

УДК 005.94 + 004.9 + 519.7

О. Є. СТРИЖАК, М. А. ПОПОВА, К. В. ЛЯШУК

*Інститут телекомунікацій і глобального інформаційного простору НАН України,
Київ, Україна*

МЕТОДИКА СТВОРЕННЯ ОНТОЛОГІЧНОГО ІНТЕРФЕЙСУ У СЕРЕДОВИЩІ WEB-ПОРТАЛУ

У статті розглядаються питання використання методів онтологічного моделювання при проектуванні та побудові науково-освітніх WEB-порталів та підтримки на їх основі досліджень у певних предметних областях. Визначаються структурні компоненти онтологій, які найбільш повно можуть бути представлені у вигляді онтологічного інтерфейсу. Викладено положення автоматизації процесу проектування інтерфейсу користувача на основі онтологічних моделей предметних областей. Запропоновано певні правила створення дизайну інтерфейсу на основі властивостей концептів, які означені в онтології предметної області. Описується методика формування онтологічного інтерфейсу на основі врахування властивостей концептів тематичних онтологій.

Ключові слова: онтологія, властивість, інтерфейс, портал, контент-аналіз.

Вступ

Інтерфейс користувача, (UI – англ. User interface) – різновид інтерфейсів, в якому одна сторона представлена людиною (користувачем), інша – машиною/пристроєм; являє собою сукупність засобів і методів, за допомогою яких користувач взаємодіє з різними, найчастіше складними, машинами, пристроями, апаратурою, програмами та їх додатками.

Найчастіше, або, точніше, майже завжди, щоб успішно користуватися системою (працювати в середовищі Порталу), людині необхідно однозначно розуміти (через інтерфейс взаємодії), як система працює. При цьому необов'язково точно розуміти сутність процесів, що відбуваються в системі, більш того, необов'язково правильно їх розуміти. Це розуміння сутності системи називається ментальною моделлю, яку треба враховувати при розробці інтерфейсу користувача науково-освітнього порталу.

Область знань, яка пов'язана з розробкою користувальницького інтерфейсу, розвивається швидкими темпами. З'являються нові елементи інтерфейсу, розширюються властивості існуючих елементів інтерфейсу, що суттєво спрощує способи взаємодії користувача з програмним засобом, яким він користується. Але завжди залишається проблема адаптації та кастомізації інтерфейсу користувача під швидку зміну версій програмного забезпечення, яке використовується в науковій та дослідницькій діяльності.

Треба звернути увагу на те, що науково-освітній портал створюється як велика онтолого-керована система, яка описує природномовні об'єкти (ПМО), як елементи загальної природномовної картини світу (ПМКС) з відповідними взаємо-

пов'язаними предметними областями (ПДО), що описуються відповідними онтологіями ПДО.

Тому одним із завдань є розгляд сучасних підходів до автоматизованої розробки онтологічного інтерфейсу користувача науково-освітнього порталу, як природно-мовного інтерфейсу (ПМІ), враховуючи те, що онтологія Порталу визначається як опис природно-мовних об'єктів (ПМО) за допомогою онтології предметної області (О ПДО).

1. Онтологічні ознаки інтерфейсу

Доцільно розглянути деякі підходи до автоматизації проектування інтерфейсу користувача на основі знань, які описуються онтологією предметної області [1, 2]. Інструментарій для розробки інтерфейсів на основі онтологій ПДО, засновано на принципі роздільного проектування та реалізації користувальницького інтерфейсу і бізнес-додатків з подальшим їх зв'язуванням, а також автоматичній генерації коду інтерфейсу за його моделлю. Для розробки пропонуються онтології, в термінах яких розробники інтерфейсу формують відповідні компоненти його моделі; генерація інтерфейсу здійснюється кількома мовами програмування на різних платформах; підтримуються як локальна, так і мережева взаємодія з бізнес-додатком; модель інтерфейсу розділена на компоненти відповідно до груп фахівців, що здійснюють його розробку (експертів предметної області, програмістів, дизайнерів) і містить всю інформацію, необхідну для його розробки і автоматичної генерації згідно з моделлю. Таким чином, модель інтерфейсу складається з наступних компонентів [2, 4]:

– моделі системи понять діалогу (описує сис-

тему термінів предметної області, в яких виражаються вхідні/вихідні дані бізнес-додатку, інформація про управління бізнес-додатком та її користувача інтерфейсом, а також про інтелектуальну підтримку дій користувача);

– моделі задач користувача (описують завдання, які він може вирішити, використовуючи програмну систему);

– моделі подання (описує структуру і властивості візуального представлення елементів інтерфейсу);

– моделі сценарію діалогу (визначає безліч можливих станів діалогу і дій, які виконуються);

– моделі зв'язку інтерфейсу з бізнес-додатком (описує спосіб взаємодії інтерфейсу і бізнес-додатків, а також програмні інтерфейси, за допомогою яких забезпечується зв'язок між ними).

Підходи та методи автоматизації проектування інтерфейсу користувача на основі онтології ПдО призначені для вирішення наступних проблем:

– забезпечення підтримки розробки складних і комплексних інтерфейсів (заснованих на великих обсягах моделей понять діалогу і завдань користувача);

– автоматизація розробки інтерфейсу з урахуванням вимог користувачів, їх досвіду, рівня володіння комп'ютером; середовища використання програмної системи; урахування принципів «юзабіліті» (зручності та простоти), різних стандартів розробки та настанов по стилю;

– забезпечення розширюваності для внесення нових знань про проектування, нових керівництв користувачів по стилю і стандартів розробки.

Використання онтологічних моделей щодо формування систем знань з предметних областей, які включаються у середовище науково-освітнього порталу, дозволяє розширити його функціональні можливості та забезпечити реалізацію механізмів семантичного пошуку інформації в мережі Інтернет [6-11]. Комп'ютерну онтологію деякої предметної дисципліни можна розглядати як загальнозначущу, відкриту базу знань, що представлена загальноприйнятою (формальною) мовою специфікації знань. В онтолого-класифікаційній схемі засобів і методів штучного інтелекту онтологічний підхід [8-10] трактується як різновид системного підходу, заснованого на знаннях. Онтологічний підхід забезпечує ефективне проектування компонентів будь-якої знання-орієнтованої інформаційної системи. На відміну від звичайного, суб'єктивного підходу при проведенні контент-аналізу різноманітних документів, системно-онтологічний підхід [9, 10] припускає строгу (наскільки це можливо на даному етапі розвитку науки) структуризацію термінів і понять предметної

дисципліни. Категоріальний рівень представляється онтологією верхнього рівня домена предметних дисциплін. Проектування онтології верхнього рівня повинне бути включене в загальний алгоритм розробки баз знань з кожної предметної області.

При зборі інформації і розробці лінгвістичних ресурсів необхідно враховувати що є об'єкт дослідження – предметна область, її тематичний розділ, процеси, властивості, функціональний опис.

Об'єкт має стан, структуру, властивості, виявляє чітку функціональність, може мати межі.

Група чи множина об'єктів, що мають зв'язки та пов'язані спільною структурою та функціональністю можуть бути об'єднані в класи.

Теми дослідження (предметні області) визначаються поставленими завданнями щодо контент-аналізу, результати якого будуть використовуватися при прийнятті рішення.

Тобто онтологічний інтерфейс забезпечує для користувачів відображення властивостей об'єктів та їх складових як при зборі даних, так і при розробці лінгвістичних ресурсів, які повинні забезпечити здобуття знань про об'єкт.

Основними положеннями автоматизації проектування інтерфейсу користувача є [2, 4]:

1. Розширення загальної моделі користувача інтерфейсу: додавання до моделі інтерфейсу нових компонентів – моделі користувачів і моделі середовища (в нашому випадку – О ПдО) використання програми (для опису різних контекстів її використання).

2. Поділ знань, необхідних для автоматизації проектування, на відповідні групи, що необхідно для подальшої ітераційної послідовності автоматизації.

