-аналітична система складається з трьох підсистем: збору та зберігання даних, аналізу даних та взаємодії з користувачем (рис. 1). Опис функцій та призначення підсистем наведено в табл. 1.

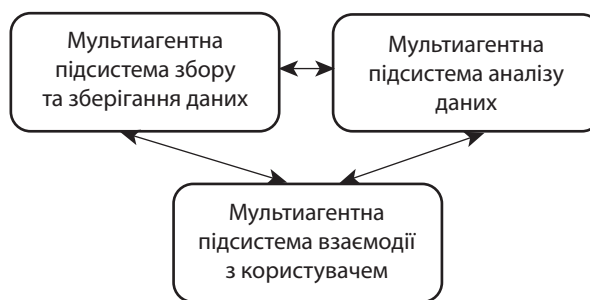


Рис. 1. Структура мультиагентної інформаційно-аналітичної системи

Таблиця 1

Структурні елементи МАС та їх призначення

Назва	Призначення
Мультиагентна підсистема збору та зберігання даних	Пошук джерел даних, автоматичне завантаження, обробка та актуалізація даних. Зберігання зібраних даних та оновлення структури до актуального стану
Мультиагентна підсистема аналізу даних	Розрахунок статистичної інформації, а також побудова прогнозів. Перелік показників залежить від набору агентів, що використовується у системі
Мультиагентна підсистема взаємодії з користувачем	Взаємодія з користувачем, інтерпретація запитів користувача у зрозумілі системі команди та візуалізація результатів аналізу

Структура підсистеми взаємодії з користувачем цілком визначається конкретними вимогами користувача, а значить, може змінюватися залежно від його вимог. Тому більш детально опишемо підсистеми збору та

зберігання даних та аналізу даних. Мультиагентна підсистема збору та зберігання даних повинна вирішувати задачі збереження та обробки нових даних. Також у ній мають враховуватися збільшення обсягу вхідних даних, первинна обробка даних, агрегація різнопланових даних. При цьому зростають кількість та обсяги інформаційних потоків, що мають бути проаналізовані та оброблені при надходженні до підсистеми, тому виникає потреба в аналізі вхідної інформації на достовірність з метою зниження ризиків завантаження недостовірної інформації до бази даних. Також значної актуальності набувають питання забезпечення збереження та безпеки інформації в базах даних, що вимагає запровадження комплексу засобів інформаційної безпеки. А для забезпечення цілісності вразливих даних виникає необхідність ведення протоколювання факту зміни даних і збереження усіх станів даних підсистеми.

Для вирішення окресленої задачі пропонуємо привести різноманітні структури вхідної інформації до однієї базової платформи з відображенням прийнятих рішень у структурі бази даних, що здатна зберігати історичну інформацію. Стратегії об'єднання можуть також відрізнятися для різних даних залежно від критеріїв класифікації визначених параметрів даних.

Важливою характеристикою також є поточне і прогнороване навантаження для мінімізації часу відклику та вимога неперевищення максимального навантаження на систему в процесі завантаження нових даних. Тому підсистема збору та зберігання даних має такі характеристики:

- ✦ дані в системі розбиваються на скінчену кількість класів за вартісною функцією атрибутів кожного елемента даних, причому при виявленні нових атрибутів даних система може їх сама класифікувати;
- ✦ для кожного класу даних існує набір агентів, які підтримують пріоритетний перелік стратегій та алгоритмів виявлення структурних змін, змін даних, відбору оновлених даних, узагальнення схем даних джерел (медіатор), оновлення структури бази даних з історичною інформацією і трансформації та збереження відібраних даних;
- ✦ для всієї системи в цілому також існують агенти аналізу та прогнозування навантаження серверів та агенти забезпечення зв'язку агентів між собою та із зовнішнім середовищем, що можуть мати декілька екземплярів, які можуть домовлятися про відповідальність кожного екземпляра.

