

МОДЕЛЮВАННЯ WEB-ОРІЄТОВАНИХ СИСТЕМ ТА НАПРЯМКИ РОЗВИТКУ WEB-РЕСУРСІВ

© Бойко Н., 2012

Розглянуто типовий приклад координації web-систем, основні проблеми, які виникають під час побудови ефективних web-сайтів та підходи до вирішення цих проблем. Зокрема досліджено основні показники ефективності сайтів та визначено методи їхнього покращення. Аналізуються напрямки розвитку інформаційних систем, які побудовані на основі web-технологій. Сформульовано основні підходи до впровадження web-технологій. Розглянуто структурно-функціональну модель програмного комплексу для проектування web-орієнтованих систем та описано структурні модулі, які входять до його складу.

Ключові слова: Інтернет, всесвітня павутина, World Wide Web, мова HTML, web-технології, web-системи, web-орієнтовані системи, сайт, мова програмування, технологія, інформаційні системи, скрипт, модуль.

In this article is a typical example of coordination web-systems. The main problems that arise when building an effective web-sites and approaches to solving these problems. In particular, studies key performance indicators defined sites and methods for improvement. Analyzes the trends of development of information systems that are based on web-technology. Formulated main approaches to implementation of web-technologies. The structural-functional model of software system for designing web-oriented systems and by a description of structural modules that are part of it.

Keywords: Internet, World Wide Web, World Wide Web, the language HTML, web-technologies, web-system, web-oriented systems, website programming language, technology, information systems, script module.

Вступ. Загальна постановка проблеми

Сьогодні існує широкий спектр програмних та технічних засобів для побудови web-систем. Проте подальший розвиток web-технологій стає неможливим без побудови ґрунтовної формально-математичної основи, насамперед без створення формальної моделі web-системи. Без такої основи неможливо розробити методики її проектування. Отже, сьогодні відсутні розширення для web-технологій усталених методів і засобів аналізу та проектування інформаційних систем (зокрема, структурних методологій). Відсутність математичної моделі web-системи унеможливує розроблення інтелектуальних засобів їх адміністрування, які ґрунтуються на алгоритмах її оптимізації та прогнозуванні поведінки в часі [1–5].

Потреба у моделюванні багаторівневої web-системи глобального характеру приводить до необхідності побудови їх формальної моделі та алгоритмів їхньої оптимізації як основи для якісної розробки ефективних систем, незалежно від їх складності та характеру.

Зв'язок висвітленої проблеми із важливими науковими та практичними завданнями теми

Багатоплановість і багатоаспектність проблеми дослідження та побудови зазначених ресурсів у глобальній інформаційній мережі зумовили появу значної кількості публікацій, що мають безпосереднє або опосередковане відношення до тематики даного дослідження. У низці робіт [1–10] розглянуто особливості формування веб-сайтів, порушено питання формування інформаційних ресурсів, зокрема створення оцифрованих ресурсів та організації пошуку в них інформації, а також деякі аспекти інформаційної технології репрезентації web-ресурсів.

Предметом дослідження є функціонування веб-сайтів, особливості формування і технічного супроводу інформаційних ресурсів, організація доступу до них, перспективи розвитку цих ресурсів в умовах інформатизації суспільства з урахуванням сучасного історичного етапу розвитку України.

Методологічною основою статті є загальнонаукові підходи до наукового пізнання (системний, функціональний, структурний, моделювання). Використано методи інформаційного аналізу, що дало можливість дослідити специфіку та розвиток веб-сайтів, окреслити комплекс актуальних науково-технологічних питань формування сайту та використання його ресурсів.

Актуальність теми зумовлена питаннями інформатизації, впровадження інформаційно-комунікаційних технологій в економічну діяльність підприємств та ознайомлення із основними моделями інформаційної бази web-орієнтованих систем.

Мета і задачі дослідження. Сьогодні існує безліч web-технологій для створення сайтів. Метою дослідження є розроблення методів та алгоритмів моделювання інформаційних систем, що реалізовані на основі web-технологій, та створення процедур оптимізації структури web-систем.

Аналіз останніх досліджень та публікацій

У публікаціях на цю тематику [1–3], переважно розглядаються окремо архітектура та технології. Проте загалом система функціонує, і потрібно розглядати на різних рівнях її функціонування та взаємодію. Це дасть змогу визначити вимоги як до програмної так і до апаратної частини інформаційної системи.

У нових умовах системи оптимальної структури будують, враховуючи глобальний, багаторівневий характер інформаційних систем та стохастичну поведінку користувачів такої системи. Ефективним математичним апаратом розв'язання таких задач є (як показано в роботах А.М. Пелешишина [8–10]) теорія координат.

