

## ЯКІСТЬ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

УДК 004.05

**Новицький О.В., Проскудіна Г.Ю.,  
Резніченко В.А., Овдій О.М.**  
Інститут програмних систем  
НАН України

# ОЦІНЮВАННЯ ЯКОСТІ ЕЛЕКТРОННИХ БІБЛІОТЕК В ВЕБ- СЕРЕДОВИЩІ

*В статті розглядається проблема оцінювання якості електронних бібліотек. За основу набору критеріїв для оцінки ЕБ взята модель DELOS DLRM. Для більшості параметрів якості запропоновано метрики, які дозволяють дати кількісну оцінку. Вибір метрик був запропонований виходячи з того, що ЕБ функціонує в веб-середовищі. Цей підхід є новим стосовно DELOS DLRM, оскільки в цій моделі не робиться акцентів, щодо функціонування ЕБ в веб-середовищі. В роботі запропонована формальна модель для оцінки зручності використання, яка базується відображення поведінки користувача на сайті через зміну станів системи. Запропонована нами методика дозволяє проводити таку оцінку більш незалежно, при цьому використовуються статистичні показники. Окрім цього запропоновані метрики для оцінювання параметрів безпеки ЕБ в веб-середовищі.*

*В статье рассматривается проблема оценки качества электронных библиотек. За основу набора критериев для оценки ЭБ взята модель DELOS DLRM. Для большинства параметров качества предложено метрики, которые позволяют дать количественную оценку. Выбор метрик был предложен исходя из того, что ЭБ функционирует в веб-среде. Этот подход является новым по отношению DELOS DLRM, поскольку в этой модели не делается акцент на, относительно функционирования ЭБ в веб-среде. В работе предложена формальная модель для оценки удобства использования, основанная отражение поведения пользователя на сайте через смену состояний системы. Предложенная нами методика позволяет проводить текущую оценку более независимо, при этом используются статистические показатели. Кроме этого предложены метрики для оценки параметров безопасности ЭБ в веб-среде.*

*The article considers the problem of assessing the quality of digital libraries. The basis for a set of criteria to assess the DL is modeled DELOS DLRM. For most quality parameters proposed metrics that allow a quantitative assessment. Selection metrics has been proposed based on the fact that EB operates in the web environment. This approach is new in relation DELOS DLRM, since this model does not focus, on the operation of digital libraries in the web environment. The paper presents a formal model for usability evaluation based reflection of user behavior on the site through a change of the system states. The proposed technique allows ongoing assessment of more independently, using the statistics. Also proposed metrics to assess security settings DL in the web environment.*

**Ключові слова:** *якість електронної бібліотеки, зручність використання сервісів, модель DELOS.*

### Вступ

Електронна бібліотека (ЕБ) як інформаційна система повинна задовольняти основні потреби користувачів. Під користувачами ми розуміємо особу, організації чи сторонній програмний сервіс, який використовує електронну бібліотеку для конкретної функції. Для того щоб дати відповідь на питання на скільки ЕБ задовольняє вимогам якості, ми повинні визначитися з переліком вимог, яким повинна відповідати ЕБ. Окрім визначення переліку вимог, необхідні метрики для оцінювання даних вимог. На сьогодні відсутності чіткі відповідні стандарти та нормативні документи, тому проблематика якості електронної бібліотеки досліджується в багатьох напрямках.

Існує багато розроблених методів оцінювання якості програмних продуктів [1], [2]. Проте ці методи відносяться як правило до оцінки якості програмного забезпечення (ПЗ) загалом. В той же

час ЕБ мають свою специфіку, яка полягає в певних особливостях інформаційних об'єктів, якими оперує ЕБ, а також набору програмних сервісів, які виконуються в середовищі ЕБ. Зважаючи на складність об'єкту дослідження за основу була прийнята модель ЕБ [3], оскільки саме в цій моделі належна увага приділена питанню якості. Зокрема в цій моделі вводиться ряд характеристик, які підлягають оцінюванню. В даній роботі проводиться аналіз цих параметрів та надаються конкретні метрики, а також пропонуються деякі покращення стосовно їх визначення. До уваги було прийнято підхід, що сучасна ЕБ функціонує в веб середовищі. Це накладає специфічні вимоги до моделі якості. Зокрема на основі уточнень характеристик пропонується такі параметри, як пропускна здатність, віднести не до функціональних характеристик, а до загальних параметрів якості системи.

### Огляд досліджень з проблем якості ЕБ

В електронних бібліотеках поняття якості може стосуватися різноманітних компонентів [4].

Такими компонентами є метадані, сервіси користувача, зручність інтерфейсів, якість програмного коду, якість інфраструктури, якість процесів управління ЕБ. Одним із самих важливих складових ЕБ є метадані, що описують інформаційні ресурси ЕБ. Зокрема в [5] розглядається проблема визначення якості метаданих. Незважаючи на актуальність та важливість даної проблеми на даний момент нема загально прийнятого підходу до поняття якості метаданих. В наслідок цього відсутні загально прийняті метрики для оцінки цього параметру. Недостатність оцінювання якості метаданих призводить до ускладнення перенесення таких метаданих між інформаційними системами. Бо коли постає питання інтеграції електронних бібліотек якісь метаданих починає відігравати значну роль [6]. Адже при переході між системами як правило якість метаданих знижується. Оскільки метрики якості, які були прийняті в початковій системі, як правило не відповідають критеріям якості в інтегрованій системі. Тому початково якісні дані при інтеграції втрачають якість. Можемо зробити висновок, що при втраті якості ми втрачаємо зміст цих метаданих тобто їх семантична значущість знижується. В різних роботах [7] [8] було запропоновано різні характеристики та метрики якості метаданих такі як повнота, точність, відповідність тощо.

В [9] була зроблена спроба розробити модель зручності використання ЕБ. Зокрема робиться висновок, що зручність використання програмного продукту є складною характеристикою. Ця зручність є певним балансом між функціональними можливостями програмного забезпечення та простотою використання даного функціоналу через інтерфейси взаємодії. Зручність використання лежить в площині взаємодії користувача та машини, і включає в себе ряд характеристик, таких як інтуїтивність інтерфейсу, технології реалізації, можливості, швидкості досягнення поставленої цілі тощо.

Актуальність вирішення проблеми якості електронних бібліотек обумовлюється масштабною їх створення та розповсюдження. Здатність електронних бібліотек утворювати, зберігати та розповсюджувати різноманітну інформацію в великих обсягах у зручному для користувача вигляді через глобальні мережі

передачі даних є передумовою для розгляду електронної бібліотеки як засобу для підвищення ефективності вирішення задач наукового, освітянського та культурного характеру.