Деталізацію архітектури підсистеми збору та зберігання даних наведено на *рис. 2*, а короткий опис функцій та призначення агентів цієї підсистеми – у *табл. 2*.

Зазначимо, що в даному контексті під базою даних розуміємо базу, яка може зберігати надзвичайно великі обсяги даних. З погляду економічності та складності побудови найбільш пристосованою до цих вимог є база, розміщена в розподіленій файлової системі. Це можуть бути як і нестандартизовані файлові дані, так і структуровані чи неструктуровані бази дані, здатні працювати в такому середовищі.

Така вимога пов'язана з тим, що агент виявлення змін даних має як список джерел, обов'язкових для обробки, так і необов'язкові джерела, які він виявляє у процесі аналізу. Наприклад, це можуть бути сайти, що публікують новини, або власні прогнози. Така інформація зберігається для подальшого аналізу іншими агентами. Тематичний перелік необхідної інформації та її обсяг регулюється користувачем, у той час як конкретні джерела можуть виявлятися в автоматичному режимі. Розглянемо докладніше зміст та функції кожного типу агентів як структурних елементів другого рівня деталізації МАС.

Агент виявлення структурних змін. Функціональним навантаженням цього типу агентів є своєчасне виявлення структурних змін у даних і джерелах їх отримання. Для забезпечення якомога швидшого збереження оновлених даних у форматі нової структури агенти, які працюють зі структурними змінами та оновленнями, повинні миттєво реагувати на оновлення структури джерел даних. Загальним методом є періодичне опитування, але така стратегія має певні недоліки, зокрема, затримку в оновленні структури на час опитування та значне навантаження при аналізі та співставленні старої та нової схем даних.

Значно перспективнішими стратегіями стає використання спеціфічних програмних інтерфейсів сповіщення про структурні зміни або використання зовнішніх джерел з уже проаналізованими схемами та загальнодоступними даними про різницю в структурах схем. У результаті аналізу має бути доступна для запиту інформація про операції, які були виконані для кожного атрибута кожного елемента структурної схеми. Також для кожної схеми має бути доступна повна історія змін для можливості перегляду цієї інформації користувачем.

Агент оновлення структури. Коли агент виявлення структурних змін для конкретної схеми проаналізував зміни, що відбулися, і сформував порядок операцій оновлення схеми для приведення її до нової форми, він отримує список даних і починає будувати стратегію оновлення поточної схеми. Основними проблемами цього агента є вибір часу оновлення структури бази даних, який залежить від зовнішнього середовища, і безпосереднє проведення самого оновлення. Додатково допускається повна відмова агента від впровадження необхідних змін при неможливості обробки даних в існуючому зовнішньому середовищі. На результат може впливати багато факторів: наявність часу та потужності на оновлення параметрів атрибута, наявність необхідного місця у виділеному просторі, де розміщується база даних, можливість обробки запропонованих даних чи структури, обсяг інформації, яку необхідно оброблювати, максимальне навантаження на мережу та багато інших.

Агент виявлення змін даних. За функціональністю цей агент схожий з агентом виявлення змін структури даних, але за алгоритмами роботи він відрізняється від «структурного» агента. У агента виявлення змін даних частина аналізу факту зміни даних не може працювати в режимі повного зіставлення, оскільки в загальному випадку це буде вимагати аналізу повної відповідності всіх даних конкретної схеми з періодичністю виконання, що

