

РОЗДІЛ V. ІНФОРМАЦІЙНО-КОМП'ЮТЕРНІ ТЕХНОЛОГІЇ

УДК 004.94

В.В. Казимир, д-р техн. наук

М.В. Харченко, асистент

Чернігівський національний технологічний університет, м. Чернігів, Україна

ТЕХНОЛОГІЯ ПОБУДОВИ ВБУДОВАНИХ ПРОГРАМНИХ МОДЕЛЕЙ БІЗНЕС-ПРОЦЕСІВ НА ОСНОВІ ЇХ XML-ОПИСІВ

В.В. Казимир, д-р техн. наук

М.В. Харченко, асистент

Черниговский национальный технологический университет, г. Чернигов, Украина

ТЕХНОЛОГИЯ ПОСТРОЕНИЯ ВСТРАИВАЕМЫХ ПРОГРАММНЫХ МОДЕЛЕЙ БИЗНЕС-ПРОЦЕССОВ НА ОСНОВЕ ИХ XML-ОПИСАНИЙ

Volodymyr Kazymyr, Doctor of Technical Sciences

Maryna Kharchenko, assistant

Chernihiv National University of Technology, Chernihiv, Ukraine

THE TECHNOLOGY OF CREATING EMBEDDED SOFTWARE MODELS OF BUSINESS-PROCESSES BASED ON THEIR XML DESCRIPTIONS

Розглянуто можливості використання мови XML для побудови програмних моделей бізнес-процесів. Запропоновано метод конвертації XML-опису концептуальної моделі бізнес-процесу в XML-опис формалізованої моделі за допомогою введення додаткових структур. Розроблено метод вбудовування програмних моделей бізнес-процесів у контур управління управляючого web-порталу.

Ключові слова: бізнес-процес, технологія, вбудована модель, управління, xml.

Рассмотрены возможности использования языка XML для построения программных моделей бизнес-процессов. Предложен метод конвертации XML-описания концептуальной модели бизнес-процесса в XML-описание формализованной модели путем ввода дополнительных структур. Разработан метод встраивания программных моделей бизнес-процессов в контур управления управляющего web-портала.

Ключевые слова: бизнес-процесс, технология, встраиваемая модель, управление, xml.

Possibilities of using XML for building software models of business processes are describe in the article. The method for converting XML-description of the conceptual model of the business process to XML-description of formalized model by adding additional structures is proposed. A method of embedding software models of business processes in a control loop of a managing web-portal is developed and described in the article.

Key words: a business process, a technology, an embedded model, a management, xml.

Постановка проблеми. Проблема комплексної автоматизації управлінської діяльності дуже актуальна для кожного підприємства. Цей процес повинен відбуватися із застосуванням певного методу, який дозволив би провести зміни з максимальною ефективністю [1]. Таким методом на сьогодні є моделювання бізнес-процесів підприємств. Пропонується для побудови моделі бізнес-процесу застосовувати такі самі етапи, що і для побудови імітаційної моделі. А саме, розроблення таких моделей, як концептуальної, формальної, програмної [2]. У роботі [3] авторами запропоновано використовувати формальний апарат управляючих Е-мереж для формалізації бізнес-процесів. Щоб мати змогу створити за отриманим формалізованим описом програмну модель бізнес-процесу і в подальшому використовувати її для реалізації концепції управляючих web-порталів [4], необхідно розробити технологію перетворення формалізованої моделі у програмний код алгоритму виконання бізнес-процесу.

Аналіз останніх досліджень і публікацій. Питання моделювання й автоматизації управління бізнес-процесами розглядається у працях Б. Старінського, Л.В. Кузнецової, А.В. Ніколаєва, О.І. Максимова, А.Є. Глухова [5] та ін. Однак, незважаючи на свою актуальність, рівень дослідження питання вбудовування моделей бізнес-процесів у контур управління управляючих web-порталів не достатній.

Мета статті. Головною метою цієї статті є розроблення інформаційної технології побудови програмної моделі бізнес-процесу, яка може бути вбудована в контур управління управляючого web-порталу.

Виклад основного матеріалу. Графічно представлені концептуальна і формальна моделі дозволяють якнайкраще відобразити логіку виконання бізнес-процесу, але їх не можна застосовувати для управління процесами. У зв'язку з цим виникає необхідність перетворення даних моделей у такий формат, який можна використовувати у подальшому для реалізації інформаційної підтримки управління бізнес-процесами.

