

І. І. Кушнірецька¹, О. І. Кушнірецька¹, А. Ю. Берко²
Національний університет “Львівська політехніка”,
¹кафедра інформаційних систем та мереж,
²кафедра загальної екології та екоінформаційних систем

АНАЛІЗ ІНФОРМАЦІЙНИХ РЕСУРСІВ СИСТЕМИ ДИНАМІЧНОЇ ІНТЕГРАЦІЇ СЛАБОСТРУКТУРОВАНИХ ДАНИХ У WEB-СЕРЕДОВИЩІ

© Кушнірецька І. І., Кушнірецька О. І., Берко А. Ю., 2015

Описано порядок аналізу інформаційних ресурсів системи динамічної інтеграції слабоструктурованих даних у web-середовищі для визначення спільних рис інформаційних ресурсів і виявлення зв'язків між ними. На основі моделі загального визначення ресурсів і мови опису правил доступу до ресурсів розглянуто процес створення об'єктного адаптера визначення спільних рис інформаційних ресурсів і виявлення зв'язків між ними за допомогою правил методу «чорної скриньки». Описано процес визначення структурно-динамічної моделі предметної області Мешап-додатка.

Ключові слова: інформаційний ресурс, динамічна інтеграція, Мешап-додаток, об'єктний адаптер, метод “чорної скриньки”, агентно-орієнтований підхід.

This paper describes the analysis of information resources of dynamic integration system the semi-structured data in a web to determine common features of information resources and identify relationships between them. The process of creating the object adapter of identify common features of information resources and identify relationships between them using the rules of the method of "black box" method has been considered, based on the model of the general definition of resource description language and rules of access to resources. The process of determining the structural and dynamic domain model of mash-up application has been described.

Key words: information resource, dynamic integration, mash-up application, wrapper of information resource, “black box” method, agent-oriented approach.

Вступ. Загальна постановка проблеми

Зростання кількості ресурсів і послуг, доступних в Інтернеті, і можливостей, що пропонує Web 2.0, спонукає кінцевих користувачів еволюціонувати від пасивних споживачів інформації до інформаційних виробників, які можуть отримати доступ до наявних інформаційних ресурсів і маніпулювати ними для того, щоб генерувати новий зміст. З погляду людино-машинної взаємодії це вимагає нових парадигм комунікацій, які повинні дозволити людям отримати доступ до контенту, перемістити його в особисті інтерактивні робочі області, де вони можуть інтегрувати його, а також, за потреби, створювати новий контент, використовуючи наявний. Розв'язання цієї задачі вимагає застосування способів і засобів динамічної інтеграції даних. Система інтеграції даних звільняє користувачів від необхідності знати, дані з яких джерел, крім інтегрованого, вони використовують, які властивості цих джерел і як здійснити доступ до них [4].

Сьогодні, внаслідок бурхливого розвитку Інтернету, накопичена і постійно зростає величезна кількість інформаційних ресурсів різноманітного характеру і змісту. Звідси виникає проблема їх оперативної, динамічної інтеграції, з виділенням змісту інформації та збереженням семантики даних. Складність і характер використовуваних методів вирішення цієї проблеми істотно залежить від рівня інтеграції, який необхідно забезпечити, властивостей окремих джерел даних і всієї сукупності джерел загалом та необхідних способів інтеграції.

Зв'язок висвітленої проблеми із важливими науковими та практичними завданнями

У зв'язку зі стрімким зростанням українських і світових інформаційних ресурсів методи визначення спільних рис інформаційних ресурсів та динамічної інтеграції інформації перебувають у центрі уваги дослідників федеративних середовищ і становлять науково-практичний інтерес для розробників сучасних розподілених інтелектуальних інформаційних систем, але численні проблеми все ж залишаються невирішеними. Тому сьогодні особливо актуальна задача розроблення нових підходів, технологій, архітектурних рішень і гнучких інструментальних засобів для аналізу інформаційних ресурсів системи динамічної інтеграції слабоструктурованих даних у web-середовищі для визначення спільних рис інформаційних ресурсів і виявлення зв'язків між ними.

