

5. ІНФОРМАЦІЙНІ ТЕХНОЛОГІЇ ГАЛУЗІ

УДК 004.82

*Аспір. Я.Р. Василюк; доц. В.М. Теслюк, д-р техн. наук;
аспір. А.Я. Зелінський – НУ "Львівська політехніка"*

АНАЛІЗ МОДЕЛЕЙ, МЕТОДІВ, ПРИНЦИПІВ І ЗАСОБІВ ДЛЯ ПОБУДОВИ ОНТОЛОГІЙ ОБЛАСТІ МІКРОЕЛЕКТРОМЕХАНІЧНИХ СИСТЕМ

Розглянуто відомі моделі, методи та інструментальні засоби інженерії онтологій, за допомогою яких створюють та об'єднують різні онтології. Досліджено основні інструментальні засоби онтологій, проведено їх порівняльний аналіз, представлено можливості їх використання для побудови онтологій мікроелектромеханічних систем.

Ключові слова: онтологія, база даних (БД), інструментальні засоби інженерії онтологій, мікроелектромеханічні системи (MEMS).

Вступ. Інформатизація суспільства, накопичення великих обсягів інформації, впровадження обчислювальної техніки в усі сфери життя та виробництва, потреба у нових промислових виробках та системах зумовлюють необхідність опрацювання великих обсягів даних та виконання низки проектувальних робіт. Під час проектування (електронних) бібліотек різного функціонального призначення виникає серйозна проблема щодо запису та опрацювання великого обсягу інформації та доступу до неї. Для вирішення цієї проблеми було розроблено складні схеми класифікації та правила категоризації під час утворення та зберігання метаданих [21]. Такий підхід спростив процедуру пошуку потрібної інформації користувачем та підвищив ефективність його роботи загалом. Проте, залишається актуальною проблема представлення метаданих у адаптованому вигляді (зрозумілому) для машинного оброблення у процесі автоматизованого проектування [34]. Для досягнення зазначеної мети запропоновано використовувати спеціальні моделі онтологій, що дасть змогу підвищити рівень автоматизації програмних засобів для аналізу та синтезу мікроелектромеханічних систем [34].

Аналіз основних принципів побудови онтологій. За побудови моделі онтологій області автоматизованого проектування MEMS будемо ґрунтуватись на таких основних принципах [17]:

1. Зрозумілість (Clarity). Терміни (і поняття) онтологій повинні відображати реалії світу. Їхнє символічне представлення має створюватись на основі загальноприйнятих правил семіотики, і вони повинні виражати встановлені значення реальних об'єктів. Ці значення отримуються із словників предметної області. Судження формалізуються на основі формального апарату у вигляді істинних за значенням аксіом.
2. Розширюваність (Extensibility). На початковій фазі ядро будь-якої онтології формується введеними поняттями та аксіомами, які описують ці поняття. В онтології повинен бути передбачений механізм розширення (об-

меження) залучених словників понять без порушення цілісності системи загалом.

3. Обґрунтованість, зв'язність (Coherency). Формулювання початкового набору понять онтології і їхнє подальше додавання повинне бути обґрунтоване і визначене переліком завдань, які мають бути вирішені. Логічні аксіоми початкового набору понять не повинні суперечити між собою щодо встановлених ними обмежень.
4. Мінімальний вплив кодування (minimal encoding bias). В онтології має бути реалізований принцип сумісного використання декількох онтологій, який передбачає специфікацію онтологій на рівні представлення, а не на рівні символічного кодування. Таку специфікацію, здійснену на загальноприйнятому рівні та платформонезалежній мові під час опису онтології, може застосовувати будь-який програмний агент чи користувач.
5. Мінімальні онтологічні зобов'язання (minimal ontological commitment). За змістом цей принцип схожий на принципи обґрунтованості та розширюваності/обмеження. Важливим фактором для будь-якої онтології є представлення структури предметної області з допомогою множини понять онтології. Процес відокремлення концептуальних знань від фактичних формує стратегію побудови онтологічних систем, онтологічних баз даних.

Аналіз моделей та основних методів представлення онтологій.

