

**Е-НАУКА НА ШЛЯХУ ДО СЕМАНТИЧНОГО ГРІД.
ЧАСТИНА 2: СЕМАНТИЧНИЙ WEB- І СЕМАНТИЧНИЙ ГРІД**

М.З. ЗГУРОВСЬКИЙ, А.І. ПЕТРЕНКО

Досліджено семантичний Web, заснований на метаданих, онтологіях і програмних агентах як фундамент для семантичного Грід, в якому інформація і послуги мають чітке визначення, що надає можливість комп'ютерам і людям працювати в кооперації зі знанням, а не з інформацією, як це робиться зараз. Розглянуто підходи до опису онтологій даних і Web- та Грід-сервісів (із станом і без стану), їхньої взаємодії, архітектури, а також й існуючі реалізації семантичних Грід.

ВСТУП

У першій частині даної роботи [1] було розглянуто еволюцію Грід-систем на шляху до семантичних Грід, в яких інформація та послуги дають можливість комп'ютерам і людям працювати в кооперації. Семантичний Грід як розширення сучасного Грід виявляється надзвичайно корисним для е-науки [2–6], оскільки його технології дають змогу легко, швидко та зручно працювати *зі знанням*, а не з інформацією, як то було дотепер. Додатки семантичних Грід можуть інтегрувати велику множину різнорідних джерел інформації і сервісів, яких залучено і об'єднано, а також людські й обчислювальні ресурси і сервіси передачі інформації.

1. СЕМАНТИЧНИЙ WEB

У даній частині головну увагу приділено семантичному Web як фундаменту семантичного Грід. Ідея семантичного Web — це концепція мережі, в якій кожен ресурс людською мовою забезпечено описом, зрозумілим комп'ютеру. Документи в мережі публікуються у форматі XML, який містить семантичні RDF-ствердження.

Згідно із планом, запропонованим Тімом Бернерсом-Лі, винахідником семантичного Web [7], для впровадження семантичного Web необхідно послідовно розробити:

- синтаксис для представлення знань, що використовує посилання на онтології (зроблено: RDF);
- мову опису онтологій (зроблено: OWL);

- мову опису Web-серверів (зроблено: WSDL, OWL-S);
- інструментарій розробки/читання документів семантичного Web (почато: Jena, Haystack, Protege);
- мову запитів до знань, записаних у RDF (зроблено: SPARQL);
- логічне виведення знань (повністю не зроблено);
- семантичну пошукову систему (почато: SHOE, Swoogle);
- агентів семантичного Web (поки що не зроблено).

У цьому переліку новим, у порівнянні з попереднім матеріалом [1], є поняття *онтології*, тому зупинимось на ньому детальніше. Онтологія визначає загальний словник для вчених, яким потрібно спільно використовувати інформацію в предметній галузі [7–10]. Вона включає формулювання основних понять предметної галузі й відношення між ними, що інтерпретуються комп'ютером.

Причини необхідності в розробці онтології:

- сумісне використання людьми або програмними агентами загального розуміння структури інформації;
- можливість повторного використання знань у предметній галузі;
- створення явних допущень у предметній галузі;
- відокремлення знань предметної галузі від оперативних знань;
- аналіз знань у предметній галузі.

Сумісне використання людьми або програмними агентами загального розуміння структури інформації є однією з найзагальніших цілей розробки онтологій [9]. Наприклад, нехай, декілька різних Web-сайтів містять інформацію з медицини або надають інформацію про платні медичні послуги, що сплачуються через Інтернет. Якщо ці Web-сайти спільно використовують і публікують одну і ту ж базову онтологію термінів, якими вони всі користуються, то комп'ютерні агенти можуть добувати інформацію з цих різних сайтів та накопичувати її. Агенти можуть використовувати накопичену інформацію для відповідей на запити користувачів або як вхідні дані для інших застосунків.

Забезпечення можливості використання знань у предметній галузі сприяло бурхливому розвитку у вивченні онтологій. Наприклад, для моделей багатьох різних предметних галузей необхідно сформулювати поняття часу, яке включає поняття часових інтервалів, моментів часу, відносних мір часу і т.ін. Якщо одна група вчених детально розробить таку онтологію, то інші можуть повторно використовувати її у своїх предметних галузях. Крім того, якщо нам необхідно створити велику онтологію, ми можемо інтегрувати декілька існуючих онтологій, які описують частини великої предметної галузі. Ми також можемо повторно використовувати основну онтологію, таку, як UNSPSC [9], і розширювати її для опису предметної галузі, що нас цікавить.

Створення явних допущень у предметній галузі, покладених в основу реалізації, надає можливість легко змінювати ці допущення при зміні наших знань про предметну галузь. Жорстке кодування припущень про світ мовою програмування призводить до того, що ці припущення не тільки складно знайти і зрозуміти, але й складно змінити, особливо непрограмістові. Крім

того, явні специфікації знань у предметній галузі корисні для нових користувачів, які мають засвоїти значення термінів з цієї галузі.

Відокремлення знань предметної галузі від оперативних знань — це ще один варіант загального застосування онтологій. Ми можемо описати завдання конфігурації об'єкту і його компонентів відповідно до необхідної специфікації і запровадити програму, яка формує цю конфігурацію незалежно від об'єкту і самих компонентів. Після цього ми можемо розробити онтології компонентів і характеристик комп'ютерів, на яких базується об'єкт, та застосувати цю програму для морфологічного синтезу нестандартних конфігурацій об'єктів.

Аналіз знань у предметній галузі можливий, якщо є декларативна специфікація термінів. Формальний аналіз термінів надзвичайно цінний як під час спроби повторного використання існуючих онтологій, так і при їх розширенні. Часто онтологія предметної галузі сама по собі не є метою. Розробка онтологій схожа на визначення набору даних і їх структури для використання іншими програмами. Методи розв'язання завдань, доменно-незалежні застосування і програмні агенти використовують як дані онтології і бази знань, побудовані на основі цих онтологій [10].

Для кожної предметної галузі існує безліч онтологій залежно від ступеня деталізації її структури. Прийнято говорити про *спектр онтологій*, який містить:

- *онтології малої структуризації* — тахономії (наприклад, ієрархія Yahoo, біологічна тахономія), схеми баз даних і схеми метаданих (ebXML, WSDL);
- *онтології середньої структуризації* — тезауруси (WordNet, CALL, DTIC) і концептуальні моделі (Моделі OO, UML);
- *онтології високої структуризації* — логічні концептуальні моделі (TOVE, CYC, семантичний Web).

Приклад фрагменту структури онтології з Грід-галузі наведено на рис. 1.

