

ПРИНЦИПИ ФОРМАЛЬНОГО МОДЕЛЮВАННЯ ІНФОРМАЦІЙНИХ СИСТЕМ НА ОСНОВІ WEB-ТЕХНОЛОГІЙ

© Бойко Н.І., 2012

Розглядаються принципи моделювання web-орієнтованих інформаційних систем та приклади їх формального моделювання на основі CASE-технологій. Аналізується структурний підхід до проектування інформаційних систем. Узагальнюються деякі поняття проектування інформаційних систем. Зокрема, описуються принципи використання web-технологій в архітектурі інформаційних web-систем та пропонується найоптимальніший варіант для її використання. Пропонується архітектура web-системи та технології, які потрібно використати для задоволення потреб підприємства. Сформульовано основні підходи до впровадження web-технологій.

Ключові слова: розроблення інформаційної системи, технічне завдання, проект, реалізація, програмне забезпечення, інформаційна система, життєвий цикл, архітектура, декомпозиція процесів, методологія побудови інформаційних систем, CASE-технологія.

The article examines the principles of modeling web-oriented information systems. Examples of formal modeling based CASE-technology. We analyze structural approach to the design of information systems. It sums up some concept design of information systems. In particular, describes how the use of web-technologies in the architecture of web-information systems and offer the best option for its use. A web-architecture systems and technologies to be used to meet the needs of the enterprise. Formulated main approaches to implementation of web-technologies.

Key words: development of an information system specification, design, implementation, software, information system life cycle, architecture, decomposition processes, methodology of building information systems, CASE-technology.

Постановка проблеми

Основною тенденцією у розвитку сучасних інформаційних систем (ІС) є всебічне застосування засобів автоматизації підтримки прийняття рішень. Адже динаміка зміни їх структури залежить від діючої парадигми функціонування, підвищення інтелектуальності як самих систем, так і апаратних, програмних, технологічних компонент. Урахування динаміки специфічних вимог оточення, складність задач системної інтеграції диктують необхідність створення методів та засобів підтримки проектування розподіленої інформаційної системи [2, 5].

Одним із популярних сьогодні напрямів розвитку інформаційних технологій є розроблення та впровадження інформаційних систем, які побудовані на основі web-технологій. Власне, впровадження web-технологій стало новим кроком у розвитку систем архітектури “клієнт-сервер”, що надає користувачу нові сервісні можливості. Однак безконтрольний розвиток аналогічних систем може призвести до значного і не завжди виправданого перевантаження як серверної, так і комунікаційної компоненти інформаційної системи, що є вкрай важливою проблемою для підприємства. Усе це актуалізує наукове осмислення нових web-технологій, розроблення засобів оцінки ефективності функціонування web-систем, їх аналізу та оптимізації. Для проведення аналізу та оптимізації web-систем необхідно розробити відповідну формальну математичну модель її подання [6].

Аналіз останніх досліджень і публікацій

Сьогодні існує безліч публікацій на цю тематику [1, 2, 3], але, як правило, розглядаються окремо архітектура та технології. Проте загалом, система функціонує і потрібно її розглядати на різних рівнях функціонування та взаємодії. Це дасть змогу визначитись з вимогами як до програмної, так і до апаратної частини інформаційної системи.

Задачу побудови інформаційних систем оптимальної фізичної структури досліджено, зокрема, у роботах Г. Йеркса [4], К. Херрінгшоу [10] та інших вчених, проте з розвитком глобальної мережі Інтернет та web-технологій виникли принципово нові види цієї задачі. У нових умовах побудова систем оптимальної структури здійснюється з урахуванням глобального, багаторівневого характеру інформаційних систем та стохастичної поведінки користувачів такої системи. Ефективним математичним апаратом розв'язання таких задач є (як показано в роботах А.М. Пелешішина [6, 7], Є.В. Бузова [1, 2]) теорія координат.

