

ВИННИК О.Ю.

(Львівський нац. ун-т ім. Івана Франка)

КАТЕГОРІЯ АВТОРИТЕТНОСТІ У СУЧАСНОМУ АНГЛОМОВНОМУ ДИСКУРСІ ПРОГРАМУВАННЯ

The paper focuses upon the ascertainment of the linguistic markers of the category of authority in modern English programming discourse. Results indicate that the author's intention to increase the authority of his narration is realized through the personalization of narration, the citation and interpretation of the utterances of well-known programming experts, the use of specific terminology and symbols, the use of visual aids, the use of expressive and emotional means, and the review of the readers' knowledge in the form of questions and exercises.

Key words: *modern English programming discourse, the category of authority, the author (subject) of discourse, the reader (addressee) of discourse.*

Наприкінці ХХ століття на перший план у мові виходить фактор людини як активного суб'єкта пізнання. Провідною у працях сучасних лінгвістів стає теза про те, що дискурсивна діяльність як різновид людської діяльності вимагає особливої уваги до особи мовця, автора, творця, суб'єкта дискурсу. З іншого боку, "будь-який дискурс існує лише заради когось і в певній ситуації" [Робен 1999, с. 184]. Тобто інтерпретація мовленнєвого повідомлення перебуває у прямій залежності не лише від особи його творця, але й від фактора адресата та ситуації спілкування. Чимало питань, пов'язаних із суб'єктом дискурсу (тексту), його адресатом, їх взаємодією у комунікації вже знайшли своє розв'язання у наукових розробках вітчизняних і зарубіжних науковців (див. напр., [Бацевич 2004; Карасик 2002; Карпчук 2005; Подолкова 2001; Радзівська 1999; Чернявская 2006]), що однак не знімає необхідності у подальшій розбудові теорії дискурсу в аспекті вивчення особливостей вербалізації людського досвіду, вираження особистості у дискурсі та організації міжособистісної мовленнєвої взаємодії.

У цьому дослідженні ми пропонуємо аналіз мовленнєвої діяльності автора англomовного дискурсу програмування, що зумовлює розгляд останнього крізь призму комунікативно-прагматичного підходу до мови. Матеріалом дослідження слугували 8 автентичних англomовних книг з програмування за 1991 – 2010 роки. Основне призначення цих книг – бути джерелом наукового знання, пов'язаного з розробкою комп'ютерного програмного забезпечення.

Як показав проаналізований матеріал, автор не просто ділиться з читачами своїми знаннями і досвідом у відповідній предметній галузі, але водночас намагається переконати їх у вірогідності й об'єктивності викладених у книзі знань, у своїй фаховій і загальній ерудиції. Тому однією з базових комунікативно-функціональних категорій англomовного дискурсу програмування ми вважаємо *авторитетність*, яка відображає прагнення мовленнєвого суб'єкта завоювати довіру та повагу інших учасників

комунікативного процесу. Як слушно зауважує О. І. Багрій, категорія авторитетності має значний вплив як на розвиток самої науки, так і на вербальний науковий дискурс. Однак дотепер нерозв'язаним залишається питання про те, за допомогою яких мовних засобів відбувається передача цієї категорії, зокрема, в англійській мові [Багрій 2007, с. 325]. Отож, **метою** цієї статті є виявлення та лінгвістичний аналіз мовних маркерів категорії авторитетності у сучасному англомовному дискурсі програмування.

Одразу слід зауважити, що закріпленню за автором інформаційної влади над читацькою аудиторією значною мірою сприяє той факт, що формування англомовного дискурсу програмування можна вважати більш-менш завершеним та усталеним, а, отже, автор книги з програмування обізнаний з характерними для цього жанру типовими дидактичними прийомами та методами представлення знань з розробки комп'ютерного програмного забезпечення. Передовсім йдеться про методи систематизації предметно-логічної інформації, а також логічної аргументації та побудови причинно-наслідкових зв'язків. Для англомовного дискурсу програмування характерним є поділ текстової інформації на частини (розділи, підрозділи тощо) та подання їх у певній логічній послідовності. У межах окремої книжної статті зв'язність реалізується шляхом використання повторів, особових і вказівних займенників, сполучників, сполучникових прислівників та їх еквівалентів тощо.

