

УНІВЕРСИТЕТ імені АЛЬФРЕДА НОБЕЛЯ
КАФЕДРА АНГЛІЙСЬКОЇ ФІЛОЛОГІЇ ТА ПЕРЕКЛАДУ

Робота допущена до захисту
Зав. кафедри
Зінукова Наталія Вікторівна
д. пед. н., доцент

КВАЛІФІКАЦІЙНА РОБОТА

магістра

**Особливості вживання англomовної комп'ютерної термінології у
науково-технічному дискурсі та особливості її перекладу**

Здобувача Литвиненко
Поліни Вячеславівни
ОПП Переклад (англійська)
Спеціальність 035 Філологія
Керівник кваліфікаційної роботи

72 бали (Задовільно) D

Степанова Анна Аркадіївна
доктор філологічних наук, професор



Дніпро
2023

АНОТАЦІЯ

Литвиненко П.В. Особливості вживання англomовної комп'ютерної термінології у науково-технічному дискурсі та особливості її перекладу. Актуальність роботи визначається малодослідженістю теми та її новизною, а також розповсюдженістю комп'ютерних термінів та способами їх перекладу. Об'єктом дослідження даної роботи є англomовний науково-технічний дискурс. Робота складається зі вступу, трьох частин, а саме теоретичної, аналітичної, та практичної, висновку, списку використаної літератури, додатків та реферату. За обсягом робота складає 84 сторінки. Список використаних джерел становить 65 позиції.

Ключові слова: комп'ютерні терміни, науково-технічний дискурс, ІТ терміни, проблеми перекладу.

SUMMARY

Lytvynenko P.V. Peculiarities of using English computer terminology in scientific and technical discourse and peculiarities of its translation. The relevance of the work is determined by the lack of research on the topic and its novelty, as well as the prevalence of computer terms and their translation methods. The object of research of this work is the English-language scientific and technical discourse. The work consists of an introduction, three parts, namely theoretical, analytical, and practical, a conclusion, a list of used literature, appendices, and an abstract. The volume of the work is 84 pages. The list of used sources is 65 items.

Keywords: *computer terms, scientific and technical discourse, IT terms, translation problems.*

ЗМІСТ

ВСТУП	3
РОЗДІЛ I. ІНФОРМАЦІЙНІ ТЕХНОЛОГІЇ ТА ЇХНІ ГАЛУЗЕВІ	6
ТЕРМІНИ	
1.1. Термінологія інформаційних технологій	6
1.2. Поняття терміну, його основні ознаки	6
1.3. Аналіз класифікації термінів	12
1.4. Комп'ютерна термінологія у науково-технічному дискурсі	18
1.5. Проблеми перекладу комп'ютерної термінології	20
РОЗДІЛ II. ОСОБЛИВОСТІ ВЖИВАННЯ КОМП'ЮТЕРНОЇ	23
ТЕРМІНОЛОГІЇ В НАУКОВО-ТЕХНІЧНОМУ ДИСКУРСІ	
2.1. Структурно-семантичний аналіз термінів інформаційних технологій	23
2.1. Особливості вживання термінів у дискурсі англomовних інформаційних технологій	27
РОЗДІЛ III. ОСОБЛИВОСТІ ПЕРЕКЛАДУ ТЕРМІНІВ ДИСКУРСУ	29
ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ	
3.1. Способи перекладу термінів	31
3.2. Аналіз особливостей перекладу англomовних термінів українською мовою	41
ВИСНОВКИ	46
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ	49
ДОДАТКИ	56

ВСТУП

Динамічний розвиток галузі комп'ютерних технологій дуже змінив сьогоденне суспільство. Основна частина людей на планеті не може уявити буденне життя без ноутбуків, смартфонів та сучасних гаджетів. Така тенденція впливає й на вокабуляр кожної людини, який наповнився ІТ лексикою, бо знайомість, навіть простої термінології, стає життєвою необхідністю. Дивлячись на велику популярність інформаційних технологій, інтерес лінгвістів до дослідження цього процесу є очевидним. Проблема розглядається в дослідженнях багатьох вітчизняних лінгвістів, але не є повністю вивченою. Це можна пояснити швидким розвитком ІТ галузі, а отже, й незупинним оновленням термінології.

На сучасному етапі розвитку України сфера інформаційних технологій має особливе значення, бо на фоні загального занепаду економічного та політичного стану, вона здатна покращити стан за допомогою стрімкого розвитку, а також забезпечення функціонування суміжних галузей. Сучасні технології вже більше не асоціюються з чимось науково віддаленим, і суспільство сьогодні дедалі більше пов'язує своє життя з ІТ технологіями.

Глобалізаційні процеси створюють сприятливі умови для розвитку ІТ сектора України. Наука також досліджує питання впливу цього сектора на покращення економіки нашої країни. Сучасні інноваційні технології в центральних-східних районах Європи взагалі, та України зокрема, досить успішно інтегруються сьогодні в ринки Європейського союзу та здійснює потужний вплив на комплексний розвиток самостійного ІТ сектора, що може стати одним з найбільших напрямків сучасної державної економіки. У сьогоденному комп'ютерному дискурсі все частіше зустрічаємося з конкретизованою лексикою інформаційних технологій, а зокрема з термінологією ІТ галузі.

Актуальність роботи визначається малодослідженістю теми та її новизною, а також розповсюдженістю комп'ютерних термінів та способами їх перекладу.

Предмет дослідження: англomовні терміни у сучасній галузі інформаційних технологій та їх відтворення українською мовою.

Мета кваліфікаційної роботи: проаналізувати способи утворення та особливості вживання англomовних термінів у науково-технічному дискурсі шляхи їх відтворення в українських перекладах.

Об'єктом дослідження даної роботи є англomовний науково-технічний дискурс.

Предметом дослідження даної роботи є проблеми перекладу англomовних комп'ютерних термінів українською мовою.

Матеріалом дослідження є спеціальна комп'ютерна література та зібрані англomовні терміни галузі інформаційних технологій.

Практичне значення кваліфікаційної роботи полягає в тому, що матеріали дослідження можуть бути використані в перекладацькій діяльності при перекладах вузькоспеціалізованих текстів комп'ютерної тематики.

Структура та обсяг дипломної роботи: робота складається зі вступу, трьох частин, а саме теоретичної, аналітичної, та практичної, висновку, списку використаної літератури, додатків та реферату. За обсягом робота складає 84 сторінки. Список використаних джерел становить 65 позицій.

РОЗДІЛ І

ІНФОРМАЦІЙНІ ТЕХНОЛОГІЇ ТА ЇХ ГАЛУЗЕВІ ТЕРМІНИ

1.1 Термінологія інформаційних технологій

Бурхливий розвиток інформаційних технологій, який ми спостерігаємо з початку ХХІ століття, впливає не лише на розвиток економіки та змін у повсякденному житті суспільства, але й стає предметом досліджень низки наукових напрямків і мовознавства зокрема велику кількість робіт сучасних лінгвістів присвячено вивченню питання перекладу англійських термінів українською мовою, серед них І.Н. Волкова, А.С. Д'яков, Т.Р. Кияк, А.М. Науменко, Г.І. Беженарь, та інші. Але на сьогоднішній день комп'ютерні термінологічні одиниці є недостатньо вивченими.

Переклад текстів науково-технічного стилю неодноразово ставав предметом уваги науковців, вагомий внесок у розробку проблеми перекладу загалом та особливостей перекладу науково-технічної літератури зокрема було внесено наступними авторами: В. І. Карабан, Л. М. Черноватий, А. Я. Коваленко, А. Л. Пумпянський, Т. Р. Кияк, А. О. Ніколаєва та інші, втім питання відтворення лексичних особливостей україномовних науково-технічних статей у галузі комп'ютерних технологій англійською мовою залишається і досі майже недослідженим.

1.2 Поняття терміну, його основні ознаки

Термінологія є одним з важливих частин лексичної мовної системи будь-якої держави. Як вважає Лепеха Т.В. – це невід'ємна складова наукової мови, яка виражає досягнення певної ділянки знання на кожному етапі розвитку суспільства. Якість і багатство термінології певної галузі свідчить про рівень прогресу в цій галузі [1, с. 5].

Термінологія – є самостійним сектором мови, який пов'язаний з фаховою діяльністю, за думкою Томіленко Л.М. В кожній окремій галузі терміни поєднані в свої терміносистеми, що є поняттєвими елементами фахового знання, які відображені мовними засобами. Мова допомагає формувати

наукові наукові принципи і фіксувати досягнення, отже термінологія є важливою частиною науки і інших галузей [2, с. 6–9].

Слово термін походить з латинського *terminus* і означає межу або кордон воно окреслює поняття, яке може бути використаним в окремій галузі з найбільш точним смисловим визначенням [3]. Термін має бути емоційно нейтральним та давати точні визначення поняттям або явищам.

Термінологія – це особлива форма звичайної щоденної мови, яка пристосована до найточнішого опису потрібної даної галузі знань. Вона відрізняється від наддіалектичної мови насамперед професійною лексикою, часто містить багато інтернаціоналізмів, синтаксисом, а також частотою використання конкретних граматичних форм.

Термінологія в основному пов'язана із переліком спеціалізованих слів у даній галузі, але з наукової точки зору сфера термінології значно більша. Термінологія – це наука про: принципи впорядкування понять у конкретних галузях знання та людської діяльності, принципи визначення цих понять та принципи відповідності термінів та спрямовані на вдосконалення процесів спілкування людей, у фахових сферах [4, с. 14]. Тому предметом термінологічних досліджень є терміни з різних галузей знань, а це означає, що вони пов'язані з багатьма науковими напрямками та технічною діяльністю.

Фахова мова використовує специфічний набір слів (лексичні одиниці), використання яких підпадає під більш обмежувальні правила, ніж у випадку розмовної мови. Сама етимологія цього слова вказує, у чому його характерна особливість. Походить від латинського слова *terminus*, що означає межа, кінець. Це пояснює те, що термін має строгі межі значення і його слід вживати лише в конкретному контексті. Кожен термін має поняття, і саме правильне визначення поняття впливає на термін, або на мовну реалізацію концептуальної реальності. Ідеальний термін – той, що не потребує контексту, оскільки його семантичне поле побудовано таким чином, що його не можна

сплутати з іншою лексичною одиницею. На жаль, цей термінологічний ідеал не завжди досягається.

Термінологія, як і її розвиток, притаманна прогресу цивілізації. Всі соціальні, політичні, економічні зміни, а також науково-технічні відкриття породжують нові концепції, процеси, пристрої та функції, які потрібно якось називати, щоб був можливий обмін інформацією та спеціальними знаннями. Технічні винаходи, такі як відео, DVD або флеш-накопичувачі, наразі не є новими, але, коли вони вийшли на ринок, терміни, що описують їх, були неологізмами англійською мови. Термінологія має бути не лише «списком слів», але організованою концептуальною системою із систематичним набором правил, що полегшують процес розуміння, створення та перекладу термінів.

У сучасній літературі термінологія визначається як галузь науки, що займається вивченням термінологічних одиниць, так і сукупністю термінів у сфері конкретної галузі спеціальних знань. Говорячи про термінологію в першому розумінні, важливо підкреслити що вона займається побудовою, створенням, розвитком та нормативними аспектами щодо використання термінологічних одиниць, тобто наукою щодо принципів впорядкування понять в окремих галузях знань та діяльності людини, принципів визначення цих понять та принципів відповідності їм термінів, спрямованих на вдосконалення комунікаційних процесів між людьми, які говорять на теми, що належать до галузей, у яких вони є фахівцями.

Серед завдань термінологічної науки на перший план виступає вивчення виразів, що використовуються конкретною групою людей тієї чи іншої мови в якій терміни – описують їх функцій, пояснюють їх сутність тощо, а також розроблюють як загальну, так і детальну теорію термінів, з яких потім можна було б отримати науково обґрунтовані знання прикладного характеру, необхідні для раціональної реалізації намічених практичних цілей у межах створення або обробки наборів термінів. Ці завдання можна розділити на три основні питання: дослідження функціонування спеціалізованої лексики, вибір

оптимальних шляхів створення нових термінологічних систем, удосконалення існуючих систем.

Мовознавство традиційно виділяє дві течії термінологічних досліджень, першою займаються фахівці, які сприймають термінологію як окрему галузь національних мов, тісно пов'язану з професійною діяльністю, другу – дослідники якої стверджують, що термінологія є складовою лексики природної мови. У роботах багатьох вчених термін сприймається як постійний елемент загальноприйнятої лексики, позбавлений змінних ознак, що прагне називатися ідеальним терміном [5, с. 203].

Відмінності між терміном і лексичною одиницею, яка не є терміном, покладаються не на специфічні особливості термінологічних одиниць, а лише на сферу їх функціонування. Як доказ того, що термін є мовною одиницею, наводиться, що він має всі характеристики такої одиниці (синонімія, омонімія, полісемія тощо), яка суперечить принципу термінологічної однозначності. Термін, який одночасно має всі мовні характеристики лексеми в той самий час прямує до однозначності в межах цілої терміносистеми, також до відсутності синонімів, до самостійності але в реальному спілкуванні він зберігає в тій чи іншій мірі всі ці ознаки [6, с. 21]. Термін існує тому що існує термінологічна система, елементом якої він є. Існування терміна як самостійної лексичної одиниці є абсолютним, існування терміна як терміна є відносним.

Об'єктом дослідження термінології є термін, тобто звичайна мовна одиниця, яка, діючи в межах певної області знань, використовується для вербалізації понять у цій галузі, при цьому утворюючи досить важливу частину лексичного складу дискурсу, терміни є найбільш наповненою інформативною частиною. Вони несуть в собі головне смислове навантаження спеціального тексту. Будучи його ядром виконують одне з найголовніших завдань дискурсу – вони несуть спеціальну інформацію та покращують процес професійно-наукового спілкування [7, с. 33].

Термінологія завжди обмежена замкнутим лексичним контекстом, межі якого встановлені конкретною організацією дійсності, тому не дивно, що

поняття терміна нероздільно з 1970–х років пов'язано з терміном LSP, або Language for Special Purposes. Цей термін був створений у західноєвропейських країнах для опису мови науки, техніки та інших галузей знань та для відділення спеціалізованої мови (LSP) від загальної природної мови (LGP), при цьому підкреслюється, що LSP – це природна мовна підсистема, тісно пов'язана зі структурою мови загальної. Отже, це своєрідна мовна підструктура, яка відокремлюється від загальної мови та функціонує як мова в межах даного професійного середовища [8, с. 219].

Наведена схема наочно показує, що технолект функціонує як автосемантична семіотична система з покладеною на неї первинною когнітивною функцією. Якщо функції природної мови множинні та різноманітні, то функції технолекта чітко обмежені. Настанови щодо термінологічної політики, розроблені з дозволу ЮНЕСКО, свідчать, що в цьому контексті ми говоримо про "спеціалізовані мови" (LSP) різних предметних спільнот, участь яких – це знання певного предмету або іншого типу знань. Представники цієї громади, як правило, погоджуються з принципами мовної конвенції, яка не завжди сумісна з принципами загальної мовної конвенції (LGP). Оскільки сучасне суспільство сильно впливає на науково-технічний прогрес, спеціалізовані мови надають все більший вплив на розвиток відповідних загальних мов.

Уже згаданий термінологічний лексикон – це сукупність термінів, що відображають стан знань у певній галузі людської діяльності. Саме лексикони окремих термінологій складають основу галузевих лексикографічних досліджень. Однак термінологічні лексикони не утворюють єдиної структури, а складаються з окремих, тематично узгоджених термінологічних полів, до яких належать термінологічні мікросистеми (парадигми / термінологічні ряди), які як прямий компонент поля характеризуються прозорою структурою. Це сукупність термінологічних одиниць, отриманих в результаті застосування «термінологічного квадрата» до конкретної термінологічної одиниці [9, с. 57].