Виділення проблем

Сьогодні бізнес висуває доволі жорсткі вимоги до побудови інформаційних систем. Щоб їх задовольнити, системі необхідно: мати розподілену модульну архітектуру; забезпечувати централізований доступ до інформації та централізоване управління; мінімізувати витрати на своє утримання (обслуговування серверів, налаштування конфігурації робочих станцій, оновлення версій тощо); підтримувати інтерфейси взаємодії з іншими системами; володіти мультиплатформеністю, мати мінімальні претензії до конфігурації робочих місць користувача; відповідати високим показникам якості, надійності, стабільності, безпеки, швидкодії [1, 8, 12].

Для сучасних умов характерне застосування вискоелективних внутрішньофірмових систем інформації, що ґрунтуються на використанні найновіших інформаційних технологій, зокрема єдиної локальної комп'ютерної мережі. Управлінська внутрішня інформаційна система – це сукупність інформаційних процесів для задоволення потреб в інформації на різних рівнях прийняття рішень. Інформаційна система містить компоненти обробки інформації, внутрішні і зовнішні канали передавання [13].

Інформація, особливо її автоматизована обробка, і тепер залишається важливим фактором підвищення ефективності діяльності будь-якої організації. Важливого значення у використанні інформації набувають способи її реєстрації, обробки, нагромадження і передавання; систематизоване збереження інформації та її виведення в потрібній формі; виробництво нової числової, графічної та іншої інформації.

Підвищують ефективність інформаційних систем за допомогою наскрізної структури і сумісності інформаційних систем, які дають змогу усунути дублювання і забезпечують багатократне використання інформації, встановлюють визначені інтеграційні зв'язки, обмежують кількість показників, зменшують обсяг інформаційних потоків, підвищують рівень використання інформації. Інформаційна система повинна підтримувати такі функції, як надання інформації (наприклад, потрібної користувачам для розв'язання науково-виробничих задач) та створення найзручніших умов для її поширення (наприклад, проведення адміністративно-організаційних, науково-дослідних і виробничих заходів, які забезпечують її ефективне розповсюдження) [14, 15].

Темпи розвитку сучасних інформаційних технологій (ІТ) вимагають від фахівців різних галузей глибоких знань ІТ, розуміння тенденцій їх розвитку, вміння передбачати нові можливості. Тому регулярний моніторинг прогресу в сфері інформаційних технологій, оцінювання їхніх перспектив у контексті економічної діяльності є актуальними.

Інтернет впевнено увійшов в наше життя, зокрема суспільне. Паралельно з процесом збільшення кількості активних користувачів мережі невідмінно поліпшується якість послуг провайдерів інтернет-послуг, насамперед це стосується регулярного здешевлення пропускну здатності у перерахунку на окремого кінцевого споживача. Все частіше в пресі йдеться про настання епохи Web 2.0 як про доконаний факт. Проте для кінцевого користувача типовий інформаційний ресурс все ще залишається статичною сторінкою з матеріалом сумнівної якості та походження [14, 15].

Власне саме поняття Web (дослівно – павутина) не має чіткого технічного визначення, скоріше, – це концепція, ідея поєднання інформаційних технологій для представлення інформації різного типу в зручному для користувача вигляді та з можливістю негайного переходу за посиланнями.

На момент масового впровадження графічного інтерфейсу користувача на платформах персональних комп'ютерів ця концепція призвела до вибухоподібного поширення мережі Інтернет у світі. Web 2.0 теж слід швидше сприймати як суму технологій (за Лемом), а не як єдину технологію. При цьому на передній план виходить ідея співпраці, колективної творчості, миттєвого поширення та отримання відгуку від споживачів інформації. Інтернет перестає бути лише інформаційним джерелом, а стає платформою, місцем для спільної праці.

Сьогодні стає все більш зрозуміло, що сучасні інформаційні системи, зокрема web-орієнтовані системи, повинні все більше спиратися на семантику предметної області, на знання про неї. Якщо основу знань системи становить онтологія, то і система, яка ґрунтується на знаннях, повинна бути онтологічно орієнтованою [11].

Функціональність інформаційних систем залежить від організаційно-управлінської структури організації, технології документообігу та всього, що необхідно для підтримки ефективного бізнесу. За сучасного рівня розвитку програмного забезпечення, існує безліч різноманітних програмних засобів обробки інформації, написаних різними мовами програмування за вищепереліченими методами. Різноманіття програмних пакетів (ПП) пов'язано із специфікою кожної галузі, в якій проводиться обробка. Наприклад, для обробки графічних зображень широко використовуються методи розпізнавання образів, криптографічні методи, що ґрунтуються на перетворенні Фур'є тощо [13].