Іншим важливим прийомом підвищення авторитетності викладу є переважне використання дієслівних форм дійсного способу теперішнього часу, що дозволяє автору подавати інформацію як вірогідну та надійну, тобто як таку, що не потребує верифікації:

*C# is a strongly typed object-oriented language whose code visually **resembles** C++ (and Java). This decision by the C# language designers **allows** C++ developers to easily leverage their knowledge to quickly become productive in C#. C# syntax **differs** from C++ in some ways, but most of the differences between these languages **are** semantic and behavioral, stemming from differences in the runtime environments in which they **execute** [AC, p. 1].*

Такі та схожі речення корелюють з психічними якостями впевненості й емоційної стабільності та відіграють важливу роль у створенні умов інформаційного домінування автора над читачем – вони допомагають авторам керувати знаннями й установками свого читача безконфліктно.

Підвищенню ступеня довіри до викладеної у тексті інформації сприяє також той факт, що у багатьох випадках автор є не просто ученим-теоретиком, а практикуючим програмістом, про що читач може дізнатись зі статті “Про автора”. Крім того, основна частина книги часто містить посилання на попередній досвід роботи автора, випадки з практики, наприклад:

You might think that this technique is outdated, but as recently as mid-2003 I worked with a client that had hundreds of thousands of lines of code written in RPG that was subject to a 6-character-variable-name limitation. These issues still come up from time to time. [CC U4, p. 10].

Окремої уваги у наведеному вище фрагменті заслуговує пряма вказівка на автора через використання особового займенника “я” (I). Зауважимо, що прямі

засоби вираження авторської позиції доволі часто зустрічаються у текстах англомовного дискурсу програмування, що є зайвим свідченням на користь того факту, що “експліцитне авторське “я” (I) є усталеною етикетною рисою сучасного англо-американського наукового дискурсу” [Ільченко 2002, с. 9].

Для того, щоб додатково обґрунтувати вірогідність власних суджень, а також ввести їх у контекст існуючих наукових уявлень автори вдаються до цитування чи інтерпретації висловлювань загальноновизнаних експертів у галузі програмування:

Alan Kay, the principal designer of Smalltalk, once said “As complexity increases, architecture dominates material” [OOP, p. 3].

Computing pioneer Edsger Dijkstra pointed out that computing is the only profession in which a single mind is obliged to span the distance from a bit to a few hundred megabytes, a ratio of 1 to 10⁹, or nine orders of magnitude (Dijkstra 1989) [CC U5, p. 7].

Цитування авторитетних фахових джерел може застосовуватись автором також з метою протиставлення їм власної думки або критики:

Some writers have asserted that projects that use iterative techniques don't need to focus on prerequisites much at all, but that point of view is misinformed. [CC U3, p. 11]

Іноколи автори вдаються до цитування людей не пов'язаних з програмуванням, виявляючи при цьому свою загальну ерудицію, високий інтелектуальний і культурний рівень. Цікава особливість спостерігається у книгах віруючих учених: у вступному слові чи епіграфі вони наводять цитати з Біблії чи висловлюють подяку Богові, наприклад:

Thanks go to:

My savior Jesus Christ, without whom I wouldn't have had the strength to complete the task of writing this book [SQL].

Українська дослідниця О. І. Багрій так коментує це явище: для вчених різних віросповідань авторитет Бога є найвищим, а власні наукові досягнення розцінюються як скромна присвята Йому – найвищому творчому принципу в житті [Багрій 2007, с. 326].

Важливою ознакою авторитетності в англомовному дискурсі програмування є вживання спеціальної термінології та символічних знаків. Зловживання цими засобами звичайно інтерпретується як прагнення показати свою ученість, тоді як їх грамотне використання оптимізує процес науково-технічної комунікації, робить повідомлювану інформацію точною, об'єктивною, достовірною.

Терміни – невід'ємна складова лексичного складу будь-якого наукового тексту, де їм належить основна роль у передачі змістово-фактуальної інформації. Термінологічну базу АДП можна умовно поділити на три групи:

1. терміни на позначення базових комп'ютерних понять (*computer, hardware, operating system, CPU, memory, input/output devices, keyboard*);
2. терміни на позначення понять програмування (*program, programmer, software, source code, source file, object code, compiler, architecture*);
3. терміни інших наук (математики, кібернетики, електроніки тощо),

які входять до складу терміносистеми програмування без зміни своїх конкретнонаукових значень (*variable, array, matrix, floating-point number, digit, infinite series, AI*).