Використання XML для опису бізнес-процесів

Кожна модель бізнес-процесу має ієрархічну структуру, яка складається із послідовності виконуваних етапів (операцій), станів, ресурсів (виконавців бізнес-процесу та інформаційних ресурсів), товарів і послуг. Як відомо, дані такого типу легко представити у вигляді структурованого опису даних. До того ж одним із об'єктів управління бізнес-процесів є документи. Згідно зі стандартом ISO 23081-1:2006 [6] для управління документами, бізнес-правилами та бізнес-процесами використовуються їх метадані. У стандарті визначено, що структура таких метаданих описується на мові XML [7].

На рис. 1 у вигляді схеми показана ієрархія структури документа XML, що містить кореневий елемент, декілька вузлових елементів, а також вузлів нижчого рівня і так далі.

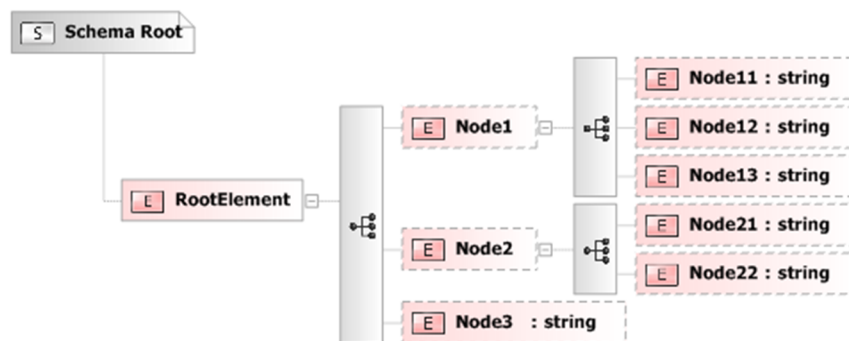


Рис. 1. Ієрархія структури документа XML

XML – це ієрархічна структура, призначена для зберігання будь-яких даних, візуально структура яких може бути представлена як дерево. Найважливіша обов'язкова синтаксична вимога – те, що документ має тільки один кореневий елемент. Текст або інші дані всього документа мають бути розташовані між єдиним початковим кореневим тегом і відповідним йому кінцевим тегом. Таким чином, як мову опису бізнес-процесів оберемо XML.

Технологічно процес побудови програмних моделей бізнес-процесів і їх вбудовування в управляючий web-портал полягає у послідовному виконанні шести етапів.

1. Визначення XML-опису концептуальної моделі бізнес-процесу.
2. Розширення XML-опису концептуальної моделі з урахуванням формального апарату управляючих E-мереж і отримання XML-опису формалізованої моделі бізнес-процесу.
3. Визначення XML-структури управляючого web-порталу.
4. Додаткова розмітка XML-документів.
5. Визначення піддокументів.
6. Відображення піддокументів.

XML-опис концептуальної моделі бізнес-процесу можна легко отримати за допомогою експорту її графічного представлення (IDEF0 та IDEF3) засобами системи моделювання бізнес-процесів, наприклад, BPWin.

Реалізація другого етапу відбувається за методом конвертації, що описується нижче.

Етапи 3–6 також описуються нижче в методі вбудовування моделей бізнес-процесів у контур управління управляючого web-порталу.

Метод конвертації XML-опису концептуальної моделі в XML-формат формалізованої моделі бізнес-процесу

Суть методу полягає в конвертації описів концептуальних моделей бізнес-процесів із стандартного формату XML у спеціалізований формат [8]. Оскільки формалізована модель будується із використанням формального апарату управляючих Е-мереж, то для перетворення використаємо запропоноване у роботі [9] розширення визначенням CEN (Contol E-Nets – управляючі Е-мережі) формату Petri Net Markup Language (PNML). Основу схеми визначення мережі складають: схема визначення позиції, схема визначення переходу і схема визначення дуги.

На рис. 2–5 наведені схеми PNML-визначень основних компонентів CEN.

