**ГЛАВА 7. ІНСТРУМЕНТАЛЬНИЙ КОМПЛЕКС ОНТОЛОГІЧНОГО**

**ПРИЗНАЧЕННЯ**

**7.1. Аналіз функціональних можливостей засобів побудови онтологій**

Методологія проектування онтології ПдО (глава 4) допускає формування множин концептів, відношень, функцій інтерпретації та аксіом. Побудова вказаних множин вручну є трудомістким процесом, як за часом, так і по кількості залучених в процес проектування висококваліфікованих фахівців. Ручне проектування онтологій мало чим відрізняється від проектування експертних систем.

Розуміння важливості створення інструментальних засобів підтримки процесу проектування онтології заданої ПдО прийшло практично одночасно з прийняттям парадигми комп‘ютерних онтологій. На сьогодні відомо більше ста інструментальних програмних систем[[1]](#footnote-1) [5, 52, 91, 93, 154], які мають наступні характеристики.

1. ***Підтримуючі формалізми та формати представлення***.

Під формалізмом розуміється деяка формальна теорія, що лежить в основі способу представлення онтологічних знань (логіка предикатів, фреймові моделі, дескриптивна логіка, концептуальні графи та ін.). Формалізм суттєво впливає на внутрішні (комп‘ютерні) структури даних і може визначати їх формат представлення.

Формат представлення онтологій задає вид їх збереження в бібліотеці та спосіб передачі онтологічних описів іншим споживачам. В якості форматів розроблені мови представлення онтологій, найбільш відомими з яких є OWL, RDFS, KIF.

Деякі з відомих редакторів онтологій підтримують роботу з декількома формалізмами представлення, але тільки один формалізм і формат являються переважними для даного (конкретного) редактора [91].

1. ***Функціональність****.*

Є однією із самих важливих характеристик редакторів онтологій, під якою розуміється множина надаваних користувачу сценаріїв роботи з онтологічними структурами.

Базовий набір функцій забезпечує:

* роботу з одним або декількома проектами одночасно;
* графічний інтерфейс з користувачем;
* редагування онтології (створення, редагування, видалення концептів, відношень, аксіом та інших структурних елементів онтології).

1. ***Архітектура додатку, місце збереження онтологій, мова програмного забезпечення, інтерфейс користувача, доступність****.*
2. ***Додаткові можливості***.

До них відносять підтримку мови запитів, аналіз цілісності, використання механізму логічного виведення, підтримку віддаленого доступу через Інтернет, документування.

Відомі три групи ІнЗ онтологічного інжинірингу [154].

*До першої групи* відносять інструменти створення онтологій, які допускають підтримку сумісної розробки й перегляду, створення онтології у відповідності з заданою (довільною) методологією, підтримку міркувань.

*До другої групи* відносять інструменти об‘єднання, відображення та вирівнювання онтологій.

Об‘єднання допускає знаходження подібностей і розходжень між вихідними онтологіями та створення результуючої онтології, яка містить елементи вихідних онтологій. Для цього ІнЗ автоматично визначають відповідності між концептами або забезпечують графічне середовище, в якому користувач сам знаходить ці відповідності.

Процедура відображення заключається в знаходженні семантичних зв‘язків різних онтологій.

Процедура вирівнювання онтологій встановлює різні види відповідності між двома онтологіями, інформація яких зберігається для подальшого використання в додатках користувача *До першої групи* відносять інструменти [155].

*До третьої групи* відносять інструменти для анотування Web-ресурсів на основі онтологій.

Змістовний огляд відомих інструментів інженерії онтологій, в якому розглянуті основні функції та можливості ІнЗ, їх достоїнства, недоліки, порівняльний аналіз і опис відомих доступних онторедакторів наведено в [154, 156–158].