Найбільш типовими для англомовного дискурсу програмування моделями уведення в текст і пояснення значення нових термінологічних одиниць є такі:

- стисле пояснення термінів словникового типу, наприклад:

The #include mechanism is a text manipulation facility for gathering source program fragments together into a single unit (file) for compilation [CPL, p. 201].

- розгорнуте описове пояснення:

The word program is used in two ways: to describe individual instructions, or source code, created by the programmer, and to describe an entire piece of executable software. This distinction can cause enormous confusion, so we will try to distinguish between the source code on one hand, and the executable on the other ... [TYS, Day 1].

- пояснення значення терміна, яке супроводжується прикладом практичного застосування поняття чи функції, яку він позначає:

The domain of a column is the set of valid values that the column is intended to store. For example, consider a column that is intended to store an employee's date of birth... [SQL, p. 13].

- пояснення значення нових термінів через проведення аналогії з реаліями повсякденного життя:

We have all given instructions to someone in English and then had that person carry out the instructions. The programming process is analogous, except that machines have no tolerance for ambiguity and must have all steps specified in a precise language and in tedious detail [CD, p. 2].

- визначення терміна через метафору:

Polymorphism is the genie in OOP, taking instruction from a client and properly interpreting its wishes [CD, p. 424].

Авторська дефініція терміна – одна з відмітних рис науково-навчального тексту. При цьому автор звичайно враховує рівень фахової компетенції читача, його життєвий досвід, психологічні особливості сприйняття. Відповідно, для пояснення значення термінів в англомовному дискурсі програмування залучаються конкретні приклади, порівняння з реаліями повсякденного життя, образні порівняння, тобто робиться опора на особистий досвід читача (життєвий і професійний).

Окрім термінів типовою ознакою англомовного дискурсу програмування є включення в основний текст повідомлення одиниць деякої мови програмування – формальної знакової системи, призначеної для передачі команд і даних від людини до електронно-обчислювальної машини. На відміну від природної чи будь-якої іншої штучної мови тексти на мові програмування можна автоматично транслювати у машинний код, що його розуміє комп'ютер. Отже, англійська мова не є мовою програмування, хоча деякі її субмножини використовуються у мовах програмування 4-го покоління. Для того, щоб відрізнити одиниці англійської мови від елементів програмного коду,

застосовуються різноманітні шрифтові засоби. Наприклад, у наведеному нижче фрагменті

*Here's the meat of the code, the part that notifies each observer that there's been a change: the **notifyObservers** method [DPD, p. 73].*

елементи тексту програми інкорпоровано в основний текст і надруковано із застосуванням іншого типу шрифту, ніж основний текст англійською мовою.

Істотну роль в інтелектуальній пізнавальній діяльності адресатів англomовного дискурсу програмування відіграють різноманітні візуальні елементи: приклади програмного коду, таблиці, графіки, діаграми, фотографії тощо. Їх розмаїття та кількість визначаються змістом і характером тексту, рівнем складності навчального матеріалу, обраною методикою навчання. Використання цих засобів оптимізує навчальну діяльність читачів, організує їх увагу та мислення, а також сприяє формуванню позитивної психологічної реакції на написане.

Одночасний вплив на інтелект читача та його почуття здійснюється за допомогою експресивно-емоційно-оцінних компонентів, які не лише виявляють відношення автора до повідомлюваних фактів, але також є ефективним засобом переконання читача. Зокрема, йдеться про використання прикметників і прислівників у вищому та найвищому ступені, слів, що передають перебільшену оцінку тощо:

*This design is **cleaner and more easily maintainable** than the traditional one – and no less efficient [CPL, p. 320].*

*Structured programming remains an **enormously successful** approach for dealing with complex problems [ТУС, Day 1].*

Автор англomовного дискурсу програмування – не лише експерт у відповідній предметній галузі, ерудит, але й учитель, наставник, порадник, який передбачає ймовірні труднощі читача, пропонує ефективні рішення, застерігає від невірних кроків. Найбільш характерним проявом цього є використання автором імперативних висловлювань типу “порада” та “пропозиція”, звернених безпосередньо до читача, наприклад:

***Remember**, a Bag object is a form of collection [ООР, p. 112].*

Ефективний навчальний процес неможливий без перевірки рівня засвоєння матеріалу. Моніторинг якості навчання в англomовному дискурсі програмування здійснюється за допомогою питань і завдань у кінці розділу.