Загальні вимоги, що висуваються до характеристик ЕБ наступні:

- вона повинна мати можливість кількісного або якісного вимірювання;
- вона має бути відкритою, тобто повинна допускати можливість внесення або виключення з неї окремих характеристик.

Метрики мають будуватися, виходячи з вимог:

- специфічності, як здатності відбивати фундаментальні відмінності вимірюваних елементарних характеристик;
- ефективності, як можливості забезпечувати найбільш економний вимір елементарних характеристик;
- можливості допускати різну точність оцінок якості ЕБ відповідно до висунутих їй вимог;
- розширюваності в залежності від вимог, що висунуті до повноти і глибини оцінювання якості ЕБ;
- технологічності використання.

### Поняття якості програмного забезпечення для управління ЕБ

Якість програмного забезпечення відіграє важливу роль в процесах життєвого циклу. З метою визначення кількісних показників якості проводять оцінку програмного забезпечення. Оцінка здійснюється за допомогою метрик. Метрика – кількісне значення міри параметрів системи, компонентів чи процесів які володіють певною властивістю. За їх допомогою оцінюють властивості складових програмного забезпечення (продуктів, процесів). Для програмних продуктів розроблено ряд метрик, які дозволяють оцінювати їх характеристики.

Наприклад в Software Engineering Body of Knowledge (SWEBOK) [10] виділяється шість базових характеристик якості програмного забезпечення:

- функціональність;
- надійність;
- зручність використання;
- ефективність;
- супроводжуваність;
- портативність.

Кожна з цих характеристик визначається властивостями та атрибутами (табл. 1).

Опис	Властивість визначення характеристик
Функціональність	Група властивостей програмного забезпечення, визначає його можливість виконувати функції у відповідності з вимогами при заданих умовах.
Надійність	Група властивостей, які описують здатність програмного продукту безвідмовно виконувати певні функції при заданих умовах протягом заданого періоду часу з досить великою ймовірністю.
Зручність використання	Набір властивостей відображають легкість з якою користувач може навчитися керувати, готувати дані для введення й інтерпретувати результати роботи системи або компонента.
Супроводжуваність	Група властивостей, що визначає зусилля, необхідні для виконання довгострокової підтримки і можливості виконувати коригування у зв'язку зі зміною стану навколишнього середовища.
Ефективність	Група властивостей, що характеризується як ступінь відповідності функціональних можливостей потребам предметної області.
Портативність	Група властивостей для забезпечення передачі програмного продукту з одного операційного оточення на інше через адаптації.

З метою оцінки якості програмних продуктів для ЕБ ми повинні мати певний набір характеристик. В якості бази таких характеристик ми обрали еталонну модель DELOS (Digital Library Reference Model, DLRM) [3]. Для того щоб оцінити певний програмний продукт для побудови ЕБ необхідно ввести ряд метрик для оцінювання кожного параметра в рамках обраної моделі.

Модель якості DELOS DLRM являє собою набір параметрів, які можна використовувати для характеристики і оцінки як сервісів електронної бібліотеки в цілому так і кожних структурних компонентів моделі окремо. Цей набір параметрів пов'язаний з іншими головними структурними компонентами моделі ЕБ DELOS, такими як Контент, Користувач, Функціональність, Політика та Архітектура, і вони в свою чергу можуть мати свої якісні атрибути. Тому якість може бути пов'язана не тільки з кожним набором інформації або множиною функцій, але і з конкретними інформаційними об'єктами або сервісом. Деякі з цих параметрів є кількісними і мають об'єктивний характер та можуть бути виміряні абсолютно, тоді як інші є якісними і суб'єктивними характеристиками та можуть

бути виміряні лише за допомогою оцінок користувачів.

Модель виділяє наступні групи якостей ЕБ: загальні параметри якості, параметри якості контенту, параметри якості політик, функціональні параметри якості, параметри якості користувача, архітектурні параметри якості.

Розглянемо якісні параметри для різних аспектів ЕБ відповідно до моделі DELOS DLRM.

#### Загальні параметри якості

Загальні параметри якості – це ті, що стосуються системи в цілому. Такі параметри якості важливі, оскільки вони дають можливість визначити якість системи загалом не використовуючи глибоко дослідження вузьких внутрішніх характеристик. З іншої сторони ці параметри доволі легко оцінити. Серед загальних параметрів якості розрізняють ті, що зображені на рис. 2.

Репутація (Reputation) – відображає ступінь надійності ЕБ, стосується «доброго імені» ЕБ, що має кредит довіри від співтовариства користувачів, а також володіє достовірною інформацією.