Необхідність співпраці техніків та лінгвістів, лексикографів та постійна потреба в стандартизації та уніфікації термінології у фаховій літературі обговорювались роками. Стосовно термінології, лінгвістика також зосереджується насамперед на питаннях правильності та проблем перекладу. Можна окремо виділити роботи в яких "мова техніки" відокремлена від національної мови, де автори зосереджуються на аналізі термінологічних одиниць.

Навіть короткий аналіз літератури з цього приводу показує, що конфронтаційне дослідження термінології створює прогалину і залишається поза зоною дослідницького інтересу лінгвістів. Цей факт зрозумілий, оскільки для таких досліджень повинні бути виконані такі умови: лінгвістичні знання, досвід роботи з галузевими текстами, знання обох мов, професійна практика перекладу на вивчених мовах, існування галузевого словника.

Кожна галузь має власну термінологією і не завжди є легким завданням поділити термінологію і загальну лексику. Серед мовознавців немає одногопогляду на питання співвідношення термінології і загальнолітературної мови. Як вважає Котелова Н. З термінологічна лексикає частиною загальнолітературної мови [10, с. 86]. В цьому випадку термін є такою ж лексичною одиницею, як і слово, що характеризується фонетичною, графічною формою і морфологічною структурою. На думку Л.М. Алексєєвої, всі терміни виходять з загальнолітературних слів [11, с. 19]. Як вважає М.В. Нікітін в семантичній структурі одного і того ж слова може поєднуватись загальноновживане і термінологічне значення [12, с. 37].

Основним ресурсом комп'ютерної лексики є англійська мова. Англійська ІТ термінологія дає можливість найбільш точно виділити сьогоднішню концептуально-лексичну ІТ картину як таку, що перетинається з глибинними механізмами результатів інтелектуальної діяльності суспільства, і це знаходить відбиття у створенні засобів вираження, в схемах та правилах її організації і в структурі самого ІТ дискурсу [13].

Протягом розвитку мовознавства вже було багато спроб системно проаналізувати сучасну термінологію ІТ дивлячись на походження термінів, як зазначає Ніколаєва А.О. швидкий розвиток сучасних технологій спонукає науковців до нових досліджень [14, с. 20].

1.3 Аналіз класифікації термінів

Існує багато класифікацій термінів. Їх можна поділити: за галуззю, в якій вони застосовуються, за структурними особливостями за способом побудови тощо. Наприклад, у своїх працях Суперанська А.В відокремлює терміни-слова (визначаються одним словом), терміни-словосполучення (розрізняють вільні та зв'язані словосполучення) та багатокomпонентні терміни (складаються з трьох і більше частин).

Як зазначає Панько Т.І, терміни-словосполучення складаються з 3 різних типів [15, с. 33]. До першого типу належать терміни-словосполучення в котрих компоненти відносяться до слів спеціального словника. Це окремі і самостійні слова, які ми можемо використовувати і за межами самого словосполучення. Вони можуть зберігати своє окреме значення. При об'єднанні у словосполучення вони утворюють нове значення, яке має свій власний зміст.

Терміни словосполучення другого типу можна поділити на 3 різних групи:

- 1) що складаються з одного технічного терміна і загально уживаного слова;
- 2) складаються з першого спеціального компонента – прикметника, специфічного має для тої чи іншої галузі науки;
- 3) утворені компонентами, де другий термін, який являється основним, у поєднанні з першим утворюють спеціальне для галузі значення. Терміни – словосполучення третього типу складаються з двох компонентів загальноновживаних слів, які поєднуючись утворюють терміни.

Терміни – це слова і їх сполучення, якими можна позначити конкретні поняття та об'єкти, що використовуються у певній галузі, науковій, технічній,

юридичній та інших. Термінами можуть бути слова, які використовуються наприклад в вузьких рамках однієї сфери і значення, що належать, як до галузі, так і до загальнонавчаних. Термін, як спеціалізована мовна одиниця, протиставляється природним мовним одиницям. Підкреслюються відмінності між термінами та загальною лексикою, а також той факт, що терміни не завжди створюються і функціонують відповідно до принципів літературних мов.

Окрім опису термінологічних одиниць, представники цього напрямку також зосереджуються на стандартизації, гармонізації та уніфікації термінологічних лексик. У свою чергу, представники описової тенденції припускають, що термін не є спеціальною лексичною одиницею, а лише одиницею, що використовується у спеціальній функції [16, с. 55]. Підтвердженням цієї тези, як зазначає Овчаренко Н., має стати той факт, що кожне слово загальнонаціональної мови можна термінологізувати, а кожен термін – детермінологізувати.

У термінології загального виразу виділяють три етапи:

- 1) претермін – початкова фіксація концепції в даній галузі;
- 2) квазітермін – термінологічна одиниця, завданням якої є пошук оптимального визначення об'єкта в межах конкретної термінологічної лексики з метою досягнення однозначності в межах цієї лексикону;
- 3) термін – умовна одиниця, що відповідає оптимальному визначенню поняття в межах певної галузі знань, однозначно в межах конкретної термінологічної лексики [17, с. 173].

Термін є передусім результатом мислення спеціалістів та інструментом професійного спілкування, метою якого є вербалізація концепції (абстрактної чи конкретної) у галузі конкретної галузі спеціальних знань. На думку філософа Флоренського, суть науки як такої полягає у формуванні відповідної термінології, оскільки "слово вільне і невизначене перетворити в відповідний термін – це і значить розв'язати проблему" [18, с. 229]. Султанов, однак, підкреслює, що "чим стисліше визначено поняття, тим краще з наукової сторони розкрита смислова структура досліджуваного предмету". Тому серед

основних термінологічних критеріїв зазвичай згадується розподіл в певній галузі професійних знань. Цей факт вже пов'язаний з етимологією слова термін, що походить з латинського граничний камінь, межа'. Однак не всі підрозділи відповідають цій вимозі однаково.

Для того, щоб поєднати мовну та спеціалізовану природу терміна, Лейчик В.М. пропонує використовувати поняття лінгвістичного субстрату. Згідно з нею термін є складне тришарове поняття, що включає:

1) природний мовленнєвий субстрат – матеріальний (звуковий або графічний),

2) логічний суперстрат – ознаки, що дозволяють терміну позначати спільне – абстрактне або конкретне поняття;

3) термінологічну сутність, тобто змістовні і функціональні ознаки, що дають терміну виконувати функції елемента теорії, що описує певну спеціальну сферу людських знань чи діяльності [6, с. 7].

Зосереджуючись на правильності терміна, Ядацька називає його правильним, якщо він відповідає таким критеріям:

- функціональність – термін повинен заповнювати відповідні прогалини в системі назв;
- логічність (концептуальність) – термін повинен пропонувати належні асоціації з визначенням, тобто його значення повинно відповідати змісту визначеного поняття;
- систематизованість – термін, як за формою, так і за значенням, повинен займати належне місце в спеціальній лексичній системі, а висловлене ним поняття повинно займати відповідне місце в системі понять у конкретній галузі;
- відтворюваність – побудова терміна повинна забезпечувати можливість побудови похідних назв;
- економність – дата повинна бути короткою та легкою для вимови;
- правильність – термін повинен відповідати мовним стандартам.

Діяльність, спрямована на вдосконалення окремих термінологічних лексиконів шляхом розробки єдиної та однозначної термінології, їх називають термінологічною роботою, яка регулюється міжнародними установами, такими як ISO (Міжнародна організація зі стандартизації), та національними установами, такими як РКН (Польський комітет зі стандартизації). Залежно від того, чи спрямована термінологічна робота на створення загальних термінологічних стандартів чи принципів, що діють у певній галузі професійних знань, розрізняють базову термінологічну роботу та спеціалізовану. Остання спрямована до конкретного адресата і полягає у визначенні меж професійних знань, конкретизації та впорядкуванні понять та назв, систематизації термінів, розробці принципів правильного визначення та конфронтації термінологічних систем різних мов.

Усі свої основні риси (однозначність, точність, відсутність емоційного маркування, систематичність тощо) спеціалізована лексика реалізує в межах конкретного термінологічного поля, за межами якого ці характеристики перестають застосовуватись [19, с. 132]. Однак навіть у точно визначеній області професійних знань спеціалізована лексика не є однорідною і однаково не відповідає зазначеним критеріям.

В межах спеціалізованої лексики, крім термінів, які є умовними одиницями, і їх основна мета полягає у вербалізації логічної моделі конкретної системи знань, тобто точно відобразити теорію та методологію певної галузі наукового та професійного життя – розрізняють також професійну лексику (так звані професіоналізми) або іншими словами фахову мову [19, с. 132–133]. Професіоналізми, сфера застосування яких зазвичай обмежується усними висловлюваннями представників окремих професій, є своєрідними синонімами термінів, які використовуються в неофіційному спілкуванні представників певної професії. Термін є особливим символом в терміносистемі, яка використовується для визначення спеціального поняття, тоді як професіоналізм – це завжди синонім, створений спеціально для заміни терміну. Основною особливістю фахової мови є «напівофіційний» статус,

позбавлений наукового відтінку та можливість стилістичного її забарвлення, якого не мають терміни.

На відміну від традиційних термінів, професійна лексика нерозривно пов'язана з національною лексикою. Традиційно розрізняють у звичайній спеціалізованій лексиці розрізняють поняття номена та терміна. Номени або номенклатурні вирази позначають реальні, матеріально експериментальні об'єкти, а також відношення: предмет – ідеальний образ предмета – вираз. Термін, згідно з цим підходом, є окреслений ідеальний об'єкт, тобто походить від інтелектуальної конструкції і позначає взаємозв'язок: об'єкт – поняття – вираз (наприклад, номінальна ефективність). Відповідно до цього підходу, термін (у вузькому значенні цього слова) означає теоретичне поняття, а номен – емпіричне. Сутність терміна відображена у визначенні, а номена, навпроти, має своє визначення, але на практиці його простіше намалювати, ніж описати. Враховуючи вищезазначені міркування, Люкшин Ю вважає, що об'єктом дослідження термінології, залежно від прийнятої методології, можуть бути [20, с. 15]:

а) терміни у вузькому розумінні (теоретичні поняття, наприклад, номінальна ефективність),

б) номени (назви номенклатури),

в) вирази (терміни та номени),

г) спеціалізована лексика (терміни, номени, фахова мова).

Терміни повинні, наскільки це можливо, дотримуватися 14 встановлених термінологічних правил:

- стабільність (не змінювати назв, які вже присвоїли),
- вітчизняність (терміни повинні базуватися на державній мові),
- міжнародність (етимологія повинна відповідати міжнародним назвам),
- однорідність (термін має бути створений або повністю з рідних джерел, або повністю з іноземних джерел),
- логічність (формулювання терміна повинно викликати якомога більше істотних асоціацій),

- систематичність (назвам декількох рівнозначних понять повинна відповідати назва вищого поняття),
- стислість (термін повинен містити лише одну інформацію для кожного явища),
- однозначність (термін повинен означати лише одне поняття),
- одноманітність (лише один термін повинен означати дане поняття),
- відтворюваність (джерела цього терміна повинні передбачати створення похідних назв), однаковість (якщо можливо, термін повинен містити джерела або структуру, спільну для споріднених термінів),
- оперативність (термін повинен бути максимально коротким і легким для вимови),
- правильність (термін повинен відповідати вимогам правильності мови),
- емоційність (термін не повинен викликати емоційних заперечень).

Деякі з цих принципів суперечать один одному, такі як принцип вітчизняності та принцип міжнародності. Однак слід прагнути вибирати такі термінологічні одиниці, які одночасно відповідатимуть якомога більшій кількості принципів термінології.

Поняття терміну включає як терміни (у вузькому розумінні цього слова), так і частини мови, вирішальним фактором тут є той факт, що обидва були створені в результаті усвідомлених дій конкретної професійної групи, а тому належать до звичайної лексики. Навпаки, професійна лексика належить до природної лексики і не є метою аналізу. Тому під терміном можна розуміти слово (словосполучення) із умовно або строго визначеною концептуальною структурою, по суті однозначною і не підлягає інтерпретації емоційного характеру, але має словотворчі здібності [20, с. 21]. Іншими словами, термін – це звичайний мовний знак зі строго визначеним значенням у межах конкретного термінологічного лексикону та без виразних конотацій. У літературі з цього приводу термін інтегрально пов'язаний з науково-технічним

прикметником (хоча тут зазвичай прийнято включати також терміни з галузі мистецтва та спорту), який прикметник функціонує як кваліфікуючий іменник.

Терміни в межах однієї термінологічної системи можна розділити на загальнонаукові, міжгалузеві та вузькоспеціалізовані. Перші два типи термінів функціонують у кількох термінологічних системах (особливо споріднених), тоді як останній тип відображає специфіку конкретної системи.

1.4 Комп'ютерна термінологія у науково-технічному дискурсі

Дискурс – одне з основних понять сучасної комунікативної лінгвістики та лінгвістики тексту. Через те, що розвиток інформаційних технологій в ХХІ ст. надав могутній поштовх до формування технологічної сфери, головною ознакою якої є доступність інформації у будь-який час у будь-якому місці земної кулі, на даному етапі існують підстави для виокремлення для лінгвістики поняття – науково-технологічного дискурсу.

Ньюмарк класифікує різні тексти на три категорії: виразний текст, інформативний текст, вокативний текст. Науково-технічні тексти належать до того, що ми називаємо типовим та відмінним інформативним типом текстів. Мова науково-технічних текстів характеризується лаконічністю, точністю, об'єктивністю, практичністю, стислістю та конкретністю. Тому вкрай важливо, щоб перекладачі цієї типології зберігали ці характеристики, коли відтворюють їх цільовою мовою [11, с. 129]. Крім того, формат науково-технічних текстів часто є стандартним, особливо в технічних текстах.

Загальний зміст функції науково-технічного стилю можна визначити як пояснення в широкому сенсі цього поняття, що зумовлює основні ознаки цього стилю - об'єктивність, логічність, доступність та узагальненість. основою мовного оформлення науково-технічних текстів є стандартизованість. До синтаксичних особливостей слід віднести синтаксичну повноту оформлення висловлювання, наявність аналітичних конструкцій, часте вживання клішованих структур, переважно іменний характер морфологічних компонентів речення, розгорнуту систему зв'язувальних елементів (сполучників, сполучних слів тощо). Складність синтаксичних конструкцій у

науково-технічних текстах пов'язана із багатоступеневим послідовним підпорядкуванням із великою кількістю засобів зв'язку між реченнями. Ці зв'язки мають лінійний характер, що відповідає послідовності наукового викладу, коли кожне наступне речення розвиває попереднє. визначальним серед релевантних для перекладу типологічних ознак тексту може бути вид домінуючої інформації. З точки зору і. Г. Гальперіна, "змістово-фактуальна інформація, як вербалізоване повідомлення про факти, певні події, процеси, явища, які відбуваються або будуть відбуватися, широко представлена у науково-технічних текстах". Фактуальність оптимізує інформаційний баланс, оскільки легше сприймається і усвідомлюється читачем та не потребує від нього застосування значних масивів допоміжної для декодування інформації. Мовні засоби оформлення фактуальної інформації у проаналізованих науково-технічних текстах знаходяться у строгих конвенційних рамках. наведемо основні з них.

- Лексичні (скорочення, загальномовні та спеціальні термінологічні), графічні засоби (дужки, двокрапки) та синтаксичні засоби (дієприкметникові звороти, що є компресивними синонімами до підрядних означальних речень). Будь-який текст цього типу відрізняється від інших потужним арсеналом допоміжних знакових систем (від умовних позначень x , y , z та формул до схем і креслень), що також є найбільш компресивним засобом вербальної актулізації інформації.

- Терміни, сукупність яких у науковому тексті є саморегульовальним лексичним апаратом, що спеціалізується на передачі фактуальної інформації. терміни однозначні, позбавлені емоційності та незалежні від контексту.