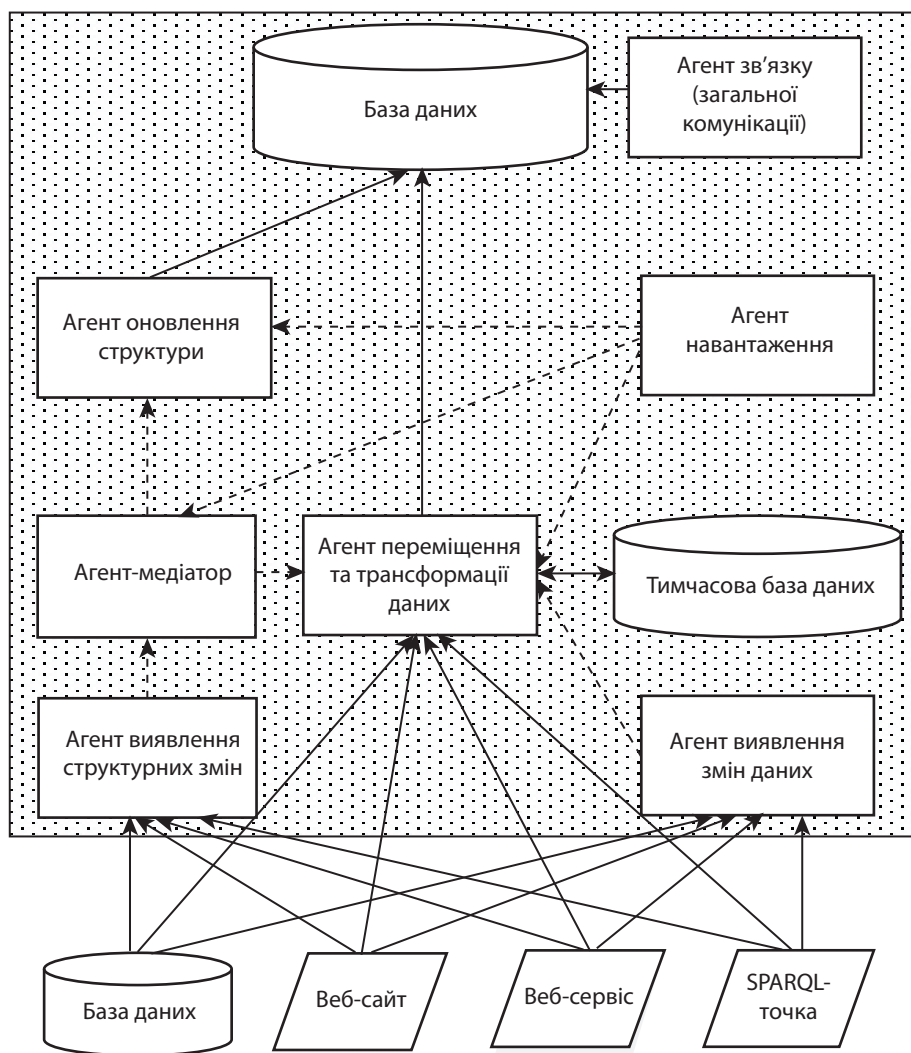


Рис. 2. Архітектура мультиагентної підсистеми збору та зберігання даних

Таблиця 2

Агенти підсистеми збору та зберігання даних

Назва	Призначення
Агент виявлення структурних змін	Виявлення та отримання нової структури кожного джерела даних
Агент оновлення структури	Побудова алгоритму оновлення структури бази даних і саме оновлення
Агент виявлення змін даних	Виявлення необхідності оновлення даних і пошук нових джерел даних
Агент-медіатор	Автоматичне створення узагальнених схем даних після аналізу поточної схеми даних і нової структури джерел даних
Агент переміщення та трансформації даних	Створення етапів процесу завантаження даних до бази згідно з оновленими схемами даних і, власне, завантаження. За необхідності проводиться трансформація даних
Агент навантаження	Балансування та паралелізація етапів завантаження. Побудова послідовності завантаження у прив'язці до агентів та/або розбиття даних на розділи
Агент зв'язку (загальної комунікації)	Єдина загальнодоступна точка комунікації підсистеми з іншими підсистемами. Довідковий каталог агентів підсистеми

співпадає з періодичністю опитування джерела. Тому якщо використання внутрішніх механізмів виявлення змін даних неможливе, потрібно використовувати інші алгоритми, наприклад аналіз обраної випадкової вибірки даних. За умови перевищення мінімального рівня ймовірності зміни даних транслюються в повідомлення про необхідність актуалізації даних у базі. Також даний

агент може виконувати пошук нових джерел даних з використанням того ж методу випадкової вибірки даних, але не з цільовими даними з даного джерела, а з усіма наявними в базі.