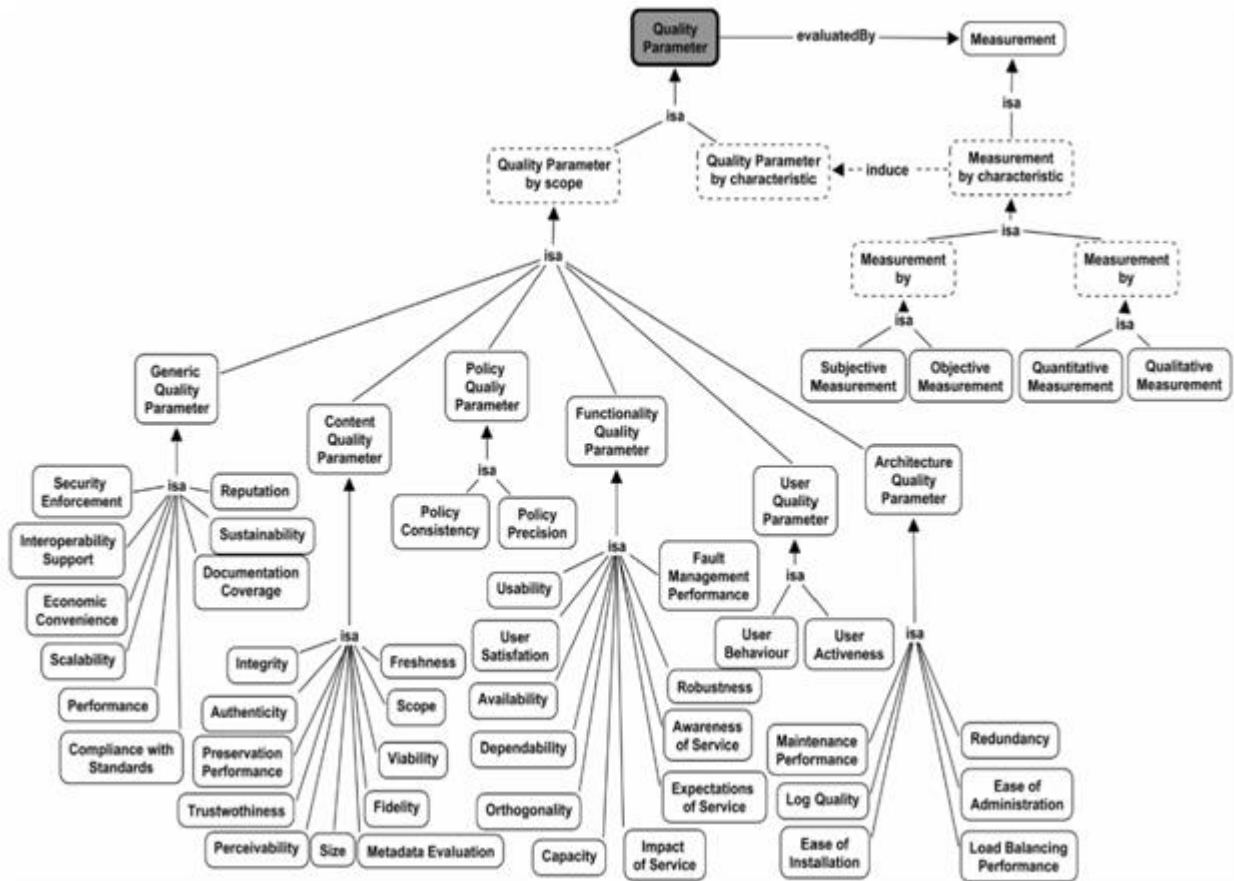


Рис. 1 Ієрархія параметрів якості ЕБ моделі DELOS DLRM.

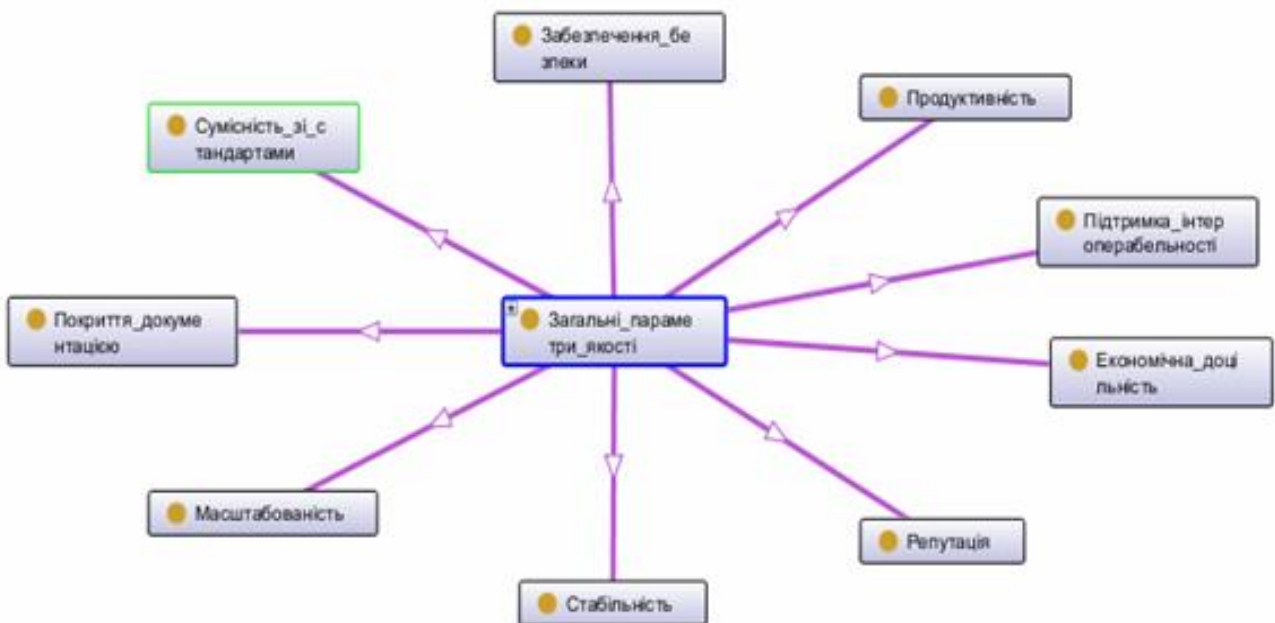


Рис. 2 Загальні параметри якості

Економічна доцільність (Economic Convenience) – відображає економічну ефективність використання ЕБ. Цей параметр оцінює економічні умови використання ЕБ, що визначає, чи достатньо вона економічно вигідна.

Стабільність (Sustainability) – відображає перспективи довгострокової працездатності та подальшого розвитку ЕБ. Стабільність повинна брати до уваги різні фактори, такі як організаційні та економічні аспекти ЕБ, а також гарантію довгострокового збереження контенту з урахуванням майбутніх інновацій.

Прикладами факторів, що впливають на стабільність є: схема фінансування, що забезпечує економічні умови підтримки; навички та готовність персоналу у рамках організації, яка забезпечує ЕБ; наявність точних планів розвитку колекцій, а також стосується програмного та апаратного забезпечення.

Забезпечення безпеки (Security Enforcement) – відображає здатність ЕБ по підтримці різних рівнів безпеки, наприклад, на рівні користувача, обмеження прав доступу до контенту та інш.

Підтримка інтероперабельності (Interoperability Support) – стосується можливості взаємодії з іншими ЕБ, а також інтеграції з існуючими системами. Відповідність стандартам суттєво впливає на підтримку інтероперабельності, оскільки їх використання дозволяє легше взаємодіяти з іншими системами. Прикладами таких стандартів є протокол OAI-PMH для обміну даних або протокол SRU/SRW для пошуку сервісів.

Покриття документацією (Documentation Coverage) – вимірює повноту, точність та ясність документації, яка описує дану ЕБ. Цей параметр охоплює документацію програмного коду, документацію для розробника, документацію для користувачів різних груп та інш. Цей параметр є важливим попри те що зазвичай він недооцінюється. Повноцінна документація надає можливість оптимально використовувати наявні ресурси.

Продуктивність (Performance) – забезпечує загальну оцінку, наскільки ефективно, швидко та результативно функціонує ЕБ. Наприклад, розрізняють наступні параметри продуктивності: час відгуку – час, виконання запиту, затримку – час, необхідний для

отримання першого байта в рамках відповіді, максимальну кількість активних користувачів.