Загальними недоліками відомих інструментальних засобів є:

* відсутність процедур автоматичного (автоматизованого) формування компонент онтології;
* англомовний інтерфейс з користувачем, в якому (для більшості ІнЗ) не передбачено присвоєння імен компонентам онтології на російській чи українській мові;
* структуризація концептів виконується тільки по одному типу відношень;
* для більшості загальнодоступних ІнЗ не передбачена робота з великими по об‘єму онтологіями (наприклад, для OntoEditFree – до 50 концептів);
* більшість інструментів зберігає свої онтології в текстових файлах, що обмежує швидкість доступу до онтологій;
* задекларовані функціональні можливості для загальнодоступних інструментів найчастіше так і залишаються нереалізованими;
* недостатньо інформації для користувачів в інструкціях.

Описані вище недоліки відомих англомовних ІнЗ відбилися і в розробці аналогічних інструментів для російського сегменту, зокрема “Багаторівневий редактор онтологій” [159] “SIMER + MIR”, “OntoEditor+”[[2]](#footnote-2).

Вказані інструментальні засоби є закритими системами, тому оцінити їх функціональні можливості в повному обсязі не представляється можливим, хоча останній інструмент висвітлений в літературі достатньо і представляє собою найбільш близький прототип до описаного в наступних розділах інструментального комплексу автоматизованої побудови онтологій предметних галузей.

Інструментальна система візуального проектування “OntoEditor+” [160] є спеціалізованою СКБД. Система призначена для ручного редагування онтологій, що зберігаються в реляційній базі даних, а також обслуговування запитів користувачів і зовнішніх програм. Розширені можливості системи забезпечуються функціональним набором “Лінгвістичний інструментарій”, посередництвом якого реалізується вбудовування прикладної онтології в лінгвістичні додатки. Найбільш типовими задачами, що вирішуються за допомогою інструментарію системи “OntoEditor+”, є: задачі корпусного дослідження (завантаження корпусу, сегментація на речення, автоматичне ведення статистики по різним об‘єктам корпусу), вивчення структурних властивостей прикладної онтології за допомогою дослідницького інструментарію, побудова лінгвістичної оболонки прикладної онтології, покриття онтологіч­ними входами, побудова виведень по прикладній онтології та підтримка протоколів інформаційного обміну системи з зовнішніми програмними модулями, в тому числі з зовнішніми інформаційними ресурсами.

Для вирішення структурно складних задач (наприклад, добування з текстів інформаційної моделі алгоритмів) в певній проблемній ситуації, контролю структурної та інформаційної цілісності виділеної схеми алгоритмів, створена спеціалізована система обробки текстових документів “ЛОТА” [161], в яку входять три взаємодіючі підсистеми: підсистема лінгвістичного аналізу технічних текстів “Аналізатор”, підсистеми “Інтегратор” і “ OntoEditor+”.

Підсистема “Аналізатор” реалізує основні етапи лінгвістичної обробки тексту (графематичний, морфосинтаксичний і частковий семантичний аналіз).

Підсистема “Інтегратор” виконує зовнішній запит на добування знань з тексту. Структура зовнішнього запиту містить компоненти інформаційної моделі алгоритму. Зовнішній запит інтерпретується при взаємодії з підсистемою “OntoEditor+” як структура, що прив‘язана до прикладної онтології. Виділення компонент інформаційної моделі відбувається на основі механізмів ототожнення елементів дерева сегментів вихідного тексту (взаємодія з підсистемою “Аналізатор”) і елементів структури запиту (взаємодія з підсистемою “OntoEditor+”).

Основними вимогами при виборі редактора онтологій для практичних цілей являються:

* безкоштовне розповсюдження;
* локальний додаток Web-інтерфейсу;
* розширення функціональних можливостей додатку;
* можливість підключення плагінів, розроблених користувачами.