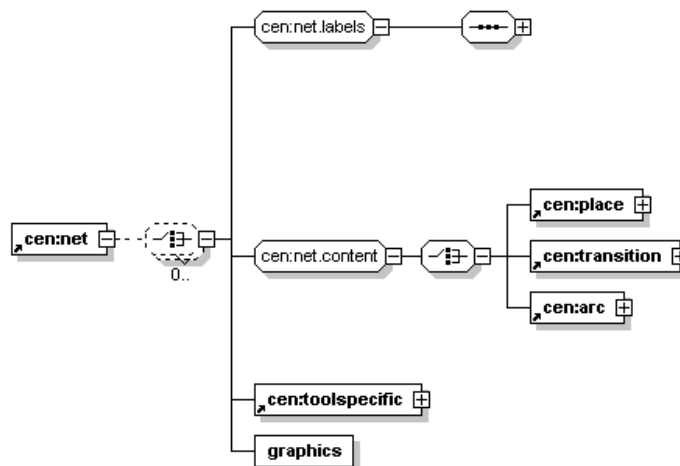


Рис. 2. Схема PNML – визначення мережі

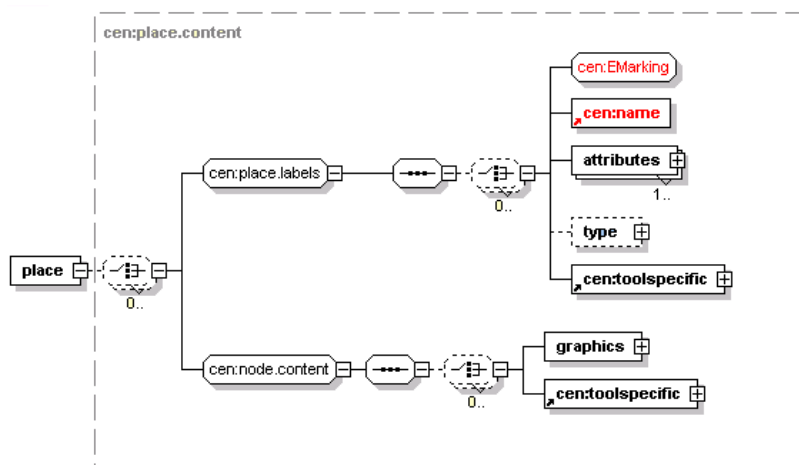


Рис. 3. Схема PNML – визначення позиції

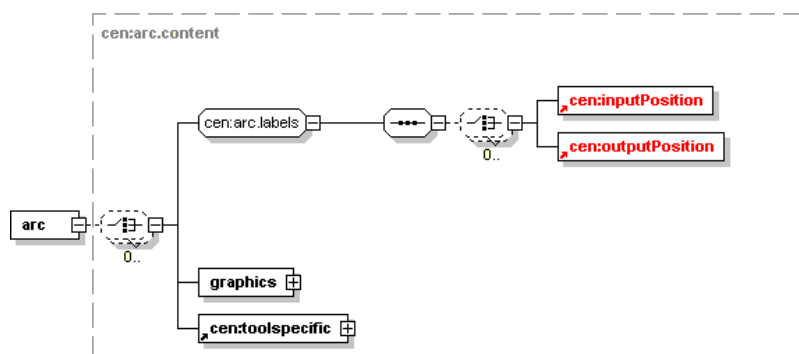


Рис. 4. Схема PNML – визначення дуги

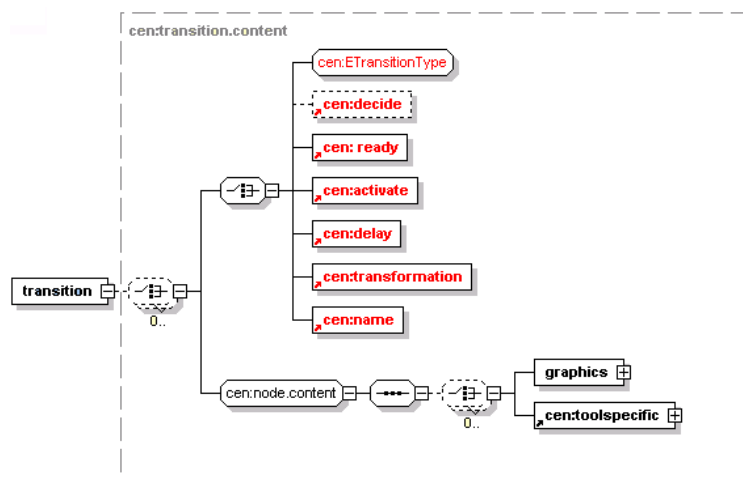


Рис. 5. Схема PNML – визначення переходу

На підставі наведених схем генерується XSD-файл, який використовується для конвертації XML-опису концептуальної моделі у XML-опис CEN-моделі (формалізованої моделі у форматі PNML).