Сумісність зі стандартами (Compliance With Standard) – відображає підтримку стандартів при проектуванні, реалізації та обслуговуванні ЕБ. Цей параметр може бути застосований до будь-якого аспекту ЕБ, а саме контенту, функцій і т.д. Він впливає на підтримку інтероперабельності, оскільки підвищує легкість взаємодії з іншими системами, а також на стабільність ЕБ, оскільки залишає її актуальною з точки зору розвитку майбутніх технологій. Наприклад, використання у ЕБ стандарту метаданих Дублінського Ядра (ДЯ) [15] для опису контенту суттєво підвищує цей показник.

Масштабованість (Scalability) – відображає здатність системи при необхідності збільшити свої робочі навантаження. В контексті ЕБ масштабованість означає здатність системи до розміщення все більшого числа об'єктів, високопродуктивну обробку зростаючих обсягів об'єктів та/або розширення мережі, системи або процесів. Наприклад, здатність системи ЕБ підтримувати зростаюче число користувачів без погіршення її продуктивності або здатність до збільшення числа запитів і отримання на них відповіді за прийнятний час.

Розглянемо більш детально деякі найбільш важливі параметри та введемо для них метрики. Всі подальші матеріали в статті стосуються веб-орієнтованої ЕБ.

#### **Забезпечення безпеки**

Забезпечення безпеки – цей параметр відображає здатність електронної бібліотеки до підтримки управління різними рівнями безпеки, на рівні користувача, доступу до контенту та обмеження прав доступу.

Для параметрів безпеки ми вводим наступну метрику  $S = \sum_{i=1} s_{ai}$ , де  $s_{ai}$  - показник ступеню вразливості.

Основний протокол взаємодії з веб-додатками є передача HTTP запитів. Зважаючи на відкритість та доступність виявлення вхідних точок для відправки HTTP запитів а також складність інформаційної систем ЕБ необхідно значну увагу приділяти оцінці безпеки ЕБ.

За даними [11] виділяється наступний ряд головних потенційних небезпек у веб застосуваннях, зокрема і ЕБ (табл. 2):

Таблиця 2  
Головні потенційні небезпеки у веб застосуваннях

Вид уразливості	Опис	Метрика
1	2	3
A1 – Injection	Ін'єкційні дефекти, такі як SQL, OS і LDAP ін'єкції, відбувається коли ненадійні дані передаються на інтерпретатор як частина команди або HTTP запиту. Ворожі дані атакуючого можуть ввести в оману інтерпретатор що в свою чергу призведе до виконання ненавмисної команди або доступу до даних без належного дозволу.	$s_{a1} = n_{a1} / N$ де $n_{a1}$ – кількість фактів компрометування системи ін'єкційними виразом, $N$ – загальна кількість спроб скомпрометувати систему.
A2 – Broken Authentication and Session Management	Прикладні функції, пов'язані з аутентифікацією та управлінням сеансами часто не виконуються правильно, що дозволяє зловмисникам скомпрометувати паролі, ключі або сесії користувача і використовувати інші недоліки реалізації з метою допустити ідентичність неавторизованих користувачів.	$s_{a2} = n_{a2} / N$ де $n_{a2}$ – кількість фактів компрометування системи, $N$ – загальна кількість спроб скомпрометувати систему.
A3 – Cross-Site Scripting (XSS)	Загроза XSS полягає в тому, що додаток отримує ненадійні дані і відправляє їх в веб-браузер без відповідної перевірки. XSS дозволяє зловмисникам виконувати скрипти в браузері жертви, які можуть викрасти користувача сесій, спотворювати веб-сайти або перенаправити користувача на шкідливі сайти.	$s_{a3} = n_{a3} / N$ де $n_{a3}$ – кількість фактів компрометування системи шляхом атаки XSS, $N$ – загальна кількість спроб скомпрометувати систему.
A4 – Insecure Direct Object References	Пряме посилання на об'єкт виникає, коли розробник надає посилання на внутрішній об'єкт реалізації, наприклад, файл, каталог або ключ бази даних, без перевірки контролю доступу або інших засобів захисту. Зловмисник може використати ці посилання для доступу до захищених даних.	$s_{a4} = n_{a4} / N$ де $n_{a4}$ – кількість фактів компрометування системи шляхом атаки XSS, $N$ – загальна кількість спроб скомпрометувати систему.
A5 – Security Misconfiguration	Якісна безпека вимагає наявність безпечної конфігурації для фреймворків розробки, сервера додатків, веб-сервера, сервера баз даних і платформи. Безпечні параметри повинні бути визначені, реалізовані і підтримуватися, оскільки як правило налаштування за замовчуванням часто не є безпечними. Крім того, програмне забезпечення має бути в актуальному стані.	$s_{a5} = \frac{\sum_{k=1} q_k}{T}$ де $q_k$ – кількість відомих потенційних вразливосте компонентів системи, $T$ – загальна кількість компонентів системи

1	2	3
A6 – Sensitive Data Exposure	Багато веб-додатків не правильно захищають конфіденційні дані, такі як кредитні карти, ПІН та дані аутентифікації. Зловмисники можуть вкрасти або модифікувати такі слабо захищених дані для проведення шахрайства з кредитними картками, крадіжки особистих даних, а також інші злочини. Конфіденційні дані заслуговують додаткового захисту, такого як шифрування, а також спеціальних застережних заходів при обміні з браузером.	$s_{a6} = \sum_{i=1} d_i / R$ <p>де <math>d_i</math> – кількість конфіденційних даних які пов'язана з певними інформаційний об'єктом і не захищені відповідним ступенем безпеки, <math>R</math> – загальна кількість об'єктів .</p>
A7 – Missing Function Level Access Control	Більшість веб-додатків перевіряють права доступу до функціонального рівня, перш ніж приймати рішення, що функціональність буде представлено в інтерфейсі. Проте, додаток повинен виконувати ті ж перевірки контролю доступу на сервері. Якщо запити не перевіряється, зловмисники зможуть підробити запити для доступу до функціональності без належного дозволу.	$s_{a7} = n_{a7} / N$ <p>де <math>n_{a7}</math> – кількість фактів компрометування системи шляхом атаки, <math>N</math> – загальна кількість спроб скомпрометувати систему.</p>
A8 – Cross-Site Request Forgery (CSRF)	Атака типу CSRF змушує браузер жертви, відправити сфальсифіковані HTTP запити, включаючи cookie сесії і будь-яку іншу автоматично включену ідентифікаційну інформацію до вразливого веб-додатку.	$s_{a8} = n_{a8} / N$ <p>де <math>n_{a8}</math> – кількість фактів компрометування системи шляхом атаки CSRF <math>N</math> – загальна кількість спроб скомпрометувати систему</p>
A9 – Using Components with Known Vulnerabilities	Компоненти програмної системи, такі як прикладні програмні модулі, бази даних, майже завжди працюють з повними привілеями. Якщо виконати атаку на такий уразливий компонент, це може призвести до серйозних втрат даних або захоплення сервера.	$s_{a9} = n_{a9} / N$ <p>де <math>n_{a9}</math> – кількість компонентів програмної системи, які працюють з повними привілеями, <math>N</math> – загальна кількість компонентів програмної системи.</p>
A10 – Unvalidated Redirects and Forwards	Веб-додатки часто перенаправляють користувачів на інші сторінки та веб-сайти, а також використовують ненадійні дані, щоб визначити, сторінки призначення. Без належної перевірки зловмисники можуть перенаправити жертву на сайти фішингу або шкідливий сайт.	$s_{a10} = n_{a10} / N$ <p>де <math>n_{a9}</math> – кількість фактів компрометування системи шляхом атаки пере направлення, <math>N</math> – загальна кількість спроб скомпрометувати систему.</p>