Водночас характерною особливістю web-орієнтованих інформаційних систем, зокрема систем управління контентом web-сайтів, є наявність великої кількості документальної інформації, причому самі ці документи часто генеруються динамічно, на основі тих чи інших процедур. Тому web-орієнтована інформаційна система, яка ґрунтується на знаннях, повинна бути орієнтована на роботу як з онтологіями, так і з документами і множинами документів.

Переважно, під час розроблення web-орієнтованих інформаційних систем, які мають використовуватися в мережі Internet чи в локальній мережі окремої установи, постає задача створення структурно складних систем. Такі системи складаються з компонентів, кожен з яких виконує свою роль у життєвому циклі системи, і задачі, що виконуються ними, є самостійними і можуть бути окремими повнофункціональними системами.

Для таких підсистем вже є готові рішення, які успішно експлуатуються. В такому випадку для економії виробничих ресурсів розробника доцільно використати вже готовий програмний продукт – платний або з відкритим кодом.

У випадку, коли технологія, на якій ґрунтується створення такої інформаційної системи, не збігається з технологією, на якій ґрунтується готове рішення певної підсистеми, постає проблема інтеграції цих продуктів з метою надання користувачам зручних засобів розв'язання задач. Як показує світовий досвід, більшість розробників вважають доцільним не зв'язувати компоненти в єдине середовище користувача, що, своєю чергою, спричиняє значні незручності.

Тому в основу моделі інформаційної бази web-орієнтованої системи необхідно покласти дві вузлові компоненти: онтологія предметної області та множина документів, а також зв'язки між цими компонентами. Іншими словами, необхідно побудувати граф, який складається не з логічно

розрізаних документів, що характерно для більшості сучасних web-орієнтованих систем, а з описів реальних класів предметної області, їх екземплярів та зв'язків між ними, а також пов'язаних з ними документів. Тобто, можна говорити про проблему “занурення” множини документів, які можуть бути статичними або генеруватися динамічно, в загальну семантику предметної області [1, 12, 13].

Детальніше формалізувати зв'язки між семантикою предметної області та множиною документів можна за формальною моделю онтології [12].

Слід також зазначити, що базу знань, яку покладено в основу інтелектуальної інформаційної системи, можна розглядати як наповнення власне онтології як концептуальної схеми [13, 14]. Формально розглядаємо модель інформаційного наповнення web-орієнтованої системи як трійку $M = \langle W^*, D, L \rangle$, де M – онтологія предметної області, W^* – розширена онтологія, наповнення онтології W конкретними екземплярами класів (фактично – база знань), D – множина документів; L – множина зв'язків між W^* та D .

На найзагальнішому рівні, в термінології UML [5] власне онтологія пов'язана з діаграмою класів, а наповнення бази знань – з діаграмою екземплярів. Але об'єктно-орієнтована реалізація онтологічного підходу, незважаючи на спорідненість цих напрямків, має свою специфіку. Зокрема, в [6] зазначається, що розроблення онтологій значно відрізняється від проектування класів і зв'язків в об'єктно-орієнтованому програмуванні.

Об'єктно-орієнтоване програмування зосереджується переважно на методах класів: програміст приймає проектні рішення за операторними методами класу, тоді як розробник онтології приймає рішення, ґрунтуючись на структурних властивостях класу.

У результаті структура класу і зв'язки між класами в онтології значно відрізняються від структури подібної предметної області в об'єктно-орієнтованій програмі.

У [11] показано, що значеннями функцій інтерпретації онтології можуть бути елементи множини документів D . При цьому мова повинна йти не тільки власне про онтологію, але і про всю базу знань, побудовану на її основі. Якщо онтологія розглядається як трійка $\langle Q, I, F \rangle$, де Q – множина класів, які відповідають поняттям предметної області; I – множина зв'язків між ними, а F – множина функцій інтерпретації, то розширена онтологія описується як трійка $\langle Q^*, I^*, F^* \rangle$, де Q^* – множина класів разом з їх екземплярами, I^* – множина зв'язків між цими елементами, а F^* – множина функцій інтерпретації, визначених у найпростішому випадку на елементах з Q^* , I^* . Тоді елементи D можуть бути значеннями функцій з F^* .

Іншими словами, будемо вважати документ d релевантним відносно W^* , якщо існують хоча б один вузол i /та функція інтерпретації f такі, що $d=f(w)$.