Традиційно [17] формальну модель онтології можна відобразити таким чином $O = \langle T, R, F \rangle$ – це три впорядковані множини, де: T – терміни предметної області (ПрО), яку описує онтологія (O); R – відношення між термінами заданої предметної області (ПрО); F – функції інтерпретації, які задані на термінах і/чи відношеннях онтології.

Проте, Аманн та Фундулак [9] запропонували іншу математичну модель представлення, яка складається також із трьох членів $O = (C, S, isa)$, де: $C = \{c_1, c_2, \dots, c_m\}$ – це набір класів, у якому кожен клас c_i відповідає певним об'єктам реального світу (екземплярам класу); $S = \{s_1, s_2, \dots, s_n\}$ – це набір слотів, де кожен слот s_i може відноситись до: а) властивості класу за значенням простого типу; б) подвійного зв'язку (представлення відношення між класами); $isa = \{isa_1, isa_2, \dots, isa_p\}$ – це набір успадкованих відношень між класами. Успадковані відношення відображають семантику підмножин (класів) та визначають частковий порядок класів, що формує одну або декілька деревоподібних структур.

Для кращого розуміння та представлення окремих екземплярів (класів) [9] пропонують ввести ще один член-елемент $I = \{i_1, i_2, \dots, i_q\}$ до вищевказаної моделі, де i_w є екземпляром певного класу $c_x \in C$. Такий екземпляр включає конкретне значення для кожного слоту s_y , що асоціюється з класом c_x чи попереднім класом. Загальновідомі методи візуалізації здебільшого представляли онтології у деревоподібному чи графовому вигляді. Вони, наприклад, використовувались для візуалізації системи організації файлів чи категоризації документів тощо.

Потрібно розуміти, що методи представлення онтологій потребують в процесі їх практичного застосування певних модифікацій відповідно до специфіки предметної області. Тому припустимо, що такі методи мають пред-

ставляти складові онтологій, а саме: класи, зв'язки (відношення), екземпляри, властивості (чи слоти). Наприклад, пряму рівнозначність між об'єктами файлової системи, категоризованими документами. Відповідно до зазначених вимог та на основі різних характеристик відображення онтологій, а саме технік взаємодії, функціональності чи розмірної візуалізації, ми пропонуємо категоризувати методи представлення онтологій таким чином: дворівневий список; деревоподібне та вузлове представлення; масштабоване представлення; просторове представлення; фокусне представлення або деформація; 3D інформаційна архітектура.

Метод дворівневого списку. Більшість систем представлення онтологій (Protege, OntoEdit, Kaon, OntoRama) при описі онтологій формує деревоподібну структуру. Такі способи репрезентації онтології відображають їх, як набір вузлів, що утворюють вертикальну чи горизонтальну структури. Таке представлення дає змогу користувачу за потреби деталізації інформації розширювати чи видаляти певні вузли і сформовані ними дерева.

Методики **методу деревоподібного та вузлового представлення** полягають у представленні онтологій як набору взаємопов'язаних вузлів, що утворюють вертикальну чи горизонтальну структури. Таке представлення дає змогу користувачу за потреби деталізації інформації розширювати чи видаляти певні вузли і сформовані ними дерева. Методики цієї категорії поділяються на дво- та трирозмірні. До перших належать:

1. OntoViz, що є модулем візуалізації редактора Protégé, який використовує бібліотеку GraphViz для побудови звичайного 2D графа [4].
2. IsaViz, що є середовищем представлення RDF онтологій з допомогою орієнтованих графів [4].

До других належать:

- 1) Cone Tree [25] – це спосіб представлення онтологій, який ґрунтується на 3D графові, що утворює за принципом "знизу-вверх" 3D дерево.
- 2) Reconfigurable Disk Tree [27] є розширеною варіацією Cone Tree, в якій користувач має змогу змінювати висоту конуса кожного піддерева з метою покращення зрозумілості та читабельності всіх вузлів.
- 3) Tree Viewer [18] утворює деревоподібну структуру у формі реального дерева.
- 4) OntoSphere [20] пропонує використовувати вузловий зв'язок між трьома різними онтологіями для їх опису та деталізації відповідно до потреб користувача.