Формулювання цілей статті

Досі не побудовано єдиної формальної моделі розподілених інформаційних систем. Проте, в деяких напрямках, проведено достатньо успішні дослідження. Однак усі вони здійснювались без наявності єдиної цілісної моделі розподілених інформаційних систем, і, як наслідок, мали частковий характер. Тому метою цієї статті є аналіз сучасних web-технологій, розвиток яких актуалізував деякі проблеми, що постають перед узагальненою моделлю розподілених інформаційних систем. Актуальність статті полягає у формалізації відкритих систем та принципів створення автономних інформаційних систем.

Виклад основного матеріалу

Сьогодні розроблено та реалізовано структурний підхід до проектування інформаційних систем, особливо на етапі отримання первинної інформації про систему. Він великою мірою орієнтований на повторне проектування ІС, що вимагає, у разі постійних змін, значних додаткових часових та грошових ресурсів [7].

Автоматизована інформаційна система, як правило, створюється для певного підприємства чи організації. Проте є багато спільних рис у структурі різних підприємств, а також у типах зв'язків (функціональних, інформаційних, зовнішніх) між елементами цієї структури. Це дає змогу сформулювати єдині принципи і шляхи побудови інформаційних систем.

Одним з ключових понять проектування інформаційних систем є життєвий цикл проекту – Project Life Cycle Management (PLCM). В загальному випадку він визначається моделлю й описується у формі методології (методу). Модель або парадигма життєвого циклу визначає загальну організацію життєвого циклу (ЖЦ) і, як правило, основні його фази та принципи переходу між ними. Адже модель життєвого циклу – це структура, що складається із процесів, робіт та задач, які охоплюють розроблення, експлуатацію і супровід програмного продукту. Модель охоплює життя системи від визначення вимог до неї до припинення її використання. А методологія (метод) визначає комплекс робіт, їх детальний зміст і рольову відповідальність спеціалістів на всіх етапах вибраної моделі ЖЦ. Вона рекомендує практики (best practices), які дозволяють максимально ефективно використовувати відповідну методологію та її модель [12].

Сьогодні вже існує структурний підхід до проектування інформаційних систем, особливо на етапі отримання первинної інформації про систему. В середовищі розробників активно використовуються об'єктні методології проектування інформаційних систем. Ці методології орієнтовані на системи, що функціонують у порівняно стаціонарному середовищі та не завжди пристосовані до динамічних середовищ.

Моделювання є важливим інструментом наукової абстракції, що допомагає виокремити, уособити та проаналізувати суттєві для певної інформаційної системи характеристики (властивості, взаємозв'язки, структурні та функціональні параметри). Розробляючи формальну модель розподіленої інформаційної системи, беремо за основу систему, котра базується на web-технологіях [5].

Для економіки, де неможливе будь-яке експериментування, особливого значення набуває математичне моделювання. Завдяки застосуванню потужного математичного апарату воно є найефективнішим і найдосконалішим методом. Своєю чергою, математичні методи не можуть застосовуватися безпосередньо щодо дійсності, а лише щодо математичних моделей того чи іншого кола явищ. Адже тенденції розвитку сучасних інформаційних технологій спричиняють постійне зростання складності інформаційних систем, які створюються у різноманітних сферах людської діяльності. Сучасні великі проекти ІС характеризуються, як правило, такими особливостями [7]:

– складність опису (велика кількість функцій, процесів, елементів даних і складні взаємозв'язки між ними), що вимагає ретельного моделювання й аналізу даних і процесів;

- наявність сукупності компонентів (підсистем), що тісно взаємодіють, виконують певні локальні задачі та цілі функціонування (наприклад додатків, пов'язаних з обробкою транзакцій і розв'язанням регламентних задач, додатків аналітичної обробки (підтримки прийняття рішень), які використовують нерегламентовані запити до даних великого обсягу);
- відсутність прямих аналогів, що обмежує можливість використання типових проектних рішень і прикладних систем;
- необхідність узгодження наявних додатків з новими розробками;
- функціонування в неоднорідному середовищі на декількох апаратних платформах;
- різномірність рівня кваліфікації та сформованих традицій використання певних наборів інструментальних засобів у групах розробників;
- істотна тривалість проекту, зумовлена, з одного боку, обмеженими можливостями колективу розробників; з іншого боку, масштабами організації-замовника і різним ступенем готовності окремих її підрозділів до впровадження ІС.