Для тестування цих параметрів якості ЕБ можливо застосовувати два методи: методи білого ящика та методи чорного ящика.

При тестуванні білого ящика (white-box testing), розробник тесту має доступ до вихідного коду і може писати код, який пов'язаний з бібліотеками тестованого ПЗ. Це

типово для юніт-тестування, при якому тестуються тільки окремі частини системи. Також даним методом можна встановити параметри A5, A6, A9.

При тестуванні за методикою чорного ящика (black-box testing), тестувальник має доступ до ПЗ тільки через ті ж інтерфейси, що і замовник або користувач або через зовнішні інтерфейси, що дозволяють іншому комп'ютеру або іншому процесу підключитися до системи для тестування. Таким підходом можливо визначити параметри A1, A2, A3, A4, A7, A8, A10.

#### **Підтримка інтероперабельності**

Ще одним важливим компонентом якості ПЗ є підтримка інтероперабельності. Цей параметр стосується можливості взаємодії з іншими електронними бібліотеками, а також можливість інтеграції з існуючими системами та рішеннями. Політики доступу до контенту та сервісів ЕБ можуть вплинути на підтримку інтероперабельності, оскільки політика визначає і контролює, як і в якій мірі електронна бібліотека повинна бути доступна.

З точки зору систематичності можна виділити кілька аспектів інтероперабельності:

1) функціональна інтероперабельність. Може розглядатися з різних сторін. Це може бути просто обмін та розповсюдження цифрового контенту. Інший підхід, дозволяє отримувати функціональне вирішення проблеми об'єднання цифрових об'єктів в загальному шарі контенту. На рівні міжсистемної взаємодії функціональна інтероперабельність вимагає надання користувачам або програмним додаткам можливості взаємодії з декількома електронними бібліотеками через уніфіковані інтерфейси (динамічні портали);

2) лінгвістична інтероперабельність (багатомовність). Може розглядатися двома різними способами: як багатомовні інтерфейси користувача до електронних бібліотек або як динамічні багатомовні методи формування інформаційних об'єктів. Тут можна виділити три типи підходів: динамічний переклад запиту на різних мовах, динамічна трансляція метаданих для відповіді на запити на різних мовах або динамічна локалізація цифрового контенту;

3) технологічні стандарти, що дозволяють різні види взаємодії і мають значну кількість реалізацій спрямованих на сумісність метаданих, такі як Z39.50/SRU+SRW, методи збору на основі OAI-PMH, підходи засновані на веб-сервісах

(SOAP/UDDI), стандартизовані в JCR (JSR 170/283) інтерфейси API;

4) семантична інтепарабельність – здатність двох або більше систем або компонентів обмінюватись інформацією, а також використовувати цю інформацію. Термін інтероперабельність є широко застосовним, зокрема, в питаннях ефективного співіснування інформаційних ресурсів. Семантична гетерогенність, як правило, відрізняється від синтаксичної гетерогенності та структурної гетерогенності.

5) синтаксична інтероперабельність пов'язана з неоднорідністю форматів даних. Стандартизація форматів даних приймається в якості підходу до вирішення проблем синтаксичної неоднорідності. Наприклад, XML використовується в якості стандартного формату для всіх видів веб-доступних даних.

6) структурна інтероперабельність пов'язані з різними моделями даних, структурами даних або схемами, наприклад, реляційних і об'єктно-орієнтованих моделей бази даних. Прикладом рішення проблеми структурної неоднорідності є використання формату RDF, який заснований на синтаксисі XML, що забезпечує уніфікований спосіб структурування джерел інформації.

Об'єктивної метрики для оцінки інтероперабельності нема. В якості суб'єктивної оцінки ми пропонуємо оцінювати наявність чи відсутність того чи іншого показника.

#### **Покриття документацією**

Під покриттям документацією розуміють параметри вимірювання точності і ясності документації, яка описує даний ресурс. Для оцінки програмного продукту вводять параметри, які відображають ступінь якості документації. Цей аспект покриває документацію програмного коду, документацію для розробника, документацію для користувачів різних груп. Цей параметр є важливим попри те що зазвичай цей параметр недооцінюється. Навпаки, повноцінна документація надає можливість оптимально використовувати наявні ресурси.

#### **Швидкодія або продуктивність**

Параметр забезпечує загальну оцінку, наскільки швидко ресурс опрацьовує запити користувачів. Цей показник є важливим для веб-орієнтованих середовищ, оскільки безпосередньо впливає на час очікування користувача при отриманні інформації з системи [12].



До параметрів швидкодії ми відносимо і пропускну здатність, яка визначає скільки одночасних запитів може бути успішно оброблено. Цей параметр може вплинути на доступність, надійність і продуктивність програмного забезпечення. Низька надійність і продуктивність може мати негативні наслідки, наприклад якщо функція не завершить виконання своїх завдань або їх виконання

займає надто багато часу. Пропускна здатність має також зв'язок з кількістю інформаційних об'єктів, що компонент ЕБ може індексувати в певну одиницю часу або максимальною кількістю користувачів, які можуть одночасно підключатися до порталу ЕБ.