3. Розробка онтології ПдО для кожної групи знань МПЮ, в термінах яких будуть модифікуватися (змінюватися) знання для кожного етапу автоматизації відповідно до вимог конкретної розробки або розвитку даної предметної області в цілому.

4. Розробка методу автоматизації проектування інтерфейсів, заснованого на використанні онтологій ПдО в системі знань для виключення модифікування самого методу при зміні знань.

Етапи автоматизації проектування користувацького інтерфейсу являють собою два основних кроки автоматизованого формування моделі інтерфейсу:

- формування моделі подання;
- формування моделі сценарію діалогу.

Ці етапи використовуються в моделюванні орієнтованому підході, так як вони дозволяють заздалегідь створити дизайн, який би задовольняв проектувальника інтерфейсу, а на наступному етапі імплементувати його в діючий інтерфейс.

Автоматизоване формування моделі представлення здійснюється шляхом виконання трьох взаємопов'язаних етапів:

– ідентифікації вікон – розподіл термінів і атрибутів моделі системи понять діалогу в одне або декілька вікон, відповідно до моделі задач користувача;

– вибір елементів інтерфейсу – вибір таких елементів взаємодії, які підходять для представлення терміна або атрибута моделі системи понять діалогу, відповідно до завдання, вимог користувача і середовища та правил «взабіліті». В даний етап також включено розташування підписів елемента інтерфейсу, опис їх тексту та встановлення значень параметрів елементів інтерфейсу;

– розташування елементів інтерфейсу – автоматизоване розташування елементів інтерфейсу в моделі, подання інформації відповідно до вікон, ідентифікованих на першому етапі.

2. Етапи створення інтерфейсу

Автоматизоване формування моделі сценарію діалогу – це формування навігації: зв'язків між вікнами, формування взаємодії інтерфейсу з прикладною програмою і низькорівневою взаємодією. Вікна зв'язуються між собою відповідно до моделі задач користувача. Формування взаємодії інтерфейсу з прикладною програмою здійснюється відповідно до моделі відображення моделі задач користувача на модель прикладної програми додаванням викликів функцій прикладної програми згідно із послідовністю вирішення завдання. Низькорівнева взаємодія – перевірка введених даних на введення за наявності обов'язковості, відповідності формату і області визначення.

Таким чином, процес автоматизованого формування моделі подання інформації та моделі сценарію діалогу полягає в наступних етапах:

1. Ідентифікація вікон.
2. Вибір елементів інтерфейсу.
3. Розташування підписів і задання їх тексту для обраних елементів інтерфейсу.
4. Задання значень параметрам обраних елементів інтерфейсу.
5. Розташування елементів інтерфейсу.
6. Створення навігації.
7. Додавання взаємодії з прикладною програмою.
8. Створення низькорівневої взаємодії.

Відповідно до запропонованого підходу, потрібно виділити параметри, необхідні для автоматизованої побудови моделі інтерфейсу. На основі аналізу їх структури має бути розроблена онтологія параметрів інтерфейсу, що визначає, яка інформація з он-

тологій ПдО користувальницького інтерфейсу на якому етапі автоматизованої побудови моделі інтерфейсу використовується, із зазначенням можливих значень цієї інформації і деякої додаткової інформації, яка характерна для кожного з етапів, з нею пов'язаних.

Основне завдання, яке вирішується при розробці такої онтології ПдО – відокремити компоненти моделі інтерфейсу від етапів, які їх використовують. Це дозволяє абстрагувати етапи від структури моделей і, відповідно, використовувати етапи в умовах як модифікації моделей, так і при появі нових моделей. При цьому досить описати параметр в термінах онтології параметрів і призначити його етапу. На підставі значень даних параметрів автоматично або автоматизовано формуються компоненти моделі інтерфейсу.

Але дуже швидке старіння програмного інструментарію є фактом сучасності. Тому виникає задача не тільки автоматизувати розробку та створення онтологічного інтерфейсу користувача, але і впровадити відповідний інструментарій розробки онтологічного інтерфейсу із властивостями розширюваності (тобто, можливості швидко змінюватися та адаптуватися до безперервного розвитку програмних засобів), що є актуальним завданням сьогодення. Вимоги до інструментальних засобів для розробки користувальницького інтерфейсу є класичні:

- простота розробки;
- модифікування і супровід;
- забезпечення якості інтерфейсу;
- забезпечення простоти та інтуїтивного розуміння інтерфейсу;
- підтримка різних типів діалогу тощо.

