

УДК 004.65

Ю.В. Рогушина, А.Я. Гладун

ІНТЕГРАЦІЯ ТЕХНОЛОГІЙ SEMANTIC WEB С СИСТЕМАМИ BUSINESS INTELLIGENCE 2.0

Розглянуті перспективи використання таких компонентів Semantic Web, як метадані, онтології, семантичні Web-сервіси та інтелектуальні програмні агенти, для інтелектуалізації засобів Business Intelligence 2.0 (BI) з метою перетворення даних у знання, а знань – у бізнес-дії для отримання певних корисних результатів. Програмні продукти BI сьогодні швидко розвиваються і широко застосовуються, тому інтеграція BI з Semantic Web впливає як на розвиток BI-платформ, так і на інтерес до практичного застосування Semantic Web.

Вступ

Сьогодні Business Intelligence (BI) сформувалося в самостійний напрям індустрії інформаційних технологій (ІТ). BI є споживачем та сферою застосування різноманітних нових технологій.

За оцінками експертів, у 2008 році доходи від додатків Business Intelligence зросли на 23 %, а загальний обсяг світового ринку систем Business Intelligence перевищив 7,1 млрд. доларів, що відповідає росту на 17,1 % у порівнянні з попереднім роком. Це свідчить про великий інтерес до цього напрямку та його потенційний вплив на інші напрями ІТ.

Ціль BI – перетворення даних у знання, а знань – у бізнес-дії для отримання певної користі [1]. Тому доцільно проаналізувати перспективи інтеграції BI з сучасними технологіями керування розподіленими знаннями, а саме – з Semantic Web.

На жаль, промислові застосунки не завжди вчасно впроваджують наукові досягнення: в сучасних комерційних BI-застосунках технології та стандарти Semantic Web практично не використовуються, а замість цього здійснюються спроби заново розробити засоби подання знань та метаописів.

Як BI, так і Semantic Web нині належать до пріоритетних напрямів ІТ, проте розвиваються паралельно, незважаючи на значну подібність задач, які вони вирішують. Тому важливо проаналізувати перспективи їх інтеграції та можливість взаємного впливу методів та застосувань.

Технології BI

BI – орієнтований на користувача процес, що забезпечує доступ і дослідження інформації, її аналіз, формування інтуїтивного розуміння, що ведуть до поліпшеного і неформального прийняття рішень. Нині BI – це інструменти для аналізу даних, побудови звітів і запитів, що можуть допомогти бізнес-користувачам обробляти великі обсяги і синтезувати з них значиму інформацію [2].

В основі технології BI лежить організація доступу кінцевих користувачів, аналіз структурованих кількісних даних і інформації про бізнес. BI забезпечує процес збору багатоаспектної інформації про досліджуваний предмет [3].

На жаль, адекватного перекладу словосполучення “Business Intelligence” українською та російською мовами немає, а ті терміни, що використовуються, тільки більше заплутують читача або є не зовсім коректною калькою з англійської («бізнес-інтелект», «бізнес-аналіз»). Можливо, варто звернутися до перекладу слова Intelligence у словосполученні Artificial Intelligence, де під Intelligence розуміється зовсім не «інтелект» і тим більше не «інтелігентність», а просто здатність до логічного виведення. Таким чином, словосполучення „Business Intelligence» можна перевести як «засоби логічного виведення, орієнтовані на бізнес-додатки».

BI – це не тільки технологічна платформа або набір програмних засобів для збору, збереження, аналізу і забезпечення доступу до даних складної структури, що

допомагає працівникам підприємства приймати кращі бізнес-рішення.

Це й здатність підприємства ефективно використовувати свої людські й інформаційні ресурси.

Основний напрям сучасних ВІ-проектів – це доставка інформації, але варто також приділити увагу аналізу нових відомостей і проблемі інтеграції з іншими сховищами даних.

Визначаючи платформу ВІ, можна виділити наступні три групи базових можливостей ВІ:

- інтеграція: інфраструктура ВІ, керування метаданими, розвиток, потоки робіт і колективна робота;
- доставка інформації: звітність, інформаційні панелі, незаплановані запити, інтеграція з Microsoft Office;
- аналіз: OLAP, розширена візуалізація, прогнозує моделювання, протоколи результатів.