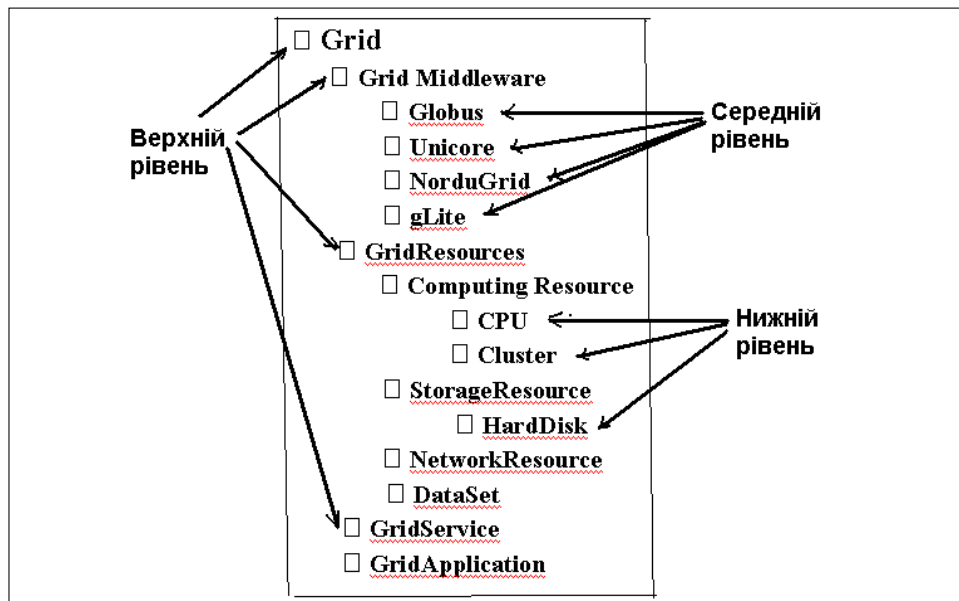


Рис. 1. Фрагмент онтології з Грід-галузі

Онтологія — це формальний явний опис понять у предметній галузі: *класів* (або *концептів*), особливостей кожного концепту (*слотів* або властивостей) і обмежень на слоти (*фасетів*, або аспектів чи ролей). Онтологія разом із набором індивідуальних зразків класів складає *базу знань*. Проте, слід пам'ятати — **для будь-якої предметної галузі не існує єдиної правильної онтології**. Проектування онтології — це творчий процес і дві онтології, розроблені різними людьми, ніколи не будуть однаковими. Потенційні додатки онтології, а також розуміння розробником предметної галузі і його точка зору на неї будуть, поза сумнівом, впливати на ухвалення рішень.

МОВИ ОПИСУ ОНТОЛОГІЙ

Для опису онтологій існують різні мови і системи, проте, найперспективнішою є мова OWL (попередня назва DALM-OIL) [11]. OWL розширює можливості з опису нових типів RDF-даних, що вже існують (наприклад, можна визначати тип класу, що є перетинанням або об'єднанням двох існуючих). OWL використовує синтаксис XML і містить команди для завдання дерева класів, команди для завдання приналежності індивідів класам, команди опису властивостей: області визначення, області значень та ін., команди опису характеристик властивостей: симетричність, транзитивність, функціональність; команди для склеювання (еквівалентності) класів.

Однією з переваг OWL-онтологій є доступність інструментів, які можуть робити *логічні висновки*. Побудова чіткої і працездатної системи логічних висновків — непроста справа, доступніше побудувати онтології. На сьогодні існують приклади онтологій (Protege-2000 [12], Ontolingua [13] і Chimaera [14]), побудовані у сферах традиційно науковомістких галузей, починаючи від хімічної обробки, і закінчуючи конструюванням машинобудівних підприємств. Крім того, для специфічних наукових галузей існує низка ініціатив щодо побудови великомасштабної онтології. Однією з таких галузей є *генетика*, де багато зусиль було спрямовано на створення спільної термінології та визначень, щоб дозволити вченим керувати їхніми знаннями [8]. Ці зусилля дають уявлення про те, як онтології можуть відігравати значну роль у підтримці науковців.

Технологія SPARQL дозволяє отримувати дані з розподілених джерел і може бути як засіб інтеграції різнорідної інформації. У специфікації SPARQL відсутні недоліки, властиві традиційним мовам запитів, зокрема, не накладаються обмеження на формат даних. Завдяки цьому стає можлива взаємодія між ресурсами різного типу. Намагатися використовувати семантичну мережу без SPARQL — це все одно, що працювати з реляційною базою даних без мови структурованих запитів SQL. Тобто, SPARQL перетворює доступ до даних у деяку подібність Web-сервісу.

Кінцева амбітна мета семантичного Web полягає у створенні такого середовища, де програмні агенти можуть динамічно виявляти і опитувати ресурси, а потім взаємодіяти з ними [15]. Агенти мають справлятися з віртуальними проблемами, які виникають в інтелектуальному середовищі, виявляти нові факти і виконувати найрізноманітніші завдання, що отримуються від людини. Семантичний Web можна представити як трирівневу структуру:

Перший рівень: фундамент, що складається з унікальної глобальної ідентифікації ресурсу; метадані для підтвердження відомостей про ресурси і обслуговування запитів на такі відомості, і простої мови опису метаданих і знань.

Метадані і знання містяться в онтологіях, що забезпечують взаєморозуміння і загальноприйнятий словник для метаданих і правил виведення нових метаданих і знань.

Другий рівень: базові сервіси для виконання стандартних висновків і цілеспрямованого перегляду метаданих і онтологій, пояснення зроблених висновків.

До цього рівня відноситься також управління довірою, агенти, пошукові машини, сервери онтологій.

Третій рівень: прикладні сервіси; наприклад, сервіс певного агента (агент науковця, агент лікаря).

Це є ідеєю, яку поки що не реалізовано, але щоб здійснити щось глобальне, необхідно почати з малого. Семантичний Web — це Web для комп'ютерів. Повноцінна реалізація ідеї якого дозволила б не лише здійснювати найефективніший комплексний пошук за допомогою інтелектуальних програмних агентів, а й знаходити приховані закономірності у масштабних розподілених наборах даних, допомагати відслідковувати активність організованої злочинності та терористів, сприяти аналізу розподілених геологічних та метеорологічних даних, допомагати у розподілі та повторному використанні важливих освітніх ресурсів тощо.

Традиційні Web-сервіси описуються, використовуючи мову WSDL на синтаксичному рівні, як колекції імен і схем XML-даних. Ці описи збагачуються згодом семантичною інформацією. Виділяють чотири галузі, де можуть використовуватися онтології для семантичного опису Web-сервісів:

- *семантика даних* для визначення значення даних, тобто вхідних і вихідних даних;
- *функціональна семантика* для визначення значення операцій і процесів, за допомогою яких вхідні дані перетворюються на вихідні;
- *семантика QoS* забезпечує рівень *якості сервісів*: наявність, рівень довіри, ціну і т.ін. Ці параметри впливають на пошук (вибір) сервісу;
- *семантика виконання* забезпечує наступні параметри: початкові стани, ефекти, шаблони взаємодії з сервісами.