Кожному елементу (позиції, переходу, дузі) формалізованої моделі підібрано шлях для збереження даних про модель у файлі PNML. Наприклад:

а) одній позиції елементу:

1) у файлі PNML відповідає шлях виду Xpath:

“pnml\content\net\content\place”;

2) Xpath у файлі XML:

“ProcessModeler\PMModel\PMDiagram_Groups\PMDiagram\PMBox_Groups\PMBox”;

б) дуга матиме вигляд:

1) Xpath у файлі PNML: “pnml\content\net\content\arc”;

2) Xpath у файлі XML:

“ProcessModeler\PMModel\PMActivity_Groups\PMActivity”;

в) перехід матиме вигляд:

1) Xpath у файлі PNML: “pnml\content\net\content\transition”;

2) Xpath у файлі XML:

“ProcessModeler\PMModel\PMJunction_Groups\PMJunction”.

Під час конвертації формалізованого опису моделі бізнес-процесу у форматі XML відбувається пошук та поєднання необхідних даних про об’єкти моделі на основі закладеної в конверторі інформації про місцезнаходження та відповідності метаданих.

У процесі читання файлу виконується послідовне визначення об’єктів:

– позицій – основних елементів діаграм;

– переходів;

– взаємозв’язків між позиціями і переходами на основі посилань, що трапляються у властивостях дуг та позицій.

Коли всі знайдені дані структуровані необхідним чином, так що є елементи, переходи та посилання між ними, виконується запис у файл PNML кожної позиції, кожної структури у такому порядку: позиції, переходи, дуги за встановленими правилами та визначеними шляхами Xpath. Усі відповідні властивості заповнюються на основі відповідностей об’єктів.

Тому позиція, що у початковому файлі XML мала опис у вигляді:

<PMBox id="{517322F2-E663-422A-AA37-9F95F3955E95}+00000000"

Name="Prog">

<PMBoxProps>

```

    <PMBoxCoordinates member_name="left">695</PMBoxCoordinates>
    <PMBoxCoordinates member_name="top">100</PMBoxCoordinates>
    <PMBoxCoordinates member_name="right">760</PMBoxCoordinates>
    <PMBoxCoordinates member_name="bottom">215</PMBoxCoordinates>
    <PMActivityRef> {0FEBE7B1-F9BD-4C83-BBE0-23E30728E745}+00000000
</PMActivityRef>
    <PMUowIdNumber>2</PMUowIdNumber>
    <PMPersistentBoxNumber />
    <PMDrawStyle>0</PMDrawStyle>
    <PMObjectFontReference> {9ACD20C4-C74C-4099-B9BF-1E6880774C22}+
00000000</PMObjectFontReference>
    <PMTextColorReference />
    <PMTitleColorReference />
    <PMBackgroundColorReference />
    <PMBitmapRef> {21B92027-26A1-4D4C-A91C-
82295C501E6E}+00000000</PMBitmapRef>
    <PMBitmapJustification>0</PMBitmapJustification>
    <PMShapeId>0</PMShapeId>
</PMBoxProps>
</PMBox>,
у новому файлі має вигляд:
<tns:pnm1 xmlns:tns="http://www.stu.chernihiv.ua/CEN"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.stu.chernihiv.ua/CEN E:\CEN_PNML.xsd" >
  <tns:place id="517322F2-E663-422A-AA37-9F95F3955E95">
    <tns:graphics>
      <tns:position>
        <attribute name="x">695</attribute>
        <attribute name="y">100</attribute>
      </tns:position>
    </tns:graphics>
    <tns:attributes>
      <tns:attribute>
        <tns:name>Prog</tns:name>
      </tns:attribute>
    </tns:attributes>
    <tns:attributes>
      <tns:attribute>
        <tns:type>Position</tns:type>
      </tns:attribute>
    </tns:attributes>
  </tns:place>
</tns:pnm1>.

```

Отже, за XSD-схемою, на яку є посилання у шапці створеного файла «E:\CEN_PNML.xsd», можна створити PNML-файл.

Запропонований метод дає змогу конвертувати XML-опис концептуальної моделі бізнес-процесу у XML-опис формалізованої моделі у форматі PNML і вбудовувати отриману програмну модель в управляючий web-портал.