ВІ забезпечує єдність метаданих для всього інструментарію та уніфіковані зручні способи пошуку, здобуття, збереження, повторного використання і публікації об'єктів метаданих. OLAP дозволяє кінцевим користувачам аналізувати дані за допомогою швидкого виконання запитів і розрахунків, а прогнозує моделювання і Data Mining дозволяє класифікувати та категоризувати інформацію, видобуваючи закономірності.

Багато організацій використовувє відразу кілька ВІ-продуктів, однак хотіли б упровадити єдиний загалькорпоративний ВІ-стандарт, що забезпечує широку функціональність й підтримує одночасно велику кількість користувачів.

Постановка задачі

Проаналізовано основні тенденції розвитку такого динамічного та комерційно успішного напрямку ІТ, як Business Intelligence, і його інтелектуалізації, розглянуто доцільність застосування технології та стандартів Semantic Web для керування знаннями в системах ВІ 2.0.

Класифікація продуктів ВІ

Програмне забезпечення ВІ підрозділяють на ВІ-інструменти та ВІ-додатки.

ВІ-інструменти застосовуються кінцевими користувачами для доступу, аналізу і генерації звітів за даними, що розташовуються в сховищі, вітринах чи оперативних складах даних [4, 5]. Серед них виділяють (рис. 1):

- генератори запитів і звітів;
- інструменти OLAP;
- корпоративні ВІ-набори EBIS;
- ВІ-платформи.

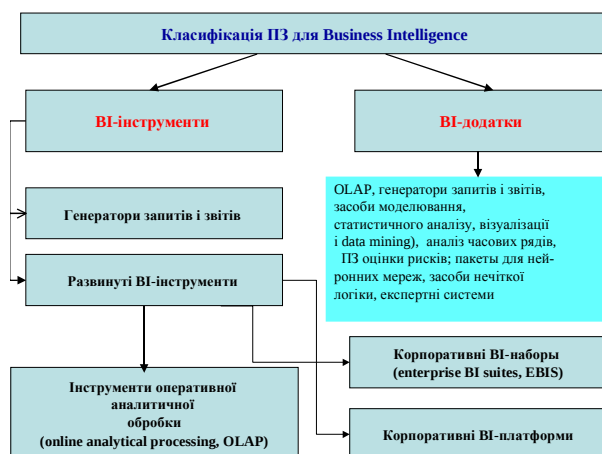


Рис. 1. Класифікація ПЗ для ВІ

Засоби генерації запитів і звітів у великому ступені поглинаються і заміщуються корпоративними ВІ-наборами. Багатомірні OLAP-механізми та реляційні OLAP-механізми є ВІ-інструментами й інфраструктурою для ВІ-платформ.

Застосунки Business Intelligence

Розглянемо програмні реалізації ВІ, щоб зрозуміти, чому саме ця сфера є цікавим та перспективним сектором для впровадження новітніх інтелектуальних засобів. У додатки Business Intelligence часто вбудовані ВІ-інструменти.

Відповідно до досліджень провідних ринкових аналітиків, таких як Forrester і Gartner, розробники програмного забезпечення ВІ можуть бути впорядковані за допомогою «магічного квадрата» (рис. 2): лідери – постачальники з широким функціоналом можливостей платформ ВІ (Cognos, Business Objects, Microsoft, Oracle, MicroStrategy, SAS); претенденти на лідерство (SAP,

Information Builders) пропонують платформи із широкою функціональністю, але в них є обмеження в частині програмного чи оточення предметних областей або каналів збуту; провідці (QlikTech, Tibco Spotfire) мають гарне представлення про платформу BI, відрізняються відкритістю і гнучкістю архітектури і пропонують добре розвинутого функціонала у своїх цільових областях, однак, діапазон пропонованих функцій не досить широкий; нишеві гравці (Actuate, arcplan, Board International, Panorama Software) – компанії, що досягли успіху тільки у конкретній області.

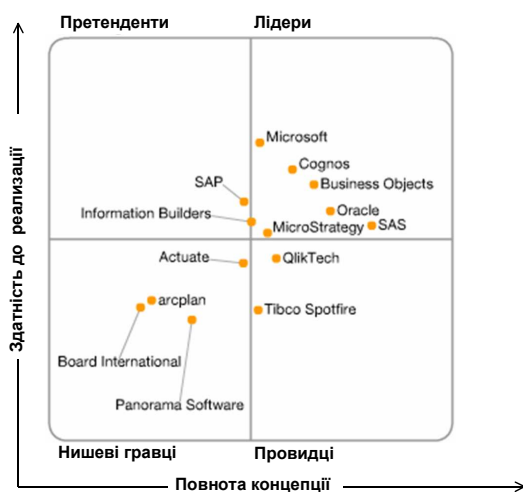


Рис. 2. «Магічний квадрат» BI

Розглянемо більш детально характеристики компаній-виробників BI-додатків і їхньої продукції.