Аналіз останніх досліджень та публікацій

Стосовно динамічної інтеграції даних у web-середовищі все більшої актуальності набувають дослідження роботи і розроблення систем інтеграції даних, що працюють, використовуючи технологію «Мешап» (англ. Mash-Up) [4]. Мешап – це такий підхід до розроблення додатків, який дає змогу користувачам об'єднувати дані з декількох джерел в один інструмент інтеграції [4]. Деякі Мешапи можуть створюватися простим поєднанням JavaScript-коду з XML, що уможливує побудову нового інноваційного web-сервісу. Інші, більші Мешапи, які є основою відповідних web-сайтів, використовують технологію таких сервісів, як Google Maps, і базу даних адрес, пов'язуючи їх і показуючи інформацію про проект на карті. На відміну від композицій web-сервісів, де акцент робиться тільки на об'єднання бізнес-сервісів, межі Mashup розширюються, що забезпечує більше функціональностей і дозволяє інтегрувати різноманітні ресурси, такі як, наприклад, сервіси передавання даних. Програми, створені за допомогою технології Mash-Up, називаються «Мешапами» (англ. Mashups), або «Мешап-додатками» (англ. Mashup applications). Дослідження показують, що існує високий рівень інтересу в межах Mashup. А через стрімкий розвиток ІТ-технологій особливої актуальності набувають потреби в динамічній інтеграції різноманітних даних у web-середовищі. Технологія «Мешап» відкриває нові й широкі можливості для передавання даних споживачам. Проте незмінним залишається той факт, що користувач Мешапу повинен знати, як мінімум, те, як писати програмний код, використовуючи мови програмування (наприклад, Java Script, XML/HTML), як використовувати різні web API. Для того, щоб вирішити цю проблему, чимало зусиль докладають для розроблення інструментів, які призначені для підтримки користувачів з незначним знанням програмування для розроблення Мешап-додатків.

Вже розроблено кілька методів та засобів для вирішення проблеми визначення спільних рис інформаційних ресурсів та автоматичного вилучення структурованої інформації з веб-сайтів і додатків, які, через специфіку характеру HTML мови, як правило, надають неструктуровану інформацію. Ці методи, як правило, розроблені на основі певних правил видобування даних, які можуть бути менш або більш витонченими, і визначаються або в напівавтоматичному режимі за допомогою демонстрації, або автоматично за допомогою машино-навчальних алгоритмів або інших методів штучного інтелекту. Огляд декількох засобів видобування web-даних подано в роботі [5].

Хоча ці підходи істотно вирішують проблему отримання даних з веб-додатків для цілей автоматичної та інтегрованої навігації по web-сторінках або для міграції в напрямку Semantic Web, інші автори представляють свій досвід з міграцією функціональних веб-додатків до web-служб. Інакше кажучи – це дослідження в напрямку створення Мешап-сервісів.

В роботі [3] пропонують техніку «білої скриньки» зворотного проектування і засіб для створення об'єктних адаптерів, що робить функціональні можливості клієнт-серверного .NET додатка доступними як web-сервіси. Цзян і Струоля [6] описують свою роботу побудови веб-сервісів з функціональних, пропонованих веб-сайтів: їх підхід оснований на аналізі пар HTTP запитів, виданих браузером і відповідних HTML відповідей, що видаються сервером для вибору функціональностей для зазначення відповідно до специфікацій WSDL. Баумгартнер та ін. [1] запропонували рішення для отримання веб-сервісів з додатків з веб-інтерфейсами: рішення містить візуальний інструмент, який дає змогу вилучити необхідну інформацію з HTML-документів і перевести їх в XML формат, який можна використовувати в контексті веб-сервісів.