Співвідношення між параметрами Web-сервісів і концептів у деякій OWL-онтології можуть бути виражені у формальний засіб через анотацію WSDL-опису з посиланнями на URL в онтології або зовнішнім описом. Існують інструменти, що стандартизують такі семантичні анотації Web-сервісів, а саме OWL-S (OWL for Services), запропонований коаліцією OWL-S [16, 17], і WSDL-S, запропонований в проекті METEOR-S [18].

Структура OWL-S онтології складається з *профілю сервісу (Profile)* для залучення і пізнання сервісів, процесуального *процесу (Process)*, який підтримує побудову сервісів, і *розташування сервісу (Grounding)*, що зв'язує концепторів профілю і процесу з розташованим нижче інтерфейсом сервісу (рис. 2). Крім того, у структуру OWL-S входять *складання (Compose process)* і *складові (Atomic Process)*, які відповідають за реалізацію складеного серві-

су. Профіль сервісу (OWL-S профіль) має функціональні і нефункціональні властивості. Функціональні властивості описують входи, виходи, початкові стани й ефекти сервісу. Нефункціональні — напівструктуровану інформацію, призначену для людини-користувача (наприклад: ім'я сервісу, опис сервісу та параметри сервісу). Параметри сервісу включають наступні вимоги: безпеку, якість сервісу, географічні межі і т.ін. Реалізація сервісу забезпечує виконання конкретного Web-сервісу шляхом перекладу абстрактних концепцій профілю OWL-S і процесу в конкретні повідомлення.

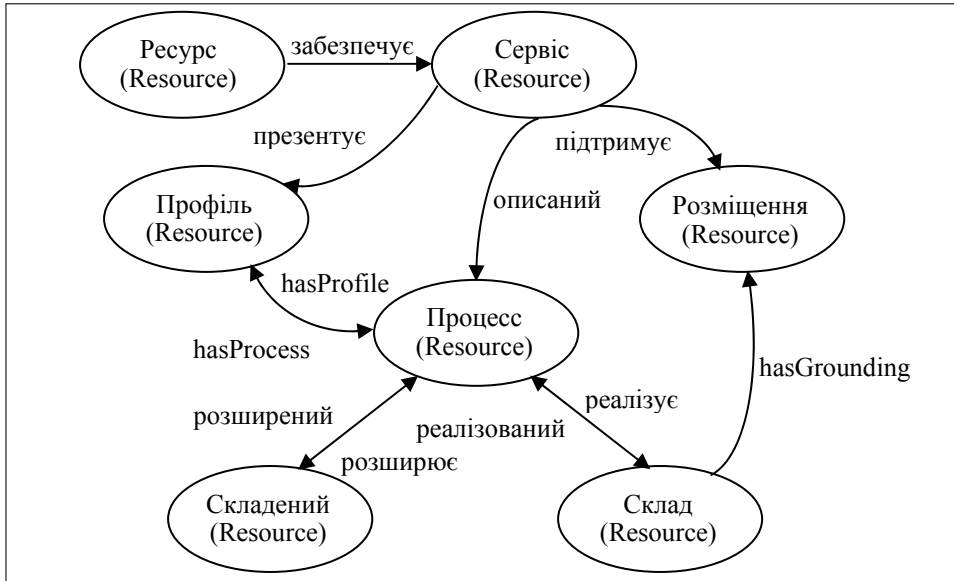


Рис. 2. Структура сервісної онтології згідно концепції OWL-S

Завдання, які вирішуються за допомогою OWL-S, включають:

- автоматичний пошук семантичного Web-сервісу;
- автоматичний виклик семантичного Web-сервісу і завантаження його параметрів;
- автоматичну композицію складу семантичного Web-сервісу і його взаємодії з іншими сервісами;
- автоматичний контроль виконання семантичного Web-сервісу.

У свою чергу, семантичні Web-сервіси можуть використовуватись з метою:

- класифікації обчислювальних ресурсів і ресурсів даних, метрик виконання, управління роботами;
- інтеграції схем описів даних і завдань та їх завантаження;
- роздрукування даних і сервісних повідомлень введення/виведення;
- вибору методів розв'язання задач і використання інтелектуальних порталів;
- інфраструктури для аутентифікації, реєстрації й управління доступом.

Базові сервіси семантичного Web можуть стати базовими Грід-сервісами. Остов семантичного Web є тим засобом, за допомогою якого в Грід можуть бути представлені метадані як відносно його інфраструктури,

так і відносно Грід-додатків. Симбіоз Грід і семантичного Web достатньо очевидний, перспективи їхнього розвитку зв'язані. Від семантичного Web Грід запозичує остов і сервіси для роботи зі своєю семантикою. Сам семантичний Web стимулюється могутнім розвитком Грід-додатків й імітує інфраструктуру Грід.

Необхідно зазначити, що й досі концепція розвитку семантичного Web не має повноцінної реалізації. Серед причин цього можна виокремити наступні:

- *Відсутність мотивації розробників.*

Необхідною умовою впровадження семантичного Web є наявність критичної кількості Web-ресурсів із коректними метаданими у відповідних форматах. Створення таких метаданих покладено на розробників Web-сайтів і є відносно складним завданням, яке не буде виправдано (зокрема фінансово) доти, доки семантичний Web не почне працювати на повну потужність.

- *Неможливість гарантувати достовірність метаданих.*

Семантичний Web надає широкі можливості для опису даних та зв'язків між ними.

- *Засоби семантичного Web є відносно складними.*

Із двох приблизно однаково ефективних підходів перевагу завжди отримує простіший. Прикладом цього є те, що формат RSS 1.0, заснований на RDF, є значно менш популярним, ніж простіший за нього RSS 2.0.

Очікується, що протягом найближчих 3–5 років ситуація докорінно зміниться. Вже функціонує програма Linking Open Data on the Semantic Web, яка збирається взагалі всю відкриту інформацію в Інтернеті перевести в RDF-формат.

СЕМАНТИЧНИЙ ГРІД

Семантичний Грід, який базується на семантичному Web, можна розглядати як спрямовану на сервіси архітектуру, в межах якої об'єкти в певному середовищі обмінюються послугами (сервісами) один із одним на певних умовах. У межах цього середовища можливе встановлення правил взаємодії об'єктів, наприклад, отримання та надання сервісів. Природно представляти власників сервісів та споживачів сервісів як автономних агентів [19].

При цьому агенти мають відповідати наступним вимогам:

- чітко ідентифікувати об'єкти, що вирішують певні задачі та мають чіткі межі й інтерфейси;
- одержувати повідомлення, пов'язані з обладнанням через датчики, і впливати на обладнання, в яке вони вбудовані, через виконавчі елементи;
- виконувати певну процедуру — у них є специфічні цілі (мета);
- бути автономними і здійснювати контроль за своїм внутрішнім станом та поведінкою;
- бути здатними гнучко розв'язувати завдання, для яких були спроектовані, тобто — реагувати своєчасно на зміни, які відбуваються в середовищі, і прораховувати свої дії з випередженням.

Обчислення, засноване на агентах, задовольняє умови динамічно змінного середовища, де автономія агентів дозволяє адаптувати обчислення до

змінних обставин [20]. Це є важливою властивістю для Грід-систем третього покоління. Однією з методик для досягнення зазначеної властивості є оперативний обмін інформацією між агентами.