Для веб-порталів більш важливими критеріями є наступні (табл. 3):

Таблиця 3  
Критерії веб-порталів

Час відгуку	Час, необхідний для повного виконання запиту. Визначається як різниця між ініціюванням запиту до часу його завершення. Як правило, показує рівень продуктивності всієї системи (веб-сервера, веб-додатку, бази даних). Цей показник найкраще вимірювати як середній час відгуку за певну кількість випробувань.
Затримка	Час до першої відповіді. Час, необхідний для отримання першого байта в рамках відповіді. Як правило, це вказує на рівень продуктивності веб-сервера. Цей показник являє собою середній час затримки при певній кількості тестів.
Користувачі	Цей показник враховує максимальну кількість одночасно працюючих на сайті користувачів.

### Сумісність зі стандартами

Ця вимога стосовно якості відноситься до будь-якої складової у ЕБ. Цей параметр відображає підтримку стандартів при розробці або реалізації ресурсу. Оскільки ЕБ є веб-орієнтованою системою, то цей параметр стосується підтримки стандартів W3C та відкритих стандартів.

Можна виділити наступні рівні сумісності зі стандартами:

- Рівень інтерфейсу користувача. Для веб-орієнтованих систем важливим параметром є підтримка валідації розмітки веб-документів. Сучасний стан справ вимагає підтримку стандартів HTML5, HTML4, CSS3 тощо.
- Рівень протоколу обміну метаданими. Для ЕБ побудованих в рамках ініціативи відкритого доступу важливою вимогою є підтримка протоколу OAI-PMH.
- Рівень пошуку інформаційних ресурсів. Оскільки мережа Інтернет це розподілене середовище, то часто виникають задачі розподіленого пошуку, для таких задач

існують протоколи SRU. В якості базових показників оптимізації є наявність карти сайту у відповідності з стандартом [16]. З точки зору семантичного веб важливим компонентом є використання мікроформатів розмітки для інформаційних ресурсів представлених в ЕБ. До таких форматів відноситься RDFa, що є рекомендацією W3C і надає набір атрибутів, які розширюють HTML та різні типи XML-документів з метою їх збагачення метаданими.

### Функціональні параметри якості

Стосуються функціонального аспекту ЕБ, серед яких розрізняють, ті що зображені на рис. 3.

Доступність (Availability) – параметр показує відношення часу, коли функція доступна до використання до загального часу існування системи;

Обізнаність про сервіси (Awareness of Service) – показує наскільки користувач ЕБ знає саму систему та її сервіси.

Потужність (Capacity) – параметр показує мінімальну кількість запитів, що виконує

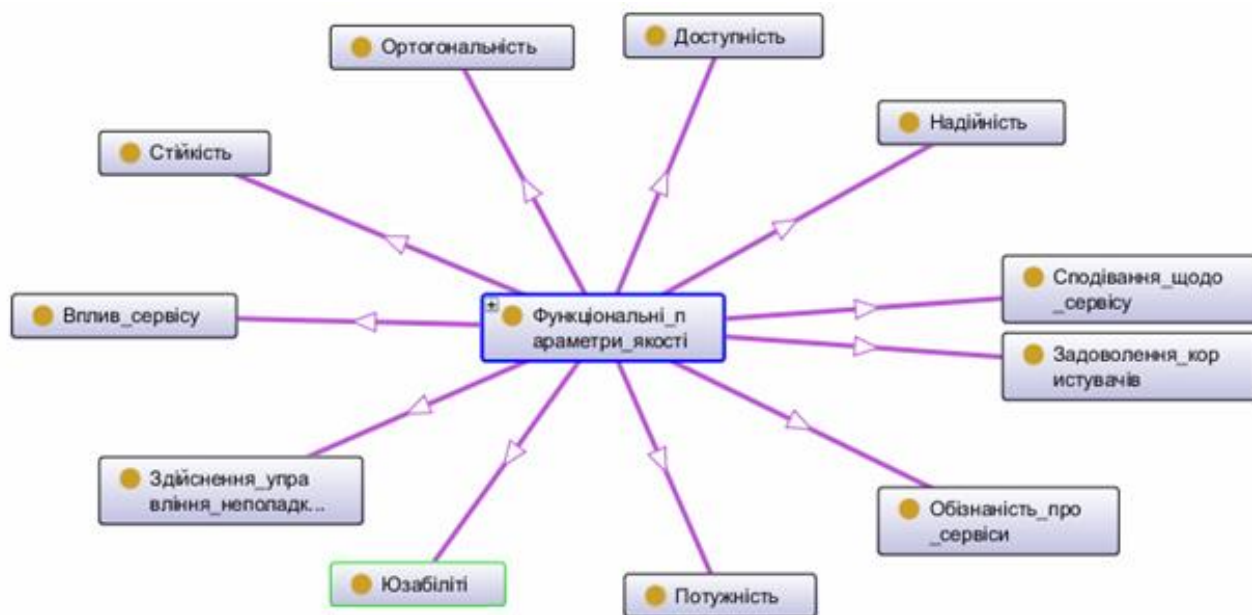


Рис. 3 Функціональні параметри якості

служба ЕБ за даний інтервал часу або кількість інформаційних об'єктів, яких компонент доступу до інформації може проіндексувати за даний час або максимальну кількість користувачів, що можуть одночасно підключитися до ЕБ.

Сподівання щодо сервісу (Expectations of Service) – параметр, який вимірює очікування користувача щодо заявленого сервісу, оскільки користувачі можуть мати різні очікування виходячи зі свого досвіду з іншими ЕБ або сервісами. Сподівання користувачів можуть бути вивчені за допомогою опитувань.

Ефективність управління неполадками (Fault Management Performance) – параметр вимірює здатність системи зреагувати на помилку та відновити свою функцію після збою.

Вплив сервісу (Impact of Service) – параметр показує, як сервіс впливає на знання та поведінку користувача.

Ортогональність (Orthogonality) – параметр показує наскільки функції є незалежними одна від одної, тобто не впливають одна на іншу. Наприклад функції управління інформаційними об'єктами та управління користувачами повинні бути ортогональними.

Надійність (Dependability) – параметр оцінює здатність ЕБ виконати функцію при певних умовах та у визначений час.

Стійкість (Robustness) – параметр вимірює гнучкість до невірно сформованого вводу інформації або помилкової послідовності виклику функцій.

Юзабіліті або зручність використання (Usability) – параметр, що показує простоту використання функцій. Стосується різних аспектів ЕБ, починаючи від інтерфейсу користувача, до засобів пошуку та доступу до релевантної інформації, представлення результатів пошуку,

підтримки складних завдань, таких як надання функціональності пошуку по зразку або функцій навігації для складних схем метаданих.