Аналіз численної літератури по онторедакторам показав, що наведеним вище вимогам задовольняє онторедактор Protégé. Він, крім того:

* найбільш широко використовується розробниками онтологій [156];
* версії додатку постійно поновлюються. Для версії 4.0 основними перевагами є: доступність більшого числа плагінів, в тому числі розуміння та генерація (англомовного) ПМТ, скріпти та ін.; конфігуровна (постійна) схема компонентів; створення, імпорт і експорт конфігуровних користувачем ярликів;
* реалізація будь-якої методології проектування онтології;
* наявність плагіну Prompt, виконуючого різні процедури маніпулювання двома онтологіями (зокрема, процедура Merge може слугувати прототипом процедури системної інтеграції онтологій предметних областей). На рис. 7.1 показані дві прості онтології з домену “Інформатика”, які об‘єднуються в онторедакторі Protégé плагіном Prompt, а на рис. 7.2 – результат вказаного об‘єднання;
* мовою програмного продукту є Java, що використовується також в оригінальних програмах інструментального комплексу, описаного нижче;
* наявність машини виведення;
* можливість присвоювати імена концептам онтології на російській або українській мовах;
* змістовна відповідність схем компонент онтології, що проектується, в Protégé і описаній в главі 4. Вказану відповідність представлено на рис. 7.3, на якому прийняті наступні скорочення:  – відношення “клас-підклас”,  – відношення “клас-властивість”,  – відношення “клас-елемент”,  – відношення “елемент-властивість”, *X*, *R*, *F*, *A* – компоненти формули (4.1).