Для реалізації агентної платформи проведено відповідні дослідження при допущенні, що у кожного власника сервісу будуть один або більше агентів, які діють від його особи. Ці агенти будуть керувати доступом до сервісів, за які вони відповідають, і гарантуватимуть виконання всіх погоджених правил (договорів). Ця діяльність залучає планування місцевих дій відповідно до доступних ресурсів та передбачає, що відповідні послуги буде надано вчасно й відповідно до контракту. Агенти будуть також діяти від імені користувачів сервісів. Залежно від бажаного ступеня автоматизації, це може бути прийняття договорів для отримання сервісу і одержання та представлення будь-яких отриманих результатів.

Відносини між власником і споживачем сервісу засновано на договорі надання сервісів. Цей договір визначає умови користування, за якими власник погоджується надати сервіс споживачу. Точна структура контракту буде залежати від природи обслуговування й відносин між власником і споживачем. Але, відповідні договори включають ціну за виклик обслуговування, інформацію, що споживач повинен надати власнику, очікуваний результат обслуговування, індикацію, коли цей результат може очікуватися, і штраф за невиконання контракту. Такі договори можуть укладатися як off-line, так і on-line, залежно від контексту.

Власники й виробники сервісів взаємодіють один із одним у специфічному контекстному середовищі. Воно може бути поширене на всі об'єкти Грід (мається на увазі, що всі об'єкти пропонують своє обслуговування на повністю відкритому ринку). В інших випадках це може бути закрите середовище, входом до якого можна керувати (об'єкти формують приватний клуб). Надалі це специфічне середовище буде називатись *ринком*, а об'єкт, що встановлює ринок — *власником ринку*.

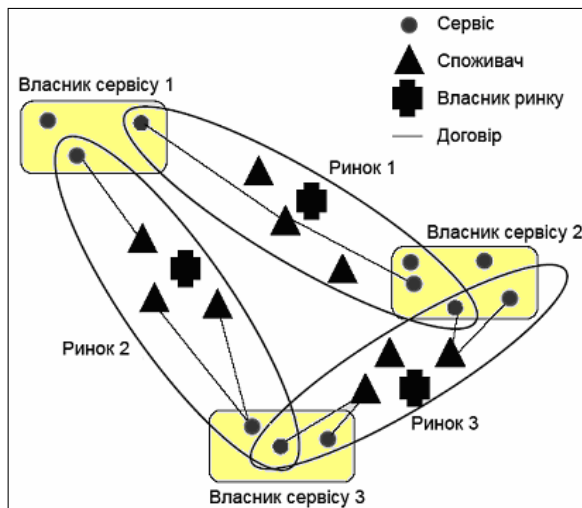


Рис. 3. Основні компоненти сервісної архітектури: округлені прямокутники — власники сервісу; заповнені кола — сервіси; заповнені трикутники — споживачі; лінії між виробниками й споживачами — договори; овали — ринок; заповнений хрест — власник ринку

На рис. 3 наведено сервісну архітектуру семантичного Грід, де власники пропонують сервіси споживачам відповідно до договорів. Кожна взаємодія власника та споживача, що має місце на цьому ринку, встановлюється власником ринку відповідно до правил. Власником ринку може бути один із об'єктів на ринку (виробник або споживач), або це може бути і нейтральна третя особа.

Сукупність ринків, кожен із яких відповідає за різні аспекти е-наукових досліджень, складає спільний науково-дослідницький

простір країни чи континенту. Дані про ринки зареєстровано у відповідній міжнародній базі даних. У цієї бази даних є власний сервіс повідомлень, користуючись яким, окремі вчені можуть реєструвати свою зацікавленість щодо певних специфічних сервісів. Коли нові сервіси створюються та з'являються на ринках, учені інформуються про це через своїх агентів.

Приклад можливої архітектури «розумної» Грід-системи, в якій поєднано програмні агенти і традиційній OGSA-підхід, приведено на рис. 4 [21]. У такій архітектурі агенти забезпечують інтелектуальні розподілені сервіси, які включають комунікацію і прийняття рішень.

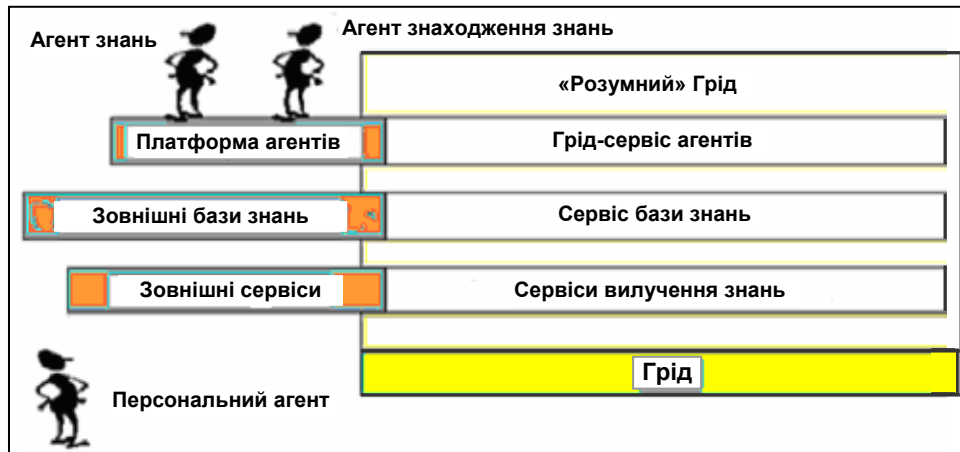


Рис. 4. Архітектура «розумного» Грід із програмними агентами

Грід-сервіс агентів відіграє головну роль при взаємодії агента і Грід-сервісу, і тому він — посередник між *платформою агентів* та іншими Грід-сервісами. Цей сервіс перетворює повідомлення агентів у точні дії й перетворює результати цих дій зворотньо в повідомлення агентів. Цей сервіс тісно співпрацює із *сервісом бази знань*, яка містить дані і їх семантику, а також інформацію про всі сервіси і ресурси, що можуть бути використані для пошуку знання.

Сервіси вилучення знань здатні виконувати пошук знань у базах даних, об'єднаних у Грід-інфраструктурі, використовуючи існуючий арсенал ефективних методів Data Mining [6].

Зовнішні сервіси розширюють систему «розумного» Грід даними, отриманими за допомогою сервісів за межами Грід-інфраструктури, наприклад, Web-сервісами.

Агент знань (A3) представляє «розумний» Грід для зовнішнього світу під час комунікації з іншими агентами. З точки зору програмного забезпечення і зовнішнього світу — це інтелектуальний агент програмного забезпечення, але з точки зору «розумного» Грід — це специфічний Грід-додаток. Фактично, цей агент працює вхідним і вихідним інтерфейсом «розумного» Грід і, тому, він — посередник (медіатор) між запитом щодо знань й інформаційними ресурсами, з іншого боку. Цей агент створено і керується Грід-сервісом агентів і може взаємодіяти з іншими агентами за допомогою повідомлень у форматі ACL [22].