Важливими елементами web-орієнтованої інформаційної системи повинні стати класи, які реалізують функції інтерпретації та можуть використовуватися для динамічного формування документів, а також класи, які забезпечують відображення отриманих документів.

Наведене співвідношення може стати основою для динамічного формування переліку споріднених документів. Для документа $d=f(w)$ спорідненими документами можуть вважатися зокрема такі:

- всі документи s такі, що $s=h(w)$, де h пов'язане з F^* (тобто документи, пов'язані з тим самим вузлом іншими функціями інтерпретації);
- всі документи s такі, що $s=g(u)$, де функції інтерпретації g пов'язані з f , вузли u пов'язані з w .

Принципово важливим є те, що зв'язки і можливі переходи між документами повинні насамперед визначатися зв'язками між класами онтології та їх екземплярами. Тому ключовим компонентом стає система навігації, яка на основі онтології забезпечує динамічне формування навігаційних графів [11], які визначають можливі переходи web-сайтом.

Якщо ввести міри, які характеризують семантичну близькість понять онтології, а також ступені важливості функцій інтерпретації, це дасть змогу проранжувати документи за мірою їх спорідненості до даного. Цікаво також дослідити, як введені так міри спорідненості співвідносяться з традиційними підходами до визначення мір схожості документів на основі аналізу власне текстів [7–12].

У цьому контексті стає очевидним, що проблема динамічного формування документів, споріднених до даного, якнайтісніше пов'язана з проблемою підвищення повноти та релевантності інформаційного пошуку. Дійсно, незалежно від того, чи користувач вийшов на певний вузол онтології в процесі навігації по сайту, чи він ввів відповідне ключове слово в полі введення – йдеться про формування переліку документів, пов'язаних з цим вузлом, і ранжування цих документів за мірою релевантності.

Провівши формальний опис інформаційних систем, визначимо типові операції над нею. Розділимо такі операції на дві основні групи.

1. До першої групи належать операції, що модифікують наявну систему шляхом зміни її внутрішньої структури – у таких операціях бере участь одна інформаційна система.
2. Друга група операцій формується на основі операцій, традиційних для теорій формальних систем [6] та абстрактних автоматів [3, 4, 7]. У таких операціях беруть участь дві або більше інформаційні системи; результатом операції є нова система, що їх об'єднує.

Розглядаючи операції 2-ї групи, вважатимемо, що вихідний потік може розщеплюватися або дублюватися та одна з його копій – передаватися в іншу систему. У результаті застосування таких операцій до двох або більше інформаційних систем утворюється нова інформаційна система, що містить як компоненти вихідні системи. Запити до цих систем уже надходять від вищого сервісу, а не від користувачів системи. Відповіді компонент спрямовуються як вхідний потік у вищу систему.

Типовими представниками першої групи є такі операції.

1. Операція визначення параметра. Визначимо перетворення інформаційної системи IS на IS' визначаючи параметр як фіксацію значення одного з параметрів $P_{j(i)}$ запиту Q_i . У разі такої операції невизначеність поведінки системи залишається такою самою.
2. Операція усунення параметра. Визначимо перетворення інформаційної системи IS у IS' шляхом усунення параметра як усунення одного з параметрів $P_{j(i)}$ запиту Q_i з розгляду. У разі застосування цієї операції зростає невизначеність поведінки системи.
3. Операцію усунення параметра доцільно застосовувати до системи з метою побудови її загальної моделі, придатної для презентації.
4. Операція визначення параметра з запиту. Визначимо операцію визначення параметра з запиту як злиття групи можливих запитів в один та визначення нового параметра в цьому запиті. Область визначення нового параметра повинна бути скінченною множиною, кожен з елементів якої однозначно відповідає одному з усунутих збігаються. Таку операцію застосовують лише у випадку, коли кортежі параметрів запитів збігаються. Тому перед застосуванням цієї операції доцільно узгодити кортежі запитів за допомогою операції усунення та визначення параметрів. Очевидно, що за такої операції зростає невизначеність поведінки системи.
5. Операція визначення запиту з параметра. Ця операція визначення запиту з параметра є оберненою до попередньої (один запит розбивається на групу запитів за допомогою усунення параметра з переліченою областю визначення). Така операція породжує групу запитів, у яких домени параметрів запитів збігаються. При такій операції зростає визначеність поведінки системи. Тому ця операція є не зовсім коректною у тому сенсі, що в системі недостатньо інформації для її здійснення. Тому необхідно проводити додаткові дослідження для уточнення поведінки системи.

Аналогічно формулюються операції для станів та відповідей інформаційної системи.