- Нейтральний лексичний фон решти лексики — лексики загальнонаукового опису. вона представляє писемну літературну норму мови та має розвинену синонімію, причому синоніми, як правило, стилістично

рівноправні (наприклад, важливий = суттєвий = значущий; вивчати = досліджувати = аналізувати тощо).

- Мовні засоби, що забезпечують об'єктивність подання фактуальної інформації: різноманітні засоби вираження пасивності у відношенні до формального підмета, неособова семантика підмета.

- Перевага теперішнього часу дієслова, використання якого дає можливість представити інформацію як абсолютно об'єктивну, таку, що знаходиться поза часом (атемпоральний характер тексту).

- Мовні засоби, що підкреслюють високий рівень абстрактності викладу: значна кількість складних слів, побудованих за словотвірними моделями з абстрактним значенням; чітка номінативність тексту (перевага іменників; вираження дії через віддієслівний іменник із десемантизованим дієсловом, наприклад, “здійснює вплив на об'єкт”).

Отже, можна зробити висновок, що метою перекладу з точки зору комунікативно-функціонального підходу, визначеною ініціатором для науково-технічного перекладу, є відтворення тексту перекладу, інформаційне наповнення якого повністю відповідає вимогам ініціатора перекладу, при цьому передбачається, що це наповнення співпадає в вихідному та перекладеному текстах.

1.5 Проблеми перекладу комп'ютерної термінології

Терміни відрізняються від слів загальної мови тим, що вони є насамперед інструментом пізнавальної, практичної роботи та засобом спеціалізованого спілкування, а загальномовні слова – це головним чином інструменти повсякденного спілкування.

Терміни відрізняються від загальномовних слів: спеціальність, тобто використання терміна конкретними користувачами в конкретних ситуаціях та з посиланням на конкретні об'єкти; умовність, що виникає внаслідок того, що термін не виникає природним шляхом, а є результатом цілеспрямованої діяльності конкретної професійної групи; системний характер, такий, що

кожен термін завжди є частиною певної термінологічної системи; точність та однозначність, що виникають внаслідок того, що кожен термін у межах певного домену чи сфери діяльності людини має конкретне визначення та системне значення; нейтральне емоційно-стилістичне маркування, що виникає внаслідок пізнавальної та практичної функцій терміну.

Проблема перекладів комп'ютерної термінології є надзвичайно важливою. Невідповідність в перекладі термінів або помилки можуть бути причиною непорозумінь між спеціалістами та, і взагалі, вплинути на кінцевий результат.

Сам переклад ІТ термінів вимагає відповідних знань у цій галузі, а також розуміння термінів англійської мови і знань термінології української мови. Під час перекладу технічних текстів велике значення має взаємодія терміну з контекстом, завдяки чому визначається значення слова.

Варто зазначити, що багато англійських термінів вже не потребують перекладу, тому що набули загального значення (наприклад диск, файл, програма). Існує, також, і багато запозичень: Windows, e-mail, host, а також «хибних друзів перекладача», наприклад data, image, rating.

В процесі роботи над технічним текстом, дуже часто зустрічаються терміни, значення яких, не відразу є зрозумілим. Тому, в цьому випадку, надзвичайно важливим є розуміння контексту речення або змісту тексту, і вже потім, термін набуває свого значення. Важливим, також є і необхідність підбору правильного методу перекладу. Переклад комп'ютерної термінології може здійснюватися різними способами та за допомогою таких методів як:

- транслітерація (printer - принтер, monitor - монітор, server – сервер, buffer – буфер)
- калькування або лексичне копіювання (access code - код доступу, current drive – поточний дисковод, disk storage – дискова пам'ять)
- описовий переклад (software – програмне забезпечення; protocol – метод передачі даних).

Отже, головним завданням перекладу комп'ютерної термінології є обрання того чи іншого прийому або способу перекладу, виходячи зі змісту тексту, щоб найточніше передати значення кожного терміна, оскільки від цього залежить успішність перекладу всього тексту в цілому.

РОЗДІЛ II

ОСОБЛИВОСТІ ВЖИВАННЯ КОМП'ЮТЕРНОЇ ТЕРМІНОЛОГІЇ В НАУКОВО-ТЕХНІЧНОМУ ДИСКУРСІ

Комп'ютерна термінологія завжди тісно пов'язана з технічним розвитком, а істотний розвиток технології змушує якомога швидше систематизувати найменування об'єктів. Спочатку в кожній мові виникає необхідність назвати явище, яке відноситься до певної технічної області, наступною необхідністю є визначення понять і величин, які використовуються для опису цих явищ, а потім слід вказати одиниці вимірювання, якими це можна виміряти [22, с. 28-30]. Тільки після завершення всіх вищезазначених заходів можна сформулювати детальні технічні вимоги (стандарти, положення). Це передбачає необхідність постійно створювати та нормалізувати галузеву термінологію кожною термінологічно забезпеченою мовою світу.

2.1 Структурно-семантичний аналіз термінів комп'ютерних технологій

Поняття у технічному тексті номінуються термінами, а увесь обсяг змісту традиційного тексту ІТ дискурсу передається у прямих значеннях мовних одиниць, то повторюваність термінів, призводить до того, що терміни логічно, семантично, формально утворюють мовленнєву структуру технічного тексту. Семантика наукового мовлення формується через семантику термінів.

Сучасний ІТ дискурс є насамперед дуже багатожанровим та вміщує функціональну різноманітність публічного монологу та діалогу речей. Для нього характерний цілий ряд конкретних комунікативних засобів. Різноманітність комп'ютерного дискурсу та застосування його користувачами вирізняється не лише у використанні професіоналізмів, а й у поєднанні лексичних одиниць, що відносяться до різних стилів та регістрів, створеними у відповідності з прагматичними установками та цілями спілкування в комп'ютерних мережах. За своїм походженням термінологія ІТ неоднорідна. Вона складається з питомих назв, чужомовних запозичень, номінацій

змішаного типу. В ході формування комп'ютерної термінології мали місце як процес термінологізації, так і процес детермінологізації.

Спеціального значення мовознавці надають вивченню структурно-семантичних особливостей термінології, бо значною кількістю лінгвістичного фонду є саме фахова лексика. За думкою Мікрюкова О.О. в мові, як в системі, є багато законів і понять, що утворюються завдяки новим засобам, щознаходяться у спеціалізованій лексиці та яким потрібна номінація [23, с. 231]. Використання багатозначних загальних слів у спеціальному напрямі є одним із широко вживаних способів номінації фахових термінів. Коли використовуються загальномовні слова з багатьох сфер науки, вони відносяться до тих самих законів та механізмів утворення різних змістів їх застосування [24, с. 20].

Структурно-семантичні особливості термінів, часто була в центрі уваги мовознавців: В. Н. Прохорової, А. В. Суперанської та інших [25, с.23-32]. На сьогоднішній день глибокого дослідження ІТ термінів не було в достатній кількості, бо галузь постійно поповнюється новими поняттями. Структура термінологічних одиниць – це дуже важливий чинник перекладу, тому що, щоб отримати відповідність перекладач має відтворити структуру слова у цільовій мові.

Більшу частину ІТ термінів англійської мови складають двокомпонентні складні слова. Двокомпонентні складні терміни представляють собою поєднання і взаємозв'язок двох основ різних частин мови і служать як засіб утворення різних частин мови, особливо іменників і прикметників [26, с. 83]. Умовою точного перекладу ІТ термінів є абсолютне розуміння значення перекладачем. Це можливо тільки у разі вивчення особливостей всіх типів морфологічного словотворення ІТ термінології. Виділяють такі типи морфологічного словотворення комп'ютерної термінології:

- афіксація (префіксація, суфіксація);
- словоскладання;
- конверсія;

- реверсія чи зворотній словотвір;
- контамінація;
- скорочення.

Для кожного типу морфологічного словотворення притаманні своєрідні особливості комп'ютерної термінології [27, с. 43]. Афіксація є способом словотворення, шляхом додавання до основи слова афікса (суфікса, префікса, інтерфікса тощо). Словоскладання – це поєднання декількох основ в одному слові (database, workstation). Незалежно від кількості слів, з яких складаються слова, утворені словотворенням, вони мають одне значення і є однією лексемою. За думкою Н. Клименко цей феномен є основокладанням декількох слів, яким характерне об'єднання двох чи більше частин без сполучних голосних [28, с. 3]. Компресія це спосіб словотворення, що включає синтаксичне зміщення слова чи словосполучення без зміни компонентного складу. Композиція – це спосіб словотворення, для якого відповідне об'єднання декількох основ слів чи основи та цілого слова, використовуючи інтерфікси або без них (software, keyboard). Конверсія – це спосіб словотворення при якому утворюється нова основа з вже існуючої, за допомогою переосмислення останньої не змінюючи її форму, особливо неафіксальний спосіб переходу слова в другу частину мови (download, backup) [29, с. 20]. Реверсія – це спосіб словотворення за допомогою відсікання афіксів (doubleclick – double – clicking). Контамінація є впливом одного слова на друге, що дає утворення нової форми, яка має ознаки обох слів (netizen, webliography).

Скорочення є одним з найбільш продуктивних способів словотворення. Скорочення — це спосіб творення слова, в результаті якого відсікається частина основи, яка збігається зі словом, чи є словосполученням, що поєднане суцільним змістом. Сучасні ІТ комунікації швидко інтегрували в різні сфери життя та діяльності суспільства, то звичайним є прагнення людей до мовного мінімалізму у зв'язку з інтенсифікацією життєвого темпу.

Слід зазначити, що серед інших термінологій англійська комп'ютерна термінологія посідає особливе місце в лінгвістичному відношенні. Її відмінною рисою можна вважати значну кількість запозиченої лексики, також мають місце і інші семантичні процеси – семантичний розвиток значень існуючих раніше слів (звуження і розширення значень). Як і в інших термінологіях, так і в комп'ютерній термінології значне місце займає загальнонародна лексика. Тобто вона перестала бути вузькоспеціальною областю знань і знаходиться в процесі активного взаємодії із загальноновживаною лексикою.

На сьогоднішній день немає однозначного погляду стосовно понять лексико-семантична та тематична групи [30, с. 131]. Ґрунтуючись на спільності цих понять, лінгвісти зазначають, що вони обидва виражають об'єктивну дійсність. Але науковці ділять ці поняття, звертаючи увагу на те, що лексико-семантична група є результатом закономірностей розвитку лексичної семантики мови, а тематична група базується на від рівні знань народу й способі класифікації явища дійсності. Як вважає Томіленко Л.М., основою для відокремлення тематичних груп є предметно-логічний зв'язок між поняттями [31, с. 36]. Межжеріна Г.В. розглядає лексико-семантичну групу як структурно-семантичну одиницю мови, для якої характерний семантичний зв'язок слів та їхніх значень [32]. Лексико-семантична група, як вважає Шматко І.В, є групуванням термінів у межах тематичних груп, спільних за лексичним значенням [33], а Микульчик Р. Б. у своїх дослідження, акцентує на умовності такого поділу [34, с. 92 – 95].

Соціальні комунікації допомагають визначити, до якої родової належності відноситься термін, а також угруповання, які виділяють одну тематичну групу серед інших. Як вважає Ментинська І. Б, до складу української ІТ термінології входять такі тематичні групи:

1. Назви галузей науки: інформатика, кібернетика, програмування, кібербезпека тощо.

2. Назви ІТ професій (на основі чинного професійного стандарту: розробник проектів у галузі ІТ, фахівець з інформаційних ресурсів , систем, із розробки програмного забезпечення, менеджер продуктів ІТ.
3. Одиниці вимірювання інформації: за національними стандартами, що відповідають системі СІ, ті що використовують двійкові множники та префікси, одиниці вимірювання простору та часу.
4. Найменування програмного забезпечення: системне ПЗ, прикладне ПЗ. 5. Символи мов програмування: C/C++, C#, Java, JavaScript, PHP, Python, Ruby, GNU bc, Euphoria, Limbo, Lua, Moу, Breeze тощо.
6. Назви типів комп'ютерів: десктоп, лептоп, неттоп, нетбук, моноблок, сервер тощо.
7. Назви комп'ютерних елементів та приладів: процесор, монітор, клавіатура, принтер, сканер, модем, відеокарта, материнська плата, жорсткий диск, оперативна пам'ять тощо.
8. Назви ІТ процесів: аналіз, тестування, автентифікація, перезавантаження, автозбереження, оптимізація, кодування тощо.
9. Найменування інтернет-комунікацій: тег, шаблон сайту, хостінг, SEO, просування сайту, сайт, соцмережа, Інтернет [35, с. 26 — 30].

2.2 Особливості вживання термінів у дискурсі англomовних інформаційних технологій

Як відомо, сьогодні швидкими темпами здійснюється комп'ютеризація всієї діяльності людини. Це само собою зачіпає мовну діяльність людини і ту частину лексики мови, яку називаємо термінологією. Вже цілком очевидно, що до цього величезного глобального процесу різні мови прийшли з різними можливостями фіксації нових ідей. Попереду всіх в цьому сенсі стала англійська мова, яка обслуговує весь сучасний електронний зв'язок. Інші мови або запозичують з англійської якісь вербальні компоненти, або, спираючись на англomовні моделі, розробляють свої вербальні засоби. Тут слід сказати, що "обраність" англійської мови мотивується не якимись винятковими рисами

мови, а тим, що він є фаховою мовою для працюючих в цій сфері. Англійська ІТ термінологія надає можливість окреслити сьогodнішню лексичну комп'ютерну картину як тісно пов'язану з глибинними механізмами розумової діяльності людини, і це має відображення у створенні засобів виразу, в формі та правилах її організації і в структурі самого ІТ дискурсу [36].

Одним з джерел запозичення термінів у всіх мовах є міжнародне співробітництво у різних сферах. Особливості розвитку лексичних систем багатьох мов на сучасному етапі пов'язані із стрімким розвитком науки та стрімким поширенням інформації завдяки розвитку технологій. Особливої уваги набула англійська мова, бо за рахунок англійських неологізмів проходить поповнення комп'ютерної лексики. Сьогodні є необхідним засвоєння мовцями англійської ІТ термінології.

Важливим викликом для ефективної реалізації ІТ проектів є повне взаєморозуміння між різними зацікавленими сторонами (власниками компаній, менеджерами, клієнтами) та ІТ спеціалістами, що реалізують це рішення. Досліджуючи масштаби цього явища та пробуючи виявити причини такої ситуації, можна розглянути тексти з точки зору правильності використання обраного терміну ІТ, який може бути неправильно зрозумілий зацікавленими сторонами та виробниками програмного забезпечення.

РОЗДІЛ ІІІ

ОСОБЛИВОСТІ ПЕРЕКЛАДУ ТЕРМІНІВ ДИСКУРСУ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ

Прогресуюча сфера інформаційних технологій диктує потребу адекватної трансляції фахових текстів, написаних на англійській мові. Це необхідно для перекладу інструкцій, літератури, інтерфейсу програмних продуктів. Складність перекладу текстів ІТ галузі полягає в тому, що велика кількість комп'ютерних термінів належить до безеквівалентної лексики, це означає, що вони не мають регулярних відповідностей в українській мові, але це не диктує неможливість їх перекладу. З часом ІТ терміни набувають перекладацької відповідності, і весь процес потребує вивчення. На сьогоднішній день, словники ІТ термінів відстають від розвитку самої галузі, або частково описують термінологію. Залишається актуальною необхідність створення стандартизованого алгоритму перекладу англійських ІТ термінів на українську мову. Такий алгоритм має назву «модель перекладу». Цей термін ввів Ю. Найда, говорячи про представлення процесу перекладу як про ряд перетворень одиниць оригіналу в одиниці тексту перекладу. Як вважав Комісаров В. Н, модель перекладу – це умовний опис послідовності розумових операцій, за допомогою яких перекладач здійснює переклад тексту або його частини [37, с. 234].