Агент знаходження знання (A33) використовується «розумним» Грід, для того, щоб знайти відповідні сайти, у випадку, коли інформація, яку за-

питують, не знаходиться в *базі знань*, але її розташування відоме. Цей агент зареєстровано в Платформі агентів і керується *Грід-сервісом агентів*, який надає агентові інструкції, як запросити бажані дані, й інформацію про розташування даних. Агент може спілкуватися з іншими агентами, а також із семантичним Web, який є іншим джерелом інформації і джерелом знання всієї системи.

Персональний агент (ПА). Користувач, який запитує про знання, генерує ПА. Цей агент може адресувати питання до «розумного» Грід і передавати відповіді користувачам. Успіх цієї діяльності залежить від порозуміння агентів ПА і АЗ, а також від конструкції запиту. ПА не є частиною системи, тому що він створюється користувачем і не реєструється в Платформі агентів. Цей агент може запросити у АЗ список доменів, про які він має знання, або ПА може інформувати АЗ про онтологію, яка описує його домен, і об'єднати її разом із онтологіями АЗ, щоб обрати один загальний домен.

Розглянуті задачі можна вирішити за допомогою мови ACLs (Agent Communication Languages — мови комунікації агентів). Зокрема, консорціум FIPA (Foundation for Intelligent Physical Agents) створює програмні стандарти для різнорідних і взаємодіючих агентів і систем, заснованих на агентах. В абстрактній архітектурі FIPA [22]:

- агенти спілкуються, обмінюючись повідомленнями, які відображають динаміку змін і закодовані мовою комунікації агента;
- сервіси надають підтримку агентам, включаючи також сервіси каталогів і сервіси транспортування повідомлень;
- сервіси можуть бути здійснені у вигляді агентів або у вигляді програмного забезпечення, до якого можна звернутися через мови програмування інтерфейсів (наприклад, Java, C++ або мови опису інтерфейсів).

База знань (БЗ) містить специфічні дані про реальні об'єкти і відносини між цими об'єктами і їх властивостями. Вона складається з двох частин: перша пов'язана з добре розробленою *онтологією*, а друга — зі зразками цієї онтології.

Оскільки багатоагентну інтелектуальну платформу ще не створено, семантичний Грід сьогодні можна базувати на вживаному проміжному програмному забезпечення (middleware), додавши до базових сервісів, характерних для Грід-систем, наступні семантичні сервіси:

- *метаданих*: збирання інформації з різних джерел та розміщення її в одному місці;
- *анотацій*: відслідкування, фільтрування, аналіз та пошук сервісів, публікація інформації про них;
- *онтологій* об'єктів та ресурсів, де онтологія — це засіб опису семантики проблемної області за допомогою словника і підібраної специфікації існуючих у ній відношень та обмежень, які забезпечують інтеграцію словника;
- *композиції*: автоматична композиція сервісу як процесу відкриття, інтегрування та виконання набору пов'язаних сервісів у належній послідовності для формування змістовного та комплексного сервісу відповідно до підходів *оркестровки* та *хореографії*. При цьому оркестровка веде до централізованої архітектури, у якій механізм оркестровки контролює виконання

завдання, а хореографія ґрунтується на співробітницькому підході, в якому кожна група процедур відіграє свою роль, виконуючи дії, задані лише для неї.

При цьому, документи в мережі публікуються в форматі XML, який містить семантичні RDF-твердження. Відношення, аксіоми та твердження про об'єкти зберігаються в онтологіях, що реалізуються мовою OWL. Запросити дані з RDF можливо за допомогою SPARQL.

У результаті архітектура OGSA перетворюється в архітектуру S-OGSA (рис. 5) [23, 24].

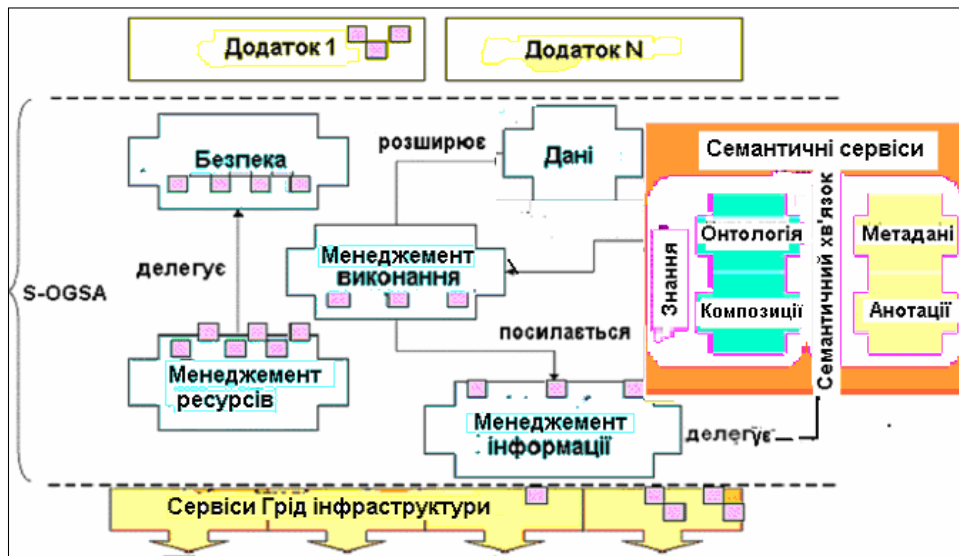


Рис. 5. Від OGSA до S-OGSA

Підхід S-OGSA намагається розширити архітектуру OGSA і забезпечити повне використання семантики разом із використанням сервісу, визначеного як *сервіс знань*. Вказаний підхід намагається реалізувати семантичний Грід методом «зверху до низу», створюючи архітектури, які мають задовольнити велику кількість додатків і вимог. Кожен Грід-сервіс переважно використовує метадані. Гнучка композиція Грід-сервісів вимагає наявності інформації про функціональність, доступність та інтерфейси самих різних сервісів. Виявлення сервісів і брокерське обслуговування використовують описи на рівні метаданих. Композиція сервісів керується і підтримується описами метаданих. Тобто, повне пов'язання Грід-сервісів із *метаданими є ключовим аспектом* архітектури S-OGSA.

На мові OWL-S можливості сервісу описуються відповідним *IOPE* (вхід, вихід, попередні стани й ефекти) [1]. Такі описи можуть бути взяті з WSDL-описів сервісу і є достатніми для сервісів, що не мають стану, сервіси зі станом, тобто WS-ресурси, складаються з сервісу і ресурсу зі станом. Властивості ресурсу (BP), як визначено в WSRF-специфікації, можуть бути динамічними. Вони можуть бути створені або знищені у часі по ходу процесу.