SAP Business Objects [6] пропонує широкий набір апробованих закінчених рішень BI, включаючи продукти з інтеграції та якості даних і аналіз тексту. Всі продукти SAP Business Objects переводяться на загальну платформу для підвищення безпеки і поліпшення адміністрування, інтеграція різних продуктів, забезпечення більш зручного обміну метаданими і переходу від одного продукту до іншого.

IBM Cognos [7] – комплекс інтегрованих програмних продуктів для керування ефективністю діяльності підприємства й управлінського обліку (Corporate Performance Management, CPM). Це одне з найсучасніших і масштабованих середо-

вищ BI. Cognos базується на єдиній платформі J2EE і надає єдиний інтерфейс для більшості продуктів.

SAS – постачальник повного комплексу рішень BI та аналітичних інструментів [8]. SAS забезпечує інтеграцію даних, аналіз тексту, інструментарій для статистичного аналізу і проактивне моделювання, а також маркетинговий аналіз і керування ризиками.

Oracle реалізувала свої ідеї щодо BI у Enterprise Edition Plus [9], що містить сервер аналітики з механізмом ROLAP (Relational OLAP) і технологією інтеграції корпоративних даних (EII).

MicroStrategy – платформа, яка охоплює весь комплекс можливостей BI, з власним механізмом ROLAP, що ефективно оптимізує різні моделі даних.

Тенденції розвитку BI

Сучасний етап розвитку IT привів до виникнення нового покоління BI, який отримав назву BI 2.0. Подібно до Web 2.0 і Semantic Web, це скоріше загальна парадигма, абстрактна концепція, а не конкретний програмний продукт.

У традиційному BI проаналізована інформація використовувалася в рішеннях, орієнтованих на бізнес (тобто інформація, що надходить, оперативно обробляється і представляється в зручній для користувача формі), а в BI 2.0 інформація допомагає приймати рішення ще до того, як відбулася певна подія. BI 2.0 – це *нове покоління програмного забезпечення BI*, що скоріше проактивне, а не реактивне, та забезпечує зворотний зв'язок прийнятих рішень і інформації, що надходить у режимі реального часу.

Основні особливості технологій BI 2.0 – засоби контролю продуктивності, інтегрованого планування і прогнозування, що вбудовуються в операції, аналітичні портали, робочі простори для колективної роботи й інтеграція з Microsoft Office, розширені технології пошуку і візуалізації даних, реалізація функцій BI як сервісів, стандартизація і розширення застосування відкритих рішень, розвиток семантичних методів, засоби роботи зі слабо структурованими даними і спеціалізовані високо-

продуктивні програмно-апаратні рішення не тільки для сховищ даних і задач BI.

Інтеграція BI з Semantic Web

Semantic Web – це злиття Web-технологій і науки про представлення знань (knowledge representation, KR), що є підобластю штучного інтелекту (artificial intelligence, AI), спрямоване на створення і підтримку потенційно складних моделей світу, що дозволяють міркувати про себе і про зв'язану з ними інформацію [10].

BI 2.0 потребує семантичного моделювання даних, користувачів та інформаційного середовища, тобто розвинутих засобів керування знаннями. Для цього доцільно використовувати такі стандарти Semantic Web [11], як мова опису метаданих Resource Description Framework (RDF), мова представлення онтологій Web Ontology Language (OWL) мова запитів до них SPARQL (SQL like query language for RDF) (рис. 3).