Категорія методу **масштабованого представлення** складається з методик, які дають змогу відобразити вузли початкових рівнів ієрархії, що входять до складу "батьківських" вузлів. До таких методик належать дво- та трирозмірні способи представлення. Розглянемо їх більш детально. До дворозмірного належать:

- 1) Grokker [7] – це репрезентаційна система карт знань. Вона графічно відображає інформацію, наприклад, як результати роботи пошукової системи чи системи пошуку файлів загалом.
- 2) Jambalaya [26] – візуалізаційний модуль Protégé, що використовує техніку представлення 2D SHriMP.

- 3) CropCircles [31] є методом представлення онтологій, що відображає ієрархію класів, як набір концентричних кіл.

До трирозмірного відносять:

- 1) Information Cube [36] використовує вкладені куби для відображення категорій об'єктів, що формують ієрархію.
- 2) Information Pyramids [1] відображають онтологію та її складові у вигляді пірамід, які мають пласку верхівку та розміщуються одна на одній.

Суть методу **просторового представлення** полягає в розподілі концептів, а саме вузлів та їх похідних вузлів у допустимих межах екрану. Розмір кожного розподілу залежить від властивості, яку відображає вузол, його розміру, числа вміщених похідних вузлів тощо. Цей метод поділяють на дво- та трирозмірні. До першої категорії належать:

- 1) Метод візуалізації TreeMaps [2] ґрунтується на способі 2D з використанням прямокутної області, що складається з прямокутних секцій, для представлення ієрархій класів-об'єктів.
- 2) SequoiaView [30] відображає ієрархію об'єктів онтологій, як деревоподібні структури TreeMap.
- 3) Техніка візуалізації Information Slices [3] використовує напівкруглі диски для більш компактного відображення великих ієрархій класів у 2D просторі.

До трирозмірних відносять BeamTrees [35], що об'єднує метод візуалізації Treemaplike та метод 3D візуалізації вузлового зв'язку.

Методики візуалізації **методу фокусного представлення або деформації** базуються на понятті деформації графу для об'єднання контексту та фокусу. Вузол фокусу є основним (батьківським) вузлом, від якого відходять вузли в послідовності зменшення їх розміру. Традиційно такий спосіб представлення описується гіперболічним рівнянням. Цей метод ґрунтується на дво- та трирозмірному представленні онтологій. До дворозмірних відносять:

- 1) 2D hyperbolic tree використовується для відображення онтології Бразильського товариства з питань сільськогосподарських досліджень. Гіперболічне дерево базується на гіперболічній трансформації. Основа дерева розміщена в середині області круга, від якої відходять похідні вузли.
- 2) OntoRama [12] є Java аплікацією, яка гіперболічно відображає структуру онтології.

До трирозмірних відносять метод візуалізації **3D Hyperbolic Tree**, якого було створено для відображення веб-сайтів та застосовувався як браузер файлів також. У середовищах віртуальної реальності досить поширеним для управління документами є поняття "інформаційної метафори", де в переліку документів їх розміщують відповідно до (зазначеного) кольору та розміру закодованих 3D об'єктів. До категорії методу інформаційної архітектури належать лише дві системи, які дещо відрізняються одна від одної: **File System Navigator (FSN)** [32], в якій висота вузла відображає кількість вміщених файлів, щодо онтологій – екземпляри; **Harmony Information Landscape** [13], створений для опрацювання гіпертекстів.

Виходячи зі специфіки автоматизованого проектування MEMS, пропонуємо використовувати такі методи представлення, як метод дворівневого списку та метод деревоподібного і вузлового представлення, що дасть змогу створити більш зрозумілішу і читабельну для користувача модель онтології.

Аналіз інструментальних засобів побудови онтологій. Процес побудови онтологій є дуже складним та часозатратним. Тому в 90-х роках почали розробляти перші середовища для процесу створення онтологій. Вони забезпечували інтерфейси, які дають змогу проводити концептуалізацію, реалізацію, перевірку розбіжностей та документування. Протягом останніх років кількість інструментів побудови онтологій значно зросла [22].