Методику «чорної скриньки» для міграції функціональностей системи до SOA запропонували Канфора і співавтори [2]. Цей метод спрямований на виявлення інтерактивних функціональних можливостей систем на основі форм у вигляді сервісів. Проблема перетворення оригінального користувацького інтерфейсу системи на запит/відповідь інтерфейс у SOA вирішується за допомогою об'єктного адаптера (Wrapper), який здатен взаємодіяти з додатком, володіючи правилами діалогу між користувачем та програмою. Ці правила визначаються користувацьким інтерфейсом (UI) моделі, що спрямована на кінцевий стан автомата (Finite State Automata), який інтерпретується автоматичною машиною і може бути отриманий реверсивною інженерною UI-технікою. Цей міграційний підхід підтверджений серією тематичних досліджень, які показали свою ефективність. Вагнер [7] пропонує агентно-орієнтований підхід для концептуального моделювання організацій та організаційних інформаційних систем під назвою агентно-об'єктне моделювання зв'язків Agent-Object-Relationship: коли деяка сутність є агентом, подією, дією, вимогою, зобов'язанням чи звичайним об'єктом і коли особливі відносини між агентами, подіями, діями, вимогами і зобов'язаннями доповнює фундаментальний зв'язок, агрегація/організація і генералізація типів відношень сутність-зв'язок (Entity-Relationship) і моделювання UML-класу.

Оскільки наповнення, природа і структура Мешап-додатка не можуть бути повністю заздалегідь відомі, адже Мешап формується як відповідь на користувацький запит, то найдоцільніше вибрати таку методику побудови функціонування системи, яка б найбільше відповідала зазначеним вище критеріям. Тому пропонуємо використати поєднання об'єктно-орієнтованого підходу до моделювання та методу «чорної скриньки».

Відповідно до методики «чорної скриньки» та агентно-орієнтованого підходу [7] виконаємо аналіз інформаційних ресурсів системи динамічної інтеграції слабкоструктурованих даних у web-середовищі для визначення спільних рис інформаційних ресурсів у web-середовищі для того, щоб, застосовуючи Мешап-технологію, динамічно інтегрувати слабкоструктуровані дані.

Відмітною рисою агентно-орієнтованого підходу є те, що в базовому знанні про середовище функціонування локального агента не повинно міститися всіх відомостей про всі інформаційні ресурси – повна модель. Завдання, алгоритми, специфікації об'єктів, необхідних для виконання завдання, завантажуються з агента-сервера у вигляді сценарію і потрібного подання – зрізу моделі предметної області. Поданням даних прийнято називати форму загального представлення знань предметної області [7]. Представлення тих самих реальних інформаційних об'єктів у різних інформаційних системах може бути різним.

Виділяють три основні класи моделей даних: модель структурованих даних (детермінована схема даних); модель неструктурованих даних (недетермінована схема даних); змішані моделі даних (частково детермінована схема). Як тип моделі для проектування мультиагентної системи з метою обміну інформацією між автономними розподіленими інформаційними ресурсами доцільно використовувати модель з частково детермінованою схемою даних. Цей тип моделі характеризують такі властивості: сукупність типів властивостей і типів взаємозв'язків об'єкта з іншими об'єктами динамічно визначається тільки в момент появи кожного реального об'єкта в «фокусі» інформаційної системи; частково детермінована схема даних; можуть відображатися не тільки спільні властивості та відношення об'єктів предметної області, а й індивідуальні; схема добудовується динамічно з надходженням нового факту, тобто довизначаються деякі властивості і відношення. Досліджувана предметна область – web-середовище, елементами якого є користувацькі логічні уявлення про семантику і структуру інформаційного об'єкта.

Концептуальна модель web-середовища повинна забезпечувати: засоби представлення даних; засоби і методи подання семантичних правил, що обмежують множину допустимих станів (простір даних); засоби ідентифікації даних, які дають змогу виділяти підмножини об'єктів інформаційної системи, необхідних для аналізу конкретних завдань; засоби маніпулювання даними.

Під структурою даних у загальному випадку розуміють множину, що складається з елементів даних, і множину зв'язків між ними. Таке визначення охоплює всі можливі підходи до структуризації даних, але в кожній конкретній задачі використовуються ті чи інші його аспекти [7]. Розгляд структури даних без урахування її подання до пам'яті комп'ютера називається абстрактною

або логічною структурою. Виділяють такі рівні глобальності логічних структур в обчислювальному середовищі: глобальні мережі; локальні мережі; окремо взятий персональний комп'ютер; логічні диски в персональному комп'ютері; логічні папки; файли; елементи файла; елементи даних.