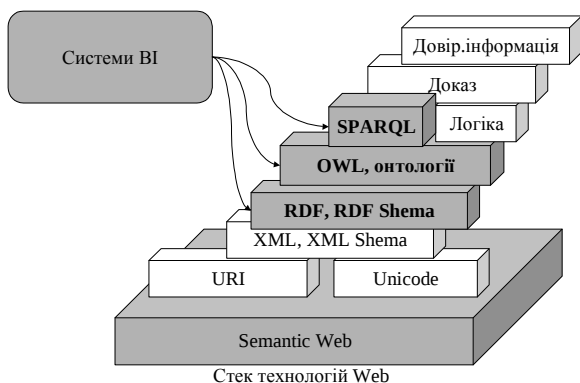


Рис. 3. Технології Semantic Web в BI

Побудова Semantic Web складається з розробки наступних елементів:

- засобів представлення знань про інформаційні ресурси (RDF);
- мови опису онтологій (OWL);
- мови опису Web-сервісів (OWL-S);
- інструментальних засобів для роботи з документами Web і онтологіями;
- мови запитів до баз знань у форматі RDF (SPARQL);
- засобів виведення на знаннях;
- семантичної пошукової системи.

Іншими важливими складовими Semantic Web є інтелектуальні Web-сервіси та інтелектуальні програмні аген-

ти (та мультиагентні системи) [12], що користуються цими сервісами.

У сучасних системах BI досить широко застосовуються метадані про інформаційні ресурси, проте вони описують тільки типи даних, а не їх семантику. Крім того, такі метаописи не використовують загальноновживаних стандартів та не пов'язуються з базами знань предметної області. Це ускладнює інтеграцію різних застосунків.

Тому з метою інтероперабельності BI-систем та повторного використання їх компонентів доцільно використовувати модель опису метаданих RDF, який дозволяє формулювати твердження у виді, придатному для обробки комп'ютером.

RDF Schema дозволяє розроблювачам визначати конкретний словник для даних RDF та вказувати види об'єктів, до яких можуть застосовуватися ці атрибути. Іншими словами, механізм RDF Schema надає базову систему типів для моделей RDF. Ця система типів використовує деякі визначені терміни, такі як Class, subPropertyOf і subClassOf, для схеми, орієнтованої на конкретне застосування.

Нині вже є певні спроби такого використання. Приміром, представники Oracle оголосили, що на сьогоднішній день уже більш ніж у 100 комерційних системах і системах з відкритим кодом використовується реалізація технології Semantic Web, розроблена Oracle для СУБД Oracle Database 10g Enterprise Edition. Для них створюються Java-розширення, що забезпечують можливість збереження даних в Oracle 10g у форматі RDF. Таке застосування має певний вплив і на розвиток Semantic Web. Наприклад, включення компанією Oracle у липні 2005 року підтримки RDF у свій продукт Spatial 10.2g забезпечило легітимність RDF.

У ході експериментів з базами даних RDF у багатьох випадках проявилися їхні безсумнівні переваги перед традиційними структурованими базами даних, особливо у відношенні вбудовування даних у Web. Як зазначила компанія Microsoft у Services Framework 3.0 Developer Guide за 2006 рік, збереження

профілів із застосуванням RDF вигідно за двома причинами: забезпечення гнучкої схеми збереження даних та підтримки створення Web-подібних відношень між даними.

ВІ-системи інтенсивно використовують знання та засоби їх обробки. Щоб забезпечити інтеоперабельність накопичених знань, слід базуватися на певних загальноновживаних стандартах представлення знань. У багатьох інших сферах ІТ для цього використовують онтології та тезауруси, для яких розроблено мову представлення OWL. Тому доцільно використовувати OWL в ВІ 2.0 для інтегрованого подання знань відповідних предметних областей. На жаль, у сучасних ВІ-застосунках OWL практично не впроваджено.

З визнанням RDF назріла потреба в стандартній мові запитів для RDF, що відіграє ту ж роль, що і SQL у реляційних даних. Протокол SPARQL і мова запитів RDF Query Language розроблені саме з цією метою. Твердження SPARQL як стандартна мова запитів для RDF дозволяє багатьом сховищам даних стати точками доступу SPARQL, забезпечуючи гнучкий обмін даними між системами.

SPARQL одночасно є як мовою запитів, так і протоколом доступу до даних, одним із ключових компонентів Web 2.0: як стандарт для підтримки гнучкої моделі даних він надає спільний механізм запитів для Web 2.0. SPARQL використовується для представлення запитів до різноманітних джерел даних, незалежно від того, зберігаються ці дані безпосередньо в RDF або представляються у виді RDF за допомогою проміжного програмного забезпечення (middleware). Використання запитів SPARQL в ВІ 2.0 забезпечує інтегрований доступ до розподілених знань.

Тенденції ВІ 2.0

Можна виділити шість *ключових факторів*, пов'язаних з переходом ВІ на новий рівень [13].