Існуючий інструментарій для розробки інтерфейсу (будівники інтерфейсу, системи управління призначенням для користувача інтерфейсом, моделює орієнтовані засоби для розробки інтерфейсу тощо) не відповідає повною мірою зазначеним вимогам. Аналіз інструментарію з точки зору вимог представлений в роботі [2]. Так, жоден із зазначених засобів не має спеціальних інструментів для розширення можливостей інструментарію розробки. Саме ця причина, як наголошується, наприклад, в [4], «гальмує» розвиток сучасних інструментальних засобів для розробки користувальницьких інтерфейсів. Для створення інструментарію, який би задовольняв усім перерахованим вище вимогам, запропоновано новий підхід до розробки користувальницького інтерфейсу на основі онтологій [2], в тому числі О ПдО [1].

Це є актуальним і доцільним в даному випадку, адже користувачі занурені в онтологічне середови-

ще, яке описує мовну картину світу із використанням онтологічних описів предметних областей. Таким чином онтологія, на базі якої треба розглядати онтологічний інтерфейс користувача, є онтологією предметної області О ПдО [1].

Запропонований підхід до розробки користувацького інтерфейсу на основі онтологій ПдО [1, 2, 3] також, як і існуючі підходи, насамперед моделює-орієнтований, виходить з роздільної розробки як модифікування інтерфейсу, так і прикладної програми поділу інтерфейсу на компоненти інструментальної підтримки проектування цих компонентів, роздільної їх модифікації, повторного їх використання, а також автоматичної генерації коду для користувача інтерфейсу за його моделлю.

Новим є визначення моделі користувача інтерфейсу. *Модель користувача інтерфейсу* – це декларативний опис, за яким автоматично генерується код користувача онтологічного інтерфейсу ПдО. Вона містить тільки ту інформацію, яка може змінитися при зміні вимог до інтерфейсу або прикладної програми. З кожним компонентом моделі інтерфейсу зв'язується своя система понять, а сам компонент моделі інтерфейсу – це інформація, яка представлена в цій системі понять таким чином, що різні компоненти моделі інтерфейсу визначаються різними системами понять. Виділяються чотири основні компоненти моделі інтерфейсу О ПдО і, відповідно, чотири класи систем понять:

1. Система понять користувача, в термінах якої він здійснює свою взаємодію з прикладною програмою. У цій системі понять виражаються вхідні і вихідні дані прикладної програми, а також інформація про інтелектуальну підтримку дій користувача.

2. Система понять подання інформації, в термінах якої визначаються різні типи діалогу. Даний клас містить три типи систем понять:

- систему понять графічного інтерфейсу користувача;
- систему понять графічних статичних сцен;
- систему понять для формування текстів.

Таким чином, кожна з систем понять підтримує проектування одного з типів діалогу.

3. Система понять для визначення сценаріїв діалогу. Вона визначає абстрактні терміни для опису реакцій на події (набори дій, які виконуються при виникненні подій, джерела подій, види режимів переходів між вікнами, способи вибору примірників вікон та ін.).

4. Система понять зв'язку, в термінах якої здійснюється зв'язок між прикладною програмою і інтерфейсом користувача. Вона визначає змінні, типи їх значень, які є загальними для інтерфейсу та прикладної програми, а також протоколи, за допомогою

яких відбувається комунікація, адреси серверів, з якими проводяться з'єднання, а також методи передачі повідомлень.

У термінах цих систем понять забезпечується інтелектуальна підтримка розробника з використанням структурних і графічних редакторів, а також визначаються зв'язки між компонентами. Інструментарій для розробки користувацького онтологічного інтерфейсу на базі онтологій ПдО включає в себе засоби для проектування та реалізації користувацького інтерфейсу, тестування та оцінювання, в тому числі вимоги до розширюваності інструментарію, які реалізуються для кожного типу засобів.