У процесі виконання сервісів, що мають стан, можливе використання спадкоємства властивостей ресурсу. Хоча точна ієрархія властивостей ресурсу може допомогти в генеруванні семантики, не існує стандарту, який описував би детально, як BP спадкоємство має бути реалізовано. Це може

викликати серйозні складнощі в синтаксичному аналізі сервісу і необхідно буде представити спеціальну документацію, щоб розібрати ВР-ієрархію. До того ж, властивості ресурсу можуть бути пристосовані тільки для моделювання відповідних входів і виходів сервісу, тобто сервіс приймає лише спеціальні завдання і може презентувати входи і виходи, як властивості завдання, ховаючи, таким чином, введення/виведення у властивостях ресурсів сервісу. Це необхідно враховувати при складанні сервісу.

Окрім складнощів у процесі пошуку сервісу, існує невелика різниця в процесі активізації сервісу. WS-ресурс складається з сервісу і ресурсу зі станом, тобто його визначено як ЕРР (end point reference), який описує не тільки адресу сервісу, але й ідентифікацію ресурсу. Ідентифікація сервісу в реалізації OWL-S має бути розширена в складнішу структуру. Для Грід-сервісів таке розширення надає можливість отримати декілька варіантів одного сервісу, розміщених на різних серверах. Проте, це можна зробити простіше шляхом генерації кількох OWL-S для сервісу.

У роботі [25] розроблено інструментарій для автоматичної генерації описів OWL-S Грід-сервісів, що мають стани, і для тих, що їх не мають, з відповідних описів Web-сервісів (WSDL).

Переклад починається з аналізу конфігурації і URL WSDL-документа. Перекладач аналізує WSDL-документ, відмічаючи операції типу портів, входів, виходів і властивості ресурсу. Потім перекладач генерує для кожної WSDL-операції скелет OWL-S документа, а також створює входи, виходи, початкові стани й ефекти та розв'язує елементи в онтологічні концептори, визначені в конфігурації. Якщо необхідно, він може створити онтологію, яка моделює властивості ресурсу даного сервісу.

Приклад перекладу WSDL-опису метеорологічного сервісу показано на рис. 5. Оскільки кожен сервіс має декілька операцій, семантичний опис генерується для кожної операції з метою дати можливість створювати робочі потоки сервісних операцій. Окрім WSDL-опису процес перекладу також вимагає знання конфігурації, яка описує відповідність введення/виведення WSDL онтологічним концепторам домену. Ці концептори можуть описувати інформацію про сервіс (наприклад, ім'я сервісу, провайдера), а також складні входи і виходи сервісів, такі, як географічне розташування, географічну інформацію, басейни річок і т.ін.). Далі вони зможуть ідентифікувати форми HTML, які представляються користувачеві, якщо необхідні додаткові вхідні дані.

Подібне перетворення описів WSDL в OWL-S здійснюється також у семантичному Грід, описаному в роботі [26].

Розглянуто підхід перетворення описів сервісів притаманний методології реалізації семантичного Грід «з низу до верху», яка намагається витягнути якомога більше користі з існуючих семантичних Web-сервісів. Існує також альтернативна можливість побудови семантичного Грід «з низу до верху», базуючись на підтримці OGSA-DAI (Open Grid Service Architecture — Data Access Interfaces) і створенні нового програмного забезпечення проміжного шару InteliGrid [27].

WSDL опис

```
<wsdl:portType name=«MM5ServicePortType»
  wsrp: ResourceProperties=«tns:MM5Properties»>
  <wsdl:operation name=«configureFromProperties»>
  <wsdl:input message=«tns:ConfigureInputMessage»/>
  <wsdl:output message0=«types:VokiOutputMessage»/>
  </wsdl:operation>
  . . .
  <wsdl:service name0:MM5Service».
  <wsdl:port name=«MM5ServicePortTypePort»>
  binding=«binding: MM5ServicePortTypeSOAPBinding»>
  <soap:address
  location=«http://localhost:8080/wsrt/setvices/»/>
```

Конфігурація

```
MM5Service:configureFromProperties.properties=http://gom.kw
fgrid.net/gom/ontology/DomainApplicationOntology/
  FFSC#MM5Properties
  MM5Service.configureFromProperties.voidResponse=
  Effect=
  http://gom.kwfgrid.net/gom/ontology/DomainServiceOntology/
  FFSC#MM5isConfigured
```

OWL-S опис

```
<service:Service
  rdf:ID=«configureFromProperties_MM5Service»>
  <service:presents>
  <profile:Profile
  rdf:ID=«configureFromProperties_MM5Profile»/>
  </service:presents>
  </service:describedBy>
  <process:AtomicProcess edf:ID=
  «configureFromProperties_MM5AtomicProcess»/>
  </service:describedBy>
  <service:supports>
  <grounding:WsdLGrounding
  <grounding:WsdLGrounding rdf:ID=
  «configureFromProperties_MM5Grounding»/>
  </service:supports>
  </service:Service>
  . . .
  <profile:Profile
  rdf:about=«configureFromProperties_MM5Profile»>
  <profile:hasInput
  <process:Input rdf:ID=«properties»>
  <process:parameterType>= http://gom.kwfgrid.net/gom/
  ontology/DomainApplicationOntology/FFSC#MM5Properties
  </process:parameterType>
  </process:Input>
```

Рис. 6. Приклад перекладу WSDL-опису метеорологічного сервісу в OWL-S опис

Якщо на сьогодні існують сотні успішних Грід-проектів із різних галузей науки і техніки для традиційного Грід-середовища [6], то проектів семантичного Грід поки що дуже мало. Інформація про існуючі пілотні розроблення семантичних Грід наведено у таблиці. Слід зазначити, що їх присвячено вирішенню насущних викликів людству, і значну частину з них пов'язано з проектами 6-ї Рамочної програми (FP-5). На сьогодні 7-й Рамочній Програмі подібних проектів значно більше. У найближчому майбутньому семантичні Грід використовуватимуться в складних медичних аналізах, при наркотичному контролі, фінансовому аналізі ризиків, кризовому управлінні, цифрових ринках та розробці виробів.