Для успішної реалізації проекту об'єкт проектування повинен бути насамперед адекватно описаний, побудовані повні й несуперечливі функціональні та інформаційні моделі ІС. Накопичений донині досвід проектування ІС показує, що це складна, трудомістка і тривала за часом робота, яка потребує високої кваліфікації фахівців, які беруть у ній участь. Однак донедавна проектування ІС виконувалося переважно на інтуїтивному рівні із застосуванням неформалізованих методів, які базуються на мистецтві, практичному досвіді, експертних оцінках і дорогих експериментальних перевірках якості функціонування ІС. Крім того, у процесі створення і функціонування ІС інформаційні потреби користувачів можуть змінюватися або уточнюватися, що ще більше ускладнює розроблення і супровід таких систем [1–4].

Традиційні підходи до побудови інформаційних систем ґрунтуються на тому, що на початку проекту складно визначити весь обсяг даних і які аналітичні задачі будуть розв'язувати кінцеві користувачі. Наприклад, методологія Oracle DWM FT (Datawarehouse Method Fast Track – метод створення сховищ даних “високошвидкісна траса”) виходить з припущення, що розробники впродовж цілого життєвого циклу інформаційної системи визначатимуть і аналізуватимуть вимоги до сховища даних. Основана на DSDM (Dynamic System Development Method – метод розроблення динамічних систем), ця методологія реалізує підхід RAD (Rapid Application Development – швидке розроблення аплікацій).

Згідно з DSDM і Oracle DWM FT, цикл проектування проходить через створення ряду прототипів доти, доки не будуть задоволені вимоги кінцевих користувачів. Розробники стверджують, що завдяки гнучкості та простоті використання інструментів Business Intelligence створення прототипів не становить труднощів. Проте застосування такого підходу є виправданим для ІТ-фахівців, які не прагнуть розширювати свої знання в цій області [3].

Методологія RAD у наш час широко застосовується у сфері створення автоматизованих інформаційних систем. Вона охоплює всі етапи життєвого циклу інформаційних систем.

Основні принципи RAD полягають в тому, що:

- використовується спіральна модель розроблення;
- повне завершення робіт на кожному етапі життєвого циклу не є обов'язковим;
- застосування CASE-засобів і засобів швидкої розробки додатків;
- тестування і розвиток проекту здійснюється одночасно з розробленням.

Засоби RAD дають можливість реалізувати нову технологію створення інформаційних систем: об'єкти формуються як деякі діючі моделі (прототипи), функціонування яких узгоджується з користувачем, а потім розробник може переходити до остаточного формування додатків, зважаючи на загальну картину проектованої системи.

Серед основних засобів методології RAD назвемо об'єктно-орієнтоване (візуальне) програмування. Метод RAD добре зарекомендував себе при створенні невеликих аплікацій. Проте, подібно до того, як при створенні складних систем транзакцій підприємства виникає потреба в перетворенні бізнес-процесів, так і при створенні інформаційних систем є необхідність у створенні структури, що орієнтована на виконання певної стратегії [8].

Підхід BSC із самого початку визначає бізнес-аспекти аналізованих даних, що дає змогу проектувати інформаційно-аналітичну систему зверху-вниз паралельно із упровадженням на підприємстві МО (Management objectives – управління, основане на досягненні меті).

Сучасні засоби візуальної розробки додатків можна розділити на дві групи – універсальні та спеціальні. Серед універсальних мов програмування найпопулярнішими є Java і C++. Серед спеціалізованих СУБД слід виділити Oracle та MySQL.