Своєю чергою, комбінації таких логічних структур утворюють складніші складові логічні структури. Розрізняють прості (базові, примітивні) та інтегровані (структуровані, композитні, складні) структури (типи) даних. З логічного погляду прості дані є неподільними одиницями. Інтегрованими називаються такі структури даних, складовими частинами яких є інші структури даних – прості або, своєю чергою, інтегровані. Перераховані логічні структури даних називають сутностями (типами) предметної області. Всі сутності перебувають у деяких відношеннях один з одним, їх кількість і кількість відношень між ними може бути досить великою залежно від погляду на предметну область. Також сутності можуть мати доволі багато властивостей – характеристик залежно від моделі предметної області. Відношення між сутностями також можуть мати як завгодно багато властивостей – характеристик.

Деякі різні (фізичні та логічні) інформаційні об'єкти можуть мати загальні характерні властивості. Тому доцільно розглядати не тільки екземпляри сутностей (конкретні інформаційні об'єкти), а й абстрактнішу сутність, яка у певний спосіб узагальнює ці примірники у деяку множину з характерними загальними для всіх елементів властивостями – типами об'єктів. Наприклад, тип текстових документів = $\{*.Doc\}, \{*.Txt\}, \dots$ – штучно створений тип, який має загальну властивість для всіх примірників – функціональне призначення = робота з текстами. Зауважимо, що той самий інформаційний об'єкт може одночасно тимчасово належати кільком типам об'єктів і по-різному брати участь в логічних зв'язках з іншими сутностями залежно від приналежності до типу. Таке уявлення приналежності структури або примірника структури до деякого типу дає змогу розглядати предметну область як динамічну, а перехід від типу до типу – як перехід між уявленнями моделі предметної області (ПО) залежно від фокусу завдання або користувача, тобто трансформацію моделі ПО.

«Чорна скринька» — це термін, який вживають у техніці й кібернетиці для позначення об'єкта чи системи, про принципи дії яких нічого не відомо, крім того, що певному вхідному сигналу відповідає певний вихідний сигнал [2].

Моделі «чорної скриньки» дають змогу відобразити ті входи та виходи системи, які необхідні для вивчення одного з аспектів її функціонування, тому називаються моделями «вхід–вихід». Під час побудови такої моделі встановлюється відношення між цими входами та виходами. Модель «вхід–вихід» відображає основні властивості системи, такі як цілісність та відносна ізольованість через наявність зв'язку із зовнішнім середовищем [2].

Під час побудови моделі «вхід–вихід» проблемою є визначення тих входів та виходів, які необхідно вводити до складу моделі, оскільки у процесі вивчення системи модель постійно модифікується. Реальна система взаємодіє із середовищем через нескінченну кількість способів, тобто через нескінченну кількість входів та виходів. Критерієм відбору цих входів та виходів є цільове призначення системи, суттєвість того чи іншого зв'язку системи із середовищем. У моделі ми вимушені відображати скінченну кількість взаємодій, тому висока ймовірність невведення саме тих входів та виходів, які найістотніше визначають властивості системи.

Проблема вибору найважливіших входів та виходів виражається ще у тому, що зв'язки із зовнішнім середовищем, які не входять у систему, не зникають, а у певний спосіб діють незалежно від нас і тим самим ускладнюють вивчення поведінки системи. Під час побудови моделей «вхід–вихід» невраховані та невідомі зв'язки із зовнішнім середовищем представляють за допомогою спрощених моделей невизначеності [2]. Сучасні підходи до побудови моделей «вхід–вихід» побудовані на відображенні однієї з таких форм невизначеності: стохастичної, теоретико-множинної або їх комбінації. У цих випадках модель системи розглядається у вигляді «чорної скриньки» (рис. 1) [2] з такими групами параметрів: керуючі (вхідні) параметри X_i , які називаються факторами; вихідні параметри Y_i , які називаються параметрами стану; W_i параметри – впливи.