Проектування інтерфейсу полягає в побудові його онтологічної моделі, тобто специфікування (класифікації та опису) всіх компонентів для користувацького інтерфейсу.

Модель інтерфейсу – це декларативний опис, сформований за допомогою структурних і графічних редакторів, керованих чотирма класами систем понять із відображенням специфіки кожного компонента моделі інтерфейсу. Класи понять постійно змінюються. Відповідно, можливість розширюваності засобів проектування припускає розширення цих систем понять з відповідною зміною структурних і графічних редакторів, які керуються даними цих систем понять.

Для забезпечення розширюваності засобів проектування пропонується виконати наступні кроки:

- системи понять представити у формі моделей онтологій ПдО;
- надати розробнику інтерфейсу структурні та графічні редактори, реалізовані як інтерпретатори моделей онтологій ПдО;
- склад моделей онтологій не фіксувати, а надати фахівцям, які здійснюють супровід інструментарію, редактори моделей онтологій ПдО.

Онтологія – явний опис концептуалізації. Вона може мати різні форми, але обов'язково включає словник термінів, специфікацію їхнього змісту, а також опис зв'язків між термінами [5]. В роботах [2, 4] описано моделі онтологій, за якими формується модель для користувача інтерфейсу.

Таким чином, пропонується дворівнева архітектура інструментального програмного комплексу. Перший рівень призначений для розробників інтерфейсу і пов'язаний з проектуванням та генерацією коду інтерфейсу за його моделлю, а другий рівень призначений для фахівців, що здійснюють його супровід (розширення) і включає в себе набір редакторів моделей онтологій ПдО.

Реалізація інтерфейсу полягає в автоматичній генерації коду для користувача інтерфейсу за його

моделлю, тобто, процес генерації коду – це процес зіставлення моделі інтерфейсу набору вихідних кодів деякою мовою програмування. Зміна систем понять (моделей онтологій ПдО), як правило, вимагає внесення змін до коду користувача інтерфейсу. Для забезпечення розширюваності коду користувацького інтерфейсу пропонується модель генерації коду, яка описує відповідності між компонентами моделі інтерфейсу та інструкціями цільової мови програмування. Таким чином, генератор коду інтерфейсу керується моделлю генерації коду. Вона реалізується у вигляді широкого набору програмних компонентів і складається зі статичної та динамічної частин. Статична частина містить шаблони файлів, що реалізують фіксовані алгоритми для управління процесом генерації коду, а динамічна частина містить алгоритми відображення компонентів моделі інтерфейсу на програмний код (інструкції мови програмування).

Новизна і відмінна особливість цього підходу до створення онтологічного інтерфейсу користувача Порталу, у порівнянні з існуючими інструментами й розробками, полягає в тому, що методи автоматизації засновані на знаннях. Знання завжди знаходяться в русі, тому вони, за рахунок використання онтології ПдО, розширюються без зміни початкового коду інструментального засобу, а, крім того, дозволяють одночасно враховувати сукупність ергономічних та технологічних факторів, що веде до поліпшення якості інтерфейсу.

При розробці інтерфейсу треба, в тому числі, систематизувати ті прийоми (як візуальну систему понять), які використовує дизайнер-розробник інтерфейсу, під час проектування інтерфейсу користувача Порталу. Розглянемо мову взаємодії та її складові частини.

Елементи

Слова. Вони можуть виконувати роль універсальних міток абсолютно на всьому. (Наприклад, бути інструкцією для нових користувачів). Слова можуть бути командою або просто виконувати функцію інформування. При вмілому використанні, слова привносять простоту в інтерфейс, при невмілому – тільки бентежать користувача. Нашарування схожих, але неоднакових за змістом слів один на одного теж веде до плутанини. Для інтерфейсу це часом є смертельним.

Призначення слів в інтерфейсі – створювати простоту та ясність.

Іконки. Це універсальна система позначень, один з найдоступніших способів працювати із асоціативним мисленням користувача, не змушуючи його вчитуватися, вдивлятися й думати. Спочатку іконки були затребувані там, де час взаємодії люди-

ни і знака був швидшим, нетривалим.