Як вважає Сарієва А. Б в перекладі комп'ютерних текстів має використовуватись ситуативна або денотативна модель. Вона описує процес перекладу як опис за допомогою використання мови перекладу тієї ж самої ситуації, що описана на мові оригіналу. Це означає, що ми передаємо українським еквівалентом той же предмет, який виступає англійським терміном. В цьому випадку найголовніше значення отримує предметна, а не комунікативна еквівалентність. Причина переваги цієї моделі обумовлена тим, що головна функція текстів ІТ дискурсу полягає в повідомлення відомостей, а не в художньо-естетичній дії [98, с. 97].

У процесі перекладу термінів ми часто стикаємося з різними труднощами, які можуть бути пов'язані з приналежністю терміна до термінології різних наук, з суміщенням в одному терміні декількох спеціальних значень або значень спеціальних і загальних тощо. Окрему проблему представляє ситуація, коли у терміні відсутня регулярна відповідність в мові перекладу.

Виконуючи переклад термінів дискурсу ІТ, перекладач має добре розумітися професійно в комп'ютерній галузі. Інформаційний світ є світом віртуальним, більшість термінів його терміносистеми мають асоціативний характер, тобто звичайне слово отримує зрозумілого комп'ютерного значення на інтуїтивному рівні [39, с.148].

Спеціальної уваги при перекладі ІТ термінів потребують так звані «фальшиві друзі» перекладача – лексичні одиниці, які подібні зовнішньо і навіть за внутрішньою формою, але викликають неправильні асоціації у зв'язку з наявністю в них зовсім іншого значення, можливістю семантичного варіювання мовної одиниці [40, с. 18].

Труднощі перекладу пов'язані не тільки з перекладом конкретних окремих слів, що зафіксовані в словнику, а з адекватною передачею змісту кожної фрази, в випадку якого не завжди відповідником є дослівний переклад. В цьому випадку знов необхідне добре розуміння галузі інформаційних технологій. Поняття змісту охоплює матеріально-логічну, ідейно-пізнавальну сферу висловлювання. Це і є складністю пошуку функціональних відповідників при перекладі. Перекласти означає повноцінно передати існуючими засобами однієї мови те, що зафіксовано засобами другої мови, не ламаючи зміст і форму.

Питання перекладу термінів було розглянуто в роботах Карабана В.І [41], Білозерської Л. П [42], Кікець І. [43], Сапальнової Н.М. [44] та інших українських і зарубіжних науковців. Велика кількість дослідників визнає, що термінологічність є характерною ознакою наукового дискурсу та його головним інформативним центром.

В ідеалі переклад термінів здійснюється за рахунок заміни слова вихідної мови словом мови перекладу. Говорячи про терміни, Рецкер Я.І. пише, що велика кількість термінів дійсно мають стійкі еквіваленти в інших мовах. Наявність еквівалентів у термінів значно полегшує процес перекладу, дозволяючи не тільки легко знайти відповідне значення слова, але і зрозуміти, про що йде мова в тексті.

Однак переклад терміну, здійснюваний за допомогою еквівалента, можливий далеко не завжди, і в таких ситуаціях доводиться вдаватися до інших способів перекладу. Як вказують лінгвісти Влахов С. і Флорін С., можна запозичувати термін з мови перекладу (вдавшись до транскрипції), або створити свій (за допомогою кальки, неологізму, складеного терміна), або надати статус терміна загальнономовній одиниці.

Як показує аналіз спеціальної літератури, при роботі з термінами перекладач може допустити і ряд інших помилок, які пов'язані з неправильним вибором відповідності, наприклад, вдатися до дослівного перекладу, коли термін або його компоненти аналогічні терміну мови перекладу, але мають інше значення, або ж прагнути знайти буквальну відповідність [45, с. 181]. Таким чином, правильне розуміння і переклад термінів залежать не тільки від високого рівня володіння мовою, а й від певних знань в тій науковій галузі, до якої вони належать, і вміння застосовувати різні способи перекладу.

3.1 Способи перекладу термінів

На сучасному етапі розвитку перекладознавства підвищена увага приділяється оптимізації перекладу термінологічних одиниць фахових мов [21, с. 126]. Переклад термінів потребує доброго знання галузі інформаційних технологій, розуміння змісту термінів англійською мовою і знання термінології українською мовою. В процесі перекладу термінології виділяють 2 етапи:

1. З'ясування значення терміну (у контексті тексту).
2. Переклад значення рідною мовою [41, с. 315 — 407].

Аналіз літератури дозволяє стверджувати, що термін, в більшості випадків є досить однозначним, він не має синонімів, а здатний лаконічно надавати значень предметам або явищам [46, с. 115]. При його перекладі, ми можемо не зосереджуватись на контексті, бо еквівалентом терміну буде тільки термін. При цьому не завжди перекладом терміну буде проста заміна англійського слова українським.

Важливим кроком під час перекладу англомовних термінів дискурсу ІТ, є підбір правильного методу перекладу, бо переклад галузевої термінології відбувається за допомогою кількох методів. Головне завдання перекладача є підбір вірного прийому, щоб як найточніше передати значення кожного терміну [39, с.151]. Розглянемо їх детальніше.

Калькування - один з прийомів перекладу, який полягає в тому, що слова і вирази однієї мови перекладаються іншою мовою шляхом точного відтворення засобами мови, на яку відбувається переклад їх морфемну або словесну структури [47, с. 326]. Цілком очевидно, що калькування як спосіб перекладу без еквівалентної одиниці може використовуватися лише тоді, коли слово має прозору словотворчу структуру і внутрішню форму. З тексту перекладу калька може входити в мову і з оказіонального слова переходити в розряд узуальних слів. Калькування – означає створення слів або висловів за зразком будови відповідного слова чи вислову іншої мови [48, с. 516].

Як вказує Л.С. Бархударов, калькування полягає в передачі безеквівалентної лексики вихідної мови за допомогою заміни її складових частин — морфем або слів (у випадку стійких словосполучень) їх прямими лексичними відповідниками в іншій мові. З думкою Бархударова Л.С. погоджується і Казакова Т.А., визначаючи калькування як відтворення комбинаторного складу слова або словосполучення, коли складові частини слова (морфеми) або фрази (лексеми) перекладаються відповідними елементами мови [49, с. 88]. Переклад-калька зазвичай застосовується при передачі значення складних слів і термінів словосполучень. Також

наголошується, що калькуванню можуть піддаватися і елементи цілих смислових груп.

Калькування також можна використовувати при перекладі англomовних аббревіатур [50, с. 225]. Через дослідження калькування, можна знайти ряд особливостей застосування зазначеного способу у галузі перекладу комп'ютерних термінів українською мовою. Спосіб калькування як один із способів запозичення іншої мовної лексики представляє собою утворення слів і виразів за допомогою копіювання, буквального (частіше допоміжного) перекладу структур іншомовних слів.

- *The HTTP Status Codes are used to indicate the status of the executed operation.* **Коди статусу** HTTP використовуються для вказівки статусу виконаної операції.

- *An **entity** consists of metainformation in the form of entity-header fields and content in the form of an entity-body.* **Сутність** складається з метайнформації у формі полів заголовка сутності та вмісту у формі тіла сутності

- *If the **port** is empty or not given, port 80 is assumed.* Якщо порт порожній або не вказаний, передбачається порт 80.

- *A request message from a **client** to a server includes, within the first line of that message, the method to be applied to the resource, the identifier of the resource, and the protocol version in use.* Повідомлення запиту від клієнта до сервера містить у першому рядку цього повідомлення метод, який буде застосовано до ресурсу, ідентифікатор ресурсу та версію протоколу, що використовується.

- *If the client sees an error status, it **SHOULD** immediately cease transmitting the **body**.* Якщо клієнт бачить статус помилки, йому СЛІД негайно припинити передачу **тіла** запита.

- *This method is often used for testing **hypertext** links for validity, accessibility, and recent modification.* Цей метод часто використовується для перевірки **гіпертекстових** посилань на валідність, доступність і останні зміни.

• *The response to a HEAD request MAY be cacheable in the sense that the information contained in the response MAY be used to update a previously cached entity from that resource.* Відповідь на запит HEAD МОЖЕ бути кешованою в тому сенсі, що інформація, що міститься у відповіді, МОЖЕ використовуватися для оновлення попередньо кешованої сутності з цього ресурсу.

• *This interim response is used to inform the client that the initial part of the request has been received and has not yet been rejected by the server.* Ця проміжна відповідь використовується для інформування клієнта про те, що початкова частина запиту була отримана і ще не була відхилена сервером.

Транскодування – це такий спосіб перекладу, де звукова та/або графічна форма слова однієї мови відтворюється засобами абетки іншої мови [51]. Транскодування термінів застосовується тоді, коли в мові перекладу немає коректного відповідного терміну, і не можливо знайти слово, яке б еквівалентно відтворило зміст терміну і відповідали більшості вимог утворення термінів [52, с. 1]. Існує 4 типи транскодування: транскрибування, транслітерація, транс кодування змішане і адаптивне [53, с. 129]. Так як під час транскодування слово має одне значення, то цей спосіб перекладу влучно застосовувати, якщо в мові перекладу є потреба чітко створити однозначний термін [42]. Нерідко використовується транскодування для перекладу термінів, якщо поняття у мові, з якої перекладається, утворюється з латинських або давньогрецьких терміноелементів [21, с. 126].

Перш ніж використовувати цей спосіб перекладу, перекладач має переконатися, що в мові перекладу немає перекладного відповідника, що перекладається, бо через транс кодування можуть виникати подібні дубляжі термінів в мові перекладу, а це заважає чіткості певної терміносистеми.

Транслітерація — це метод переписування (перетворення) тексту написаного знаками (літерами) одного фонетичного письма (голосного, приголосного, алфавітно-силабічного), використовуючи знаки (літери) іншого

фонетичного письма [43, с. 63]. Це літерна подібність оригінальної форми терміну. Вона заснована на принципі суворої відповідності букв: одна графема (букві чи букві з діакритикою, або константна група з двох і більше літер) данної системи письма завжди узгоджується з однією і тою самою графемою іншої системи. Транслітерація дозволяє відтворити оригінальні позначення (ретранслітерація), що зазвичай неможливо при фонетичній транскрипції. Вона використовується в наукових дослідженнях, паперових та електронних бібліотечних та архівних каталогах, в географічних атласах тощо.

Транскрибування або транскрипція — це система написання звуків однієї мови за допомогою графічних символів іншої. Зокрема, його можна використовувати для запису слів однієї мови у формі, яка дозволяє легко відтворити звук людині, яка цього не знає — тоді звуки однієї мови записуються символами іншої мови, але заданий звук не завжди повинен відповідати лише одному символу, а деякі символи можуть бути використані для позначення кількох різних звуків у вихідній мові [55, с. 144]. Інколи транскрипція може обслуговувати одержувачів за допомогою різних мов, найчастіше використовуючи символи міжнародного фонетичного алфавіту — це так звана фонетична транскрипція.

Транскрибування, на відміну від транслітерації, не завжди відображає заданий символ одного алфавіту з одним символом другого алфавіту, оскільки це спосіб передачі звуків, а не літер. З цієї причини, наприклад, символ «j» кириллицею може подаватись за буквами «дж». Завдання транскрипції — відтворити звук мови оригіналу, а транслітерація – оригінального написання, використовуючи символи іншої писемної системи, наприклад: provider – провайдер, fax – факс. Цьому способу перекладу відповідає така особливість у перекладі українською мовою, як відтворення в будь-якому випадку літери r, наприклад, driver – драйвер [54, с. 144].

У випадку змішаному транскодуванні використовуються принципи транскрибування і транслітерація. Для перекладу однієї частини слова може використовуватись відторення буквенної системи слова, а для іншої

застосовуватись передача звуків і навпаки, наприклад *interface* – інтерфейс, *context* – контекст [55, с. 110]. Під час змішаного транскодування, більша частина слова відтворює його звучання у вихідній мові, але поруч з тим передаються деякі елементи його графічної форми.

Ще одним видом транскодування у перекладознавстві є спосіб адаптивного транскодування, в якому форма слова вихідної мови адаптується до фонетичної та/або граматичної структури мови перекладу [56, с. 21]. При адаптивному транскодуванні літерами мови може передаватися вся форма, або велика її частина. Адаптивне транскодування є самим поширеним типом транскодування. Це пояснюється відмінністю у фонетичній та граматичній структурах мов: англійська – в більшості аналітична, українська – синтетична. У деяких випадках переклад терміна адаптується до фонетичної, а в других – до граматичної структури української мови [54, с. 144].

- *After your service provider configures your VLAN attachment for a Partner Interconnect connection, you must **activate** it before the attachment can start passing traffic* [57]. Після того, як ваш постачальник послуг налаштує ваше приєднання VLAN для підключення Partner Interconnect, ви повинні **активувати** його, перш ніж приєднання почне пропускати трафік.

- *Staged means that you have marked a modified file in its current version to go into your next **commit** snapshot* [58]. Індексований стан виникає тоді, коли ви позначаєте змінений файл у поточній версії, щоб ці зміни ввійшли до наступного знімку коміту.

- *Before making changes to your **resources**, you can generate a change set, which is a summary of your proposed changes* [58]. Перш ніж вносити зміни до ваших **ресурсів**, ви можете створити набір змін, який є підсумком запропонованих змін.

- *In Spring, the objects that form the backbone of your application and that are managed by the Spring IoC container are called **beans*** [57]. У Spring об'єкти, які

утворюють основу вашої програми та якими керує контейнер Spring IoC, називаються **бінами**.

- *A bean is an object that is instantiated, assembled, and managed by a Spring IoC **container*** [57]. Bean — це об'єкт, який створюється, збирається та керується **контейнером** Spring IoC.

- *The configuration **metadata** is represented in XML, Java annotations, or Java code* [59]. **Метадані** конфігурації представлені в XML, анотаціях Java або кодів Java.

- *Your application **classes** are combined with configuration metadata so that, after the `ApplicationContext` is created and initialized, you have a fully configured and executable system or application* [57]. **Класи** вашої програми поєднуються з метаданими конфігурації, щоб після створення та ініціалізації `ApplicationContext` у вас була повністю налаштована виконувана система або програма.

- *The `ApplicationContext` is the **interface** for an advanced factory capable of maintaining a registry of different beans and their dependencies* [57]. `ApplicationContext` — це інтерфейс для вдосконаленої фабрики, здатної підтримувати реєстр різних **бінів** та їх залежностей.

Описовий переклад або експлікація — це відтворення значення лексичної одиниці за допомогою розгорнутих словосполучень, що розкривають істотні ознаки позначуваного даного лексичною одиницею явища, за допомогою її дефініції, в випадку якщо жоден з словесних прийомів підбору відповідності не задовольняє ситуації, перекладачі вдаються до опису, наприклад `freeware` — безкоштовне програмне забезпечення, `shareware` — умовно безкоштовне програмне забезпечення. Описовий — це прийом перекладу нових лексичних елементів вихідної мови, де простий термін замінюється під час перекладу словосполучення, що адекватно передає зміст простого терміна [41, с. 36].

Описовий переклад, зазвичай, вживається паралельно з транскрипцією і застосовується при перекладі термінів, унікальних об'єктів тощо, якщо опис як перекладацький прийом зазвичай супроводжує слово, представлене в будь-якій простій формі, або навіть вживається в тексті замість самого слова, то перекладацький коментар, як правило, виноситься за межі тексту і потрапляє або в виноску на тій же сторінці, або приводиться в наприкінці тексту в якості примітки.

Використовуючи описовий переклад потрібно звертати увагу на те, щоб вираз в українській мові повноцінно і точно відтворював основні ознаки терміну в англійській мові. Щоб правильно застосовувати описовий переклад, потрібно перевірити, чи існує перекладний відповідник – еквівалент терміну, продивитися нову літературу цієї галузі, щоб не виникало термінологічних дублетів в українській мові.