Агент переміщення та трансформації даних. Мета цього агента – за сприятливих зовнішніх умов (на-явність даних для завантаження, завантаження серверів,

наявність вільного простору для зберігання тощо) виконання операції перевантаження відфільтрованого набору даних (якщо є можливість його отримати), чи усього набору даних, якщо немає достовірного фільтра. Разом з перевантаженням даних може виконуватись операція трансформації. Складність трансформації визначається агентом та може бути розбита на декілька простіших, схеми для яких агент отримує в агента-медіатора. У загальному випадку як джерело чи приймач даних виступає тимчасова база даних, але при простих трансформаціях, де можливо оброблювати потокові дані, етап збереження даних до тимчасової бази даних за домовленістю агентів може бути опущений.

Агент узагальнення схем даних (медіатор). Цей агент відповідає за створення схем даних, у які завантажуються дані, причому як проміжні, так і цільові. При побудові цього агента основними проблемами є класифікація нових атрибутів даних в автоматичному режимі та оновлення даних при ручному коригуванні класу даних, побудова схеми з урахуванням класу даних, залежно від якого обираються різні варіанти трансформації вихідних даних: від створення нового атрибута до повного ігнорування цих даних, і побудова не тільки цільової схеми, але й можливості розбити проведену трансформацію до рівня одиничних операцій за запитом.

Агент навантаження. Основною метою є балансування та паралелізація етапів завантаження. Це досягається побудовою прогнозу параметрів навантаження агентів, які використовуються при плануванні операцій завантаження та трансформації, що дозволяє створювати послідовності завантаження у прив'язці до агентів.

Також цей агент може прогнозувати кількісні оцінки процесу завантаження та трансформації для виявлення необхідності розбиття даних на розділи.

Агент зв'язку. У межах відповідальності цього агента перебуває вся зовнішня комунікація підсистеми та підтримка актуальної бази агентів підсистеми. Одним із завдань є підтримка операцій управління монопольним доступом у рамках підсистеми до будь-якого елемента, з якими працює підсистема: чи то даних, чи то схем даних. Також агент повинен підтримувати як загальнодоступну точку входу в межах підсистеми, так і точки виходу ззовні як для запитів підсистеми взаємодії з користувачами, так і для управління та контролю стану підсистеми. Окрім того, до його обов'язків належить підтримка топології підсистеми та трансляція запитів до відповідних агентів чи бази даних.

Підсистема аналізу даних призначена для обробки та аналізу вхідних даних для побудови прогнозів, на основі яких підсистема формує набір рекомендацій. Підсистема також повинна знати та використовувати специфіку діяльності як самого користувача, так і його зовнішнього середовища. Оскільки поєднання і моделей прогнозування, і специфічних умов та обмежень щодо роботи є достатньо складною системою, підсистема аналізу даних деталізується до другого рівня. До архітектури підсистеми аналізу даних входять як модулі, так і окремі агенти (рис. 3). Короткий опис модулів і агентів підсистеми аналізу даних наведено в табл. 3. Розроблена архітектура враховує найбільш загальні напрями аналітичної роботи, актуальні для будь-якого користувача, який є суб'єктом ринку.