Сьогодні впевненому користувачеві достатньо кількох секунд для того, щоб оцінити Портал і своє бажання користуватися ним (або знайти інший). Тому іконки дозволяють доставити візуальну інформацію прямо в мозок за секунди.

Іконки – досить уніфікована річ, тому, йдучи вбік зайвої креативності у винаході образів іконок, можна дуже серйозно пошкодити інтуїтивність інтерфейсу.

Кольори. Сама чіпка метафора – все той же світлофор. Розшифровка кольорів є відповідною:

- червоний – відмова, помилка, стоп, видалення, скидання, небезпека;
- жовтий – очікування, модерація, пауза, норма;
- зелений – схвалення, ОК, підтвердження, прийом, безпека.

Ще один приклад використання кольорів в інтерфейсах – це принцип «контрастних кольорів». Якщо більша частина сайту представлена в одній кольоровій гамі, а конкретна ділянка – пофарбована у контрастний колір, то користувач зверне увагу саме на неї. Можна підсилити увагу, додавши анімацію, рух.

Ще одна кольорова схема – температурна. Фіолетовий і синій сприймаються нами як щось холодне, стисле, похмуре. Водночас помаранчевий і червоний означають тепло, наповнення, комфорт. Прийом часто використовується у багатокрокових формах, де потрібно показати прогрес заповнення анкети і підштовхнути користувача заповнювати її далі.

Суцільні кольори застосовуються для позначення активних об'єктів, напівпрозорі – для неактивних. У конкретному інтерфейсі може застосовуватися прийом колірної кодування – колір повинен полегшувати навігацію користувача.

Форми. Це елемент комунікації, який останнім часом трохи втратив актуальність. Буквально в 2012-му році кнопки прагнули до максимальної правдоподібності, зараз – фактично стали різнокольоровими прямокутниками. Це відбулося багато в чому через ідеологічну війну двох китів – Microsoft і Apple. А сучасні користувачі досить спокушені – і з легкістю вгадують кнопку в кольоровому прямокутнику.

Звуки. Тут, на відміну від іконок, немає чіткої системи: звуки поділяються за принципом «приємний/неприємний» (користувачі Windows легко згадають кожен звук зі стандартної звукової схеми).

Звуки дуже часто стають предметом брендування. Призначення звуків в інтерфейсах – привернути увагу або спрямувати та супроводити дію користувача.

В інтерфейсах web-додатків користуватися звуками не прийнято. Якщо в окремих випадках звуковий супровід і робиться, то відключеним за замовчуванням.

Жести. Коли в наш побут входить незвична технологія, завжди комусь доводиться пояснювати принципи її використання. Самий універсальний спосіб – вдатися до допомоги жестів (замість того, щоб писати інструкції на всіх мовах світу).

Можна обмежитися мінімалістичними позначеннями. Наприклад, зауважте, що кнопки, які відповідають за «збільшення» завжди знаходяться праворуч або зверху, а за «зменшення» ліворуч або знизу. Зазвичай такі пари кнопок знаходяться в спеціальній «зоні», окремо від інших кнопок. Це дозволяє будь-якому користувачеві не дивлячись «намацати» їх і, відповідно, зменшити або додати гучність, переключити на наступний чи попередній канал тощо.

Взаємозв'язки

Зонування. Фактично – поділ елементів управління або контенту на «родини».

Пріоритет. Одного погляду на інтерфейс зазвичай достатньо для того, щоб зрозуміти, на що слід звернути пильну увагу, а де знаходиться другорядне.

Для розстановки пріоритетів використовуються такі елементи, як колір, розмір, форма, спеціальні стрілки-показчики, розташування – практично будь-який з названих нами раніше прийомів.

Акцентування за допомогою кольору особливо широко застосовується у web-дизайні для позначення бажаної дії. Також поширений спосіб створити фокус на сайті – використовувати персонажа з жестами, положенням особи тощо.