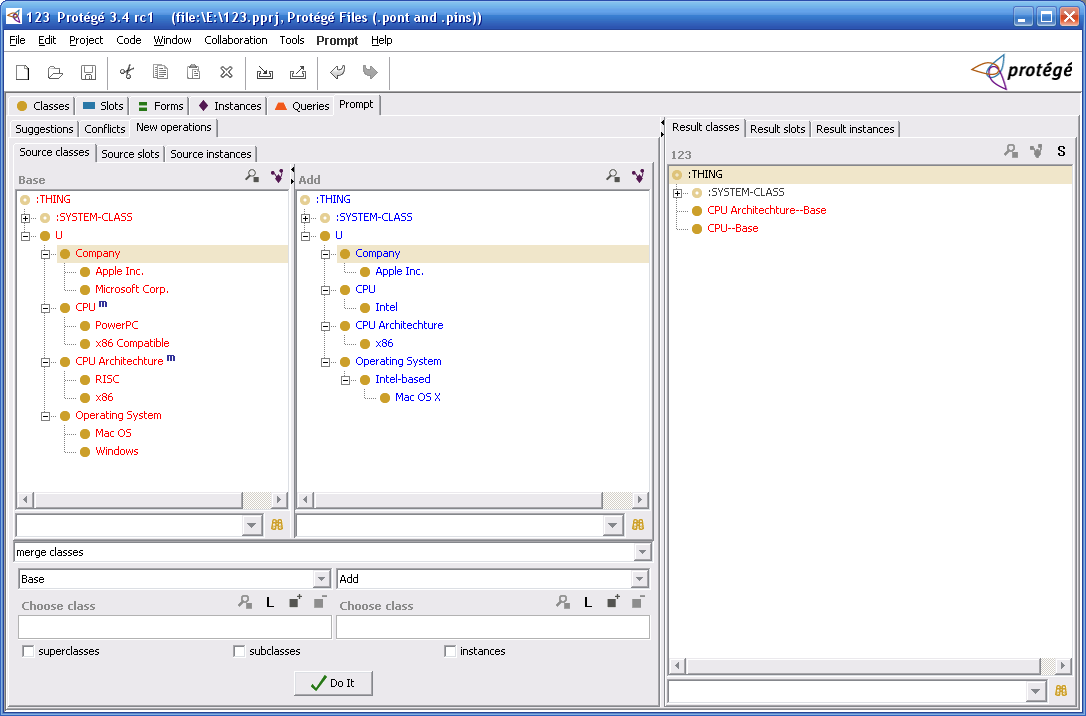


Рис. 7.1. Інтегруєма (base) та інтегрована (add) онтології в Protégé

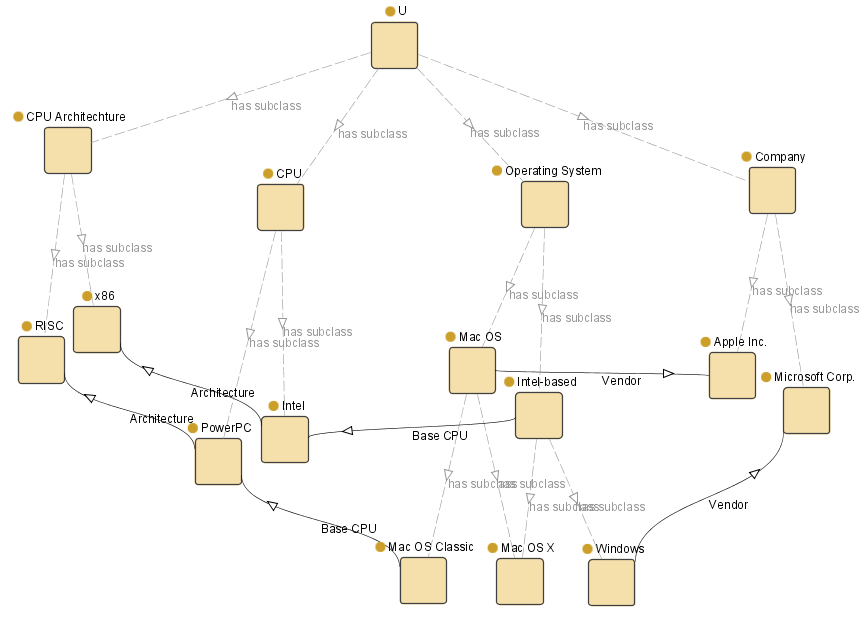
Рис. 7.2. Результат об‘єднання онтологій в Prompt



Рис. 7.3. Відповідність схем компонент проектуємої онтології в Protégé і (3.1)

**7.2. Інструментальний комплекс автоматизованої побудови  
 онтологій ПдО**

Інструментальний комплекс онтологічного призначення для автоматизованої побудови онтології в будь-якій предметній області є системою, що реалізує одне з напрямків комплексних технологій *Data & Text Mining*, а саме – аналіз і обробку великих об‘ємів неструктурованих даних, зокрема лінгвістичних корпусів текстів на українській і/або російській мові, добування з них предметних знань з послідуючим їх представленням у вигляді системно-онтологічної структури або онтології предметної області. ІКОП призначений для реалізації множини компонентів єдиної інформаційної технології:

* пошук в мережі Internet і/або в інших електронних колекціях (ЕлК) текстових документів (ТД), релевантних заданій ПдО, їх індексацію та збереження в базі даних;
* автоматична обробка природномовних текстів (Natural Language Processing);
* добування з множини ТД знань, релевантних заданій ПдО, їх системно-онтологічна структуризація та формально-логічне представлення на одній (або декількох) із загальноприйнятих мов опису онтологій (Knowledge Representation). Крім того, всередині цієї технології реалізується процедура побудови, візуалізації й перевірки семантичних структур синтаксичних одиниць ТД і понятійних структур заданої ПдО у вигляді несильно зв‘язаного онтографа, названого *початковою онтологією ПдО* (ПО ПдО);
* створення, накопичення та використання великих структур онтологічних знань у відповідних бібліотеках;
* системна інтеграція онтологічних знань як одна з основних компонент методології міждисциплінарних наукових досліджень;
* інші процедури, зв‘язані з автоматизацією придбання знань із множини природномовних об‘єктів.

ІКОП складається з трьох підсистем і представляє собою інтеграцію різного роду інформаційних ресурсів (ІР), програмно-апаратних засобів обробки та процедур природного інтелекту (ПІ), які, взаємодіючи між собою, реалізують сукупність алгоритмів автоматизованої, ітераційної побудови понятійних структур предметних знань, їх накопичення і/або системної інтеграції. Узагальнена блок-схема ІКОП представлена на рис. 7.4.