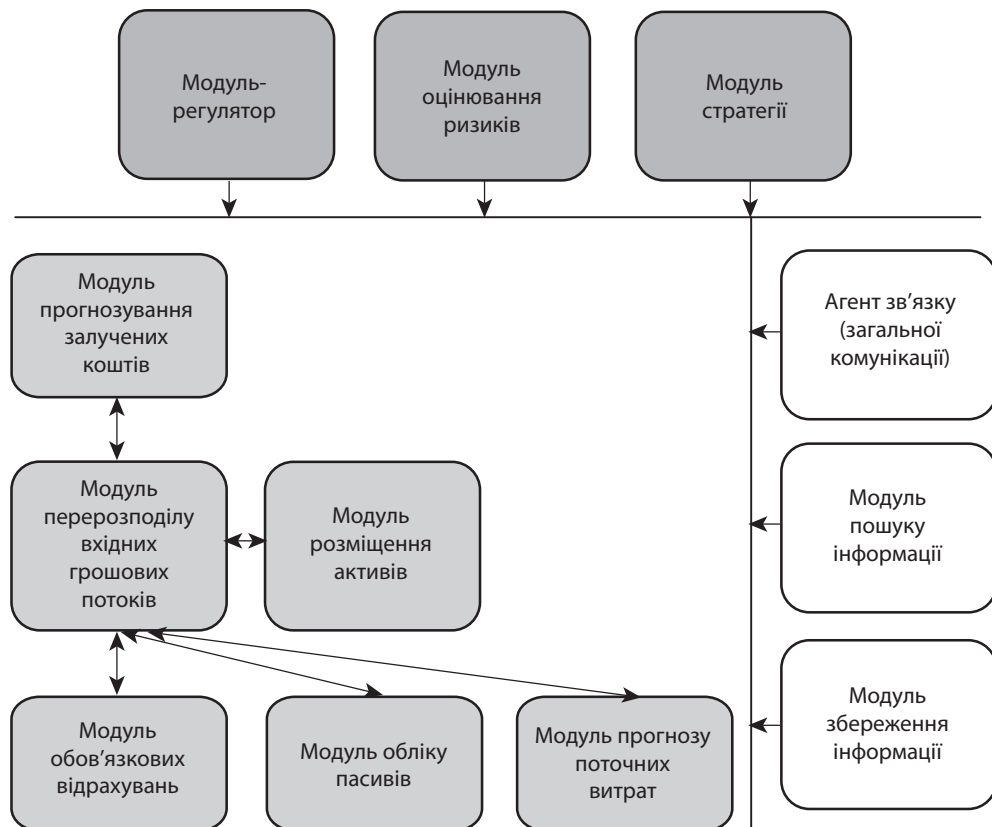


Рис. 3. Архітектура підсистеми аналізу даних

Модулі та агенти підсистеми аналізу даних

Назва	Призначення
Модуль-регулятор	Впровадження та підтримка безумовних обмежень
Модуль оцінювання ризиків	Оцінка та прогноз інтегральних показників ризику
Модуль стратегії	Впровадження та підтримка параметрів та обмежень, що диктуються обраною стратегією
Модуль прогнозування залучених коштів	Побудова прогнозу потоку коштів, залучених з урахуванням обмежень
Модуль перерозподілу вхідних грошових потоків	Прогноз повного вхідного потоку коштів та розподіл витрат згідно з пріоритетністю та діючою стратегією
Модуль розміщення активів	Розподіл ресурсів між можливими напрямками вкладення коштів
Модуль обов'язкових відрахувань	Прогноз вхідного потоку коштів, що не можуть бути відкориговані
Модуль обліку пасивів	Облік пасивів та нарахування необхідних для них відрахувань
Модуль прогнозу поточних витрат	Побудова прогнозу поточних витрат і проведення відрахувань у поточному періоді
Модуль пошуку інформації	Конвертація запитів підсистеми аналізу даних у запити підсистеми збору та зберігання даних
Модуль збереження інформації	Джерело даних підсистеми збору та зберігання даних, що може зберігати дані в постійній та тимчасовій у форматі, зручному для інших модулів підсистеми
Агент зв'язку (загальної комунікації)	Єдина загальнодоступна точка комунікації підсистеми з іншими підсистемами. Довідковий каталог агентів підсистеми