Ефективність експлікації у випадку перекладу термінів українською мовою можна роз'яснити неподібністю у шляхах утворення окремих слів та словосполучень. Можна зазначити, що в англійській мові більшість складають безприйменникові багатоконпонентні словосполучення. Така ситуація не властива українській мові, тому утворюються особливі труднощі під час перекладу. Але ці складнощі можна усунути за рахунок калькування, у якому змінена черга компонентів словосполучення. Деінде експлікація може взаємодіяти з калькуванням.

The **default value** to use for substitution, which shall be sent if an alternate value is not supplied. Значення за замовчуванням для заміни, яке має надсилатися, якщо не надається альтернативне значення.

A **map** between a property name and its encoding information. Відповідність між назвою властивості та інформацією про її кодування.

The **runtime** type of a specific bean is non-trivial to determine. Визначити тип **середовища виконання** конкретного bean-компонента нетривіально.

Еквівалентність – це збереження відносної рівності змістовної, семантичної, стилістичної й функціонально-комунікативної інформації, що знаходиться в оригіналі й перекладі. Еквівалентність залежить від ситуації походження тексту оригіналу і його відображення в мові перекладу. Вона є одним з основних і складних питань теорії перекладу, бо саме рівень еквівалентності різномовних текстів дозволяє нам оцінювати успішність перекладу [60]. Еквівалентність означає необхідність більшої тотожності змісту оригіналу і перекладу, в межах, сумісних з нормативними вимогами, які забезпечують адекватність перекладу [61, с. 23].

Еквівалентним перекладом можна назвати переклад, який відповідає поняттю комунікативної рівності. Взагалі на базі такої рівності і може виконуватись переклад будь-якого виду. Еквівалентність – це шлях передачі основного змісту мовою оригіналу, інформативність, яку передає текст, зберігаючи різні функції мовних одиниць. Однозначно еквівалентність є більш широким поняттям відносно «точності перекладу», бо в другому випадку розглядається відповідність «предметно-логічного змісту» відносно оригіналу. Інакше кажучи, еквівалентність диктує правило максимальної орієнтованості в напрямку оригіналу. Одне з найважливіших завдань перекладача — це досягнення еквівалентності перекладу і як найближчій передачі змісту тексту оригіналу [62].

- *The identifier is the name of the **variable*** [63]. Ідентифікатор - це ім'я змінної.
- *A **one-dimensional array** is, essentially, a list of like-typed variables*[63]. Одновимірний масив, по суті, є списком однотипних змінних.
- *In Java, **multidimensional arrays** are actually arrays of arrays* [63]. У Java багатовимірні масиви насправді є масивами масивів.
- ***Arithmetic operators** are used in mathematical expressions in the same way that they are used in algebra* [63]. Арифметичні оператори використовуються в математичних виразах так само, як і в алгебрі.

- *The basic arithmetic operations — **addition, subtraction, multiplication, and division** — all behave as you would expect for all numeric types [63].* Основні арифметичні операції — додавання, віднімання, множення та ділення — поведуться так, як ви очікували, для всіх числових типів.
- *If a class includes an interface but does not fully implement the methods defined by that interface, then that class must be declared as **abstract** [63].* Якщо клас містить інтерфейс, але не повністю реалізує методи, визначені цим інтерфейсом, тоді цей клас має бути оголошений як абстрактний.
- *This small program includes an **expression** that intentionally causes a *divide-by-zero error* [63].* Ця невелика програма містить вираз, який навмисно викликає помилку ділення на нуль.
- *The **Runnable** interface must be implemented by any class that will initiate a separate **thread** of execution [63].* Інтерфейс Runnable має бути реалізований будь-яким класом, який буде ініціювати окремий потік виконання.
- *You modify files in your **working tree** [58].* Ви редагуєте файли у своїй робочій директорії.
- *You selectively stage just those changes you want to be part of your next commit, which adds only those changes to the **staging area** [58].* Вибірково надсилаєте до індексу лише ті зміни, що їх ви бажаєте зберегти в наступному коміті, і лише ці зміни буде збережено в індексі.
- ***Checkboxes** are used to let a user select one or more options of a limited number of choices [63].* Прапорці використовуються, щоб дозволити користувачеві вибрати один або кілька варіантів із обмеженої кількості варіантів.

3.2 Аналіз особливостей перекладу англомовних термінів українською мовою

При дослідженні вибірки англомовних термінів дискурсу ІТ технологій потрібно звертати увагу на її важливі особливості:

1. Процес аналізу ІТ термінів ускладнюється великою кількістю неологізмів і неоднорідністю термінології (професіоналізми, арготизми, сленгізми).
2. Термінологія ІТ галузі у своєму словотворі підпорядкована законам української мови. Зокрема, поширеними способами словотворення термінів і є афіксальний, безафіксний, а також лексико-семантичний.
3. Для ІТ дискурсу характерним є використання засобів мовної експресії.
4. Серед причин великої кількості англомовних запозичень є особливий розвиток ІТ технологій в англомовних країнах, а найбільше – у США.

При перекладі англомовних ІТ термінів використовують кілька способів перекладу. Ми визначили 4 головних: транскодування, калькування, описовий переклад, еквівалентний переклад.

Транскодування застосовується в перекладі тоді, коли в мові перекладу відсутнє відповідне поняття і перекладний еквівалент, і не можна якісно підібрати слово в мові перекладу, які б передавали зміст поняття і відповідали нормам термінотворення [53, с. 130].

Транскрипція та транслітерація не часто використовуються у чистому вигляді. Може застосовуватись транскрипція зі частинами транслітерації. За допомогою транслітерації / транскрипції перекладаються, наприклад, такі англійські терміни: Alpha – Альфа, byte – байт, monitor – монітор, click – клік, тощо. Серед прикладів чистої транскрипції можна навести терміни code – код, file – файл, interface – інтерфейс. Приклади транслітерації зустрічаються рідше: script – скрипт, server – сервер, тощо. Для адекватного перекладу термінології, що передається одним словом, потрібно враховувати те, що подвоєння приголосних поміж голосними літерами не передається, ось наприклад account – акаунт. Серед особливостей можна зазначити, що літера r

у кінці слова передається як звук [r], незалежачи від того, чи вимовляється він у англійському вихідному слові, наприклад: server – сервер. Літера [x] в українській мові передається за допомогою літер к і с, наприклад, syntax – синтаксис. Транскодована термінологія, яка міцно оселилася в мові перекладу, в словниках передається без додаткового описового перекладу: tag – тег, transaction – транзакція.

Перевагою транслітерації та транскрипції серед способів перекладу термінів є їх надійність. Транскодування у цьому випадку передає лише звукову оболонку слова, при перекладі малознайомих, не добре вивчене поняття. А зміст такого терміну розуміється через контекст, і є можливість уникнути інтерпретації його спеціальних характеристик на мові перекладу. Така модель перекладу на теперішній момент дуже розповсюджена під час перекладу англійських ІТ термінів. В такому випадку англійське поняття набуває українську морфологію, парадигму відмінювань, зберігаючи з цим вихідний корінь при написанні кирилицею. Транслітерація та транскрипція – це дуже продуктивні способи перекладу, що збагачують як стандартну лексику, так і сленгову, наприклад duser – юзер, shareware – шаровари (ресурс спільного використання) [64, с. 375]. Коли транслітерація і транскрипція застосовуються, при перекладі окремих слів термінів, переклад словосполучень проходить частіше за допомогою калькування та експлікації.

Калькування як прийом перекладу застосовується не тільки при перекладі складних слів, а і при перекладі одного з компонентів складеного терміна. Не рідко калькування використовується в перекладі складних термінів, якібули створені при використанні розповсюджених загальнонародних слів. У деяких випадках застосування способу калькування супроводжується зміною калькованих елементів. Часто під час перекладу транскрипція і калькування застосовуються одночасно, наприклад, composite key – композитний ключ, file system – файлова система. В цих випадках українські відповідники англійських ІТ термінів створені при використанні вибору першого словникового відповідника кожного з термінів. Калькування є

виправданим під час перекладу термінівсловосполучень де складові частини цих словосполучень уже отримали певне місце у терміносистемі мови перекладу і є доступним для розуміння фахівців [65, с.54-59]. Додатковою перевагою способу калькування є стислість отриманого терміну і співвіднесеність з вихідним словом. Калькування може використовуватись тоді, коли перекладений термін не порушує норм вживання і сполучуваності слів.

У тому випадку, якщо словосполучення утворюється з термінів, що не входять у використання в деякій ІТ галузі в мові перекладу і вимагають свого пояснення, використовують описовий переклад [43, с. 178].

Це лексико-граматичне перетворення, в якому слово мови-оригіналу заміщується словосполученням, що дає пояснення цієї одиниці. Експлікація є дуже продуктивним шляхом перекладу ІТ термінів, бо бурхливий розвиток цієї сфери багатьом термінам допомагає знайти потрібні еквіваленти і закріпитися в мові перекладу [41, с. 180]. Використовуючи експлікацію перекладають багатокомпонентні словосполучення.

Продуктивність експлікації при перекладі англomовних термінів українською мовою пояснюють розбіжностями у шляхах утворення слів та словосполучень.

В англomовній термінології переважають багатокомпонентні безприйменникові словосполучення, які не притаманні українській мові і це утворює деякі труднощі під час перекладу [41, с. 181]. Такі труднощі часто усувають використовуючи калькування зі зміною компонентів словосполучення: BIOS (BasicInput/OutputSystem) – базова система введення-виведення, DPB (DiskParameterBlock) – блок параметрів диска. Серед переваг калькування перед експлікацією є багатослівність експлікації, а при калькуванні кожній частині словосполучення у мові оригіналу відповідає один елемент словосполучення у мові перекладу, тобто відтворюється умова економії мовних засобів. Деколи описовий переклад використовується

сумісно з калькуванням, наприклад, MS-DOS – дискова операційна система фірми Microsoft.

Щоб правильно використовувати описовий спосіб перекладу, треба добре знати предметну сферу тексту, який перекладається, для того, щоб правильно відтворити зміст поняття. З дослідженої вибірки можна розглянути такі приклади *schema statements* - SQL вирази управління схемою даних, *bigint* - ціле число великого діапазону.

Коли, під час перекладу значення англomовного терміну цілком співпадає з українським, то такий спосіб є еквівалентним. Такий спосіб перекладу є досить розповсюдженим під час перекладу англomовних ІТ термінів, хоча насправді українська мова має не таку вже й велику кількість аналогів у цій сфері, наприклад *desktop* – робочий стіл, *notepad* – записна книжка. Еквівалентність є комунікативної рівноцінністю, за допомогою якої текст і перетворюється на переклад. Еквівалентність є передача змісту тексту в перекладі, яку розглядають як інформацію, щоє в тексті з стилістичними, образними, естетичними функціями лексичних одиниць. Еквівалентність є одним з найважливіших завдань перекладача, яке відображається в найповнішій передачі змісту тексту, і справжня відповідність змісту оригіналу і перекладу дуже важлива. Еквівалентність залежить також від походження тексту оригіналу і його відображення в мові перекладу.

Терміни словосполучення нерідко застосовуються у вигляді аббревіатур/акронімів, велика кількість яких є загальноновживаними в ІТ дискурсі. Аббревіатури, як спосіб словотворення є одним з найпродуктивніших в сьогodнішній англійській мові, що пояснюється прискоренням темпу життя і бажанням економії мовних засобів. Акроніми є ознакою, яка притаманна ІТ дискурсу. Застосування аббревіатур є можливим якщо адресат володіє спільним з автором тезаурусом. В ІТ дискурсі розповсюджене вживання акронімів спричинено великою кількістю термінів словосполучень, скорочення яких є потрібним для економії місця, наприклад: RAM – Random Access Memory.

Якщо перекладач зустрічає термін, в перекладі якого він сумнівається, то важливо перевірити, чи не належить він до термінів, що не перекладаються. Проблема стандартизації перекладу термінів ІТ дискурсу перекликається зі створенням адекватних глосаріїв перекладацьких відповідностей понять для уживання в системах Translation Memory. Терміни мають перекладатися одноманітно і однозначно, можливо при правильному використанні добре сформульованої моделі перекладу.

ВИСНОВКИ

Інформаційні технології все більше впливають на сучасне суспільство. Вони наповнюють наукове, технічне і приватне життя кожної людини. Використання інформаційних технологій спрощує процес навчання, покращує його якість і створює умови для віддаленої.

Слово термін означає поняття, що може бути використаним в окремій галузі з найбільш точним смисловим значенням і бути емоційно нейтральним. Термінологія є спеціальною формою щоденної мови, яка адоптована до найбільш точного опису необхідної галузі знань. Вона відрізняється насамперед фаховою лексикою, синтаксисом, а також частотою використання конкретних мовних граматичних форм. Розвиток термінології притаманний процесу розвитку цивілізації. Всі світові зміни і науково-технічні відкриття утворюють нові концепції, процеси та функції, що потребують найменування, для обміну інформацією та фаховими знаннями. Технічні винаходи, що наразі не є новими, але на початку їх існування вони були англійськими неологізмами. Термінологія має бути не лише «переліком слів», але організованою системою з набором правил, що полегшують розуміння, утворення та перекладу фахових термінів.

Комп'ютерні терміни завжди тісно пов'язані з технічним розвитком, а істотний розвиток ІТ галузі змушує якомога швидше систематизувати найменування об'єктів. В кожній мові виникає необхідність назвати явище, яке відноситься до конкретної технічної області, а потім необхідністю є визначення понять і величин, що використовуються для опису цих явищ, далі слід вказати одиниці вимірювання, якими це можна виміряти. Тільки після завершення всіх вищезазначених заходів можна сформулювати детальні технічні вимоги. Це передбачає необхідність постійно систематизувати фахову термінологію.

Джерелами розширення термінології є поява нових речей і концепцій у зв'язку з прогресом та спроби освоїти та організувати все більш численні виробничі та комерційні процеси. З одного боку, розвиток технічної

термінології на окремих національних мовах вельми динамічний, з іншого боку, прогрес є результатом роботи міжнародних дослідницьких, а технічні винаходи швидко імпортуються, що вимагає міжнародного професійного спілкування.

Для сучасного науково-технічного дискурсу характерний цілий ряд комунікативних засобів. Його різноманітність та застосування відрізняється не лише застосуванням професіоналізмів, а й поєднанням лексичних одиниць, що відносяться до різних стилів, що створені відповідно до цілей спілкування у віртуальних мережах. За своїм походженням комп'ютерна термінологія неоднорідна. Вона утворюється з питомих назв, чужомовних запозичень, номінацій змішаного типу. В ході формування ІТ термінології мали місце як процес термінологізації, так і процес детермінологізації.

Структурно-семантичні особливості термінів, звертали на себе увагу мовознавців. Структура термінологічних одиниць – це дуже важливий чинник перекладу, бо щоб отримати відповідність перекладач має відтворити структуру слова у цільовій мові.

Переклад термінів є однією з найскладніших проблем перекладознавства, бо термінологія постійно розвивається. Термінологія ІТ має попит у фахівців та вимагає особливої уваги. Деколи виникають труднощі в процесі їх перекладу, бо переклад є неможливим без спеціальних знань, що пов'язані з особливостями перекладу термінів. Переклад потребує доброго знання ІТ галузі, розуміння змісту англійських термінів і знання термінології українською. В процесі перекладу термінології виділяють етапи з'ясування значення терміну і переклад значення рідною мовою. При перекладі терміну, можна не зосереджуватись на контексті, бо еквівалентом терміну буде тільки термін, але не завжди перекладом терміну буде проста заміна англійського слова українським. Існує декілька способів перекладу ІТ термінів: калькування, транс кодування, описовий та еквівалентний переклад.

У випадку калькування слова і вирази однієї мови перекладаються іншою мовою шляхом точного відтворення засобами мови, на яку відбувається

переклад їх морфемну або словесну структури. Калькування як спосіб перекладу без еквівалентної одиниці може використовуватися коли слово має прозору словотворчу структуру і внутрішню форму.