Підсистема ***Інформаційний ресурс*** включає блоки формування лінгвістичного корпусу текстів, баз даних мовних структур і бібліотек понятійних структур. Перший компонент представляє собою різні джерела текстової інформації, що поступає на обробку в систему. Другий компонент представляє собою різні бази даних обробки мовних структур, частина з яких формується (наповнюється даними) в процесі обробки ТД, а друга частина формується

Рис. 7.4. Узагальнена блок-схема ІКОП

до процесу побудови О ПдО і, по суті, є ЕлК різних словників. Третій компонент представляє собою сукупність бібліотек понятійних структур різного рівня представлення (від наборів термінів і понять до високоінтегрованої онтологічної структур міждисциплінарних знань) і являється результатом реалізації деякого проекту (проектування онтології ПдО і/або системної інтеграції онтологій).

Підсистема***Програмно-апаратні засоби*** включає блоки обробки мовних і понятійних структур і керуючу графічну оболонку (КГО). Остання, взаємодіючи з інженером по знанням, здійснює загальне керування процесом реалізації зв‘язаних інформаційних технологій.

Підсистема***Природний інтелект*** здійснює підготовку та реалізацію процедур попереднього етапу проектування, а на протязі всього процесу здійснює контроль і перевірку результатів виконання етапів проектування, приймає рішення про ступінь їх завершеності (і в разі необхідності – повторення деяких із них).

**7.2.1. Архітектурно-структурна організація та опис компонент ІКОП**

Архітектурно-структурна організація ІКОП представляє собою композицію трьох підсистем, зв‘язаних між собою керуючою графічною оболонкою й загальною інформаційною шиною (ЗІШ), і показана на рис. 7.5.

Підсистема***Інформаційний ресурс*** підрозділяється на зовнішній і внутрішній ІР. В свою чергу, зовнішній ІР підрозділяється на два види: перший із них створюється на основі першоджерел, передуючи процесу автоматизованої побудови онтології ПдО (ЕлК енциклопедичних словників, тлумачних словників і тезаурусів), другий – підключається в процесі створення О ПдО (мережа Internet, інші джерела ТД і розподілені бази знань).



Рис. 7.5. Архітектурно-структурна організація ІКОП

Внутрішній ІР також підрозділяється на два види. Перший з них створюється як результат реалізації алгоритмів автоматизованої побудови О ПдО (бібліотеки онтологій ПдО, онтологій ТД і термінів і понять домену ПдО), другий – створюється та використовується в процесі побудови О ПдО і представлений базою даних ТД.

***Електронні колекції енциклопедичних і тлумачних словників*** являються (за визначенням) загальнозначущими ресурсами знань в заданому домені прикладних областей. Вони містять, як правило, декілька визначень для кожного поняття з врахуванням смислових відтінків і орієнтованих на широке коло споживачів інформації. Таке представлення знань відповідає основним принципам побудови формальної онтології, а їх смислова інтерпретація є основним джерелом формування онтології ПдО для інженера по знанням.

Онтологія ПдО розробляється безвідносно до конкретної прикладної задачі і, тому, представляє собою деяку узагальнену (з точки зору ієрархії понять) і усереднену (з точки зору множин функцій інтерпретації понять) понятійну структуру предметних знань. З одного боку (перевага) онтологія ПдО є відкритою, загальнозначущою системно-онтологічною структурою, доступною до поповнення новими знаннями і/або адаптації до конкретного кола задач. З другої сторони (недолік) загальнозначущі структури не являються оптимальними для конкретних прикладних задач і потребують прикладення серйозних зусиль розробників, щоб стати такими. Наслідком вказаної вище переваги формальної онтології ПдО є її багатократне використання для різних наборів типових задач. Іншими словами, один раз побудована формальна онтологія ПдО придатна для вирішення будь-якого набору задач. Коли ж мова йде про пошук оптимальної онтологічної системи, необхідно адаптувати онтологічну базу знань до цільового застосування (виключити ланцюжки понять в онтографі, що не використовуються, конкретизувати множини функцій інтерпретації та обмежити значення ознак).

Друга складова зовнішнього ІР представляє собою сукупність джерел з природномовною та аналітичною (формально описані онтологічні бази знань) формами представлення інформації. Вони згруповані стосовно до задачі системної інтеграції технологій відповідно: автоматизованої побудови О ПдО та міждисциплінарних наукових знань. Причому, частина блоків являються загальними для обох технологій.