За побудови онтологій потрібно зважати і пам'ятати про певну визначену послідовність кроків, які пов'язані з: а) процесом розроблення онтологій; б) життєвого циклу онтологій; в) методів та методик побудови онтологій; г) набору інструментальних засобів та мов для побудови і підтримки онтологій. У наш час відомо багато інструментів для створення і підтримки онтологій, які, окрім загальних функцій редагування і перегляду онтологій, здійснюють підтримку документування онтологій, імпорт та експорт онтологій різних форматів і мов, підтримку графічного редагування, управління бібліотеками онтологій тощо. Тому пропонуємо розглянути та провести порівняльний аналіз найбільш поширених інструментів інженерії онтологій, які дають змогу підвищувати якість роботи внутрішніх систем, скорочувати терміни автоматизованого проектування складних об'єктів та систем, зменшувати матеріальні та людські витрати.

Система Ontolingua [14] була першим інструментом інженерії онтологій. Вона складається із сервера та мови представлення знань. Сервер Ontolingua організований у вигляді онтологій, які є програмними засобами WEB. Вони є певною побудовою над системою представлень знань Ontolingua. Окрім редактора онтологій, сервер Ontolingua включає мережеві програмні засоби Webster (отримання певних концептів), сервер OKBC (доступ до онтологій Ontolingua через протокол OKBC) та Chimaera (аналіз, об'єднання, інтегрування онтологій). Всі програмні засоби, за винятком OKBC, створені на основі форм HTML [14]. Система представлення знань реалізована на Lisp [14]. Сервер Ontolingua також представляє архів онтологій, вміщує велику кількість онтологій різних предметних областей, що уможливорює створення нових онтологій на основі вже створених. Сервер підтримує функцію створення онтологій кількома користувачами та розробниками. Система включає графічний браузер, який дає змогу переглянути ієрархію концептів та екземплярів. Збережені на сервері онтології можна перевести в різні формати, що дає змогу використовувати їх іншими програмними засобами та імпортувати з інших мов у мову Ontolingua.

Protégé [23] – це локальна, доступна Java-програма, яка призначена для побудови (створення, редагування та перегляду) онтологій прикладної області. Першочергове призначення цієї програми – це допомогти розробникам програмного забезпечення у створенні та підтримці наявних моделей предметної області, та включити такі моделі в програмний код. До складу

Protégé входить редактор онтологій, який дає змогу проектувати онтології, розширюючи ієрархічну структуру абстрактних чи конкретних класів, чи слотів. Структура онтології схожа на ієрархічну структуру каталогу. Protégé створений на основі фреймової моделі представлення знань OKBC (Open Knowledge Base Connectivity) і містить певну кількість плагінів [14], що дає змогу адаптувати його для редагування моделей різних форматів (стандартний текстовий, в базі даних JDBC, UML, мов XML, XOL, SHOE, RDF, RDFS, DAML+OIL, OWL).

OntoEdit [33] – це онтологічний редактор, який дає змогу здійснювати перевірку, перегляд, кодування та модифікацію онтологій. OntoEdit підтримує такі мови представлення знань: Flogic, включно з машиною виводу, OIL, розширену та внутрішню RDFS, яка ґрунтується на XML. Сerialізацію моделі онтології здійснює OXML – мова представлення знань OntoEdit (OntoEdit's XML-based Ontology representation Language). Для кодування концептуальних структур (концептів, ієрархій концептів, відношень, атрибутів) OntoEdit використовує GUI (Graphical User Interface). OntoEdit – це автономна Java-програма, яку можна локально встановити на будь-якому комп'ютері.

OilEd [6] – автономний графічний редактор онтологій, розроблений в межах європейського IST проекту On-To-Knowledge [24]. Цей інструмент розроблений за допомогою мови OIL (зараз адаптований до DAMP+OIL, у перспективі OWL), який поєднує фреймову структуру та дескриптивну логіку з сервісами розміркування. Це дало змогу створити зрозумілий та інтуїтивний стиль інтерфейсу для користувача та вивести логічні суперечності між класами та прихованими відношеннями підкласів.