До другої групи належать такі операції:

1. Послідовне з'єднання двох ІС. З'єднання відбувається послідовно (відповідно до технології “клієнт-сервер”). У такому випадку вихідний символ IS_1 формує запит IS_2 .

Так отримуємо інформаційну систему IS , яка є послідовним об'єднанням двох IS_1 та IS_2 . Її вхідний алфавіт збігається з вхідним алфавітом IS_1 , а множина вихідних символів з вхідним алфавітом – IS_2 . Невизначеність поведінки кожної з систем IS_1 та IS_2 залишається незмінною, а поведінку нової системи прогнозують так: вхідні сигнали для IS_2 є вихідними для IS_1 , тому імовірність вхідного символу IS_2 отримуватиметься так:

$$\frac{1}{N} \sum_{j=1}^N \Pr_j(A_j^{(1)}) \xrightarrow{N \rightarrow \infty} \Pr(Q_i^{(2)}), \quad (1)$$

де $\Pr_j(A_j^{(1)})$ – імовірність вихідного символу A_i для IS_j на j -й ітерації.

Імовірність кожного стану та переходу в межах підсистеми IS_1 залишається незмінною, а для підсистеми IS_2 коригується згідно із змінами ймовірностей вхідних символів. Така операція характерна для серверів переадресації, довідкових систем, електронних бібліотек тощо.

2. Паралельне з'єднання двох ІС. З'єднання відбувається паралельно (два сервера об'єднуються в один). У такому випадку отримуємо інформаційну систему IS , яка є паралельним об'єднанням двох IS_1 та IS_2 . Її вхідний алфавіт утворюється як об'єднання вхідних алфавітів IS_1 та IS_2 , а вихідний алфавіт є об'єднанням вхідних алфавітів IS_1 та IS_2 . Невизначеність поведінки кожної із систем IS_1 та IS_2 залишається незмінною, а поведінку нової системи прогнозують так: вхідні сигнали IS є вхідними для IS_1 або IS_2 , тому імовірність вхідного символу IS отримують за виразом:

$$\Pr(Q_i) = \Pr(Q_i^{(k)}) \Pr(IS_k), \quad (2)$$

де $\Pr(Q_i^{(k)})$, $k = 1, 2$ – імовірність вхідного символу Q_i для IS_k ; $\Pr(IS_k)$, $k = 1, 2$ – імовірність, що вхідний символ належить до алфавіту підсистеми IS_k .

3. Замикання ІС. Вихідний потік спрямовується як вхідний у ту саму систему (сервер “замикається” сам на себе). Поведінка нової системи буде загалом схожа на поведінку вихідної.

Одним з наслідків побудови моделі web-системи є можливість формального опису її поведінки. Такий опис має імовірнісний характер та відображає асимптотичну поведінку системи. Досліджуючи поведінку системи, розглядають абстрактний часовий параметр. Розглянемо проблему визначення часового параметра. Формальний підхід, описаний в [6], ґрунтується на виділенні певної впорядкованої множини, що розглядається як множина моментів часу. Елементи множин вхідних сигналів та реакцій системи є результатами застосування певних функцій до моментів часу.

При моделюванні систем архітектури “клієнт–сервер”, враховуючи те, що будь-яка дія системи визначається запитом клієнта, за формальний параметр часу доцільно приймати номер за порядком запиту клієнта. Множина номерів запитів – множина натуральних чисел. Існує істотна відмінність між реальним фізичним часом та формальним часом інформаційної системи: так система, що функціонує коротший час, може бути формально старішою – опрацювати більшу кількість запитів. У межах аналізу поведінки web-систем (зокрема конкретних апаратних та мережових рішень) доцільно проводити зіставлення реального та формального часу. Основний акцент під час моделювання перспективної поведінки системи роблять на терміни формального часу.

Одним з головних напрямків аналізу поведінки системи є визначення її поведінки в перспективі – тобто при зростаючому до безмежності числі запитів системи. Інформаційна система є динамічною системою в розумінні теорії формальних систем [6]. Для визначення реакції системи на отриманий запит достатньо лише інформації про стан системи (вмісту її бази даних).