Задоволеність користувачів (User Satisfaction) – показує ступінь задоволеності від виконання певної функції. Параметр може визначатися явно шляхом використання досліджень та опитувань користувачів, або неявно шляхом спостереження того, на скільки часто використовується та чи інша функція.

#### Доступність

Доступність визначає частку часу протягом якого система є функціональною та доступною. Вона може бути виміряна як відсоток від загального простою системи протягом заданого періоду. Доступність може бути порушена внаслідок системних помилок програмного забезпечення, проблеми інфраструктури, атак або завантаження системи. Основними шляхами підвищення доступності системи є наявність гнучкого методу виявлення помилок під час роботи системи. Також важливим моментом є використання надійної інфраструктури (розміщення ЕБ в центрах обробки даних). Одним із методів виключення системних помилок є функціональне автоматичне тестування програмного коду, наприклад, за допомогою RHPUnit

#### Зручність використання

Електронна бібліотека як програмна система повинна задовольняти вимоги користувачів не тільки змістом інформаційних ресурсів але й певними сервісами. В роботі [13] розглядається різні аспекти дослідження зручності використання ЕБ. Та в зв'язку зі складністю формалізації оцінки зручності використання поки немає єдиного підходу до цієї проблеми. Ми пропонуємо для оцінки зручності та ефективності

програмного забезпечення використати таку метрику як показник досягнення цілі  $C$ . Цей показник визначається як відношення кількості користувачів, які виконали певні цільові дії до загальної кількості користувачів. Як правило показник  $C$  вираховується за певний проміжок часу.

Для електронної бібліотеки цільові дії можуть бути різного типу. Оскільки сучасна ЕБ має ряд сервісів ЕБ, то слід оцінювати якість програмного продукту з точки зору кінцевого користувача. Для такого користувача можливо виділити цільові дії  $i$  в якості яких фіксувати використання сервісів ЕБ. Досягнення цілі на сайті відбувається через атомарні дії. Під атомарною дією ми будемо розуміти перехід системи з одного стану в інший єдиним можливим способом. Нехай стан системи в певний момент часу буде позначатися через  $s_0(t_0)$ , при певній дії на систему, вона переходить в стан  $s_1(t_1)$ . Атомарна дія це функція  $A = a(s_i(t_i), s_{i+1}(t_{i+1}))$ , яка переводить систему зі стану  $s_i(t_i)$  в стан  $s_{i+1}(t_{i+1})$ . Цілі, які можна досягнути на сайті, позначимо через  $g_i$ ,  $i = 1 \dots n$ . Досягнення кожної цілі можливе через атомарні дії  $g_i = A_1 \rightarrow A_2 \rightarrow \dots A_m$ , де  $m = 1 \dots t$ . Процес досягнення цілі відбувається завдяки впливу користувача на систему  $u_j(A_1, A_2, \dots A_m) \rightarrow g_i$ . З усієї множини користувачів, які відвідали ЕБ  $U = \{u_j | j = \overline{1, k}\}$  можливо виділити множини користувачів, що не перетинаються, за певними ознаками.  $U'_{g_i} = \{\forall u_j, \exists i, u_j \rightarrow g_i | j = \overline{1, k}, i > 0\}$  – група користувачів яка досягла певної цілі. В

загальному випадку показник досягнення для цілі

$g_i$  визначається відношенням  $c_{g_i} = \frac{|U'_{g_i}|}{|U|}$ . Для

загальної оцінки  $C$  використаємо формулу

$$C = \sum_{i=1}^n w \cdot c_{g_i}, \text{ де } w = [0,1] - \text{ваговий показник}$$

який виставляється групою експертів. Проте такий показник в цілому залежить від кількості цілей, для яких він визначається. Тому вводимо ваговий показник досягнення цілі

$$C_{wn} = \frac{\sum_{i=1}^n w \cdot c_{g_i}}{n}, C_{wn} = [0,1].$$

Показник  $C$  є залежним від кількох параметрів, з одного боку він залежить від зручності використання програмного забезпечення, а також від довжини цілі. Проте показники  $C, C_{wn}$  можуть бути використані для оцінки зручності сервісів ЕБ.

Користувачів веб-ресурсів як правило можна розділити на дві великі групи, це нові користувачі та користувачі які постійно використовують ресурс. Показник  $C$  для постійних користувачів є відмінним стосовно нових користувачів.

#### Параметри якості контенту

Параметри, які стосуються контенту, що міститься у ЕБ, серед них розрізняють ті, що зображені на рис.4.

Автентичність (Authenticity) – відображає властивість інформаційних об'єктів бути тим, що декларується тобто їх достовірність. Наприклад, за допомогою електронного підпису можна засвідчувати, що документ не змінювався та є оригінальним.

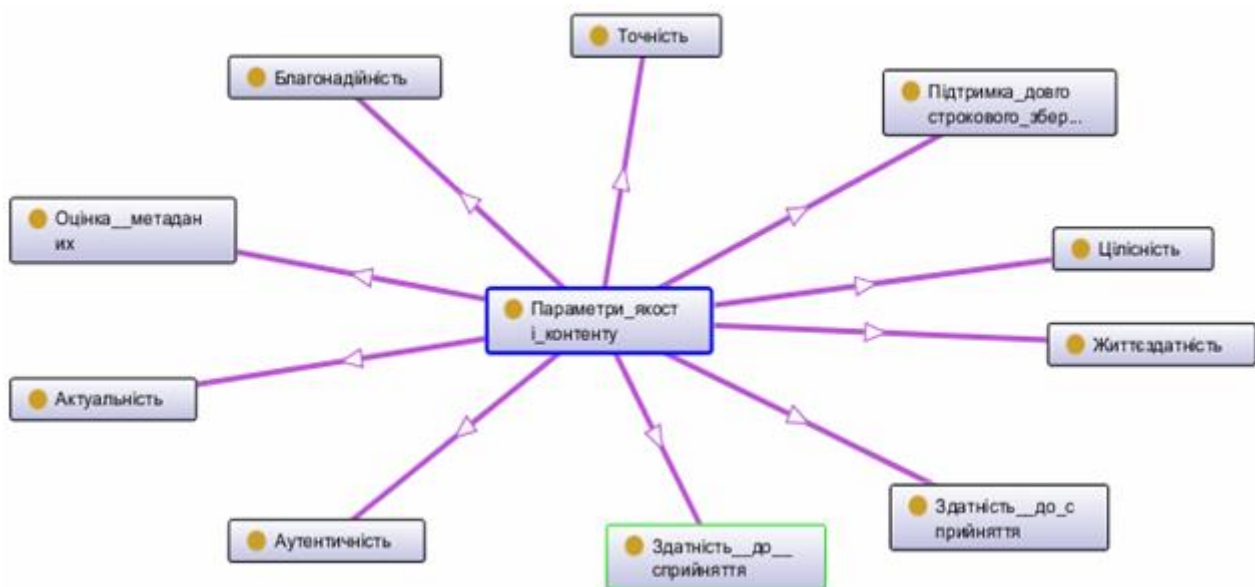


Рис. 4 Параметри якості контенту

Благонадійність (Trustworthiness) – засвідчує довіру до ресурсу на основі надійності творця цього ресурсу.

