

Лариса Яненко
(Київ, НУХТ, Україна)

Мовні проблеми розвитку інтегральної науки

Сучасний стан людства на планеті Земля вимагає оперативної перебудови старих основ господарювання, які досі називають природокористуванням, тобто оренди у Природи певних ресурсів. Коли почали швидко закінчуватися більшість з таких ресурсів, то не тільки різко посилилася боротьба за них, але й Природа почала питати з людей оплати самої оренди. На планеті почалися всілякі природні катаклізми, що посилюються з кожним днем. Саме тому постала проблема термінового наукового рішення для відпрацювання оперативної перебудови основ біосферного існування людства та його капіталістичних принципів господарювання. Все більше людей почали згадувати ідеї нашого першого президента Академії наук України Володимира Івановича Вернадського щодо створення і розбудови людством Ноосфери Землі, яку він ще називав цивілізацією знань та праці [1,4].

Таким чином маємо чітку задачу інтеграції всіх знань людства у єдину інтегральну науку, що зможе оперативно і безпомилково почати формувати надійні бази даних та знань, які вже сьогодні потрібні для розбудови Ноосфери. Проте тут маємо багато проблем щодо створення інтегральної науки, для якої потрібна нова модель Природи, чітка система реальних Законів Природи та потужна кількісна мова для інтегрального опису, формування моделей, їх аналізу, розробки проектів рішень щодо наступних дій і прогнозування наслідків. Легко бачити, що ні звична для науки класична математика, ні відомі сьогодні комп'ютерні алгоритми, ні, тим більше, просто звичайна англійська мова не здатні забезпечити оперативне рішення цієї проблеми. Зрозуміло, що людство чекає від когнітивної лінгвістики оперативної розробки теоретичних основ такої мови та її практичної реалізації для потреб світової інтегральної науки [6].

Нагадаємо про одну з перших спроб у 90-і роки створення універсальної мови UML для моделювання всього (Unified Modeling Language) [5]. Непростий шлях до створення цієї мови візуального моделювання показує складність таких проблем. Ще у 70-і роки почалася формалізація ідей структурного аналізу (Structured Analysis) та проектування для систем реального часу (Real-Time Structured Design) у вигляді таких мов програмування, як Кобол Фортран та ряду інших. Але створені за їх допомогою системи автоматизованого проектування (CASE) мало відрізнялися від звичайних генераторів звітів, проте були досить громіздкими і коштовними. Далі світ перейшов на об'єктно-орієнтований підхід і першими загально визнаними мовами тут були Simula-67, Objective C, Eiffel, CLOSS, C++ та інші. Для забезпечення цього планетарного явища було написано багато монографій та підручників, починаючи з книги Греді Буча «Об'єктно-орієнтований аналіз та проектування» [7], протягом 90-х років свої книги надрукували ще Джеймс Рамбо, Уільям Лоренсен, Ребекка Вирфс-Брок, Фредерік Едді та інші. Також у формуванні методології об'єктно-орієнтованого підходу особливу роль відіграла у 1990 році монографія Айвара Якобсона «Objectory» (назву перекладають як «створювач об'єктів»). Далі почалися пошуки уніфікації методології, для чого Дерек Коулмен (Derek Coleman) запропонував метод Fusion (злиття), що включав в себе концепції з Object Modeling Technique (OMT) та CRC-карти. Нарешті, у 1995 році Джеймс Рамбо, Греді Буч та Айвар Якобсон запропонували світу Уніфіковану мову моделювання UML [5]. Ще у 1989 році був створений міжнародний консорціум Object Management Group (OMG), яка на сьогодні налічує сотні авторитетних компаній і займається стандартизацією мов програмування. Більше 20 авторитетних фахівців від відомих світових ІТ-компаній взяли участь у стандартизації першої офіційної версії UML. Потім з'явилися більш допрацьовані версії цієї мови – UML 2 [2]. Особливістю цієї мови є графічна візуалізація при програмуванні потрібних Вам алгоритмів у вигляді різноманітних стандартизованих діаграм:

- класів (class);
- варіантів використання (use case);
- композитної структури (composite structure);
- додаткові структури (element, package, object, slot, instance);
- послідовностей (sequence);
- діяльності (activity);
- взаємодії (communication);
- кінцевого автомату (state machine);
- компонентів (component);
- розгортання (deployment).

З таких діаграм формують стандартні шаблони для швидкого проектування і відправляють графічні (візуальні) результати до підсистем кодування на мови типу C++ або інші більш абстрактні мови [3]. Далі чудова компанія VisualParadigm створила сучасну CASE-систему, яка не тільки забезпечує пошук і виправлення помилок візуального програмування, але й дозволяє швидко отримати самостійно працюючий програмний модуль – власне потрібну комп'ютерну модель, базу даних або знань. В останні роки створені ряд спрощених автоматизованих систем для прийняття рішень типу Mindjet MindManager, XMind тощо, візуальна мова яких значно спрощує творчий пошук людини.

Таким чином, людство поступово відпрацьовує потрібні мовні елементи для майбутньої інтегральної науки, що сьогодні стає завданням номер один світової науки, яка, нажаль, розсипалася по окремим комірчинам і не забезпечує вчасну підтримку оптимальних рішень для подальшого процвітання нашої планети.

ЛІТЕРАТУРА

1. Какічев С.В., Ліщитович Л.І. Новий світовий лад – електронна цивілізація: [монографія]. Какічев С.В., Ліщитович Л.І. – К.: КВІЦ, 2017. – 200с.
2. Леоненков А.В. Самоучитель UML 2. – СПб.: БХВ-Петербург, 2007. – 576 с.

3. Лерман Крег Применение UML и шаблонов проектирования: уч.пос. – М.: Изд. Дом «Вильямс», 2001, – 496 с.
4. Ліщитович Л.І. Основи Геобіономіки (до статуту члена Природи), т.1. – К.: Фітосоціоцентр, 2012. – 184 с.
5. Рамбо Дж., Якобсон А., Буч Г. UML: специальный справочник. – СПб.: Питер, 2002. – 656 с.
6. Яненко Л.П. Про мову ноосферних комунікацій. Таврійські філологічні наукові читання: Матеріали міжнар. наук.-практ. конф., (м.Київ, 26-27 січня 2018) - К.: ТНУ ім. В.І. Вернадського, 2018. – С.193.
7. Booch Grady. Object-oriented Analysis and Design Applications, Redwood City, Calif., 1994.