Таблиця. Пілотні проекти з семантичного Грід

	Назва	Опис	Призначення
1.	K-Wf Grid [25]	Розроблюється методологія компонування сервісів і додатків, підтримки паралельних обчислень, програмування Грід через визначення міжкомпонентних залежностей. Кожен компонент має декілька можливих моделей обчислень представлених у Web або Грід-сервісах, наприклад: метеорологічні методи (Aladin, MM5); гідрологічні методи (HSPF, NLC і т.ін.). Система містить автоматичний перекладач WSDL-опису сервісу в OWL-S	K-Wf Grid використовують для передбачення повеней
2.	InteliGrid [27]	Проект InteliGrid присвячено об'єднанню в Грід-системі семантичної сумісності віртуальних організацій і Грід-технологій для потреб промислових організацій у співпраці при використанні виробничої інфраструктури і комунікацій, обробці інформації. Було розроблено спеціальну платформу з різними сервісами як послугами управління документообігом. Платформа забезпечує доступ до серверів із промисловими зразками, а також забезпечує використання високопродуктивної обчислювальної інфраструктури	Архітектура InteliGrid має задовольняти велику кількість додатків і вимог
3.	myGrid [28, 8]	Адаптивна інфраструктура, яку заснувано на поняттях Грід-систем, семантичного Web і онтологій, призначена для потреб біоінформаційних і біомедичних застосувань. Вона містить сервіси та реєстратори даних, засновані на RDF; використовує RDF-метадані для експериментальних компонентів і засновані на RDF-графі походження даних; експлуатує для бази даних керовані словники на OWL і засновану на OWL-процедуру інтеграції даних	Призначено для галузі біоінформатики, зокрема, дослідження генів
4.	Insurance Grid [23]	Один із перших проектів у галузі використання технологій знань з метою збільшення та розширення архітектури можливостей Грід-систем. Демонструє вагомий приклад застосування Грід у бізнесі й забезпечує ефективні сервіси знань	Insurance Grid використовується в страхових компаніях для спрощення процесу страхових виплат й інтеграції послуг ланцюжка партнерів, залучених у процес

5.	Earth System Grid (ESG) [29]	Грід ESG є віртуальним середовищем для досліджень у галузі науки про Землю і аналізу глобальних змін клімату. Мета ESG — забезпечити пошук і безпечний доступ до величезних обсягів даних, накопичених наукою про Землю. Пошук даних за допомогою метаданих став головним аспектом ESG. Було розвинуто схеми метаданих, прототипи онтологій, сервіси пошуку і відкриття даних, засновані на семантиці	Користувачі — це вчені, які досліджують клімат у національних лабораторіях, університетах та інших урядових організаціях, забезпечуючи експертизу для Міжурядової Групи зі змін клімату
6.	Molecular GRID [30]	Властивості молекул мають дуже добре визначену семантику і дозволяють створення семантичних Грід. Було створено мову CML (Chemical Markup Language) і онтології, засновані на словниках, для підтримки широкого кола додатків, зокрема публікації даних про безпечні хімічні суміші. Велика кількість властивостей молекул може бути розрахована програмами з квантової механіки (зокрема, геометрія молекул, їх енергія, диполь, заряд, частота і т.ін.). Час моделювання для кожної молекули різниться в мільйон разів (від 0,3 сек до 4 днів)	250,000 молекул були описані, перетворені на XML, і розщеплені на 500 партій. Кожну партію оброблено одним завданням для програми Condor
7.	MONET [31]	Забезпечує формування і вирішення математичних задач у термінах домену MKMnet (Mathematical Knowledge Management NET-work — менеджмент математичних знань у мережі) за допомогою спеціальної мови OpenMath (Markup Language For Mathematics — мова розмітки для математиків); аналізує сутність математичної задачі через технологію автоматичного доказу Calculemus (Mathematical Reasoning Network — мережа математичних доказів), ідентифікує і складає замовлені сервіси з сервісного WSIL. Директорія UDDI завантажує програмні компоненти до апаратних ресурсів OGSA/Semantic Grid	Створено відповідну математичну онтологію і реалізовано мережеве використання програми Maple
8.	saBIG [32]	saBIG інтегрує дані, що збираються різноманітними постачальниками (наприклад, лабораторіями, клініками) у різноманітних форматах і з багатодисциплінарних концептуальних точок зору. Для saBIG розроблений домен моделей онтологій і метаданих, який формує основу для інтеграції даних. Послуги saBIG вирішують узгодження ресурсів (наприклад, баз даних, файлів) і даних з інтерфейсами, побудованими на Web-послугах, і дозволяють створювати черги, використовуючи стандартні мови для черг даних	Система saBIG вільно об'єднує дослідників, лікарів і пацієнтів для розповсюдження інформації про засоби запобігання і лікування раку

Інтерфейс користувача, як і в традиційних Грід, може бути забезпечено через спеціальні семантичні Web-портали знань, за допомогою яких:

- користувач створює проксі-сертифікат, щоб отримати дозвіл на роботу в Грід-системі;
- користувачу забезпечується можливість редагування при підготовці текстів, і «показчиків» Грід-сервісів і даних. При клацанні клавiші миші контекстне меню сервісів показується користувачу, за допомогою якого він контактує із директорією сервісів і поновлює сервіси, які йому доступні;



Рис. 6. Три рівні програмування в семантичному Грід

- користувач переглядає сервіси за категорією або шукає їх за ключовими словами. Після того, як користувач знайшов сервіс, надалі він здійснює пошук даних, які необхідні сервісу. Дані, що запрошуються, розміщені у файлах і відповідають вибраній RDF-онтології;
- сервіс із відповідними даними запускається на віддаленому сервері. При цьому брокер ресурсу, базуючись на розташуванні або вимогах сервісу (визначених як фрагмент із онтології сервісу), інформації про ресурс (визначену, як фрагмент онтології ресурсу) і розташування даних, знаходить оптимальне місце, щоб виконати сервіс. На екрані порталу створюється область для друку результатів або їх візуалізації;
- сервіс виконується і результати запам'ятовуються для подальшої обробки або повертаються до користувача. Користувач може дочекатися обробки обчислення, або зберегти поточний результат. Щойно обробку завершено, результати відображаються у визначеному місці екрану порталу.

Необхідно зазначити, що при розробці і використанні семантичного Грід і Web-порталу знань, програмування самих семантичних сервісів і засобів їх використання для нього, ведеться на кількох рівнях із залученням сучасного інструментарію програмування (рис. 6). На думку Білла Гейтса, у міру повсюдного проникнення Інтернету все більшу популярність набуває концепція «програмне забезпечення як сервіс». Тобто, користувачі будуть працювати з необхідним ПЗ через Web і Грід, отримуючи на свої комп'ютери готові результати. Таким чином, такі програми як PHP, ASP, JSP-скрипти, JavaBeans, COM-об'єкти та всі інші засоби програмування вже зараз можуть звертатися до будь-якої програми, яка працює на іншому сер-

вері (тобто до Web-сервісу), і використовувати відповідь, отриману від неї на своєму Web-сайті або додатку.

Відповідно, необхідність у великих локальних обчислювальних потужностях частково відпаде, що сприятиме зростанню попиту на недорогі комп'ютери з низьким енергоспоживанням.

Прототип такого Web-порталу знань зараз розроблюється в Інституті прикладного системного аналізу НАНУ і МРНУ (ІПСА) згідно із проектом № ІТ / 558–2009 від 20 липня 2009 р. Державної програми ІКТ на 2006–2010 роки.

ВИСНОВКИ

Семантичний Грід як розширення сучасного Грід надзвичайно корисний для е-науки, оскільки його технології дають змогу легко, швидко та зручно працювати зі знанням, а не з інформацією, як це робиться зараз. Це дає можливість ученим, інженерам і бізнесменам працювати за фантастичними на сьогодні сценаріями.