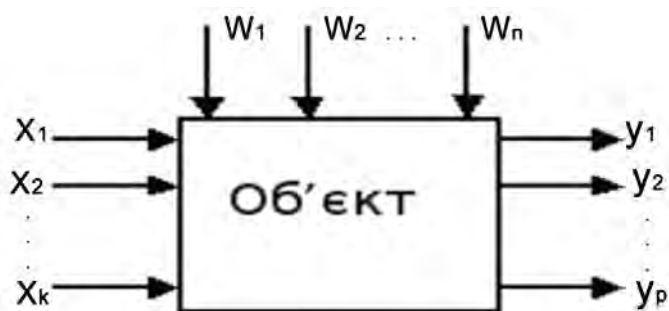


Рис. 1. Модель системи у вигляді «чорної скриньки» [2]

Як зазначено у [2], для побудови моделі використовують методи теорії ідентифікації. У загальному випадку завдання ідентифікації формулюється так: на основі результатів спостереження за вхідними та вихідними змінними системи потрібно побудувати оптимальну в деякому розумінні математичну модель. Основні етапи ідентифікації такі: вибір класу і структури моделі та мови її опису; вибір класу і типів вхідних впливів X ; обґрунтування критеріїв подібності системи та моделі; вибір методу ідентифікації та розроблення відповідних алгоритмів оцінювання параметрів системи; перевірка адекватності отриманої в результаті ідентифікації моделі.

Виділення проблем

Створення моделі системи динамічної інтеграції слабоструктурованих даних у web-середовищі з використанням мешап-технології вимагає, передусім, роботи, що стосується аналізу інформаційних ресурсів Мешап-системи для визначення спільних рис і виявлення зв'язків між ними, з використанням правильної методології. Актуальність завдання визначення спільних рис інформаційних ресурсів і виявлення зв'язків між ними, розподілених у web-середовищі, підтверджується існуванням великого класу задач. Це задачі моніторингу, інтеграції, підтримки обмежень цілісності регіонального інформаційного простору, зокрема проблеми, пов'язані з динамічною інтеграцією гетерогенних інформаційних ресурсів. Отже, актуальна задача розроблення нових підходів, технологій, архітектурних рішень і гнучких інструментальних засобів для динамічної інтеграції слабоструктурованих даних у web-середовищі. Тому впровадження нових методологій визначення спільних рис інформаційних ресурсів і виявлення зв'язків між ними з метою полегшення зберігання, організації, доступу та аналізу інформації в оперативному режимі видається перспективним і конструктивним у плані вирішення поставлених проблем.

Формулювання мети

Мета роботи полягає у використанні відомих підходів для формування послідовності аналізу визначення спільних рис інформаційних ресурсів і виявлення зв'язків між ними для того, щоб створити модель системи, яка дає змогу кінцевим користувачам отримувати уніфіковану інформацію з різних гетерогенних джерел даних.

Аналіз отриманих наукових результатів

Робота з інформаційними ресурсами як із сукупністю даних, що зберігаються в комп'ютерних системах, з метою подальшого ефективного використання – ідентифікації, вилучення, обробки, аналізу, обміну, в межах завдань власників інформаційних ресурсів, охоплює: зберігання; пошук необхідного інформаційного ресурсу; автоматизацію доступу до інформаційного ресурсу; вилучення інформації в уніфікованому вигляді; обробку даних (виконання завдань); аналіз даних; визначення спільних рис інформаційних ресурсів; обмін інформацією; контроль наявності всіх необхідних ресурсів; контроль виконання всіх етапів роботи з інформаційним ресурсом (моніторинг транзакцій); висновок – результат в заданій формі (дані/завдання).

Обмін результатами виконаних завдань передбачає: визначення учасників обміну (задача ідентифікації); визначення існування ресурсу і повноважень доступу до нього (задача моніторингу); перевірка успішності завершення транзакції обміну.

Завдання ідентифікації інформаційних ресурсів або деякого інформаційного об'єкта (далі – об'єкта) поділяється на кілька етапів, серед яких можна виділити основні:

- завдання умови виділення деякого об'єкта серед всієї сукупності об'єктів – визначення пошукового образу;
- процес розпізнавання об'єкта – перевірка задоволення умовами пошукового образу характеристик об'єкта сканування.

До основних засобів, що використовуються для забезпечення інтеграції інформаційних ресурсів, належать конвертори даних, що інтегрують моделі даних, механізми відображення моделей даних, об'єктні адаптери (Wrappers), посередники (Mediators), онтологічні специфікації, засоби інтеграції схем та інтеграції онтологічних специфікацій, а також архітектура, що забезпечує взаємодію засобів, що використовуються в конкретній системі інтеграції ресурсів.