У побудованій архітектурі модулі та агента доцільно поділити на три основні класи: модулі глобального аналізу, модулі локального аналізу та технічні модулі та агенти. Модулі глобального аналізу мають вплив на будь-який інший модуль та реалізують глобальні для системи стратегії. До цього класу належать модуль-регулятор, модуль оцінювання ризиків, модуль стратегії. Модулі локального аналізу взаємодіють лише з визначеними модулями та мають на меті завдання, що впливають лише на обмежене коло модулів. До цього класу належать: модуль прогнозування залучених коштів, модуль перерозподілу вхідних грошових потоків, модуль обов'язкових відрахувань, модуль обліку пасивів, модуль прогнозу поточних витрат, модуль розміщення активів. Технічні модулі забезпечують технічні потреби роботи підсистеми. Конкретне наповнення, завдання та призначення кожного зі структурних елементів залежить від специфіки діяльності користувача.

Очевидно, що вже на даному рівні абстракції кожен модуль мультиагентної системи представляє собою не пряме відображення один до одного в реальній сутності, а є прототипом для створення кількох однакових модулів, які при взаємодії зможуть вирішувати задачі або паралельно, або працювати над різними задачами одночасно.

Представлена архітектура підсистеми призначена для вирішення кількох задач:

- ✦ *розподіл навантаження на програмному рівні.* При виборі ключового набору елементів даних їх збереження та обробку можна проводити на різних логічних вузлах, що зменшить навантаження при обробці даних великих обсягів;
- ✦ *адаптивна апаратна обробка даних.* За наявності масиву даних, який може бути ефективно оброблений з використанням специфічного

апаратного обладнання, є сенс створити окремий набір модулів, призначених для роботи на цій платформі, причому вимоги для таких модулів залишаться незмінними;

- ✦ *геолокальна обробка даних.* Для запобігання затримок при передачі великого обсягу даних у мережі виникає потреба обробки даних якомога ближче до джерела їх надходження (на фізичному рівні). Наприклад, при високошвидкісних торгах на біржі сервери для автоматичних торгів намагаються розмістити фізично якомога ближче до апаратного обладнання біржі.

При застосуванні такої архітектури підсистеми до функцій агента зв'язку додається функція перенаправлення даних до необхідного елемента системи. Тому при використанні описаної вище архітектури в межах одного структурного елемента мають бути реалізовані агенти з реалізацією таких алгоритмів, як розбиття даних на розділи за важливими ознаками та маршрутизації в межах системи до потрібного кінцевого вузла. Причому за необхідності перелік функцій агента можна розширити функцією зменшення фрагментації даних у тому сенсі, що будь-яка зміна активної топології мережі впливатиме на відповідність цих даних функції їх розподілу між вузлами.

Кожен модуль на цьому рівні абстракції представляє собою досить складний компонент, який, хоча й реалізує невелику кількість функцій, але ці функції достатньо широкі для того, щоб складність реалізації окремих модулів була досить великою.

ВИСНОВКИ

Розроблена МАС представляє собою досить складну систему. Структурні елементи, які формуються цю

систему, функціонують паралельно та можуть знаходитися фізично в більш пристосованому для їх роботи середовищі. Тобто загалом розроблену мультиагентну систему є можливість розмістити для роботи в гетерогенному хмарному середовищі зі спеціалізацією окремих частин для роботи специфічних агентів. При розробці усіх компонентів системи вона не лише здатна керувати усіма напрямками роботи користувача, але й має властивість легкого розширення функціоналу та заміни застарілих компонентів системи. ■