Актуальність (Freshness) – засвідчує, що контент постійно та своєчасно оновлюється.

Цілісність (Integrity) – засвідчує, що контент є повним та коректним.

Підтримка довгострокового зберігання (Preservation Performance) – можливість маніпуляцій або дій, що слід виконати з даними, щоб гарантувати їх довгострокове зберігання.

Точність (Fidelity) – параметр використовується для оцінки того, в якій мірі окреме подання даного ресурсу відрізняється від його початкового подання (представлення).

Життєздатність (Viability) – параметр, що показує непошкодженість файлів та можливість їх читання або маніпулювання протягом часу за допомогою існуючого програмного та апаратного забезпечення.

Здатність до сприйняття (Perceivability) – параметр, оцінює наскільки легко можна зрозуміти ресурс та як він сприймається користувачем.

Оцінка метаданих (Metadata Evaluation) – вимірює характеристики метаданих. Це є складний показник і може в свою чергу складатися з повноти, точності, походження, відповідності очікуванням, актуальності, задоволеності користувачів, здатності до сприйняття та інш. Ця комбінація залежить від мети оцінки метаданих. Наприклад повнота показує, чи підтримується у записах метаданих їх мінімальний набір.

### Висновки

В статті розглядається модель якості ЕБ, яка охоплює показники якості її ключових аспектів. Проте, незважаючи на широке розповсюдження ЕБ, оцінка їх якості все ще знаходиться на початковій стадії розробки. Для деяких аспектів запропоновані метрики. Зокрема запропоновано формальну метрику для оцінки такого параметру групи якісних функціональних особливостей, як зручність використання. Ця характеристика зазвичай є суб'єктивним показником, а її оцінювання робилося групою спеціалістів. Запропонована нами методика дозволяє проводити таку оцінку більш незалежно, при цьому можливо використовувати статистичні показники, а не суб'єктивні оцінки експертів.

### Список використаних джерел

1. Орлов С. Технологии разработки программного обеспечения: современный курс по программной инженерии : [по специальности "Программное обеспечение вычислительной техники и автоматизированных систем"]. – С.-Пб, 2012. 358 с.
2. Gillies A. Software Quality: Theory and Management (3rd edition), 2011.
3. Candela et al., D3.2b The Digital Library Reference Model, 2010.
4. Bertot J.C. Assessing digital library services: Approaches, issues, and considerations. Intl Symposium on Digital Libraries and Knowledge Communities in Networked Information Society, University of Tsukuba, 2004.
5. Alice Tani L.C.D.C., "Dealing with metadata quality: The legacy of digital library efforts," // Information Processing & Management. –Vol. 49. – No. 6. – 2013.
6. Barton J., Currier S., and Jessie M. N. H. Building Quality Assurance into Metadata Creation: an Analysis based on the Learning Objects and e-Prints Communities of Practice // Dublin Core Conference: Supporting Communities of Discourse and Practice - Metadata Research and Applications. Seattle. 2003.
7. Bruce T.R., Hillmann D.I., "The Continuum of Metadata Quality: Defining, Expressing, Exploiting," // Metadata in Practice., 2004.
8. Когаловский М.Р. Информационные ресурсы, наукометрические показатели и показатели качества метаданных системы Соционет // 9-я Всероссийская научная конференция «Электронные библиотеки: перспективные методы и технологии, электронные коллекции» – RCDL'2007, Переславль-Залесский, Россия, 2007.
9. Jeng J., "What Is Usability in the Context of the Digital Library and How Can It Be Measured," // Information Technology and Libraries. – Vol. 24. – No. 2. – 2005.
10. IEEE Computer Society. Guide to the Software Engineering Body of Knowledge, Version 3.0. IEEE Computer Society, 2014.
11. OWASP. [2014].: <https://www.owasp.org>.
12. Butkiewicz M., V. Madhyastha H., and Sekar V. Understanding Website Complexity: Measurements, Metrics, and Implications // Proceedings of the 2011 ACM SIGCOMM Conference on Internet Measurement Conference. Berlin, 2011. – P. 313 – 328.

13. Jeng J., "What Is Usability in the Context of the Digital Library and How Can It Be Measured," // Information Technology and Librarie. Jeng J., "What Is Usability in the Context of the Digital Library and How Can It Be Measured," // Information Technology and Libraries. – Vol. 24. – No. 2. – 2005.

14. Lenzerini M. Data integration: a theoretical perspective // 21st ACM SIGMOD-SIGACT-SIGART Symposium on Principles of Database Systems (PODS 2002). – New York, 2002.

15. Galanis L., Wang Y., R. Jeffery S., and J. DeWitt D. Locating data sources in large distributed systems. // 29th International Conference on Very Large Data Bases (VLDB 2003). – P. 874 – 885.

16. Manolescu I., Florescu D., and Kossmann D. Answering XML queries over heterogeneous data sources // 27th International Conference on Very Large Data Bases (VLDB 2001). – P. 241 – 250.

### Відомості про авторів:



**Новицький Олександр Вадимович** – молодший науковий співробітник Інституту програмних систем НАН України. Наукові інтереси: електронні бібліотеки, технології Semantic Web, онтології.

**E-mail:** alex@zu.edu.ua



**Проскудіна Галина Юрївна** – науковий співробітник Інституту програмних систем НАН України. Наукові інтереси: бази даних, електронні бібліотеки, інформаційний пошук, технології Semantic Web, онтології.

**E-mail:** gupros@isofts.kiev.ua



**Резніченко Валерій Анатолійович** – провідний науковий співробітник Інституту програмних систем НАН України. Наукові інтереси: бази даних, електронні бібліотеки, інформаційний пошук, технології Semantic Web, онтології.

**E-mail:** vreznichenko\_47@mail.ru



**Овдій Ольга Михайлівна** – молодший науковий співробітник Інституту програмних систем НАН України. Наукові інтереси: електронні бібліотеки, інформаційний пошук, технології Semantic Web, онтології.

**E-mail:** olga.ovdiy@gmail.com