Об'єктні адаптери (Wrappers) – компоненти, пов'язані з джерелами даних для вирішення проблем технічної неоднорідності та неоднорідностей метамodelей. Їх основні функції: приймають запити до джерела на деякій мові, перетворюють запит мовою джерела, виконують запит, відправляють результати посереднику (Mediator). Відповідно до запропонованого у роботі [2] підходу методу «чорної скриньки» аналізуємо інформаційні ресурси Мешап-системи у web-середовищі для того, щоб визначити спільні риси інформаційних ресурсів і сформулювати зв'язки між ними. Мешапи, зазвичай, інтегрують дані з різнорідних джерел даних, таких як: бази даних, пошукові системи або локальні файли. Адаптери перетворюють дані на конкретні джерела і самоописи XML-структури для рівномірного доступу до даних у структурі. Як правило, отримана XML-структура описує набір об'єктів, таких як люди, продукти, публікації або посилання (наприклад, для відображення використанням Google Maps). Об'єкти подані у вигляді набору атрибутів, кожен з яких має значення або набір значень. Крім того, складні об'єкти можуть містити набір простих об'єктів, наприклад, людина може мати одну або кілька адрес.

На основі моделі загального визначення ресурсів і мови опису правил доступу до ресурсів розглянемо процес створення об'єктного адаптера за допомогою правил методу «чорної скриньки». Автоматичний об'єктний адаптер має в основі три модулі:

1. Інтерпретатор опису ресурсу – використовується для генерації структури даних ресурсу всередині системи, читаючи зовнішній опис ресурсу.
2. Пристрій відображень – слугує для вироблення внутрішніх правил відображення, читаючи конфігурацію порівнянь.
3. Пристрій перехоплень – використовується для перехоплення операцій системи відповідно до ресурсної та картографічної інформації, доступу до нового ресурсу за допомогою трансформаційних правил і, нарешті, для повернення результатів відображень до системи.

Розглянемо детальніше роботу об'єктного адаптера для забезпечення динамічної інтеграції інформаційних ресурсів. Зображений на рис. 2 об'єктний адаптер складається з двох підсистем: об'єктної компоненти і компоненти конфігурації опису перетворення ресурсу. Об'єктна компонента виконує багаторазове добування об'єктів або системні міграції відповідно до ресурсної і картографічної інформації та автоматично генерує повторно використовувані об'єкти або перетворену систему. Компонента конфігурації використовується для генерації опису ресурсу та правил відображення, які потрібні об'єктній компоненті. Рис. 2 показує архітектуру об'єктного адаптера визначення спільних рис інформаційних ресурсів і виявлення зв'язків між ними і описує співпрацю і основний потік даних між модулями. Інструмент конфігурації зовні адаптера використовується для генерації опису ресурсу та правил відображення. Відповідно до відмінностей у функціональності, бізнес-логіці, джерелах даних тощо, програмні системи мають різні поведінки доступу до ресурсу. Навіть з тією ж функціональністю і бізнес-логікою програмні системи ще варіюються залежно від відмінностей у досвіді, уміннях та особистих практиках дизайнерів та розробників. Проте, якими б складними і різноманітними не були web-системи, операції на системному рівні узгоджені, послідовні і відповідають певним стандартам.