WebOnto, програмний продукт, що розроблений для Tabzebau, [10] є інструментальним засобом інженерії онтологій, який призначений для підтримки функцій перегляду, створення та редагування онтологій. Цілі даного інструменту: простота використання, надання необхідних засобів для масштабування та створення великих онтологій. Для моделювання онтологій WebOnto використовують мову OCML (Operational Conceptual Modeling Language) [8]. У WebOnto користувач може створювати структури, включно з "наслідуючими" класами. Всі слоти зберігають закріплені за ними значення коректно при переході (процесі унаслідкування). Цей інструмент перевіряє всі вхідні дані за допомогою цілісності коду OCML. WebOnto наділений багатьма корисними особливостями: збереження структури діаграм, відокремлений перегляд відношень, класів, правил; спільну роботу декількох користувачів над онтологією, використання діаграм, функцій передачі і прийому тощо.

OntoSaurus [15] є Web-браузером для баз знань LOOM [19]. Він складається з двох модулів: сервера онтологій та Web-браузера для редагування та перегляду онтологій LOOM за допомогою HTML-форм, забезпечуючи для них графічний інтерфейс. OntoSaurus наділений обмеженими засобами редагування онтологій, проте його основна функція полягає в перегляді онтологій.

Конструктор онтологій ODE (Ontological Design Environment) [5] взаємодіє з користувачами на концептуальному рівні на відміну від таких інструментів як OntoSaurus, що співпрацюють на символічному рівні. ODE

надає змогу користуватися набором таблиць (концептів, атрибутів, відношень) та автоматично генерує для них код в LOOM, Ontolingua і Flogic [15]. ODE є частиною методики повного життєвого циклу побудови онтологій відповідно до Methontology [16]. ODE стало частиною WebODE [5], що інтегрує всі сервіси ODE в одну архітектуру, зберігає власні онтології в реляційній базі даних, забезпечує додаткові сервіси (машину виводу, побудова аксіом, збір онтологій, генерація каталогів).

KADS22 [28] – це інструмент підтримки розроблення моделей знань відповідно до методики CommonKADS. Онтології є частинами таких моделей знань (інша частина – це моделі виводу). Моделі CommonKADS створюються на основі мови CML (Conceptual Modeling Language). KADS22 – це інтерактивний графічний інтерфейс із такими функціональними можливостями, як: синтаксичний аналіз файлів CML, друк, перегляд гіпертексту, пошук, генерація словника та HTML.

Зважаючи на специфіку автоматизованого проектування мікроелектромеханічних систем та складних MEMC елементів, пропонуємо для створення онтологій мікроелектромеханічних систем – редактор Protégé.

Висновки. Проведено порівняльний аналіз принципів, моделей, методів та інструментальних засобів для побудови онтологій MEMC. З результатів аналізу випливає, що під час роботи з великою кількістю неопрацьованої та некатегоризованої інформації виникає необхідність використання онтологій. Разом із тим, запропоновано в процесі автоматизованого проектування онтологій MEMC використати основні принципи побудови онтологій, застосувати презентаційний метод подвійного списку або метод вузлового та деревоподібного представлення даних, що реалізовані на основі редактора Protégé. Як результат, процес побудови моделі даних та знань з допомогою запропонованих методів представлення та інструментальних засобів онтологій дає змогу побудувати моделі онтологій MEMC у коротші часові терміни з більш зрозумілим представленням об'єктів та зв'язків між ними порівняно з концептуальними моделями даних та знань, створеними з використанням традиційних способів.

Література

1. **Andrews K.** Information PyramidsTM: A new approach to visualizing large hierarchies / J. Wolte, M. Pichler // In Proceedings of the IEEE Visualization '97, Phoenix, Arizona. – 1997. – P. 52-49.
2. **Andrews K.** Visual exploration of large hierarchies with information pyramids / K. Andrews // In Proceedings of the Sixth International Conference on Information Visualization (IV'02). IEEE Computer Society Press, London, England. – 2002. – P. 798-793.
3. **Andrews K.** Information slices: Visualizing and exploring large hierarchies using cascading, semicircular discs / K. Andrews, H. Heidegger // In Proceedings of the IEEE Information Visualization Symposium, Carolina. – 1998. – P. 12-9.
4. **Amann B.** Integrating ontologies and thesauri to build RDF schemas / B. Amann, I. Fundulaki // In Proceedings of the Third European Conference on Research and Advanced Technology for Digital Libraries. – 1999. – P. 234-253.
5. **Arpirez J.C.** WebODE: a scalable workbench for ontological engineering / J.C. Arpirez, O. Corcho, M. Fernandez-Lopez, A. Gomez-Perez // First Intern. Conf. Knowledge Capture (KCAP2001). Victoria. Canada, Oct. – 2001.