Отже, до основних вербальних і візуальних засобів вираження категорії авторитетності в сучасному англomовному дискурсі програмування належать особова манера викладу, посилення на авторитетні фахові джерела у галузі програмування, використання спеціальної галузевої термінології та символічних знаків, використання різноманітних візуальних елементів, використання експресивно-емоційно-оцінних компонентів та перевірка знань читача у вигляді питань і завдань у кінці розділу. Усі вони сприяють переконанню адресата у вірогідності й об'єктивності наданої йому інформації, у професіоналізмі автора, його фаховій і загальній ерудиції.

Перспективу подальших досліджень ми вбачаємо у дослідженні еволюції англо-американського наукового дискурсу, а також виявленні структурно-

композиційної та смислової специфіки реалізації категорії авторитетності в інших типах дискурсу на матеріалі різних мов.

Література

Багрий О.І. Категорія авторитетності в сучасному науковому дискурсі (на матеріалі англійської мови) // *Мова і культура*. – К.: Видав. Дім Дмитра Бураго, 2007. – Вип. 9. – Т II (90). – С. 324 – 330. *Бацевич Ф.С.* Основи комунікативної лінгвістики: Підручник. – К.: Видавничий центр “Академія”, 2004. – 344 с. *Льченко О.М.* Етикетизація англо-американського наукового дискурсу: Автореф. дис... д-ра філол. наук: 10.02.04 / Київ. нац. ун-т ім. Т.Шевченка. – К., 2002. – 37 с. *Карасик В.И.* Языковой круг: личность, концепты, дискурс. – М.: Гнозис, 2004. – 390 с. *Карпчук Н.П.* Адресованість в офіційному та неофіційному англомовному дискурсі (комунікативно-прагматичний аналіз): Автореф. дис... канд. філол. наук: 10.02.04 / Харк. нац. ун-т ім. В.Н.Каразіна. – Х., 2005. – 20 с. *Подолкова С.В.* Засоби реалізації комунікативності у науково-технічних текстах (на матеріалі текстів технічної реклами і анотацій науково-експериментальних статей): Автореф. дис... канд. філол. наук: 10.02.04 / Харк. нац. ун-т ім. В.Н.Каразіна. – Х., 2001. – 19 с. *Радзієвська Т.В.* Комунікативно-прагматичні аспекти текстотворення: Автореф. дис... д-ра філол. наук: 10.02.15 / НАН України. Ін-т укр. мови. – К., 1999. – 33 с. *Робен Р.* Анализ дискурса на стыке лингвистики и гуманитарных наук: вечное недоразумение. // *Квадратура смысла: французская школа анализа дискурса: Пер. с фр. и португ. / Общ. ред. и вступ. ст. П. Серио; предисл. Ю. С. Степанова*. – М.: ОАО ИГ “Прогресс”, 1999. – С. 184 – 196. *Чернявская В.Е.* Дискурс власти и власть дискурса: проблемы речевого воздействия: учеб. пособие / В.Е. Чернявская. – М.: Флинта: Наука, 2006. – 136 с.

Список джерел ілюстративного матеріалу

- AC Nash, T. *Accelerated C# 2010*. – Apress, 2010. – 627 p.
 CC McConnel, S. *Code Complete*. – 2nd ed. – Microsoft Press, 2004. – 868 p.
 CD Pohl, I. *C++ by Dissection*. – Addison-Wesley, 2002. – 501 p.
 CPL Stroustrup, B. *The C++ Programming Language*. – 3rd ed. – AT&T, 1997. – 912 p.
 DPD Holzner, S. *Design Patterns For Dummies*. – Wiley Publishing, Inc., 2006. – 308 p.
 OOP Voss, G. *Object-Oriented Programming: An Introduction*. – McGraw-Hill, Inc. – 1991. – 584 p.
 SQL Davidson, L., Kline, K., Windisch, K. *Pro SQL Server 2005 Database Design and Optimization*. – Apress, 2006. – 644 p.
 TYC Liberty J. *Teach Yourself C++ in 21 Days*. – 2nd ed., 1997 – Ел. ресурс: nmm-club.ru/forum/viewtopic.php?t=2319.