Іншою важливою вимогою до web-орієнтованих інформаційних систем є необхідність використання сучасних стандартів і технологій з огляду на логіку та перспективи їх розвитку. Ідеться насамперед про проект Semantic Web [10], орієнтований на обмін даними в середовищі WWW та на їх повторне використання і особливо – на опис семантики, змісту даних, які зберігаються на web-ресурсах. Semantic Web ґрунтується передусім на таких рекомендаціях, затверджених консорціумом W3C [10]:

- розширена мова розмітки XML;
- структура опису ресурсу RDF [11] (Resource Description Framework). Ця структура забезпечує можливість для взаємодії застосувань на основі аналізу описів web-документів. RDF Schema, своєю чергою, допомагає поєднувати ці описи в єдиний словник. Специфікація RDF надає потужну інфраструктуру для підтримки обміну знаннями в Інтернеті;

- мова web-онтологій OWL [12] (Ontology Web Language), що надає інструментарій для побудови структурованих web-орієнтованих онтологій, які легко інтегруються та забезпечують можливість взаємодії даних між різними групами.

З огляду на описані вище принципи проектування web-орієнтованих інформаційних систем, які повинні брати до уваги зв'язки між онтологією предметної області та множиною документів, а також з використанням наведених вище рекомендацій, розроблено прототип системи управління вмістом web-ресурсів, який забезпечує такі можливості [11]:

1. Система генерує навігаційні графи на основі онтологічного опису предметної області, може переходити від однієї навігаційної структури до іншої та від одного опису відповідного поняття предметної області до іншого залежно від завдання потрібної системи відношень та функцій інтерпретації. При цьому залишається можливість жорсткого завдання системи навігаційних посилань на основі відношення “рубрика/підрубрика”.

2. Система працює з онтологією, яка містить три логічно виокремлені рівні:

- рівень даних, або онтологія предметної області: описує базові поняття, класи та зв'язки між ними, а також конкретні сутності предметної області (ПрО).
- рівень структури, що забезпечують навігацію користувача по сайту;
- рівень представлення даних, або онтологія структурних шаблонів, що керує відображенням інформації у вигляді, зручному для користувача.

3. Система спирається на XML, RDF та OWL.

Для того, щоб описати модель системи управління web-сайтом, що ґрунтується на базі знань, необхідно спочатку описати вимоги, що накладаються на онтологію. Онтологічна база знань разом з редактором онтологій має відповідати такому переліку вимог, сформульованих на основі [14]:

1. Структури: онтологія має відображати всю структуру інформаційного ресурсу.

2. Вмісту: онтологія має надавати можливість акумулювати різні типи інформації, а також дозволяти посилання на інші онтології.

3. Імпорту та експорту: вміст онтології має легко імпортуватись або експортуватись в систему і з системи відповідно.

4. Життєвого циклу інформації: онтологія має надавати можливості для точного представлення та ілюстрації всіх етапів життєвого циклу.

5. Відображення: онтологія має забезпечувати легкість внесення змін до представлення інформації на сайті.

Структура сайту визначається екземплярами одного класу – класу сторінок (Pages). Отже, до неї можна звернутись як до дерева, кореневому елементу якого відповідає початкова сторінка. Завданням онтології є змодельовати ієрархічну деревоподібну структуру і вказати для кожного елемента його рівень та позицію. Для цього в класі сторінок вводяться додаткові параметри (властивості), такі як:

1. Батьківський елемент: тобто кожен елемент структури, крім кореневого, має елемент, який визначає рівень вкладеності його у навігаційній структурі.
2. Позиція: властивість, що відповідає за порядок відображення певного елемента на визначеному рівні.
3. Ресурс: URI ресурсу (класу або індивіду онтології, що відображається в цьому розділі структури).

Система з web-інтерфейсом, незалежно від архітектури, спочатку є розподіленою і надає можливість спільної паралельної роботи з інформацією. Повна вартість wb-системи (включаючи витрати на утримання), як правило, виявляється нижче за рахунок централізованого оновлення, відсутність необхідності конфігурації та обслуговування користувача робочих місць. Вся робота з системою відбувається через web-браузер, який входить в набір стандартних програм будь-якої операційної системи. Ні до програмної, ні до апаратної частини робочого місця користувача не висувають жодних вимог, окрім як організувати мережне з'єднання з сервером і забезпечити роботу

web-браузера (технологія тонкого клієнта). Сучасний web-інтерфейс забезпечує прийнятний для інформаційних систем рівень швидкодії та зручності використання.

Реалізація web-додатків на платформі NET Framework (технологія ASP.NET):

- скорочує термін розроблення за рахунок використання сучасних візуальних засобів і багатого набору базових компонентів;
- озброює його найпоширенішими технологіями доступу до даних і служб (ADO.NET, робота з XML, Web-Service), методами роботи з файлами різного формату (зокрема закриті формати MS Office);
- захищає вихідний код системи від небажаного втручання за допомогою його компіляції, здійснює моніторинг і захист від хакерських атак;
- не поступається, а в деяких випадках і перевершує аналогічні рішення за показниками швидкодії;
- надає можливість захисту даних шляхом організації шифрованого каналу або обмеження доступу за адресами користувачів.