6. **Bechhofer S.** OilEd: A Reason-able Ontology Editor for the Semantic Web / S. Bechhofer, I. Horrocks, C. Goble, R. Stevens // Joint German/Austrian conf. on Artificial Intelligence (KI'01). Lecture Notes in Artificial Intelligence LNAI 2174, Springer-Verlag, Berlin. – 2001. – P. 408-396.
7. **Bosca A.** OntoSphere: more than a 3D ontology visualization tool / A. Bosca, D. Bomino, and P. Pellegrino // In Proceedings of SWAP, the 2nd Italian Semantic Web Workshop, Trento, Italy, December 14-16, CEUR. Workshop Proceedings, ISSN 1613-0073. – 2005. [Electronic resource]. – Mode of access <http://www.ceur-ws.org/Vol-166/70.pdf>.
8. **Ceccaroni L.** Ontowedss – An Ontology-based Environment Decision Support System for Management of Wastewater Treatment Plants / L. Ceccaroni // Ph.D. Thesis. Politechnic University of Catalonia. – 2001. – P. 45-9.
9. **Ceravolo P.** Introduction to Ontology Engineering / P. Ceravolo, E. Damiani // Semantic Knowledge Management: An Ontology-Based Framework, 2009. – P. 73-47.
10. **Domingue J.** Tadzebao and WebOnto: Discussing, Browsing, and Editing Ontologies on the Web / J. Domingue // Proc. of the Eleventh Workshop on Knowledge Acquisition, Modeling and Management, KAW'98, Banff, Canada. – 1998.
11. **Chaudhri V.** OKBC: A Programmatic Foundation for Knowledge Base Interoperability / V. Chaudhri, A. Farquhar, R. Fikes, P. Karp, J. Rice // Fifteenth National Conf. on Artificial Intelligence. AAAIPres/The MIT Press, Madison. – 1998. – P. 607-600.
12. **Eklund P.W.** OntoRama: Browsing an RDF ontology using a hyperbolic like browser / P.W. Eklund, N. Roberts, S.P. Andgreen // In Proceedings of the First International Symposium on CyberWorlds (CW2002). Theory and Practices, IEEE press. – 2002. – P. 411-405.
13. **Eyl M.** The Harmony Information Landscape: Interactive, Three Dimensional Navigation Through an Information Space / M. Eyl // Master's thesis, Graz University of Technology, Austria. – 1995.
14. **Farquhar A.** The Ontolingua server: A tool for collaborative ontology construction / A. Farquhar, R. Fikes, J. Rice // International Journal of Human-Computer Studies, 46 (6). – 1997. – P. 728-707.
15. **Fernandez M.** A Building a Chemical Ontology Using Methontology and the Ontology Design Environment / M. Fernandez, A. Gomez-Perez, J. Pazos // IEEE Intelligent Systems, Jan./Feb. – 1999. – P. 46-37.
16. **Fernandez M.** Methontology: From Ontological Art Towards Ontological Engineering / A. Gomez-Perez, N. Juristo // AAAI-97 Spring Symposium on Ontological Engineering, Stanford University. – 1997. – P. 144-147.
17. **Gruber T.R.** A translation approach to portable ontology specifications / T.R. Gruber // Knowledge Acquisition. – 1993. – Vol. 5 (2). – P. 200-199.
18. **Jeong C.** Reconfigurable disc trees for visualizing large hierarchical information space / C. Jeong, A. Pang // In Proceedings of Information Visualization. – 1998. – P. 25-19.
19. **Kifer M.** Logical Foundations of Object-Oriented and Frame-Based Languages / G. Lausen, J. Wu // Journal of the ACM. – 1995. – Vol. 32. – P. 121-129.
20. **Kleiberg E.** Botanical visualization of huge hierarchies / Van de Wetering H., Van Wijk J. // In Proceedings of the IEEE Symposium on Information Visualization (InfoVis'2001). IEEE Computer Society Press. – 2001.
21. **Maedche A.** Tutorial on Ontologies: Representation, Engineering, Learning and Application / Staab S. // ISWC'2002.
22. **Motta E.** Reusable Components for Knowledge Modeling // Ph.D. Thesis. The Open University, 1997.
23. **Musen M.** Domain Ontologies in Software Engineering: Use of Protege with the EON Architecture // Methods of Inform. in Medicine. – 1998. – P. 550-540.
24. **Project "ON-TO-KNOWLEDGE"** Tools for content-driven knowledge management through evolving ontologies (ON-TO-KNOWLEDGE) // IST Project, 2000. [Electronic resource]. – Mode of access <http://www.cordis.europa.eu>.
25. **Protege project.** Stanford University. [Electronic resource]. – Mode of access <http://www.protege.stanford.edu>.
26. **Rivadeneira W.** A Study of Search Result Clustering Interfaces: Comparing Textual and Zoomable Interfaces / B.B. Bederson // University of Maryland HCIL Tech. Rep. HCIL. – 2003. – P. 36.
27. **Robertson G.G.** Cone Trees: Animated 3D visualizations of hierarchical information / G.G. Robertson, J.D. Mackinlay, S.K. Card // In Proceedings of the CHI '91 Human Factors in Computing Systems. ACM, New York, 1991. – P. 202-189.