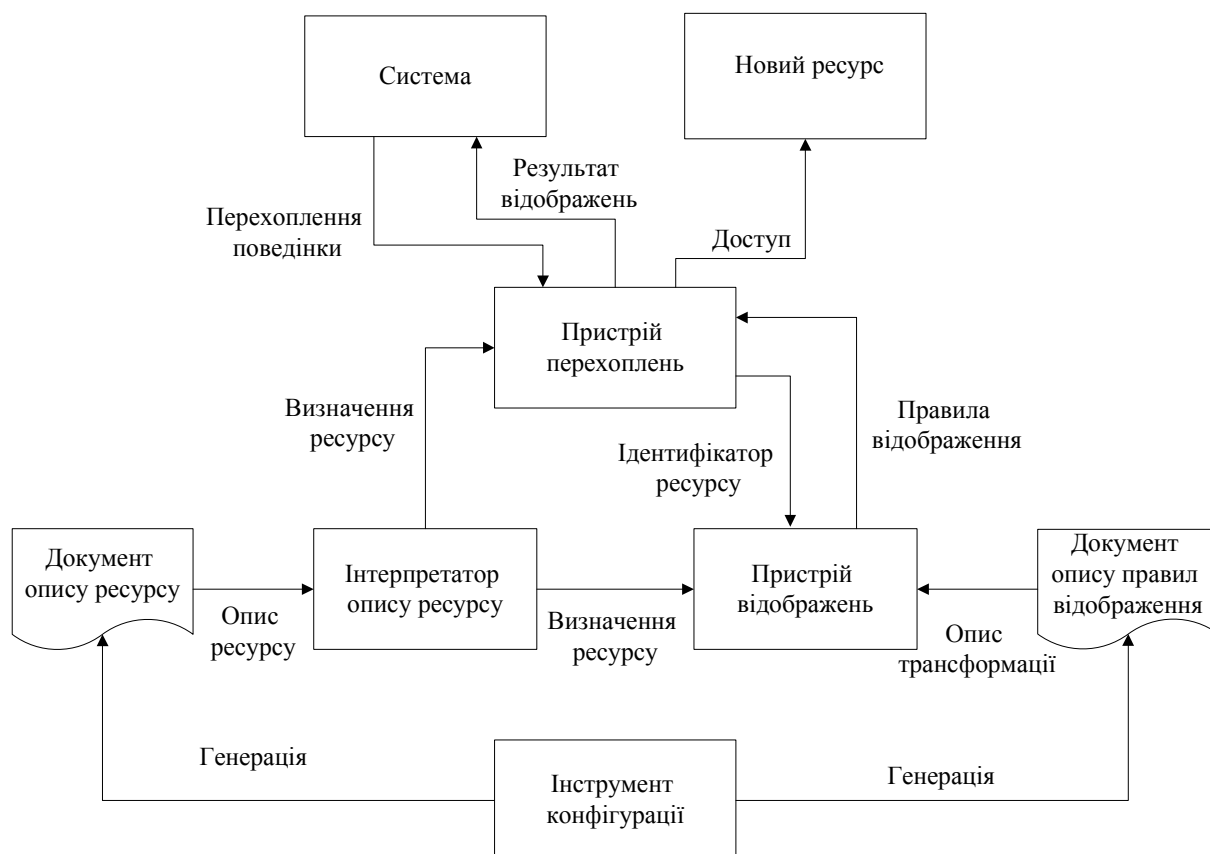


Рис. 2. Архітектура об'єктного адаптера визначення спільних ресурсів і виявлення зв'язків між ними

Операційна система повинна забезпечити повний набір операцій для отримання доступу до всіх ресурсів на платформі – набір ключів операцій. Набір ключів операцій повинен враховувати різні операційні системи, наприклад, набір дій у вигляді системних викликів на Unix/Linux або у вигляді API у файлі системного рівня DLL на Windows. Отже, проблему перехоплення поведінки програмного ресурсу можна вирішити перехопленням набору ключів операцій. Відповідно до методики «чорної скриньки» та агентно-орієнтованого підходу зобразимо структурно-динамічну модель предметної області Мешап-додатка в дискретний момент часу t . Нехай інформаційним об'єктом є інформаційний ресурс. Нехай ми маємо i -й інформаційний ресурс з такими характеристиками:

$$O_i = \langle D_i, T_i, L_i, F_i \rangle, \tag{1}$$

де D – напрям, з набором допустимих значень: {input, output}; T – тип інформаційного ресурсу з набором допустимих значень: {pipe, file, network, db}; L – розташування інформаційного ресурсу й інформація аутентифікації доступу до ресурсу; F – формат інформаційного ресурсу, що використовується для опису синтаксису і семантики інформаційного ресурсу.

Формат інформаційного ресурсу визначаємо так:

$$F_i = \langle F_{standard}, F_{description} \rangle, \tag{2}$$

де $F_{standard}$ – вказівка на стандартний тип ресурсів. Його значення NULL, якщо формат ресурсів не належить до стандартного типу, $F_{description}$ – використовується для опису формату, визначеного користувачем ресурсу. Його значення NULL для ресурсу в стандартному форматі.