Транскодування є способом перекладу, де звукова та/або графічна форма слова однієї мови передається засобами іншої мови. Транскодування використовується тоді, коли в мові перекладу немає коректного відповідного терміну, і не можливо знайти слово, яке б еквівалентно відтворило зміст терміну і відповідали більшості вимог утворення термінів. Існує 4 типи транскодування: транскрибування, транслітерація, транс кодування змішане і адаптивне.

Описовий переклад є відтворенням значення лексичної одиниці за допомогою розгорнутих словосполучень. Вони розкривають важливі ознаки позначуваного лексичною одиницею явища, за допомогою його дефініції, якщо жоден з словесних прийомів підбору відповідності не задовольняє ситуації.

Еквівалентність – це збереження відносної рівності змістовної та семантичної інформації, що знаходиться в оригіналі й перекладі. Еквівалентність залежить від походження тексту і його відображення в мові перекладу. Еквівалентним перекладом можна назвати переклад, який відповідає поняттю комунікативної рівності.

При дослідженні вибірки термінів важливо звертати увагу на її особливості:

1. Процес аналізу ІТ термінів ускладнюється великою кількістю неологізмів і неоднорідністю термінології.
2. Термінологія ІТ галузі у своєму словотворі підпорядкована законам української мови. Поширеними способами словотворення є афіксальний, безафіксний, а також лексико-семантичний.
3. Для ІТ дискурсу характерним є використання засобів мовної експресії.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Лепеха Т. В. Лексико-семантичні та словотвірні-структурні особливості судово-медичної термінології: дис. ... канд. філол. наук : 10.02.01. Дніпро, 2000. 5 с.
2. Томіленко Л. М. Термінологічна лексика в сучасній тлумачній лексикографії української літературної мови: монографія. Івано-Франківськ : Фоліант, 2015. 160 с.
3. Ішмуратов А. Термін. Філософський енциклопедичний словник. редкол.: В. І. Шинкарук та ін. НАН України, Інститут філософії імені Григорія Сковороди. Київ : Абрис, 2002. 636 с.
4. Nowicki W. O ścisłości pojęcia kulturze słowa w technice. Warszawa, 1978. 22 p.
5. Сидоренко О. А. Теоретичні засади суспільно-політичної термінології. Мова і культура. Київ, 2013. Вип. 16. 203 с
6. Лейчик В. М. Терминоведение: предмет, методы, структура. Москва, 2006. 21 с.
7. Володина М.Н. Научно-техническая терминология и общеупотребительная лексика. Вестн. Российской риторической ассоц. Красноярск, 2000. 33 с.
8. Будкова С. С. Термины радиационных и плазменных технологий в рамках научно-технической терминологии : материалы VII междунар. науч.–практ. конф. «Прикладная филология: Идеи, концепции, проекты». Томск, 2009. С. 219.
9. Lukszyn J., Zmarzer W. Teoretyczne podstawy terminologii. Warszawa : Katedra Języków Specjalistycznych UW, 2006. 189 p.
10. Котелова Н. З. До питання про специфіку терміну. Лінгвістичні проблеми науково-технічної термінології. Москва : Наука, 1970. 86 с.
11. Алексеева Л. М. Проблемы терміну та терміноутворення: навч. посіб. Пермь : ПГУ, 1998. 120 с.

12. Нікітін М. В. Курс лінгвістичної семантики : навч. посіб. 2-ге вид., перероб. та доп. СПб. : Вид-во РДПУ ім. А.І. Герцена, 2007. 819 с.
13. Мойсеєнко С. М. Особливості англійської комп'ютерної термінології : матеріали всеукр. наук.- практ. конф., м. Полтава, 2 квітня 2014 р. Полтава, 2014. URL:
14. Ніколаєва А. О. Структурно семантична характеристика термінології програмування комп'ютерних мереж та захисту інформації : автореф. дис. ... канд. філол. наук : 10.02.01. Харків, 2002. 20 с.
15. Панько Т. І. Українське термінознавство. Львів: Світ, 1994. 33с.
16. Романова О. О. Спеціальна лексика української мови як об'єкт лінгвістичного дослідження: термін і номен. Термінологічний вісник. Харків, 2011. Вип. 1. 55 с.
17. Овчаренко Н. І. Теоретичні передумови дослідження терміносистем (на матеріалі сучасної української термінології). Лінгвістичні студії. 2010. Вип. 20. С. 172-175.
18. Флоренский П. А. У водоразделов мысли. Москва : Правда, 1990. 229 с.
19. Терских Н.В. О специфике основных разрядов профессиональной лексики. Вестн. Иркутского гос. лингвистического ун-та. Иркутск, 2010. № 46. С. 131-136.
20. Lukszyn J., Zmarzer W. Teoretyczne podstawy terminologii. Warszawa : Katedra Języków Specjalistycznych UW, 2006. 189 p.
21. Абабілова Н. М., Білокамінська В. Л. Особливості перекладу термінів українською мовою. Молодий вчений. 2015. № 2 (17). С. 126-128.
22. Pawluk K. Perypetie i zawłósci terminologii z zakresu najnowszej techniki. Nowa Elektrotechnika. 2007. № 5. P. 28–30.
23. Мікрюков О. О. Структурно-семантичні особливості комп'ютерних термінів словосполучень. Педагогічна освіта: теорія і практика : зб. наук. пр. Кам'янець-Подільський, 2014. Вип. 1(16). С. 231-236.

24. Козловська Л. С. Семантично-структурний аналіз термінології комп'ютерної сфери (словотвірний аспект). Гуманітарний вісн. Іноземна філологія. Черкаси : ЧДТУ, 2003. С. 18-25.
25. Прохорова В. Н. Семантика терміна. Вестник МГУ. Филологія. Москва : ЦИИ МГУ, 1981. № 3. С. 23-32.
26. Мусоямов З. М. Структурно-семантичний аналіз термінів програмного забезпечення комп'ютера: автореф. дис. ... канд. філол. наук : 10.02.20. Душанбе, 2017. 83 с.
27. Богачик М. С. Особливості словотворення англійської комп'ютерної термінології. Наукові записки нац. ун-ту «Острозька академія». Сер. Філологічна : зб. наук. пр. Рівне, 2015. Вип. 52. С. 42-44.
28. Клименко Н. Ф. Основи морфеміки сучасної української мови: навч. посіб. Київ : ІЗМН, 1998. 3 с.
29. Андрусак І. В. Англійські неологізми кінця ХХ століття як складова мовної картини світу : автореф. дис. ... канд. філол. наук : 10.02.04. Київ, 2003. 92 с.
30. Кобець Л.К. Лексико-семантична група як складник лексико-семантичної системи. Мова і культура. Київ, 2012. Вип 15. Т. 4. С. 129-135. URL: http://nbuv.gov.ua/UJRN/Mik_2012_15_4_23(дата звернення: 16.01.2021).
31. Томіленко Л. М. Термінологічна лексика в сучасній тлумачній лексикографії української літературної мови: монографія. Івано-Франківськ : Фоліант, 2015. 160 с.
32. Межжеріна Г. В. Структурна організація семантичних одиниць (поле – лексико-семантична група – слово). Актуальні проблеми української лінгвістики: теорія і практика : зб. наук. пр. Київ : Вид.-полігр. центр Київ, 2002. Вип. 5. С. 114–126.
33. Шматко І. В. Особливості системної організації українських бджільницьких термінів за тематичними і лексико-семантичними групами.

- Термінологічний вісник : зб. наук. пр. / відп. ред. В. Л. Іващенко. Київ, 2015. Вип. 3(2). С. 198–210.
34. Микульчик Р. Б. Тематична класифікація фізичних термінів-епонімів. Вісн. нац. ун-ту Львівська політехніка. Проблеми української термінології. Львів, 2016. № 842. С. 92–95.
35. Ментинська І. Б. Тематична та лексико-семантична класифікація українських комп'ютерних термінів. Вчені записки ТНУ імені В. І. Вернадського. Філологія. Соціальні комунікації. Львів, 2020. Т. 31 (70). № 2. Ч. 1. С. 26-30.
36. Мойсеєнко С. М. Особливості англійської комп'ютерної термінології : матеріали всеукр. наук.- практ. конф., м. Полтава, 2 квітня 2014 р. Полтава, 2014.
URL: http://www.kamts1.kpi.ua/sites/default/files/files/02_Moyseyenko_Osoblyvosti.pdf (дата звернення: 19.01.2021).
37. Комиссаров В. Н. Теория перевода (лингвистические аспекты) : учеб. пособ. Москва :Высшая школа, 1990. 328 с.
38. Сарієва А. Б. Особливості перекладу комп'ютерних термінів. Наукові записки нац. ун-ту «Острозька академія». Сер. Філологічна : зб. наук. пр. 2012. Вип. 25. С. 97-98.
39. Кивлюк В. О. Модель перекладу англійських комп'ютерних термінів на українську мову. Освітній дискурс : зб. наук. пр. Київ : Гілея, 2017. Вип. 1. С. 145-154.
40. Кияк Т. Р. Фальшиві друзі перекладача як проблема міжкультурної комунікації. Наукові записки. Сер. Філологічні науки. Кіровоград : РВВ КДПУ ім. В. Винниченка, 2010. Вип. 89 (1). С. 17-20.
41. Карабан В. І. Граматичні труднощі лексичні, термінологічні та жанрово-стилістичні проблеми. Переклад англійської наукової і технічної літератури. Вінниця : Нова книга, 2002. 407 с.
42. Казакова., Возненко Н. В., Радецька С. В. Термінологія та переклад: навч. посіб.Вінниця : Нова книга, 2010. 232 с.

43. Кікец І., Кікец Н. До питання про деякі труднощі при перекладі термінів і шляхи їх подолання. Вісн. держ. ун-ту «Львівська політехніка». Львів : Львівська політехніка, 2000. № 402. 420 с.
44. Сапальова Н. М. Використання в українському перекладі науково-технічних термінів іншомовного походження. Культура народів Причорномор'я. 2001. № 13. С. 129–133.
45. Герасімова О. М. Особливості перекладу термінів (на прикладі прикордонного дискурсу). Наук. вісн. міжн. гуманітарного ун-ту. Сер. Філологія. 2016. № 22. С. 180-182.
46. Реформатский А. А. Терминология. Введение в языковедение / за ред. В. А. Виноградова. Москва : Аспект Пресс, 1996. 536 с.
47. Скопенко О. І. Сучасний словник іншомовних слів : слов. термінів. Київ, 2006. 789 с.
48. Бусел В. Т. Великий тлумачний словник сучасної української мови / уклад. і гол. ред. В. Т. Бусел. Київ : Перун, 2007. 1728 с
49. Казакова Т. А. Практические основы перевода English-Russian. Изучаем иностранные языки. СПб. : Изд-во Союз, 2001. 320 с.
50. Кукаріна А. Д. Методологія та методика способів перекладу абревіатур англomовного та україномовного медичного дискурсу. Молодий вчений. 2016. № 9 (36). С. 224-228.
51. Борщовецька В. Д. Навчання студентів-економістів англійської фахової лексики : автореф. дис. ... канд. пед. наук : 13.00.02. Київ, 2004. 20 с.
52. Кучман І. М. Переклад англійських термінів у галузі комп'ютерних технологій. Вісн. Житомирського держ. ун-ту ім. Івана Франка. Сер. Філологічні науки. Житомир, 2005. № 23. С. 164-166.
53. Шевцова О. В. Способи еквівалентного перекладу лексичних одиниць текстів конвенцій з англійської та французької мов українською. Наукові записки нац. ун-ту «Острозька академія». Сер. Філологічна : зб. наук. пр. 2012. Вип. 25. С. 129-131.

54. Гузь А. М. Транскодування як спосіб перекладу англійської авіаційної термінології. Наука і молодь. Сер. Гуманітарна. 2012. № 11-12. С. 143-146.
55. Черноватий Л., Шех В. Проблеми передачі вузькоспеціальної англійської термінології українською мовою. Наукові записки Кіровоградського держ. пед. ун-ту ім. Володимира Винниченка . Сер. Філологічні науки : зб. наук. пр. 2009. Вип. 81(4). С. 109-116.
56. Карабан В. І. Лексичні, термінологічні та жанрово-стилістичні труднощі. Переклад англійської наукової і технічної літератури : навч. посіб. Вінниця : Нова книга, 2001. Ч. 2. 303 с.
57. Spring framework documentation. URL: <https://spring.io/>
58. Pro git. URL: <https://git-scm.com/book/en/v2>
59. OpenAPI Specification. URL: <https://swagger.io/specification/>
60. Гордієнко Н. М. Поняття перекладацької еквівалентності як центральна проблема теорії художнього перекладу : матеріали VIII Міжн. наук.- практ. конф. Прага : Education and Science, 2012. С. 2.
61. Ктитарова Н. К., Воронова З. Ю. Теорія перекладу : навч. посіб. Дніпродзержинськ : ДДТУ, 2013. 310 с.
62. Ковалевська Т. І., Царук А. О. Способи забезпечення еквівалентності та адекватності перекладу англійськомовних складних термінів в написанні проектів європейського стандарту : матеріали VI всеукр. наук.-практ. конф. м. Вінниця, 26 лют. 2019 р. Вінниця, 2019. С. 2.
63. Herbert Schildt. The Complete Reference. Java. Eighth Edition. URL: <https://srikarthiks.files.wordpress.com/2016/07/java-the-complete-reference-8th-edition.pdf>
64. Новый російсько-українсько-англійський тлумачний словник з інформатики : словник термінів / ред. М. Д. Гінзбург; уклад.: М. Д. Гінзбург, Л. І. Білоусова, І. М. Корніловська. Харків: Корвін, 2002. 656 с.