До проблем, що постають при формальному описі інформаційних систем, належить використання та розвиток засобів автоматизованого проектування (CASE-систем). Модель, що розробляється, повинна передбачати можливість використання традиційних для CASE-систем моделей та пропонувати можливі їх розширення для повнішого відображення структури системи [9, 10].

Зазначені вище фактори сприяли появі програмно-технологічних засобів спеціального класу – CASE-засобів, що реалізують CASE-технологію створення і супроводу ІС. Термін CASE вживають у наш час у досить широкому сенсі. Первісне значення терміна CASE, обмежене питаннями автоматизації розробки тільки програмного забезпечення (ПЗ), сьогодні набуло нового сенсу, що охоплює процес розробки складних ІС загалом. Тепер під терміном “CASE-засіб” розуміють програмні засоби, що підтримують процеси створення і супроводу ІС, враховуючи аналіз і формулювання вимог, проектування прикладного програмного забезпечення (ПЗ) (додатків) і баз даних, генерацію коду, тестування, документування, забезпечення якості, конфігураційне керування і керування проектом, а також інші процеси. CASE-засоби разом із системним ПЗ та технічними засобами утворюють повне середовище розробки ІС [5].

Появі CASE-технології та CASE-засобів передували дослідження з методології програмування. Програмування набуло рис системного підходу з розробленням і впровадженням мов високого рівня, методів структурного і модульного програмування, мов проектування і засобів їхньої підтримки, формальних і неформальних мов описів системних вимог і специфікацій тощо.

CASE-технологія являє собою методологію проектування ІС, а також набір інструментальних засобів, що дозволяють у наочній формі моделювати предметну область, аналізувати цю модель на всіх етапах розробки і супроводу ІС і розробляти додатки відповідно до інформаційних потреб користувачів. Більшість сучасних CASE-засобів основані на методологіях структурного (переважно) або об’єктно-орієнтованого аналізу і проектування, що використовують специфікації у вигляді діаграм або текстів для опису зовнішніх вимог, зв’язків між моделями системи, динаміки поведінки системи та архітектури програмних засобів [4].

Засіб XTG Data Modeller (XTGDM) є CASE-засобом для структурного моделювання даних та розроблення баз даних. Він здатний моделювати як прості структури даних, так і складні інформаційні системи. Це видається важливим, якщо досягнуто моменту, коли модель даних є настільки складною, що простих засобів – ручки і паперу недостатньо. Система XTGDM забезпечує моделювання структури даних інформаційної системи.

Структурний аналіз (проектування) є сьогодні найпоширенішою методологією аналізу (проектування) розподілених інформаційних систем. Головними інструментами структурного аналізу є спеціальні діаграмні техніки (ERD – діаграми “сутність – співвідношення”, DFD – діаграми потоків даних, FHD – діаграми функціональної ієрархії, LCD – діаграми з’єднань розташувань). В основу структурного аналізу покладено: принцип ієрархічного впорядкування та принцип “поділяй та владарюй” [11]. Ці принципи декларують побудову моделі системи як деревоподібної ієрархічної структури. Кожен рівень структури має власну семантику, яка деталізується на нижчому рівні [8].

Структурні методології зарекомендували себе як потужний засіб аналізу, проектування та розроблення розподілених систем. Як наслідок, за допомогою цих методологій створено велику кількість систем, напрацьовано багато типових рішень та розроблено ряд спеціалізованих корпоративних технологій. Більшість засобів автоматизації системного аналізу та проектування (CASE-засобів) значною мірою орієнтовані на підтримку структурних методологій. Формальна модель повинна максимально використовувати та інтегруватися з наявними моделями, створеними на базі структурних методологій.

Використовуючи описані вище підходи, перейдемо до розгляду формального опису ІС. Виділимо компоненти системи – її входи та виходи. Входом системи є множина запитів клієнтів системи, виходами – множина відповідей системи на запити. У загальному випадку реакція визначається не лише поточним запитом, а й усіма попередніми. Для того щоб не обробляти щоразу усі попередні запити клієнта, вводиться поняття стану інформаційної системи. Стан інформаційної системи є агрегованою історією запитів до системи [3, 9].