ЛІТЕРАТУРА

1. **Codd E. F., Codd S. B., Salley C. T.** Providing OLAP (On-Line Analytical Processing) to User-Analysts: An IT Mandate. Codd & Associates, 1993. 26 p.
2. Database Mediation using Multi-Agent Systems/A. P Luiz., P. Leme, M. A. Casanova, K. K. Breitman, A. L. Furtado. Annual IEEE Software Engineering Workshop. IEEE, 2009. P. 125–133.
3. **Marks Ch. G.** Extensible Multi-Agent System for Heterogeneous Database Association Rule Mining and Unification. Biblioscholar, 2012. 108 p.
4. Інформаційні системи в економіці: монографія/за заг. ред. д-ра екон. наук, проф. С. В. Устенка. Київ: КНЕУ, 2012. 425 с.
5. **Гужва В. М.** Мультиагентні системи в економіці: суть, елементи реалізації. *Економіка та підприємництво*: зб. наук. праць КНЕУ. 2006. № 17. С. 214–223.
6. **Булаев В. В.** Мультиагентные системы защиты баз данных. *Искусственный интеллект*. 2003. № 3. С. 432–435.
7. Tropos: An Agent-Oriented Software Development Methodology/P. Bresciani, A. Perini, P. Giorgini, F. Giunchiglia, J. Mylopoulos. *Autonomous Agents and Multi-Agent Systems*. 2004. Vol. 8. P. 203–206.
8. **Schreiber G., Weilinga B., Breuker J.** KADS: A Principled Approach to Knowledge-based Systems. London: Academic Press, 1993. 457 p.
9. **Padgham L., Winikoff M.** The Prometheus Methodology. *Methodologies and Software Engineering for Agent Systems. Series: Multiagent Systems, Artificial Societies, and Simulated Organizations*. 2004. Vol. 11. P. 217–234.

10. **Bernon C., Gleizes M.-P., Peyruqueou S., Picard G.** ADELFE: A Methodology for Adaptive Multi-agent Systems Engineering. *Engineering Societies in the Agents World III. Series: Lecture Notes in Computer Science*. 2002. Vol. 2577. P. 156–169.

REFERENCES

- Bulayev, V. V. "Multiagentnyye sistemy zashchity baz danykh" [Multi-agent system for the protection of databases]. *Iskusstvennyy intellekt*, no. 3 (2003): 432–435.
- Bresciani, P. et al. "Tropos: An Agent-Oriented Software Development Methodology". *Autonomous Agents and Multi-Agent Systems*. Vol. 8 (2004): 203–206.
- Bernon, C. et al. "ADELFE: A Methodology for Adaptive Multi-agent Systems Engineering". *Engineering Societies in the Agents World III. Series: Lecture Notes in Computer Science*. Vol. 2577 (2002): 156–169.
- Codd, E. F., Codd, S. B., and Salley, C. T. *Providing OLAP (On-Line Analytical Processing) to User-Analysts: An IT Mandate*. Codd & Associates, 1993.
- Huzhva, V. M. "Mulyahentni systemy v ekonomitsi: sut, elementy realizatsii" [Multi-agent system in the economy: essence, elements of the implementation]. *Ekonomika ta pidpriemnytstvo*, no. 17 (2006): 214–223.
- Informatsiini systemy v ekonomitsi* [Information systems in economy]. Kyiv: KNEU, 2012.
- Luiz, A. P. et al. "Database Mediation using Multi-Agent Systems". In *Annual IEEE Software Engineering Workshop*, 125–133. IEEE, 2009.
- Marks, Ch. G. *Extensible Multi-Agent System for Heterogeneous Database Association Rule Mining and Unification*: Biblioscholar, 2012.
- Padgham, L., and Winikoff, M. "The Prometheus Methodology". *Methodologies and Software Engineering for Agent Systems. Series: Multiagent Systems, Artificial Societies, and Simulated Organizations*. Vol. 11 (2004): 217–234.
- Schreiber, G., Weilinga, B., and Breuker, J. *KADS: A Principled Approach to Knowledge-based Systems*. London: Academic Press, 1993.