1. Поширення ВІ за межі окремого підприємства через об'єднання оперативних і аналітичних додатків

2. Підтримка аналітичної культури й автоматизація прийняття рішень.

3. Готовність до взаємодії з новим поколінням працівників, які, на відміну від своїх попередників, впроваджують технологію ВІ у своє персональне життя

4. Використання семантичних технологій, що використовують метадані й онтології для інтеграції взаємодії між людьми і процесами.

5. Адоптація стандартів і технологій Web 2.0 та Semantic Web для спільної роботи, масштабування і прискорення створення ВІ-продуктів і додатків.

6. Висока продуктивність ВІ.

ВІ 2.0 містить кілька важливих нових концепцій, що стосується використання інформації у бізнесі, організаціях і урядових структурах. Цей термін пов'язаний з ВІ у реальному часі, технологією, що керується зовнішніми подіями, але основна ідея ВІ 2.0 – в застосуванні цих методів до бізнес-процесів. ВІ 2.0 – це розширення технології ВІ, що включає динамічне й автоматизоване прийняття рішень.

Засоби ВІ нового покоління дозволяють керуватися подіями і зможуть виявляти аномальні ситуації і виникаючі проблеми, для цього в них будуть вбудовані відповідні інтелектуальні й адаптивні можливості. Як джерела даних для аналізу будуть служити не тільки сховища, але й дані найрізноманітніших внутрішніх і зовнішніх для компанії джерел і програмних агентів (що безпосередньо корелюється з таким напрямом Semantic Web, як LinkedData).

Linked Data Project створює загальноновживані ієрархії класів, словники загальних і власних імен, а також допомагає власникам масивів даних поєднувати їхні бази знань в одну зв'язну систему знань. У багатьох випадках учасники проекту поєднують уже наявні великі бази, допомагаючи один одному установити відповідність між ідентифікаторами однієї і тієї ж речі в різних базах. Якщо якась база знань заповнюється деякою автоматичною процедурою, то ця процедура може почати використовувати імена, вже використовувані іншими учасниками, а якщо вона

пишеться вручну, тоді автори можуть заглядати в Dbpedia, GeoNames, WordNet чи Yago як у словник, одночасно з цим перетворюючи свої дані в “замітки на полях” великої енциклопедії.

Наявність перехресних зв'язків між різними базами знань не тільки робить ці бази більш корисними – часто самі знання очищаються від помилок. Дуже швидко зростає не тільки сумарний обсяг баз проекту – одночасно росте і кількість SPARQL-запитів до бази, що створює багато проблем.

У технологіях нового покоління переважними стануть колективні методи роботи в мережі з використанням віртуальних форумів, блогів, вики, соціальних мереж. Реалізація додатків буде спиратися на SOA із застосуванням загальнодоступних рішень і активних елементів Web 2.0 (наприклад, AJAX) і функціонально багатого зовнішнього інтерфейсу.

Використання інтелектуальних Web-сервісів в BI

Останнім часом багато компаній стали представляти свої сховища даних і BI-системи як Web-сервіси для використання іншими додатками і процесами, пов'язаними сервіс-орієнтованою архітектурою (SOA) або ПЗ проміжного рівня, таким як корпоративна сервісна шина ESB, для інтеграції систем BI підприємства [14].

Web-сервіси розгортаються в межах підприємства, щоб активувати більш ефективну інтеграцію операційних процесів. Інтеграція додатків (Business-to-Business, B2B) усередині Extranet на основі використання Web-сервісів нині активно досліджується. Більшість додатків BI нині є Web-базованими, і технологія Web-сервісів розширює сферу дій BI поза мережами Extranet. Web-сервіси дозволяють корпораціям обмінюватися BI-інформацією через стандартні протоколи зв'язку між програмами і централізувати сервіси. Web-сервіси дозволяють багаторазово використовувати функціональні можливості компонентів BI-додатків, використовуючи стандартний Web-базований протокол. Це дозволяє компонентам, напи-

саним різними мовами і на різних платформах, динамічно зв'язуватися один з одним.

Використання SOA як основи для побудови BI-системи дозволяє керувати подіями, виконувати багато задач у режимі реального часу, автоматизувати аналіз і обробку інформації, створювати системи, що легко масштабуються та інтегруються.

Web-сервіси базуються на широко розповсюджених і відкритих протоколах: HTTP, XML, UDDI, WSDL і SOAP. Але для того, щоб автоматизувати інтеграцію Web-сервісів на семантичному рівні, слід використовувати OWL-S.