Враховуючи формулу (1), фазова траєкторія (стан) i -го інформаційного ресурсу в кожен дискретний момент часу t матиме вигляд:

$$\varphi_i(t) = O_i(t)(D_i(t), T_i(t), L_i(t), F_i(t)). \tag{3}$$

Оскільки фазовий простір інформаційного ресурсу може бути поданим як множина станів інформаційного ресурсу, які в кожен момент часу t описують кортежем значень характеристик інформаційного ресурсу, структурно-динамічна модель предметної області Мешап-додатка в дискретний момент часу t матиме такий вигляд:

$$\langle O(t), R(t), B(t) \rangle = \langle O(t), R(O(t)), B(O(t)) \rangle, \quad (4)$$

де O – множина типів об'єктів; R – множина відношень між типами об'єктів; B – множина допустимих операцій над типами об'єктів, причому можуть виникати нові типи об'єктів, нові відношення, нові правила поведінки в момент часу t .

Опис інформаційного об'єкта може характеризуватися набором властивостей:

- повний (абстрактний) набір всіх можливих властивостей – універсум властивостей;
- підмножина універсуму властивостей об'єкта, істотних для розв'язання деякої задачі.

Кожна задача визначає набір фіксованих властивостей об'єкта, тим самим визначає і вибір типу об'єкта. Параметр – це логічна реалізація властивості інформаційного об'єкта, тому варто розрізняти значення параметра і його тип. Ту саму властивість можна зобразити різними типами (інтерпретаціями).

Над усіма логічними структурами можуть виконуватися чотири базові операції:

- створення – полягає у виділенні пам'яті для структури даних;
- знищення – протилежна за діями до операції створення;
- вибір – для доступу до даних усередині самої структури;
- оновлення – дає змогу змінити значення даних у структурі даних.

Висновки і перспективи подальших наукових розвідок

Описано порядок аналізу інформаційних ресурсів системи динамічної інтеграції слабко-структурованих даних у web-середовищі. Дано характеристику інформаційному об'єкту як інформаційному ресурсу і наведено опис найважливіших його характеристик. Описано роботу об'єктного адаптера та надано рекомендації щодо його використання для процесу визначення спільних рис інформаційних ресурсів і виявлення зв'язків між ними. Відповідно до методики «чорної скриньки» та агентно-орієнтованого підходу зображено структурно-динамічну модель предметної області Мешап-додатка в дискретний момент часу t .

1. Baumgartner R. *Interactively adding web service interfaces to existing web applications* / R. Baumgartner, G. Gottlob, M. Herzog, and W. Slany // *Int. Symposium on Applications and the Internet*, 2004. – P. 74–80. 2. Canfora G. *Migrating interactive legacy systems to web services* / G. Canfora, A. R. Fasolino, G. Frattolillo, and P. Tramontana // *IEEE CS Press, editor, European Conference on Software Maintenance and Reengineering*, 2006. – P. 23–32. 3. Guo H. *Wrapping client-server application to web services for internet computing* / H. Guo H., C. Guo, F. Chen, and H. Yang // *International Conference on Parallel and Distributed Computing, Applications and Technologies*, 2005. – P. 366–370. 4. Kushniretska I. I. *Application of MashUp Technology for Dynamic Integration of Semi-Structured Data* / I. I. Kushniretska, A. Y. Berko // *Матеріали VI Міжнародної конференції молодих вчених CSE-2013*. – Львів: Видавництво Львівської політехніки, 2013. 5. Laender A. *A brief survey of web data extraction tools*. / A. Laender, B. Ribeiro-Neto, A. Silva // *SIGMOD record*, 31(2), 2002. 6. Jiang Y. *Towards reengineering web sites to web-services providers* / Y. Jiang and E. Stroulia // *IEEE CS Press, editor, European Conference on Software Maintenance and Reengineering*, 2004. – P. 296–305. 7. Wagner G. *The Agent-Object-Relationship Metamodel: Towards a Unified View of State and Behavior Information Systems* G. Wagner, 2003.