Таке джерело як ***мережа Internet*** не потребує додаткового пояснення, специфіка обробки добутої з мережі інформації проявляється в організації роботи пошукової системи та лінгвістичного процесора. В якості *інших джерел ТД* використовуються електронні бібліотеки, що вміщують файли монографій, науково-технічних статей та інших описів предметних знань.

***Розподілені бази знань***, технологія їх використання та обміну між зацікавленими учасниками, що входять в підсистему ПІ, представляють самостійну науково-технічну проблему. Концепція розвитку міждисциплінарних наукових досліджень, що торкається, в тому числі, і вказані вище питання, наведена в [99].

***Бібліотеки*** внутрішнього ІР є кінцевим результатом проекту (проектів), представляють собою глибоко структуроване сховище знань домену прикладних областей, яке шляхом застосування технології системної інтеграції перетворюється в ***загальний інтегрований простір трансдисциплінарних знань***.

Бібліотеки онтологій ПдО і ТД складаються з розділів, кожен з яких, в свою чергу, розбитий на дві частини – для збереження онтографа та формально-логічного опису. Бібліотека термінів і понять структурована по предметним областям. У відповідних записах понять проставлені індекси їх належності до відповідного рівня ієрархії (метаонтологія, онтологія домену або онтологія ПдО).

З накопиченням об‘ємів інформації в бібліотеках розширюються і їх функціональні можливості. Зокрема, системи різноманітних наукових знань, представлених системно-онтологічними структурами, можуть суттєво покращити якість наповнення електронних курсів (дистанційного) навчання, спроектованих на основі онтологічного підходу.

В подальшому, при досягненні в бібліотеках представницького об‘єму інформації, слід реалізувати “зворотній зв‘язок” для бібліотеки онтологій ТД і бібліотеки термінів і понять домену ПдО, коли результати роботи ІКОП стають джерелом вихідної інформації, що займає (по структурованості) проміжне місце між джерелами онтологій і джерелами ПМТ.

Другий вид внутрішнього ІР зв‘язаний з обробкою текстової інформації в лінгвістич­ному процесорі та пошуковій системі.

***База даних ТД*** підключена до лінгвістичного процесора (і пошукової системи) і зберігає оригінали текстових документів, проіндексованих у внутрішньому індексному форматі ІКОП. Вона також призначена для збереження множини ЛКТ різних ПдО домену.

Підсистема***Природний інтелект*** представлена процедурами підтримки роботи інженера по знанням і експерта в заданій ПдО. Інженер по знанням у взаємодії з керуючою графічною оболонкою (менеджером проекту) вибирає та запускає на виконання основні процедури інформаційної технології обробки текстової інформації. Експерт приймає участь в побудові категоріальних знань (початкової онтології) заданої ПдО і прийняття рішення про закінчення (або повернення до відповідного ітераційного циклу) процесу побудови онтографа ПдО.

Підсистема***Програмно-апаратні засоби*** представляє собою сукупність як оригінальних програмних модулів, що реалізують відповідні алгоритми етапів проектування О ПдО, так і відомих інструментальних засобів, що вільно розповсюджуються в мережі Internet, допускаючих розширення й вбудовування в системи користувачів, зокрема, інструментальна система Protégé [163]. Система Protégé використовується як один із основних компонентів модулів візуального проектування онтографа початкової онтології, перевірки онтографів, формування формалізованого опису проектованої онтології ПдО.

Окрім вказаного вище розподілу програмно-апаратних засобів на оригінальні та відомі, ця підсистема включає програмну оболонку *Менеджер проектів*, у склад якої входить ЗІШ, пошукова система, когнітивний лінгвістичний процесор [138] і сукупність програмних модулів, що реалізують процедури обробки мовних і предметних знань із заданого домену прикладних областей.

1. Такі системи в літературі зазвичай іменуються редакторами онтологій. [↑](#footnote-ref-1)
2. “OntoEditor+” розроблений групою дослідників Казанського університету (Росія) під керівництвом проф. Невзорової О.А. [↑](#footnote-ref-2)