Ціль розробки OWL-S полягає в тому, щоб уможливити логічне виведення для Web-сервісів, планування компонування Web-сервісів, автоматичне використання сервісів програмними агентами. Використовуючи OWL-S, Web-сервіс може повідомляти про свої функціональні можливості потенційним користувачам.

OWL-S забезпечує декларативні описи властивостей Web-сервісів на основі онтологій і можливості, що можуть використовуватися для автоматичного пошуку сервісу, декларативний API для його автоматизованого виконання Web-сервісів [15].

Для інтероперабельного опису сервісів розроблено модифікацію OWL для OWL-S (Web Ontology Language for Services). OWL-S забезпечують онтологічний опис Web-сервісу.

Ціль розробки OWL-S полягає в тому, щоб уможливити використання логічного виведення для Web-сервісів, планування компонування Web-сервісів, автоматичне використання сервісів програмними агентами. OWL-S забезпечує декларативні описи властивостей Web-сервісу і можливості, що можуть використовуватися для автоматичного виявлення сервісу.

Щоб використовувати Web-сервіс, програмні агенти мають потребу у визначенні сервісу, що інтерпретується комп'ютером, і засобів доступу до цього визначення. OWL-S [16] – це онтологія сервісів, призначена для того, щоб допомогти користувачам і пошуковим агентам

виявляти, викликати, компонувати і контролювати Web-сервіси.

Щоб використовувати Web-сервіс, слід автоматизовано інтерпретувати визначення сервісу і засоби доступу до нього. OWL-S – це онтологія сервісів, що реалізує цю функціональність. Поштовхом до його розробки став проект Semantic Web, що прагне забезпечити доступ до інформаційних ресурсів Web не за ключовими словами, а за контентом. Одна з цілей Semantic Web – створити засоби, що дозволяють автоматично користувачам вибирати, використовувати, компонувати і відслідковувати Web-сервіси. Важливим результатом у цьому напрямку стала розробка мов розмітки Web, таких як OWL і його попередник DAML+OIL. Ці мови дозволяють створювати онтології кожної предметної області (ПрО) і встановлювати зв'язки з цими онтологіями Web-сайтів для їхнього опису.

Сервіси можуть бути *простими* у тому розумінні, що вони активізують тільки якусь одну доступну через Інтернет програму, сенсор чи пристрій, не використовуючи інші Web-сервіси, і тому взаємодія між користувачем і сервісом, крім простого виклику, відсутня. Сервіси можуть бути і *комплексними*, що складаються з кількох простих. Вони звичайно вимагають взаємодії чи діалогу між користувачем і сервісами, тобто користувач може здійснювати вибір або накладати певні умови.

OWL-S призначений для підтримки обох категорій сервісів, але комплексні (складені) сервіси пред'являють більше вимог до властивостей мови. Кожна доступна через Web програма, сенсор чи пристрій, що оголошені сервісом, можуть розглядатися як сервіс. OWL-S не виключає оголошення сервісом простих, статичних Web-сторінок.

Однак основною метою OWL-S є підтримка більш складних задач. OWL-S вирішує задачі чотирьох типів:

1. Автоматичне дослідження Web-сервісів містить у собі автоматичне місцезнаходження Web-сервісів, що надають певну послугу і задовольняють накладеним обмеженням. Наприклад, корис-

тувач хоче знайти сервіс із продажу авіаквитків між двома містами, що приймає його кредитну карту. Нині такий сервіс має виконуватися людиною, що може використовувати пошукову машину для знаходження сервісу, прочитати знайдену Web-сторінку і виконати сервіс вручну, упевнившись, що він відповідає накладеним умовам. Якщо сервіс розмічений OWL-S, то інформація, необхідна для знаходження сервісу, теж буде семантично розмічена і інтерпретована комп'ютером на сайті Web-сервісу, і пошукова машина, розширена онтологією, зможе знайти такий сервіс автоматично.