У системі виділяються окремі рівні, де на кожному міститься не менше від однієї автономної підсистеми (яка фактично є окремою системою). На кожному з рівнів (крім верхнього та нижнього) традиційно розглядаються три вхідних потоки – потік зовнішніх збурень, потік розпоряджень з вищого рівня, потік реакцій нижчого рівня. Перші два потоки в моделі розподілених інформаційних систем ототожнюються, оскільки вища система є клієнтом системи нижчого рівня. Отже, запити з системи вищого рівня, з одного боку, можуть розглядатися як вхідні збурення, а з іншого – як запити вищого рівня [6].

Формальна модель повинна забезпечувати можливість ієрархічного з'єднання систем та їх компонент. Таке з'єднання базується на спрямуванні вихідного потоку нижчої системи у вищу систему, що разом з вхідним збудженням утворює вхідний сигнал для вищої системи. Тому, враховуючи вищесказане, розробляючи формальну модель розподіленої інформаційної системи, беремо за приклад системи, які базуються на web-технологіях. Подальший розвиток цих технологій обмежений або взагалі неможливий без єдиного формального вирішення усіх вищезгаданих проблем [7].

Постає питання, які технології та яка архітектура інформаційних web-систем відповідатимуть таким умовам: простота у супроводженні – користуванні, поширеність (легко знайти технічних спеціалістів для супроводження, підтримки системи, модернізації, розширення тощо). Вибір систем, орієнтованих на web-доступ, пояснюється досить просто – крос-платформність, спрощена система використання “Logon and Go” – потрібен лише браузер, логін та пароль, доступ з корпоративної локальної мережі (Intranet) – будь-яке підключення; якщо потрібен доступ з мережі Internet – доступ до Internet, і системою вже можна користуватись [11].

Будуючи формальну модель web-системи, слід розглядати наявні методи та засоби опису протоколів інформаційних мереж та взаємодії відкритих систем. Виділимо класи засобів та методологій, які можуть бути відправною точкою для подальшого розроблення формальної моделі розподілених інформаційних систем взагалі та web-систем зокрема [6, 7]:

- засоби опису мережевих протоколів;
- засоби та методи структурного аналізу, CASE-системи.

Оглянувши основні проблеми, що постають перед моделюванням розподілених ІС типу web-систем, перейдемо до формального опису таких ІС (використовуючи описані вище підходи). Для проведення аналізу та оптимізації web-систем доцільно використовувати відповідну формальну математичну модель її подання. У деяких напрямках дослідження розробки проводилися достатньо успішно. Серед цих напрямів виділимо такі [7]:

- дослідження та розроблення засобів моделювання та системного аналізу інформаційних систем, зокрема їх мережевої та функціональної компоненти;
- дослідження та розроблення засобів моделювання протоколів інформаційних систем та мереж;
- дослідження фізичних характеристик функціонування розподілених інформаційних систем найрізноманітніших архітектур;
- дослідження поведінки розподілених систем, що складаються з автономних компонентів;
- дослідження проблем обробки гіпертекстових та гіпермедійних даних.

Однак усі ці дослідження проводилися без єдиної цілісної моделі розподілених інформаційних систем, і, як наслідок, мали частковий характер. Адже розвиток інформаційних технологій актуалізував деякі проблеми, що постають перед узагальненою моделлю розподілених інформаційних систем. Зокрема, це формалізація відкритих систем та принципів створення систем з автономних компонентів (визначення моделі стандартних об'єднань компонентів у систему), формалізація та узагальнення архітектури “клієнт-сервер”, опис систем з практично необмеженою кількістю компонентів (у глобальних мережах), оптимізація систем з гетерогенною мережевою компонентою тощо [7].