Принципи

Ясність. Основний принцип: інтерфейс повинен нести осмислення, а не заплутувати. Якщо в інтерфейсі і виникають якісь місця, незрозумілі одразу, завдання дизайнера – передбачити питання користувача і дати йому підказку.

Повідомлення, яке несе інтерфейс, має зчитуватися без зусиль.

Будь-яка дія користувача має приносити результати.

Немає нічого гіршого за непрацююче посилання.

Перспективність. Маючи справу з інтерфейсами, завжди потрібно усвідомлювати, що не варто слідувати моді. Завдання дизайнера – вирішити конкретну задачу.

Несподіванка. Створюйте інтерфейси, що викликають захват! Треба передбачати вимоги користувачів.

Заключення

Такий підхід до розв'язку проблеми створення інтерфейсу користувачів у середовищі WEB-порталів виходить із природної здатності людей думати логічно й творчо, визначити події й встановлювати відношення між ними. Зазначимо, що людині властиві дві характерні ознаки аналітичного мислення: перша – вміння спостерігати й аналізувати спостереження; інша – здатність встановлювати відношення між спостереженнями, оцінюючи рівень взаємозв'язків між відносинами, а потім синтезувати ці відносини в загальне сприйняття спостережуваного. Перераховане вище дає уявлення про принцип ідентичності й декомпозиції, принципи дискримінації, порівняльного судження й синтезу. Застосування онтологічних описів у цьому технологічному ланцюгу забезпечує динамічне формування відповідних множин властивостей та критеріїв щодо інформаційних ресурсів порталу, які вивчаються та досліджуються, на основі використання властивостей концептів предметних областей. Коректність та адекватність самого процесу взаємодії користувача з інформаційним середовищем порталу повністю залежить від коректності й адекватності кожної онтологічної моделі, яка складає функціональність інтерфейсу. Тоді до інформаційно-аналітичного супроводу різних інформаційних процесів у середовищі порталу можливе підключення різних груп експертів, що забезпечує семантичну повноту і коректність його наповнення.

Кожен, хто проектував інтерфейси самостійно, зрозуміє, наскільки важко часом донести потрібне повідомлення до користувача. І це незважаючи на те, що в розпорядженні дизайнера стільки «виразних засобів»: кольори, форми та інші графічні елементи, маса можливих взаємозв'язків та інструментів.

Література

1. Величко, В. *Автоматизированное создание тезауруса терминов предметной области для локальных поисковых систем [Текст]* / В. Величко, П. Волошин, С. Світла // «Knowledge – Dialogue – Solution» International Book Series «INFORMATION SCIENCE & COMPUTING», Number 15. – FOI ITHEA Sofia, Bulgaria. – 2009. – Р. 24-31.
2. Грибова, В. В. *Использование методов искусственного интеллекта для проектирования пользовательского интерфейса [Текст]* / В. В. Грибова, А. С. Клещев // *Информационные технологии.* – 2005. – № 8. – С. 58-62.
3. *Про один підхід до аналізу та розуміння природомовних об'єктів [Текст]* / О. В. Палагін, С. Ю. Світла, М. Г. Петренко, В. Ю. Величко // *Комп'ютерні засоби, мережі та системи.* – 2008. – № 7. – С. 128-137.

4. Puerta, A. R. *Issues in Automatic Generation of User Interfaces in Model-Based Systems [Електронний ресурс]* / A. R. Puerta // *Computer-Aided Design of User Interfaces*. – Namur : Universitaires de Namur, 1996. – P. 323–325. – Режим доступу: <http://www.arpuerta.com/pdf/cadui96a.pdf>. – 07.04.2014.

5. Uschold, M. *Knowledge Level Modeling: Concepts and Terminology [Text]* / M. Uschold // *In The Knowledge Engineering Review*. – 1998. – Vol. 13:1. – P. 5–29.

6. Найханова, Л. В. *Основные аспекты построения онтологий верхнего уровня и предметной области [Текст]* / Л. В. Нахайнова // *Интернет порталы: содержание и технологии*, Выпуск 3. / Редакт. : А. Н. Тихонов (пред.) и др. ; ФГУ ГНИИ ИТТ "Информатика". – М. : Просвещение, 2005. – С. 452-479.