На сьогодні відбувається активна розробка семантичного Web, але *достатнього впровадження його сервісів поки що немає*. Семантичний Web часто представляють як глобальну базу знань, особливо у світлі нещодавно розпочатих у Web-мережі проектів побудови семантичних енциклопедій, баз знань, лексичних баз розмовних мов та інших. Ідея семантичного Web — це концепція мережі, в якій кожен ресурс на людській мові забезпечений описом, зрозумілим комп'ютеру. Синтаксична взаємодія людей і комп'ютерів, та комп'ютерів між собою, полягає в коректному синтаксичному аналізі даних і вимагає побудови відповідностей між термінами, яка, в свою чергу, вимагає аналізу контексту за допомогою формальних явно заданих специфікацій моделей доменів (*онтологій*), які визначають використані терміни та зв'язки між ними.

Семантичний (або «розумний») Грід знаходиться на початковому етапі і для своєї реалізації потребує випереджаючого розроблення *багатоагентної інтелектуальної платформи*, хоча вже зараз можна розробляти і досліджувати його архітектуру і можливі властивості. Оскільки таку агентну платформу ще не створено, перші експериментальні зразки семантичного Грід будуються на базі існуючого Грід-середовища, яке базується на вживаному проміжному програмному забезпечення (middleware). Однак у middleware типу NorduGrid і gLite3, що використовуються в Україні та всій Європі, не реалізований стандарт WSRF сумісності Грід- і Web-сервісів, тому необхідно буде в національній Грід-інфраструктурі перейти на використання американського middleware типу Globus Tools 4 (GT-4) або дочекатися поки вийде наступна версія gLite-5, яка зараз розробляється в Європі із врахуванням стандарту WSRT.

Дослідження з можливої архітектури і властивостей семантичного Грід передбачено новою Державною програмою впровадження Грід-технологій на 2009–2013 роки, зокрема, в її розділі, присвяченому «розробленню і впровадженню методів ефективного використання існуючих Грід-ресурсів за допомогою інтелектуальної обробки даних (DataMining)» [6]. Семантич-

ний Web-порталу знань, що розробляється зараз в ПСА, призначено для дослідження заходів, необхідних для побудови національних реєстраторів схем метаданих і онтологій, заснованих на міжнародних стандартах, алгоритмів вилучення знань з даних та їх подальшого ефективного використання в національній Грід-інфраструктурі й мережі Світових Центрів Даних (СЦД), в якій успішно функціонує український СЦД з геоінформатики та сталого розвитку.

ЛІТЕРАТУРА

1. Згуровський М.З., Петренко А.І. Е-наука на шляху до семантичного Грід. Частина 1: Об'єднання Web- і Грід-технологій // Системні дослідження та інформаційні технології. — 2010. — № 1. — С. 26–38.
2. Згуровський М.З., Петренко А.І. Grid-технології для е-науки і освіти // Наукові вісті НТУУ «КПІ». — 2009. — № 2. — С. 10–17.
3. De Roure D., Jennings N.R., Shadbolt N.R. The Semantic Grid: Past, Present, and Future // Proceedings of the IEEE. — 2006. — **93**, № 3. — P. 669–681.
4. Петренко А.І. Семантичний Грід для гнучкого оброблення даних. — Матеріали конф. «Системний аналіз та інформаційні технології», 26–30 травня 2009 року. — С. 16–17.
5. Петренко А.І. Застосування Грід-технологій в науці і освіті. — Київ: Політехніка, 2009. — 145 с.
6. Петренко А.І. Grid і інтелектуальна обробка даних // Системні дослідження та інформаційні технології. — 2008. — № 4. — С. 97–110.
7. Berners-Lee T., Hendler J., Lassila O. The Semantic Web // Scientific American. 2001. — May. — P. 598–602.
8. The Gene Ontology. — <http://www.geneontology.org/>.
9. Natalya F. Noy & Deborah L. McGuinness. Ontology Development 101: A Guide to Creating Your First Ontology // Stanford Knowledge Systems Laboratory Technical Report KSL-01-05 and Stanford Medical Informatics Technical Report SMI-2001-0880, March 2001. — 25 p.
10. Онтологія UNSPSC. — <http://www.unspsc.org/>.
11. OWL Web Ontology Language Overview. — <http://www.w3.org/TR/owl-features/>.
12. The Protégé Ontology Editor and Knowledge Acquisition System. — <http://protege.stanford.edu/>.
13. Ontolingua (1997). Ontolingua System Reference Manual. — <http://www-ksl-svc.stanford.edu:5915/doc/frame-editor/index.html>.
14. Chimaera Ontology Environment. Chimaera (2000). — <http://www.ksl.stanford.edu/software/chimaera>.
15. The Foundation for Physical Agents. — <http://www.fipa.org/>.
16. OWL-based Web Service Ontology. — <http://www.daml.org/services/owl-s/>.
17. Mindswap OWL-S API. — <http://www.mindswap.org/2004/owl-s/api/>.
18. METEOR-S / WSDL-S. — <http://lsdis.cs.uga.edu/Projects/>.
19. Semantic Grid Community Portal. — <http://www.semanticgrid.org/>.
20. Jennings N.R. On agent-based software engineering // Artificial Intelligence 117 (2000). — P. 277–296.
21. A. Min Tjoa, Brezany P. and Janciak I. Towards Grid Based Intelligent Information Systems. Vienna University of Technology, Vienna, Austria, 2006. — 12 p.
22. Foundation for Intelligent Physical Agents. Fipa agent management specification. — <http://www.fipa.org/specs/fipa00023/>, 2000.

23. *Alper P., Corcho O., Kotsiopoulos I., Missier P., Bechhofer S., Goble C.* S-OGSA as a Reference Architecture for OntoGrid and for the Semantic Grid // GGF16 Semantic Grid Workshop. Athens, Greece, February 2006. — P. 87–94.
24. *M. Antonioletti et al.* The Design and Implementation of Grid Database Services in OGSA-DAI. *Concurrency and Computation: Practice and Experience*, 17(2–4), 2005. — P. 357–376.
25. *Knowledge-based Workow System for Grid Applications* FP6 IST project. — <http://www.kwfgrid.net>.
26. *Mirko Niinimaki.* Grid Resources, Services and Data — Towards a Semantic Grid System. (PhD Thises). Department of Computer Science, University of Tampere, Finland, January 2006. — 106 p.
27. *Dolenc M., Klinc R., Turk Ž., Katranuschkov P., Kurowski K.* Semantic Grid Platform in Support of Engineering Virtual Organisations // *Informatica*. — 2008. — № 32. — P. 39–49.
28. *Сайт проекту myGrid.* — <http://www.mygrid.org.uk>.
29. *The Gene Ontology.* — <http://www.geneontology.org/>.
30. *Murray-Rust P. and Rzepa H.S.* *Journal of Chemical Information and Computer Science*. — 2003. — № 43. — P. 234–238.
31. *MONET: Mathematical service discovery and composition.* — <http://monet.nag.co.uk/>.
32. *Cancer Biomedical Informatics Grid.* — <http://cabig.cancer.gov/index.asp>.

Надійшла 10.09.2009