28. **Schreiber G.** The CommonKADS Methodology / H. Akkermans, A. Anjewierden, R. Hog, N. Shadbolt, W. Van de Velde, B. Wielinga // Knowledge engineering and management. MIT press, Massachusetts. – 1999.

29. **SequoiaView**. [Electronic resource]. – Mode of access <http://www.win.tue.nl/sequoiaview/>.

30. **Shneiderman B.** Tree visualization with tree-maps. A2-d space filling approach. ACM Trans. Graph. – Vol. 11, No. 1, September. – 1992. – P. 99-92.

31. **Storey M.-A.** Jambalaya: Interactive visualization to enhance ontology authoring and knowledge acquisition in Protege / Mussen M., Silva J., Best C., Ernst N., Ferguson, R., Noy N. // In Proceedings of Workshop on Interactive Tools for Knowledge Capture, K-CAP-2001, Victoria, BC, Canada, – 2001. [Electronic resource]. – Mode of access <http://www.thechiselgroup.org/jambalaya>.

32. **Strasnick S.L.** Method and Apparatus for Displaying Data Within a Three-Dimensional Information Landscape / Tesler J.D. // US Patent 5,528,735, Silicon Graphics, Inc. – 1996.

33. **Sure Y.** OntoEdit: Collaborative ontology development for the Semantic Web / M. Erdmann, J. Angele, S. Staab, R. Studer, D. Wenke // In Proc. of the Inter. Semantic Web Conference (ISWC 2002), Sardinia, Italia. – 2002.

34. **Tae-Sul S.** Sharin CAD models based on the feature ontology of command history / Yon-sook Lee, Sang-Uk Cheon, Lalit Patil // International Journal of CAD/CAM, 1598-1800. – 2005. – Vol. 5. – No 1. – P. 47-39.

35. **Van Ham F.** Beamtrees: Compact visualization of large hierarchies / J.J. Vanwijk // In Proceedings of the IEEE Conference on Information Visualization. IEEE CS Press, 2002. – P. 100-93.

36. **Wang T.** Cropcircles: topology sensitive visualization of owl class hierarchies / B. Parsia // In Proceedings of the International Semantic Web Conference (ISWC 06). [Electronic resource]. – Mode of access <http://www.mindswap.org/papers/2006/cropcircles-iswc.pdf>.

Василюк Я.Р., Теслюк В.М., Зелинский А.Я. Анализ методов, моделей, принципов и средств для построения онтологий области микроэлектромеханических систем

Рассмотрены известные модели, методы и инструментальные средства инженерии онтологий, при помощи которых создаются и объединяются разные онтологии. Исследованы основные инструментальные средства онтологий, проведен их анализ, представлены возможности их использования для построения онтологий микроэлектромеханических систем.

Ключевые слова: онтология, база данных (БД), инструментальные средства инженерии онтологий, микроэлектромеханические системы.