2. Автоматичний виклик Web-сервісу (invocation) містить автоматичне виконання ідентифікованого Web-сервісу комп'ютерною програмою чи агентом. Наприклад, користувач може запросити покупку з певного сайту квитка на конкретний рейс. У даний час користувач має зайти на цей сайт, заповнити певну форму і натиснути на кнопку для виконання сервісу. Альтернативний підхід полягає у тому, що користувач може відправити запит з відповідними параметрами за протоколом HTTP безпосередньо до сервісу. В обох випадках необхідна участь користувача. Виконання Web-сервісу можна розглядати як набір викликів функцій. Розмітка OWL-S для Web-сервісу забезпечує декларативне, інтерпретоване комп'ютером API для виконання цих функціональних викликів. Програмний агент має бути здатним інтерпретувати розмітку, щоб зрозуміти, які потрібні вхідні дані для виклику сервісу, яка інформація буде повернена і як виконати сервіс автоматично. Таким чином, OWL-S має забезпечувати декларативні API для Web-сервісів, що потрібні для їхнього автоматичного виконання.

3. Автоматична композиція і взаємодія Web-сервісів містить автоматичний вибір, композицію і взаємодію Web-сервісів для виконання певної задачі, обумовленої високорівневим описом завдання. Наприклад, користувач хоче сформулювати всі заходи для поїздки на конференцію. Нині користувач має вибрати Web-сервіси, вручну задати їхню компо-

зицію й упевнитися, що все програмне забезпечення, необхідне для взаємодії, відповідає його вимогам. За допомогою розмітки OWL-S Web-сервісу інформація, необхідна для вибору і композиції сервісів, може бути подана на сайті Web-сервісу. Може бути написане програмне забезпечення для маніпулювання цими поданнями разом зі специфікаціями цілей задачі, призначене для автоматичного виконання задачі. Таким чином, OWL-S має забезпечувати декларативні специфікації попередніх умов і наслідків виконання використання окремих сервісів, необхідні для автоматичної композиції і взаємодії сервісів.

4. Автоматичний моніторинг виконання Web-сервісів. Окремі сервіси, і, більш того, композиції сервісів, звичайно вимагають якийсь час на повне виконання. Користувач може захотіти в цей період знати, який статус його запиту, або ж його плани можуть змінитися і зажадати змін у діях програмного агента. Наприклад, користувач хоче упевнитися в тому, що резервування готелю успішно виконано. Для цього було б корисно мати можливість з'ясувати, на якому етапі знаходиться процес виконання запиту і де виникли які-небудь непередбачені перешкоди. Таким чином, OWL-S має забезпечувати декларативні описи для стану виконуваних сервісів. На останню задачу варто звернути увагу, тому що в ранніх версіях OWL-S вона не розглядалася. Онтології OWL-S структурують відомості про сервіси на три категорії (рис. 3):

- профіль сервісу: клас SERVICE представляє SERVICEPROFILE;
- модель сервісу: клас SERVICE описується в SERVICEMODEL;
- база сервісу: клас SERVICE підтримується в SERVICEGROUNDING.

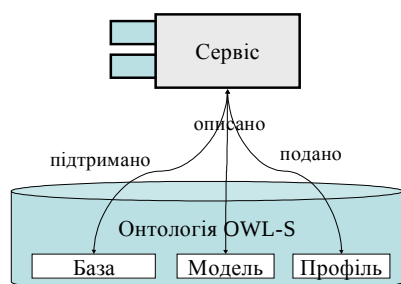


Рис. 3. Онтологія Web-сервісу

Для семантичного обґрунтування параметрів Web-сервісів використовують онтології різного рівня: онтології застосування, онтології Про і онтології верхнього рівня. Доцільно мати загальний словник (тезаурус), що містить базові терміни Про, що використовуються в онтологіях застосування (рис. 4). Кожне поняття в такому тезаурусі можна зрозуміти через набір інших понять, що приводить до появи семантичного поля.

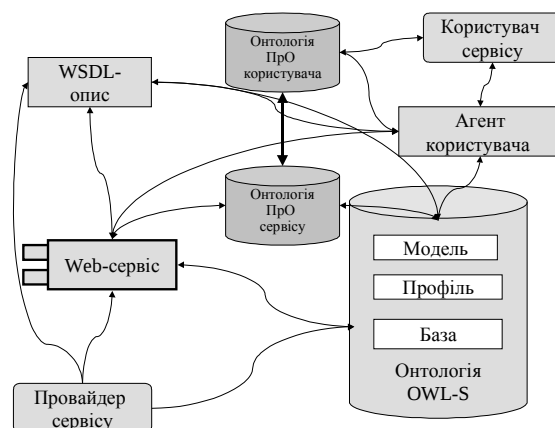


Рис.4. Семантика Web-сервісу

OWL-S не нав'язує які-небудь обмеження на моделі профілів та процесів, тобто обидва визначення можуть бути несумісними без впливу значущості виразу OWL. Тому саме OWL-S може стати основою опису сервісів в системах BI 2.0 на семантичному рівні.