Під час моделювання систем архітектури “клієнт-сервер”, враховуючи те, що будь-яка дія системи визначається запитом клієнта, за формальний параметр часу доцільно приймати номер за порядком запиту клієнта. Множина номерів запитів – множина натуральних чисел. Існує суттєва відмінність між реальним фізичним часом та формальним часом інформаційної системи: так, система, що функціонує коротший час, може бути формально старішою – обробити більшу кількість запитів. У межах аналізу поведінки web-систем (зокрема конкретних апаратних та мережових рішень) доцільно проводити зіставлення реального та формального часу. Основний акцент у моделюванні перспективної поведінки системи повинен ставитись на терміни формального часу.

Одним з головних напрямів аналізу поведінки системи є визначення її поведінки в перспективі – тобто якщо зростає до нескінченності кількість запитів системи. Інформаційна система є динамічною системою в розумінні теорії формальних систем [6]. Для визначення реакції системи на отриманий запит достатньо лише інформації про стан системи (вміст її бази даних).

Системи, які не відповідають умові стійкості ІС від початкових запитів, мають погано прогнозовану поведінку і, як наслідок, їх важко оптимізувати. Тому такі ІС слід перекомпоновувати так, що вони були стійкими відносно початкових запитів. Стійку щодо початкових запитів систему, в якій немає практично недосяжних станів, за аналогією з [6] називатимемо повною динамічною. У повній динамічній системі всі стани є доступними на будь-якій ітерації, достатньо далекій від початку функціонування системи.

Під час створення будь-якої системи потрібно приділяти увагу вибору архітектури та технології, які будуть використовувати. Проблема вибору web-технологій для інформаційної системи, орієнтованої на web-доступ (Web-Based), є актуальною. Адже оптимальний вибір забезпечить розширений термін життєвого циклу системи, просту підтримку та зручність у її використанні в короткостроковій та довгостроковій перспективі.

Висновки та перспективи подальших досліджень

Сьогодні web-системи стали одним з основних видів інформаційних систем, які функціонують у глобальних мережах. Проте універсальних підходів до проектування, розробки та їх функціонування досі не створено. Зокрема, чітко не визначено критеріїв їх якості та ефективності.

Аналіз формальної моделі web-системи дає змогу зробити висновок про принципову схожість класичних інформаційних систем, що будуються на технологіях баз даних, та web-систем. Фактично, йдеться про узагальнення понять баз даних у моделях найрізноманітніших інформаційних систем (таких як web-системи). База даних системи у такому разі розглядається у ширшому розумінні, ніж просто набір спеціальних об’єктів (наприклад таблиць), що обробляє та підтримує певна система управління базами даних (СУБД). Навіть набір статичних web-сторінок, що зберігаються як окремі файли у файловій системі сервера, розглядається як база даних. Модель даних у будь-якому випадку повинна належно документуватись та проектуватись. Адже інформаційна система є динамічною системою в розумінні теорії формальних систем і для визначення реакції системи на отриманий запит достатньо лише інформації про стан системи (вміст її бази даних).

1. Буров Є.В. *Аналіз та оптимізація інформаційних потоків у сучасних internet та intranet системах* / Є.В. Буров, А.М. Пелецишин // *Вісник Держ. ун-ту “Львівська політехніка”*. – 1998. – № 330. – С. 27–34.
2. Буров Є.В. *Оптимізація розміщення даних у Web-системах* / Є.В. Буров, А.М. Пелецишин // *Вісник Держ. ун-ту “Львівська політехніка”*. – 1998. – № 330. – С. 17–27.
3. Дмитриева М.В. *JavaScript. Экспресс-курс [Текст]* / М.В. Дмитриева. – СПб: БХВ-Петербург, 2005. – 336 с.
4. Йеркса Г. *Механизмы, балансирующие нагрузку Web-узлов* / Г. Йеркса // *Сети и системы связи*. – 1999. – № 5. – С.120–126.
5. Кін. А. *Секрети Інтернету : навч. посіб.* / А. Кін. – Ростов н/Д.: Фенікс, 2009. – 320 с.
6. Пелецишин А.М. *Методи та алгоритми оптимізації Web-систем* / А.М. Пелецишин // *Вісник Держ. ун-ту “Львівська політехніка”*. № 406. – Львів: 2000. – С. 199–211.
7. Пелецишин А.М. *Принципи формального моделювання інформаційних систем на базі Web-технологій* / А.М. Пелецишин, Є.В. Буров // *Вісник Держ. ун-ту “Львівська політехніка”*. – 1999. – № 383. – С. 178–201.
8. Хантер Д. *XML. Базовий*