65. Єнікєєва Є. М. Особливості перекладу комп'ютерних термінів на українську мову. Вісник СумДУ. Суми, 2001. №5 (26). С. 54-59

**ПРИКЛАДИ ПЕРЕКЛАДУ
КОМП'ЮТЕРНОЇ ТЕРМІНОЛОГІЇ
УКРАЇНСЬКОЮ МОВОЮ**

	Терміносполука англійською мовою	Терміносполука українською мовою	Спосіб перекладу
1.	After your service provider configures your VLAN attachment for a Partner Interconnect connection, you must activate it before the attachment can start passing traffic.	Після того, як ваш постачальник послуг налаштує ваше приєднання VLAN для підключення Partner Interconnect, ви повинні активувати його, перш ніж приєднання почне пропускати трафік.	транскодування
2.	Staged means that you have marked a modified file in its current version to go into your next commit snapshot.	Індексований стан виникає тоді, коли ви позначаєте змінений файл у поточній версії, щоб ці зміни ввійшли до наступного знімку коміту.	транскодування
3.	You modify files in your working tree .	Ви редагуєте файли у своїй робочій директорії.	еквівалент
4.	You selectively stage just those changes you want to be part of your next commit, which adds only those changes to the staging area .	Вибірково надсилаєте до індексу лише ті зміни, що їх ви бажаєте зберегти в наступному коміті, і лише ці зміни буде збережено в індексі.	еквівалент
5.	Checkboxes are used to let a user select one or more	Прапорці використовуються, щоб дозволити користувачеві	еквівалент

	options of a limited number of choices.	вибрати один або кілька варіантів із обмеженої кількості варіантів.	
6.	Infrastructure provisioning, orchestration, and deployment should also support the use of the infrastructure as code .	Надання інфраструктури, оркестровка та розгортання також повинні підтримувати використання інфраструктури як коду.	еквівалент
7.	Sometimes these scripts were stored in version control systems or documented step by step in text files or run-books.	Іноді ці сценарії зберігалися в системах контролю версій або документувалися крок за кроком у текстових файлах або run-books.	еквівалент
8.	This results in the creation of new environments not always being repeatable, reliable, or consistent.	Це призводить до створення нових середовищ, які не завжди є повторюваними, надійними чи послідовними	калькування
9.	A CloudFormation template is deployed into the AWS environment as a stack.	Шаблон CloudFormation розгортається в середовищі AWS як стек.	еквівалент
10.	Before making changes to your resources , you can generate a change set, which is a summary of your proposed changes.	Перш ніж вносити зміни до ваших ресурсів, ви можете створити набір змін, який є підсумком запропонованих змін.	транскодування
11.	Change sets enable you to see how your changes might impact your running resources, especially for	Набори змін дозволяють побачити, як ваші зміни можуть вплинути на ваші поточні ресурси, особливо	калькування

	critical resources, before implementing them.	для критичних ресурсів, перед їх впровадженням.	
12.	A construct represents a cloud component and encapsulates everything AWS CloudFormation needs to create the component.	Конструкція представляє хмарний компонент і інкапсулює все, що потрібно AWS CloudFormation для створення компонента.	калькування
13.	You can set up alarms based on the metrics collected by Amazon CloudWatch Metrics.	Ви можете налаштувати сповіщення про загрози на основі показників, зібраних Amazon CloudWatch Metrics.	описовий
14.	The container then injects those dependencies when it creates the bean.	Контейнер потім вставляє ці залежності під час створення компонента.	калькування
15.	In Spring, the objects that form the backbone of your application and that are managed by the Spring IoC container are called beans .	У Spring об'єкти, які утворюють основу вашої програми та якими керує контейнер Spring IoC, називаються бінами.	транскодування
16.	A bean is an object that is instantiated, assembled, and managed by a Spring IoC container .	Бін — це об'єкт, який створюється, збирається та керується контейнером Spring IoC.	транскодування
17.	The configuration metadata is represented in XML, Java annotations, or Java code.	Метадані конфігурації представлені в XML, анотаціях Java або кодї Java.	транскодування
18.	Several implementations of the <code>ApplicationContext</code>	Кілька реалізацій інтерфейсу <code>ApplicationContext</code>	транскодування

	interface are supplied with Spring .	постачаються разом із Spring.	
19.	Your application classes are combined with configuration metadata so that, after the <code>ApplicationContext</code> is created and initialized, you have a fully configured and executable system or application.	Ваші класи програми поєднуються з метаданими конфігурації, щоб після створення та ініціалізації <code>ApplicationContext</code> у вас була повністю налаштована виконувана система або програма.	транскодування
20.	The <code>ApplicationContext</code> is the interface for an advanced factory capable of maintaining a registry of different beans and their dependencies.	<code>ApplicationContext</code> — це інтерфейс для вдосконаленої фабрики, здатної підтримувати реєстр різних бінів та їх залежностей.	транскодування
21.	These identifiers must be unique within the container that hosts the bean.	Ці ідентифікатори мають бути унікальними в контейнері, який містить компонент.	транскодування
22.	An OpenAPI definition can then be used by documentation generation tools to display the API, code generation tools to generate servers and clients in various programming languages, testing tools, and many other use cases.	Визначення OpenAPI потім може використовуватися інструментами створення документації для відображення API, інструментами створення коду для створення серверів і клієнтів різними мовами програмування,	Вживається мовою оригіналу

		інструментами тестування та багатьма іншими випадками використання	
23.	The HTTP Status Codes are used to indicate the status of the executed operation.	Коди статусу HTTP використовуються для вказівки статусу виконаної операції.	калькування
24.	The order of the tags can be used to reflect on their order by the parsing tools.	Порядок тегів можна використовувати для відображення їхнього порядку за допомогою інструментів аналізу.	транскодування
25.	This URL supports Server Variables and MAY be relative, to indicate that the host location is relative to the location where the OpenAPI document is being served.	Ця URL-адреса підтримує змінні сервера та МОЖЕ бути відносною, щоб вказати, що місце розташування хоста є відносним до розташування, де обслуговується документ OpenAPI.	описовий
26.	The information transferred as the payload of a request or response.	Інформація, що передається як пейлоад навантаження запиту або відповіді.	транскодування
27.	An entity consists of metainformation in the form of entity-header fields and content in the form of an entity-body	Сутність складається з метаінформації у формі полів заголовка сутності та вмісту у формі тіла сутності	калькування

28.	A proxy MUST implement both the client and server requirements of this specification.	Проксі має реалізувати як клієнтські, так і серверні вимоги цієї специфікації.	транскодування
29.	Gateway is server which acts as an intermediary for some other server.	Gateway - це сервер, який діє як посередник для іншого сервера.	Вживається мовою оригіналу
30.	Once active, a tunnel is not considered a party to the HTTP communication, though the tunnel may have been initiated by an HTTP request.	Після активації тунель не вважається стороною зв'язку HTTP, через тунель може бути ініційовано запитом HTTP.	калькування
31.	A cache stores cacheable responses in order to reduce the response time and network bandwidth consumption on future, equivalent requests.	Кеш зберігає відповіді, які можна кешувати, щоб зменшити час відповіді та споживання пропускну здатності мережі на майбутні еквівалентні запити.	еквівалент
32.	Comments can be included in some HTTP header fields by surrounding the comment text with parentheses.	Коментарі можна включити в деякі поля заголовків HTTP, узявши текст коментаря в дужки.	еквівалент
33.	If the port is empty or not given, port 80 is assumed.	Якщо порт порожній або не вказаний, передбачається порт 80.	транскодування
34.	Multiple message-header fields with the same field-name MAY be present in a	Кілька полів заголовка повідомлення з однаковою назвою поля МОЖУТЬ бути	описовий

	message if and only if the entire field-value for that header field is defined as a comma-separated list	присутніми в повідомленні тоді і тільки тоді, коли все значення поля для цього поля заголовка визначено як список, розділений комами	
35.	The message-body (if any) of an HTTP message is used to carry the entity-body associated with the request or response.	Тіло повідомлення (якщо таке є) повідомлення HTTP використовується для перенесення тіла сутності, пов'язаного із запитом або відповіддю.	Вживається мовою оригіналу
36.	A request message from a client to a server includes, within the first line of that message, the method to be applied to the resource, the identifier of the resource, and the protocol version in use.	Повідомлення запиту від клієнта до сервера містить у першому рядку цього повідомлення метод, який буде застосовано до ресурсу, ідентифікатор ресурсу та версію протоколу, що використовується.	транскодування
37.	The exact resource identified by an Internet request is determined by examining both the Request-URI and the Host header field.	Точний ресурс, ідентифікований в Інтернет-запиті, визначається шляхом аналізу Request-URI і поля заголовка Host.	Вживається мовою оригіналу
38.	When a client or server wishes to time-out it SHOULD issue a graceful close on the transport connection.	Коли клієнт або сервер бажає завершити час очікування відповіді, йому	описовий

		має бути закрито транспортне з'єднання.	
39.	If the client sees an error status, it SHOULD immediately cease transmitting the body .	Якщо клієнт бачить статус помилки, йому СЛІД негайно припинити передачу тіла запита.	описовий
40.	The GET method means retrieve whatever information (in the form of an entity) is identified by the Request-URI.	Метод GET означає отримання будь-якої інформації (у формі сутності), ідентифікованої Request-URI	Вживається мовою оригіналу
41.	The HEAD method is identical to GET except that the server MUST NOT return a message-body in the response.	Метод HEAD ідентичний методу GET , за винятком того, що сервер НЕ ПОВИНЕН повертати тіло повідомлення у відповідь.	Вживається мовою оригіналу
42.	This method is often used for testing hypertext links for validity, accessibility, and recent modification.	Цей метод часто використовується для перевірки гіпертекстових посилань на валідність, доступність і останні зміни.	транскодування
43.	The response to a HEAD request MAY be cacheable in the sense that the information contained in the response MAY be used to update a previously cached entity from that resource.	Відповідь на запит HEAD МОЖЕ бути кешованою в тому сенсі, що інформація, що міститься у відповіді, МОЖЕ використовуватися для оновлення попередньо кешованої сутності з цього ресурсу.	транскодування

44.	The POST method is used to request that the origin server accept the entity enclosed in the request as a new subordinate of the resource identified by the Request-URI in the Request-Line	Метод POST використовується для запиту, щоб вихідний сервер прийняв сутність, укладену в запиті, як новий підлеглий ресурс, визначений Request-URI у Request-Line	Вживається мовою оригіналу
45.	Responses to this method are not cacheable, unless the response includes appropriate Cache-Control or Expires header fields.	Відповіді на цей метод не можна кешувати, якщо відповідь не містить відповідних полів заголовка Cache-Control або Expires.	Вживається мовою оригіналу
46.	The PUT method requests that the enclosed entity be stored under the supplied Request-URI.	Метод PUT надсилає запит, щоб вкладена сутність зберігалася під наданим Request-URI.	Вживається мовою оригіналу
47.	That resource might be a data-accepting process , a gateway to some other protocol, or a separate entity that accepts annotations.	Цей ресурс може бути процесом, що приймає дані, шлюзом до іншого протоколу або окремою сутністю, яка приймає анотації.	описовий
48.	The DELETE method requests that the origin server delete the resource identified by the Request-URI.	Метод DELETE запитує вихідний сервер на видалення ресурсу, визначеного Request-URI.	Вживається мовою оригіналу
49.	The TRACE method is used to invoke a remote,	Метод TRACE використовується для виклику віддаленого	Вживається мовою оригіналу

	application-layer loop-back of the request message.	зациклення на прикладному рівні повідомлення запиту.	
50.	This interim response is used to inform the client that the initial part of the request has been received and has not yet been rejected by the server.	Ця проміжна відповідь використовується для інформування клієнта про те, що початкова частина запиту була отримана і ще не була відхилена сервером.	калькування
51.	The request has been fulfilled and resulted in a new resource being created.	Запит виконано, і створено новий ресурс.	транскодування
52.	The 202 response is intentionally non-committal .	Відповідь 202 навмисно необов'язкова.	калькування
53.	The returned metainformation in the entity-header is not the definitive set as available from the origin server, but is gathered from a local or a third-party copy.	Повернена метаінформація в заголовку сутності не є остаточним набором, доступним із вихідного сервера, а зібрана з локальної або сторонньої копії.	транскодування
54.	The default value to use for substitution, which SHALL be sent if an alternate value is not supplied.	Значення за замовчуванням для заміни, яке МАЄ надсилатися, якщо не надається альтернативне значення.	описовий
55.	Each Media Type Object provides schema and examples for the media type identified by its key.	Кожен об'єкт медіа типу надає схему та приклади типу медіа, визначеного його ключем.	калькування

56.	A map between a property name and its encoding information.	Відповідність між назвою властивості та інформацією про її кодування.	калькування
57.	The runtime type of a specific bean is non-trivial to determine.	Визначити тип середовища виконання конкретного bean-компонента нетривіально.	описовий
58.	Dependency injection (DI) is a process whereby objects define their dependencies (that is, the other objects with which they work) only through constructor arguments, arguments to a factory method, or properties that are set on the object instance after it is constructed or returned from a factory method.	Впровадження залежностей (DI) — це процес, за допомогою якого об'єкти визначають свої залежності (тобто інші об'єкти, з якими вони працюють) лише за допомогою аргументів конструктора, аргументів фабричного методу або властивостей, які встановлюються для екземпляра об'єкта після його створення або повернуто із заводського методу.	калькування
59.	Constructor-based DI is accomplished by the container invoking a constructor with a number of arguments, each representing a dependency.	DI на основі конструктора виконується контейнером, який викликає конструктор із низкою аргументів, кожен з яких представляє залежність.	калькування

60.	Calling a static factory method with specific arguments to construct the bean is nearly equivalent, and this discussion treats arguments to a constructor and to a static factory method similarly.	Виклик статичного фабричного методу зі специфічними аргументами для побудови bean майже еквівалентний, і в цьому обговоренні аргументи конструктора та статичного фабричного методу розглядаються однаково.	Транскодування
61.	Setter-based DI is accomplished by the container calling setter methods on your beans after invoking a no-argument constructor or a no-argument static factory method to instantiate your bean.	DI на основі сетера виконується шляхом виклику контейнером методів сетера ваших компонентів після виклику конструктора без аргументів або статичного фабричного методу без аргументів для створення екземпляра вашого компонента.	описовий
62.	The <code>ApplicationContext</code> is created and initialized with configuration metadata that describes all the beans.	<code>ApplicationContext</code> створюється та ініціалізується за допомогою метаданих конфігурації, які описують усі компоненти.	транскодування
63.	The Spring container validates the configuration of each bean as the container is created.	Контейнер Spring перевіряє конфігурацію кожного компонента під час створення контейнера.	калькування

64.	The first form is preferable to the second, because using the idref tag lets the container validate at deployment time that the referenced, named bean actually exists.	Перша форма є кращою перед другою, тому що використання тегу idref дозволяє контейнеру під час розгортання перевірити, що названий bean-компонент, на який посилається, дійсно існує.	Вживається мовою оригіналу
65.	The Spring Framework supports six scopes , four of which are available only if you use a web-aware ApplicationContext	Spring Framework підтримує шість областей, чотири з яких доступні, лише якщо ви використовуєте ApplicationContext, що підтримує роботу в мережі	калькування
66.	Only one shared instance of a singleton bean is managed, and all requests for beans with an ID or IDs that match that bean definition result in that one specific bean instance being returned by the Spring container.	Керується лише один спільний екземпляр єдиного можливого компонента, і всі запити на компоненти з ідентифікатором або ідентифікаторами, які відповідають цьому визначенню, призводять до того, що контейнер Spring повертає цей конкретний екземпляр компонента.	описовий
67.	Allocate a new data buffer, optionally specifying capacity upfront, if known, which is more efficient even though	Виділіть новий буфер даних, за бажанням заздалегідь вказавши ємність, якщо вона відома,	Вживається мовою оригіналу

	implementations of DataBuffer can grow and shrink on demand.	що ефективніше, навіть якщо реалізації DataBuffer можуть збільшуватися та зменшуватися за вимогою.	
68.	Object-oriented programming (OOP) is at the core of Java.	Об'єктно-орієнтоване програмування (ООП) лежить в основі Java.	калькування
69.	All computer programs consist of two elements: code and data .	Усі комп'ютерні програми складаються з двох елементів: коду та даних.	еквівалент
70.	An object-oriented program can be characterized as data controlling access to code	Об'єктно-орієнтовану програму можна охарактеризувати як дані, що контролюють доступ до коду	описовий
71.	A powerful way to manage abstraction is through the use of hierarchical classifications.	Потужним способом керування абстракцією є використання ієрархічних класифікацій	транскодування
72.	Encapsulation is the mechanism that binds together code and the data it manipulates, and keeps both safe from outside interference and misuse	Інкапсуляція — це механізм, який об'єднує код і дані, якими він маніпулює захищає від зовнішнього втручання та неправильного використання	транскодування
73.	Access to the code and data inside the wrapper is tightly controlled through a well-defined interface	Доступ до коду та даних всередині оболонки суворо контролюється через чітко визначений інтерфейс	транскодування

74.	It encapsulates hundreds of bits of information about your engine, such as how much you are accelerating, the pitch of the surface you are on, and the position of the shift lever	Він інкапсулює сотні бітів інформації про ваші потужності, наприклад, скільки ви прискорюєтеся, нахил поверхні, на якій ви перебуваєте, і положення важеля перемикачів передач.	транскодування
75.	Inheritance is the process by which one object acquires the properties of another object.	Спадкування — це процес, за допомогою якого один об'єкт набуває властивостей іншого об'єкта.	калькування
76.	This is known as a subclass of animals, where animals are referred to as mammals' superclass	Це відомий як підклас тварин, де тварини називають супер-класом ссавців	калькування
77.	A deeply inherited subclass inherits all of the attributes from each of its ancestors in the class hierarchy	Глибоко успадкований підклас успадковує всі атрибути від кожного зі своїх предків в ієрархії класів	калькування
78.	Polymorphism (from Greek, meaning “many forms”) is a feature that allows one interface to be used for a general class of actions	Поліморфізм (з грецької означає «багато форм») — це функція, яка дозволяє використовувати один інтерфейс для загального класу дій.	транскодування