Інформаційні web-системи отримують особливу перевагу там, де є розподіл мережі користувачів системи. Це може бути філіальна структура організації, наявність віддалених співробітників або партнерів. Так само web-система виявляється вигіднішою в разі необхідності мобільного доступу, наприклад, коли користувач працює з різних ПК чи йому необхідно мати доступ до системи зі свого портативного комп'ютера, але не бути територіально прив'язаним. У цьому випадку вигода полягає у відсутності необхідності установки клієнтського програмного забезпечення та організації доступу до системи (за винятком доступу до Інтернету).

Web-технології так само не стояли на місці та набули кращих можливостей побудови інтерактивного інтерфейсу, сучасних середовищ швидкого розроблення web-додатків, можлива взаємодія з уже існуючими системами і сервісами. Звичайно, якою б досконалою не була технологія, сама вона не в змозі вирішити 100 % завдань, і web тут не виняток. Але якщо ж web-рішення задовольняє вимоги проекту, то робота з цією технологією користувача не розчарує.

Формулювання мети

На основі викладеного вище матеріалу постає питання: які технології та яка архітектура інформаційних web-систем відповідатимуть таким умовам: простота у супроводженні, користуванні, поширеність (легко знайти технічних спеціалістів для супроводження, підтримки системи, модернізації, розширення тощо). Це визначає мету, завдання та актуальність цієї проблематики. Адаптивність систем орієнтованих на web-доступ, пояснюється доволі просто – крос-платформеність, спрощена система використання “Logon and Go” – потрібен лише браузер, логін та пароль, доступ з корпоративної локальної мережі (Intranet) – будь-яке під'єднання; якщо потрібен доступ з мережі Internet – доступ до Internet, і системою вже можна користуватись. Тому завдяки описаним вище властивостям структура web-сайту стає повністю керованою і підпорядкованою адміністратору. Більше того, додавання нових елементів до структури полягає лише у введенні нового елемента та визначенні для нього вищезгаданих параметрів.

Аналіз отриманих наукових результатів

Аналізуючи запропонований вище підхід до експлуатації web-орієнтованих інформаційних систем, можна вирішити актуальні завдання, що постають в процесі інтеграції систем, створених на основі різних технологій.

За описаним підходом можна:

- мінімізувати дублювання інформації у БД;
- зменшити вимоги до апаратних ресурсів;
- вирішити деякі принципи незручності, які можуть виникнути в процесі користування, завдяки створенню єдиної системи авторизації та реєстрації, розміщенню компонентів в єдиному домені та встановленню правил заміни посилань;

- зменшити затрати виробничих ресурсів на створення такої системи, створювати програмний код та скрипти для багаторазового використання;
- підвищити надійність системи завдяки існуванню єдиної системи ведення облікових записів.

Особливу увагу необхідно звернути на захищеність web-орієнтованої інформаційної системи, що є в багатьох випадках ключовим питанням, особливо коли така система є комерційним сервісом. У цьому випадку дані про облікові записи користувачів повинні бути особливо захищеними. Безперечно, захищеність такої системи визначається рівнем захисту найвразливішого до атак зловмисників компонента. Тому, коли треба вибрати створювати власні компоненти системи чи використати вже готові, слід спочатку оцінити доцільність кожного варіанта. У випадку розглянутої двокомпонентної системи захищеність форуму як вибраного вже готового компонента не важлива, оскільки дані з БД форуму не є закритою інформацією. Щоби не втратити інформацію, періодично копіюють вміст БД. Проте конфіденційна інформація міститься в БД ZODB сервісу. СУБД ZODB не дає змогу отримати віддалений доступ до інформації, яка зберігається в ній, без встановлених відповідних програмних засобів, і заволодіти інформацією можна тільки, отримавши адміністративні права на сервері чи виконавши Python – програму на боці сервера.

Висновки і перспективи подальших наукових розвідок

Моделювання – це одна з основних категорій пізнання. На ідеї моделювання ґрунтується будь-який метод наукового дослідження – як теоретичний (коли використовуються абстрактні моделі), так і експериментальний (що використовує предметні моделі). Системність діяльності полягає у тому, що вона організовується за певним планом, або алгоритмом. Алгоритм є моделлю діяльності, що планується. Ціль моделювання визначає, які властивості оригіналу і якою мірою (з якою точністю) повинні бути відображені у моделі.