Висновки

BI 2.0 – це нове покоління програмного забезпечення, що допомагає приймати рішення ще до того, як відбулася певна подія, призначення якого – перетворення даних у знання, а знань – у бізнес-дії.

Аналіз сучасних засобів BI та тенденцій їх розвитку показує доцільність використання сервіс-орієнтованої архітектури та технологій Semantic Web для підвищення рівня інтелектуальності таких застосунків [17].

Інтеграція BI з Semantic Web впливає як на розвиток BI-платформ, так і на інтерес до практичного застосування таких компонентів Semantic Web, як метадані, онтології, семантичні Web-сервіси та

інтелектуальні програмні агенти. В цілому, використання в системах ВІ технологій Semantic Web забезпечує новий рівень взаємодії між інформаційними ресурсами.

1. *Sanjay Mehta* BI 2.0 Technology – MAIA Intelligence Perspective. – www.maia-intelligence.com.
2. *Business intelligence definitions*. – http://searchdatamanagement.techtarget.com/sDefinition/0,,sid91_gci213571,00.html.
3. *Вэллс Д.* Десять основных преимуществ Microsoft Business Intelligence, 2008, (Пер. с англ.) – <http://www.w3.org/TR/html4/strict.dtd>.
4. *Neches R., Fikes R., Finin T., Gruber T., Patil R., Senator T., Swartout R.* Enabling Technology for Knowledge Sharing // *AI Magazine*. – 1999. – Vol. 12., № 3.
5. *Уотсон Х., Вискон Б.* Современное состояние бизнес-аналитики (Business Intelligence, BI), <http://www.osp.ru/>.
6. *SAP BusinessObjects Business Intelligence Solutions*. – <http://www.sap.com/solutions/sapbusinessobjects/large/business-intelligence/index.epx>.
7. *Business Intelligence from IBM*. – <http://www-01.ibm.com/software/data/businessintelligence/>.
8. *SAS Business Intelligence*. – <http://www.sas.com/technologies/bi/>.
9. *Oracle Enterprise Performance Management and Business Intelligence*. – http://www.oracle.com/solutions/business_intelligence/index.html.
10. *Semantic Web*. – <http://www.semanticweb.org>.
11. *Рогущина Ю.В., Гладун А.Я.* Технологии Semantic Web и их использование при разработке интеллектуальных приложений // *Проблемы програмування*. – 2008. – № 2-3. – С. 385–394.
12. *Плескач В.Л., Рогущина Ю.В.* Агентні технології // *Монографія*. – К.: Київськ. нац. торг.-екон. ун-т, 2005. – 338 с.
13. *Артемьев В.* Что такое BUSINESS INTELLIGENCE? // *Открытые системы*. – 2003. – № 4. – <http://www.osp.ru/os/2003/04/020.htm>.
14. *Эверет Д.* Web-сервисы на службе у Business Intelligence. – <http://www.iso.ru/journal/articles/572.html>.
15. *Рогущина Ю.В., Гладун А.Я., Штонда В.М.* Онтологический анализ Web-сервисов в интеллектуальных сетях // *Proc. of The XIII-th International Conf. "Knowledge-Dialogue-Solution"*, ITHEA, Sofia, V. 2, 2007. – С. 451–459.
16. *OWL-S: Semantic Markup for Web Services*. The OWL Services Coalition. – <http://www.daml.org/services/owl-s/1.0/owl-s.html>.
17. *Рогущина Ю.В., Любич А.А., Плескач В.Л.* О выборе критериев оценки интеллектуальности информационной системы // *УСиМ*. – 2005. – № 1. – С. 3–7.

Отримано 12.11.2009

Про авторів:

Рогущина Юлія Віталіївна,
кандидат фізико-математичних наук,
старший науковий співробітник, доцент,

Гладун Анатолій Ясонович,
кандидат технічних наук,
старший науковий співробітник, доцент.

Місце роботи авторів:

Інститут програмних систем
НАН України, 03187, Київ-187,
Проспект Академіка Глушкова, 40.

Міжнародний науково-навчальний центр
інформаційних технологій та систем
НАН України і МОН України.