Vasylyuk Ya.R., Teslyuk V.M., Zelinsky A.Ya. The analysis of models, methods, principles and tools for ontology design of microelectromechanic systems

Well-known models, methods, principles and tools on ontology engineering are observed, with help of which various ontologies are designed and unified. The main ontology instrumental tools are examined and analyzed. The possibilities of their usage for ontology construction of microelectromechanical systems are presented.

Keywords: ontology, data base (DB), instrumental tools of ontology engineering, microelectromechanical systems.

УДК 681.3

Проф. Ю.І. Грицюк, д-р техн. наук; доц. Т.Є. Рак, канд. техн. наук – Львівський ДУ БЖД

ПРОБЛЕМИ ЗАХИСТУ ІНФОРМАЦІЇ У СТРУКТУРНИХ ПІДРОЗДІЛАХ МНС УКРАЇНИ

Проаналізовано специфіку підготовки фахівців з інформаційної безпеки у Львівському ДУ БЖД, виявлено потенційні джерела загроз та рівні їх небезпек для структурних підрозділів МНС України. Встановлено, що багато дослідників у своїх роботах приділяють значну увагу аналізу змісту поняття "інформаційна безпека",

водночас такі поняття, як джерела загроз та рівень їх небезпек розглядаються ними дещо спрощено, здебільшого у відірваному від контексту поняття "інформаційна безпека", часто не пов'язаному із контекстом видового поняття "інформаційна загроза". Сучасні процеси функціонування систем інформаційної безпеки вказують на потребу удосконалення системи адміністративно-правового регулювання інформаційної безпеки, яка вимагає вироблення нових засобів, методів і способів забезпечення інформаційної безпеки різних органів управління, моніторинг інформаційного середовища, наявності джерел загроз і рівнів їх небезпек.

Вступ. За останні декілька десятиліть відбулися якісні зміни в процесах управління на всіх ієрархічних рівнях через інтенсивне впровадження сучасних інформаційних технологій. Їх швидкий розвиток призвів до зростання відносних цінностей суспільства взагалі й окремої людини зокрема – наявності конфіденційної інформації. Водночас стала зростати небезпека втручання в роботу інформаційних систем для несанкціонованого зчитування інформації. Значення та вагомість наслідків таких втручань з часом збільшилися настільки, що навіть розвинені держави, їх промислові та фінансові структури стали заручниками своїх інформаційних технологій. Саме тому в Україні все більше уваги приділяється проблемі не тільки захисту інформації, але й проблемі пошуку шляхів *управління інформаційною безпекою* [2].

Якщо раніше проблема захисту інформації була актуальною тільки для спеціальних служб, то згодом вона стала актуальною для всіх організацій та підприємств, так чи інакше пов'язаних з науковими чи виробничими здобутками, а також тих, які мали справу з комерційними чи банківськими таємницями. Окрім цього, навіть поверхневий аналіз надзвичайних ситуацій природного характеру, які часто появляються в Україні та за її межами, показує тенденцію зростання їх кількості та масштабів можливих наслідків. У багатьох випадках вони провокують складні техногенні аварії та глобальні катастрофи (Японія, 2011 р.). Їхнє прогнозування, завчасне попередження, інформування населення про наявні джерела загроз та рівень їх небезпеки – основне завдання структурних підрозділів МНС України. Водночас, швидка ліквідація надзвичайних ситуацій з найменшими витратами сил і засобів має немаловажне значення для фізичного і морального стану населення, а також економіки самої країни. У будь-якому випадку усі ці стратегічні та тактичні дії супроводжуються складними інформаційними процесами, більшість з яких мають конфіденційний характер.

Найбільш ефективне вирішення питань інформаційної безпеки у структурних підрозділах МНС України зводиться до постійної та систематичної роботи компетентних фахівців у кожному з підрозділів залежно від масштабів вирішуваних завдань. Хоча проблема підготовки фахівців з захисту інформації донедавна була актуальною для спеціальних служб силових відомств [1], проте на сьогодні, в силу специфіки виконуваних робіт і вирішуваних завдань, вона стосується і навчальних закладів МНС України, одним із яких є Львівський ДУ БЖД. Запроваджені тут методики навчання поряд із традиційними методами і засобами захисту інформації пропонують курсантам і студентам вивчати сучасні технології забезпечення безпеки інформаційних ресурсів і комунікаційних систем. Насамперед це пов'язано зі збере-