Моделювання є найефективнішим способом дослідження складних систем різного призначення: технічних, економічних, екологічних, соціальних, інформаційних – як на етапі їх проектування, так і в процесі експлуатації. Можливості моделювання систем далеко не вичерпані, тому постійно з'являються найновіші методи та технології моделювання. Сучасні технології моделювання не тільки полегшили і прискорили процес побудови та дослідження моделі, але й значно наблизили сприйняття інформації спеціаліста з моделювання систем і спеціаліста, що працює у галузі, яка моделюється.

Сучасна тенденція розвитку програмних систем така, що частка мережових віддалених, зокрема і web-орієнтованих рішень постійно збільшується. Це зумовлено насамперед меншою вартістю експлуатації таких систем, поліпшенням якості каналів зв'язку і зниженням їх вартості. Однак, розвиток інформаційних технологій відбувається настільки бурхливо, що вони не встигають реалізовуватися в конкретних програмних продуктах. Поява нових технологій призводить до миттєвого “старіння” та втрати здатності до конкуренції старих інструментів. Тому слід постійно вивчати стан ринку програмного забезпечення. Адже вивчення, аналіз та порівняння можливостей web-орієнтованих систем – довгий та ресурсомісткий процес, який потребує подальшого дослідження. Загалом, теоретичні дослідження нових технологій значно випереджають практичне їх застосування, а це вказує на актуальність цієї проблематики і глибокого її вивчення.

Буч Г. UML. / Г. Буч, А. Якобсон, Дж. Рамбо. – СПб.: Питер, 2006. – 736 с. 2. Гаврилова Т.А. Базы знаний интеллектуальных систем / Т.А. Гаврилова, В.Ф. Хорошевский . – СПб.: Питер, 2000. – 384 с. 3. Гриценко В.И. Поисковый сервис. Проблемы, технологии, перспективы / В.И. Гриценко, К.К. Духновская, А.А. Урсатьев // УсиМ – 2006. – №2. – С. 81–92. 4. Жежнич П.І. Основні правила побудови семантично відкритих інформаційних систем / П.І. Жежнич, Р.Б. Кравець, В.В. Пасічник, А.М. Пелецишин // Інформаційні системи та мережі: Вісник Нац. ун-ту “Львівська політехніка –

1999. – №383. – С. 84–95. 5. Жежнич П.І. Семантично відкриті інформаційні системи / П.І. Жежнич, Р.Б. Кравець, В.В. Пасічник, А.М. Пелецишин // Інформаційні системи та мережі: Вісник Нац. ун-ту "Львівська політехніка". 1999. – №383. – С. 73–84. 6. Ландэ Д.В. Поиск знаний в Интернет / Д.В. Ландэ. – М.: Изд. дом "Вильямс", 2005. – 272 с. 7. Олецкий О.В. Застосування формальних моделей онтологій для формалізації інформаційних потоків у системах управління контентом / О.В. Олецкий // Теоретичні та прикладні аспекти побудови програмних систем: Матер. міжнар. конф. ТАAPSD'2005, Київ, 7–9 грудня 2005 р. – С 26–29. 8. Пелецишин А.М. Використання апарату абстрактних автоматів для моделювання Web-систем / А.М. Пелецишин // Інформаційні системи та мережі: Вісник Нац. ун-ту "Львівська політехніка". – 1998. – №330. – С. 188–201. 9. Пелецишин А.М. Методи та алгоритми моделювання Web-систем / А.М. Пелецишин // Інформаційні системи та мережі: Вісник Нац. ун-ту "Львівська політехніка". – №406. – 2000. – С. 199–211. 10. Пелецишин А.М. Побудова формальної моделі Web-системи / А.М. Пелецишин // Інформаційні системи та мережі. Задачі та методи прикладної математики. Вісник Львівського Нац. ун-ту ім. І. Франка. – 1998. – С. 182–185. 11. Проскудіна Г.Ю. Онтології в інформаційних системах / Г.Ю. Проскудіна., О.М. Овдій //Теоретичні та прикладні аспекти побудови програмних систем: спец. випуск Вісника Київськ. нац. ун-ту ім. Т.Г. Шевченка, 2004. – С. 164–169. 12. Noy N., McGuinness D. *Ontology Development 101: A Guide to Creating Your First Ontology*// Stanford Knowledge Systems Laboratory Technical Report KSL-01-05 and Stanford Medical informatics Technical Report SMI-2001-0880, March 2001. 13. Suren A. *powL - A Web Based Platform for Collaborative Semantic Web Development* // powi.sourceforge.net/swc/powL_usage.pdf *Semantic Web Development Platform - powL*, www.powLsourceforge.net. 14. <http://www.w3.org/2004/OWL/>. 15. <http://www.w3.org/RDF/>