79.	In Java , a source file is officially called a compilation unit.	У Java вихідний файл офіційно називається модулем компіляції.	Вживається мовою оригіналу
80.	To compile the Example program, execute the compiler, javac , specifying the name of the source file on the command line	Щоб скомпілювати приклад програми, запустіть компілятор javac , вказавши назву вихідного файлу в командному рядку	транскодування
81.	The Java if statement works much like the IF statement in any other language.	Оператор Java if працює так само, як оператор IF у будь-якій іншій мові.	Вживається мовою оригіналу
82.	Java allows two or more statements to be grouped into blocks of code, also called code blocks	Java дозволяє згрупувати два або більше операторів у блоки коду, які також називають блоками коду	калькування
83.	There are 50 keywords currently defined in the Java language	Зараз у мові Java визначено 50 ключових слів	калькування
84.	The keywords const and goto are reserved but not used.	Ключові слова const і goto зарезервовані, але не використовуються.	Вживається мовою оригіналу
85.	Java defines eight primitive types of data: byte, short, int, long, char, float, double, and boolean.	Java визначає вісім простих типів даних: byte, short, int, long, char, float, double і boolean.	еквівалент
86.	A literal character is represented inside a pair of single quotes	Літеральний символ представлено всередині пари одинарних лапок	транскодування

87.	The identifier is the name of the variable .	Ідентифікатор - це ім'я змінної.	еквівалент
88.	Although the preceding examples have used only constants as initializers, Java allows variables to be initialized dynamically , using any expression valid at the time the variable is declared.	Хоча в попередніх прикладах використовувалися лише константи як ініціалізатори, Java дозволяє динамічно ініціалізувати змінні, використовуючи будь-який вираз, дійсний на момент, коли змінна оголошується.	калькування
89.	To create a conversion between two incompatible types, you must use a cast .	Щоб створити перетворення між двома несумісними типами, необхідно використовувати приведення.	калькування
90.	A one-dimensional array is, essentially, a list of like-typed variables.	Одновимірний масив, по суті, є списком однотипних змінних.	еквівалент
91.	In Java, multidimensional arrays are actually arrays of arrays.	У Java багатовимірні масиви насправді є масивами масивів.	Вживається мовою оригіналу
92.	The String type is used to declare string variables	Тип String використовується для оголошення змінних рядка	Вживається мовою оригіналу
93.	If you are an experienced C/C++ programmer, then you	Якщо ви досвідчений програміст на C/C++, то ви	Вживається мовою оригіналу

	know that these languages provide support for pointers.	знаєте, що ці мови підтримують покажчики.	
94.	Arithmetic operators are used in mathematical expressions in the same way that they are used in algebra.	Арифметичні оператори використовуються в математичних виразах так само, як і в алгебрі.	еквівалент
95.	The basic arithmetic operations — addition, subtraction, multiplication, and division —all behave as you would expect for all numeric types.	Основні арифметичні операції — додавання, віднімання, множення та ділення — поведуться так, як ви очікували, для всіх числових типів.	еквівалент
96.	The continue statement performs such an action.	Оператор <code>continue</code> виконує таку дію.	еквівалент
97.	As with the break statement, continue may specify a label to describe which enclosing loop to continue.	Як і оператор <code>break</code> , <code>continue</code> може вказати мітку для опису циклу, який потрібно продовжити.	Вживається мовою оригіналу
98.	The return statement is used to explicitly return from a method.	Оператор <code>return</code> використовується для явного повернення з методу.	Вживається мовою оригіналу
99.	For example, you already know that access to a private member of a class is granted only to other members of that class.	Наприклад, ви вже знаєте, що доступ до приватного члена класу надається лише іншим членам цього класу.	калькування
100.	In a Java source file, import statements occur immediately following the package	У вихідному файлі Java оператори імпорту з'являються безпосередньо	калькування

	statement (if it exists) and before any class definitions.	після оператора пакета (якщо він існує) і перед будь-якими визначеннями класів.	
101.	Using the keyword <code>interface</code> , you can fully abstract a class' interface from its implementation .	Використовуючи ключове слово <code>interface</code> , ви можете повністю абстрагувати інтерфейс класу від його реалізації.	еквівалент
102.	If a class includes an interface but does not fully implement the methods defined by that interface, then that class must be declared as abstract .	Якщо клас містить інтерфейс, але не повністю реалізує методи, визначені цим інтерфейсом, тоді цей клас має бути оголошений як абстрактний.	еквівалент
103.	A nested interface can be declared as <code>public</code> , <code>private</code> , or <code>protected</code> .	Вкладений інтерфейс може бути оголошений як публічний, приватний або захищений.	еквівалент
104.	One interface can inherit another by use of the keyword extends .	Один інтерфейс може успадкувати інший за допомогою ключового слова <code>extends</code> .	Вживається мовою оригіналу
105.	This chapter examines Java's exception-handling mechanism.	У цій главі розглядається механізм обробки помилок Java.	описовий
106.	All exception types are subclasses of the built-in class Throwable .	Усі типи помилок є підкласами вбудованого класу <code>Throwable</code> .	Вживається мовою оригіналу

107.	The other branch is stopped by Error , which defines exceptions that are not expected to be caught under normal circumstances by your program.	Інша гілка зупиняється помилкою, яка визначає винятки, які не очікується, щоб ваша програма перехопила за звичайних обставин.	еквівалент
108.	This small program includes an expression that intentionally causes a divide-by-zero error	Ця невелика програма містить вираз, який навмисно викликає помилку ділення на нуль	калькування
109.	When the Java run-time system detects the attempt to divide by zero, it constructs a new exception object and then throws this exception .	Коли система виконання Java виявляє спробу поділу на нуль, вона створює новий об'єкт помилки, а потім дає можливість показати її користувачу.	описовий
110.	Although the default exception handler provided by the Java run-time system is useful for debugging, you will usually want to handle an exception yourself.	Незважаючи на те, що обробник помилок за замовчуванням, наданий системою виконання Java, корисний для налагодження, зазвичай ви захочете обробити виняток самостійно.	калькування
111.	To guard against and handle a run-time error, simply enclose the code that you want to monitor inside a try block .	Щоб захиститися від помилок під час виконання та впоратися з ними, просто вставте код, який потрібно контролювати, у блок try.	Калькування та транскодування

112.	Using generics , it is possible to create a single class, for example, that automatically works with different types of data.	Використовуючи дженерики, можна створити єдиний клас, наприклад, який автоматично працює з різними типами даних.	транскодування
113.	To handle these (and other) situations, Java provides type wrappers , which are classes that encapsulate a primitive type within an object.	Для обробки цих (та інших) ситуацій Java надає обгортки типів, які є класами, які інкапсулюють примітивний тип в об'єкт.	калькування
114.	Your application can create subclasses that extend ClassLoader , implementing its methods.	Ваша програма може створювати підкласи, які розширюють ClassLoader, реалізуючи його методи.	Вживається мовою оригіналу
115.	The Runnable interface must be implemented by any class that will initiate a separate thread of execution.	Інтерфейс Runnable має бути реалізований будь-яким класом, який буде ініціювати окремий потік виконання.	калькування
116.	SecurityManager is a class that your classes can subclass to create a security manager.	SecurityManager — це клас, який ваші класи можуть створювати підкласи для створення менеджера безпеки	Вживається мовою оригіналу
117.	The StackTraceElement class describes a single stack frame, which is an individual element of a stack trace when an exception occurs.	Клас StackTraceElement описує один кадр стека, який є окремим елементом трасування стека, коли виникає виняткова ситуація.	транскодування

118.	Enum defines several methods that are available for use by all enumerations	Enum визначає кілька методів, доступних для використання всіма перерахуваннями	Вживається мовою оригіналу
119.	The Collection interface is the foundation upon which the Collections Framework is built because it must be implemented by any class that defines a collection.	Інтерфейс Collection є основою, на якій будується Collections Framework, оскільки він має бути реалізований будь-яким класом, який визначає колекцію.	Вживається мовою оригіналу
120.	The List interface extends Collection and declares the behavior of a collection that stores a sequence of elements	Інтерфейс List розширює Collection і оголошує поведінку колекції, яка зберігає послідовність елементів	Вживається мовою оригіналу
121.	The SortedSet interface extends Set and declares the behavior of a set sorted in ascending order.	Інтерфейс SortedSet розширює Set і оголошує поведінку набору, відсортованого в порядку зростання.	Вживається мовою оригіналу
122.	The NavigableSet interface extends SortedSet and declares the behavior of a collection that supports the retrieval of elements based on the closest match to a given value or values.	Інтерфейс NavigableSet розширює SortedSet і оголошує поведінку колекції, яка підтримує пошук елементів на основі найбільшої відповідності заданому значенню або значенням.	Вживається мовою оригіналу

123.	The Queue interface extends Collection and declares the behavior of a queue, which is often a first-in, first-out list.	Інтерфейс Queue розширює Collection і оголошує поведінку черги, яка часто є списком першим увійшов, першим вийшов.	Вживається мовою оригіналу
124.	The Deque interface extends Queue and declares the behavior of a double-ended queue.	Інтерфейс Deque розширює Queue і оголошує поведінку двосторонньої черги.	Вживається мовою оригіналу
125.	Returns the element at the tail of the deque, removing the element in the process. It returns null if the deque is empty.	Повертає елемент у хвості двох рядів, видаляючи елемент у процесі. Він повертає null , якщо двочерга порожня.	Вживається мовою оригіналу
126.	Because LinkedList implements the Deque interface, you have access to the methods defined by Deque .	Оскільки LinkedList реалізує інтерфейс Deque , ви маєте доступ до методів, визначених Deque .	Вживається мовою оригіналу
127.	HashSet does not define any additional methods beyond those provided by its superclasses and interfaces.	HashSet не визначає жодних додаткових методів, крім тих, що надаються його суперкласами та інтерфейсами.	Вживається мовою оригіналу
128.	It creates a queue that is prioritized based on the queue's comparator	Він створює чергу, яка має пріоритет на основі компаратора черги	еквівалент
129.	Passing a known number of arguments can be faster than	Передача відомої кількості аргументів може бути	еквівалент

	using a vararg parameter when the number of arguments is small.	швидшою, ніж використання параметра vararg, коли кількість аргументів невелика.	
130.	This means that you cannot cycle through a map using a for-each style for loop	Це означає, що ви не можете циклічно переходити по карті за допомогою циклу for-each	еквівалент
131.	The Map interface maps unique keys to values.	Інтерфейс Map відображає унікальні ключі для значень.	Вживається мовою оригіналу
132.	A key is an object that you use to retrieve a value at a later date.	Ключ — це об'єкт, який ви використовуєте для отримання значення пізніше.	калькування
133.	The NavigableMap interface extends SortedMap and declares the behavior of a map that supports the retrieval of entries based on the closest match to a given key or keys.	Інтерфейс NavigableMap розширює SortedMap і оголошує поведінку карти, яка підтримує пошук записів на основі найбільшої відповідності заданому ключу або ключам.	Вживається мовою оригіналу
134.	Returns a NavigableMap that includes all entries from the invoking map that have keys that are less than upperBound	Повертає NavigableMap, що включає всі записи з мапи виклику, ключі яких менші за upperBound	Калькування та транскодування
135.	It maintains a linked list of the entries in the map, in the order in which they were inserted .	Він підтримує пов'язаний список записів у мапі в тому	транскодування

		порядку, в якому їх було вставлено.	
136.	Returns true if the elements were added and false otherwise.	Повертає true, якщо елементи були додані, і false в іншому випадку.	Вживається мовою оригіналу
137.	Stack is a subclass of Vector that implements a standard last-in, first-out stack.	Стек — це підклас вектора, який реалізує стандартний стек «останній увійшов, перший вийшов».	калькування
138.	Given a key and value, you can store the value in a Dictionary object.	Отримавши ключ і значення, ви можете зберегти значення в об'єкті Dictionary.	Вживається мовою оригіналу
139.	One important point: Like the map classes, Hashtable does not directly support iterators.	Один важливий момент: як і класи мапи, Hashtable не підтримує безпосередньо ітератори.	Вживається мовою оригіналу
140.	Beginning with version 1.4, Java has provided a second I/O system called NIO (which is short for New I/O).	Починаючи з версії 1.4, Java надала другу систему вводу/виводу під назвою NIO (що скорочено від New I/O).	Вживається мовою оригіналу
141.	The NIO system is built on two foundational items: buffers and channels .	Система NIO побудована на двох основних елементах: буферах і каналах.	калькування
142.	The current position is the index within the buffer at which the next read or write operation will take place.	Поточна позиція — це індекс у буфері, за яким відбудеться наступна	еквівалент

		операція читання або запису.	
143.	The limit is the index value one past the last valid location in the buffer.	Обмеження – це значення індексу на один кінець останнього дійсного розташування в буфері.	еквівалент
144.	The capacity is the number of elements that the buffer can hold.	Ємність - це кількість елементів, які може вмістити буфер.	еквівалент
145.	You can encode a sequence of characters into bytes using an encoder	Ви можете закодувати послідовність символів у байти за допомогою кодувальника	еквівалент
146.	The two most common windows are those derived from Panel , which is used by applets, and those derived from Frame , which creates a standard application window	Два найпоширеніших вікна — це ті, що походять від Panel , яка використовується аплетами, і ті, що походять від Frame , який створює стандартне вікно програми	Вживається мовою оригіналу
147.	Component is an abstract class that encapsulates all of the attributes of a visual component.	Компонент — це абстрактний клас, який інкапсулює всі атрибути візуального компонента.	еквівалент
148.	The Graphics class defines a number of drawing functions.	Клас Graphics визначає ряд функцій малювання.	Вживається мовою оригіналу
149.	Synchronizers offer high-level ways of synchronizing the interactions between multiple threads.	Синхронізатори пропонують високорівневі способи синхронізації	еквівалент

		взаємодії між кількома потоками.	
150.	Executors manage thread execution	Екзекутори управляють виконанням потоків	транскодування
151.	One of the most interesting features of the concurrent API is the Callable interface.	Однією з найцікавіших особливостей паралельного API є інтерфейс Callable.	еквівалент
152.	A Java Bean is a software component that has been designed to be reusable in a variety of different environments.	Java Bean — це програмний компонент, розроблений для багаторазового використання в різноманітних середовищах.	Вживається мовою оригіналу
153.	A Bean may be visible to an end user	Bean може бути видимим для користувачів клієнта	описовий
154.	The properties , events, and methods of a Bean that are exposed to another application can be controlled.	Можна керувати властивостями, івентами та методами Bean, доступними для іншої програми.	еквівалент
155.	A method in this interface is invoked when a bound property is changed.	Метод у цьому інтерфейсі викликається, коли змінюється зв'язана властивість.	еквівалент
156.	The Introspector class provides several static methods that support introspection.	Клас Introspector надає кілька статичних методів, які підтримують самоаналіз.	Вживається мовою оригіналу
157.	When the DownloadManager is instantiated, all of the GUI's	Коли DownloadManager створюється, усі елементи керування GUI	Вживається мовою оригіналу

	controls are initialized inside its constructor	ініціалізуються в його конструкторі	
158.	Now that you've created, merged, and deleted some branches , let's look at some branch-management tools that will come in handy when you begin using branches all the time.	Тепер, коли ви створили, об'єднали та видалили деякі гілки, давайте розглянемо деякі інструменти керування гілками, які стануть у нагоді, коли ви почнете постійно користуватися гілками.	еквівалент
159.	PooledDataBuffer is an extension of DataBuffer that helps with reference counting which is essential for byte buffer pooling .	PooledDataBuffer — це розширення DataBuffer, яке допомагає підраховувати посилання, що є важливим для об'єднання байтових буферів.	калькування
160.	When you are planning to transfer large amounts of data for a short period of time, for example, for a data migration project.	Якщо ви плануєте передати великі обсяги даних протягом короткого періоду часу, наприклад, для проекту міграції даних.	калькування