курс [Текст]: [пер. с англ.] / Д. Хантер, Дж. Рафтер, Дж. Фаусетт, Е. Ван дер Влиет. – 4-е изд. – М: ТОВ “Вильямс”, 2009. – 1344 с. 9. Храмов П.Б. Основы Web-технологий / П.Б. Храмов, С.А. Брик, А.М. Русак. – СПб: БХВ-Петербург, 2007. – 480 с. 10. Хэррингшоу К. Оптимизация сервера Web / К. Хэррингшоу // LAN: журнал сетевых решений. – 1997. – № 8. 11. [Электронный ресурс]. – Режим доступа: <http://www.osp.ru/lan/1997/08/111.htm>. 12. [Электронный ресурс]. – Режим доступа: http://www.rusnauka.com/1_NIO_2011/Informatica/78189.doc.htm.

УДК 681.3; 004.896

О.Ю. Бочкарьов

Національний університет “Львівська політехніка”,
кафедра електронних обчислювальних машин

ПРОБЛЕМА ОРГАНІЗАЦІЇ АДАПТИВНИХ ВИМІРЮВАЛЬНО-ОБЧИСЛЮВАЛЬНИХ ПРОЦЕСІВ У АВТОНОМНИХ РОЗПОДІЛЕНИХ СИСТЕМАХ

© Бочкарьов О.Ю., 2012

Розглянуто проблему організації адаптивних вимірювально-обчислювальних процесів у автономних розподілених системах. Запропоновано відповідну модель, механізм координації та алгоритм управління адаптивним вимірювально-обчислювальним процесом.

Ключові слова: адаптивний вимірювально-обчислювальний процес, автономна розподілена система.

The problem of organization of adaptive measurement and computational processes in the autonomous distributed systems is considered. The corresponding model, coordination mechanism, and control algorithm are proposed.

Key words: adaptive measurement and computational processes, autonomous distributed system.

Вступ

Інтенсивний розвиток та впровадження інформаційних технологій в усіх сферах людської діяльності супроводжується збільшенням автономності та масштабу відповідних комп'ютерних та програмних систем. Відтак дослідження та розробки в області автономних розподілених систем (АРС) набувають все більшої актуальності [1–7]. Одним з ключових моментів у роботі таких систем є організація вимірювально-обчислювальних процесів (ВО-процесів), які: 1) збирають інформацію про оточення АРС, про її об'єкт управління (цільове середовище) та про роботу самої АРС; а також 2) виконують попередню обробку зібраної інформації для її подальшого використання системами управління та іншими компонентами АРС. При цьому внаслідок автономності та розподіленості таких систем виникає потреба у використанні методів адаптації ВО-процесів до збурень у оточенні АРС, недетермінованих змін об'єкта управління та змін у роботі самої АРС (викликаних, наприклад, частковою відмовою її вузлів чи зменшенням запасу енергії). У найцікавіших випадках йдеться про делегування значної частини повноважень із прийняття рішень щодо роботи АРС адаптивним ВО-процесам на основі концепції інтелектуального автономного агента (intelligent autonomous agent) та технологій багатоагентних систем (multi-agent systems) [8, 9]. У межах цього підходу окремий вимірювально-обчислювальний вузол АРС отримав назву “вимірювальний агент” [1–7], поведінку якого реалізує відповідний адаптивний ВО-процес. У цій роботі розглянуто актуальну проблему організації адаптивних ВО-процесів як відображення та результату колективної поведінки автономних вимірювальних агентів.