7. Белоногов, Г. Г. *Языковые средства автоматизированных информационных систем [Текст]* / Г. Г. Белоногов, Б. А. Кузнецов. – М. : Наука. Главная редакция физико-математической литературы, 1983. – 288 с.

8. Попова, М. А. *Онтологичний інтерфейс як засіб представлення інформаційних ресурсів в ГІС-середовищі [Текст]* / М. А. Попова, О. Є. Стрижак // *Ученые записки Таврического национального университета им. В. И. Вернадского. Серия: География*. – 2013. – № 1, Том 26 (65). – С. 127-135.

9. Палагін, А. В. *Системно-онтологический анализ предметной области [Текст]* / А. В. Палагін, Н. Г. Петренко // *УСiМ*. – 2009. – № 4. – С. 3–9.

10. *Про один підхід до аналізу та розуміння природномовних об'єктів [Текст]* / О. В. Палагін, С. Ю. Світла, М. Г. Петренко, В. Ю. Величко // *Комп'ютерні засоби, мережі та системи*. – 2008. – № 7. – С. 128-137.

11. Стрижак, О. Є. *Комп'ютерні тезауруси як технологічна платформа створення авторських методик викладання предметних дисциплін [Текст]* // О. Є. Стрижак // *Актуальні проблеми психології: Психологічна теорія і технологія навчання / за ред. С. М. Максименко, М. Л. Смольсон*. – К. : Вид-во НПУ ім. М. П. Драгоманова, 2009. – Т. 8, вип. 6. – С. 259-266.

Надійшла до редакції 7.04.2014, розглянута на редколегії 19.05.2014

Рецензент: д-р техн. наук, професор, професор каф. проектування радіоелектронних систем ЛА Г. Я. Красовський, Національний аерокосмічний університет ім. М. Є. Жуковського «ХАІ», м. Харків.

МЕТОДИКА СОЗДАНИЯ ОНТОЛОГИЧЕСКОГО ИНТЕРФЕЙСА В СРЕДЕ WEB-ПОРТАЛА

А. Е. Стрижак, М. А. Попова, К. В. Ляшук

В статье рассматриваются вопросы использования методов онтологического моделирования при проектировании и построении научно-образовательных WEB-порталов и поддержки на их основе исследований в определенных предметных областях. Определяются структурные компоненты онтологий, которые наиболее полно могут быть представлены в виде онтологического интерфейса. Изложены положения автоматизации процесса проектирования интерфейса на основе онтологических моделей предметных областей. Предложены определенные правила создания дизайна интерфейса на основе свойств концептов, которые указаны в онтологии предметной области. Описывается методика формирования онтологического интерфейса на основе учета свойств концептов тематических онтологий.

Ключевые слова: онтология, свойство, интерфейс, портал, контент-анализ.

CREATION METHOD OF ONTOLOGICAL INTERFACE IN WEB-PORTAL

A. Ye. Stryzhak, M. A. Popova, K. V. Liashuk

The article deals with usage of ontological modeling methods in the design and construction of scientific and educational WEB-portals and research support on it base in specific subject areas. Ontology structural components that best may be presented in the form of ontology interface are defined. The interface design process automation provisions based on the ontological domain models are described. Some rules of creating interface design based on the domain ontology concepts properties are suggested. The method of the ontological interface forming based on consideration of the properties of the thematic ontologies concepts is described.

Keywords: ontology, property, interface, web-portal, content-analysis.

Стрижак Олександр Євгенійович – канд. техн. наук, провідний науковий співробітник, Інститут телекомунікацій і глобального інформаційного простору НАН України, Київ, Україна, e-mail: sae953@gmail.com.

Попова Марина Андріївна – молодший науковий співробітник відділу інформаційно-комунікаційних систем, Інститут телекомунікацій і глобального інформаційного простору НАН України, Київ, Україна, e-mail: pma1701@gmail.com.

Ляшук Костянтин Володимирович – аспірант, Інститут телекомунікацій і глобального інформаційного простору НАН України, Київ, Україна.