Метод вбудовування моделей бізнес-процесів у контур управління управляючого web-порталу

Цей метод ґрунтується на запропонованому у статті [10] підході, за яким функціонування управляючого web-порталу описується формальними моделями бізнес-процесів, представленими у форматі XML. У загальному випадку весь процес роботи web-порталу можна представити у вигляді ієрархічної структури виду:

```

<root><!-- суперкористувач-->
  <admin><!--адміністратор системи -->
    <post><!-- посада-->
    <doctype><!-- тип документа-->
    <department><!-- відділ -->
      <user><!-- співробітник-->
        <file><!--завантажений файл -->
        </file>
      </user>
      <customer><!--співробітник, що ініціює процес -->
      </customer>
    </department>
    <model>
      <alias><!-- екземпляри документів, що будуть переміщатись у моделі-->
        <doctype><![CDATA[link to /root/admin/doctype]]></doctype>
      </alias>
      <defination><!--визначення структури основного документа моделі -->
      <config><!--налаштування моделі (початковий етап, атрибути і файли,
доступні після завершення роботи моделі)-->
      <action><!--дія (етап)-->
        <subaction><!-- піддія-->
          <action/><!--посилання на наступну дію -->
        </subaction>
        <responsible><!-- виконавець дії-->
        </responsible>
      </action>
      ...
    </model>
  </admin>
</root>.

```

Згідно із запропонованим методом наведена структура абстрактного XML-документа розділяється на окремі складові, які зберігаються у БД. Застосовуючи технологію XML, один запис БД зберігає разом дані і метадані з проекції назв об'єктів і їх атрибутів на XML-теги, а даних – на значення тегів. Оскільки об'єкти (складові) можуть бути пов'язаними, передбачено створення посилань між ними. Для цього у БД створена окрема таблиця. Таким чином, є БД, в якій у вигляді текстових записів зберігаються XML-структури, що містять відомості про моделі процесів, документи і дані користувачів, а також у БД зберігаються зв'язки між цими даними. Нижче як приклад наводиться запис про модель:

```

<name>Модель рецензування</name>
<author>Науково-дослідна частина </author>
<description>Модель процесу рецензування авторефератів, дисертацій та науково-методичної літератури </description>.

```

Наведена структура явно не пов'язана з іншими даними, але використовуючи таблицю зв'язків створюється абстрактний XML-документ, структура якого була наведена вище.

Визначення значень складових тегу `model` відбувається завдяки конвертації XML-опису формалізованої моделі бізнес-процесу. Так, атрибути тегу `definition` визначаються із значення атрибутів мітки в початковій позиції формалізованої моделі. Атрибути тегу `config` отримують свої значення із початкового переходу і кінцевої позиції. Тегу `action` відповідають дії, що виконуються під час переходу, тег `subaction` визначається управляючою функцією переходу. Тег `responsible` отримує своє значення з атрибуту «виконавець» мітки управляючої Е-мережі.

Оскільки у БД дані зберігаються у форматі XML, інтерфейс користувача також оперує даними у форматі XML. Для цього краще всього підходять дві технології – XSL та XSLT [8].

Використовуючи розроблену БД та технологію зберігання об'єктних даних, розроблення інтерфейсу користувача можна поділити на два етапи: формування даних у форматі XML згідно з моделлю відображення; перетворення підготовлених даних у формат HTML.

Формування даних являє собою підготовку DOM-документа, що містить увесь необхідний набір даних, який буде в подальшому перетворений у web-сторінку. Для побудови ієрархічного документа був розроблений набір спеціальних інструкцій для формату XSL, після оброблення яких здійснювалось підключення інших DOM-документів і їх об'єднання в один XML-документ.

Сформовані дані передаються в XSLT-шаблон, який їх обробляє та виконує перетворення у web-сторінку.

Приклад інструкцій вибірки даних з БД і формування XML-документа наведено нижче. В ньому формується список моделей.

```
<xsl:stylesheet xmlns:xsl="http://www.w3.org/1999/XSL/Transform" version="1.0">
  <xsl:template match="/">
    <xsl:element name="models">
      <xsl:attribute name="id"><xsl:value-of select="*/@id" /></xsl:attribute>
      <xsl:element name="model">
        <xsl:attribute name="action">link</xsl:attribute>
        <xsl:attribute name="tmpl">short</xsl:attribute>
        <xsl:attribute name="doctype">model</xsl:attribute>
        <xsl:element name="criteria"> ( type='model' ) order by cdate ASC
        </xsl:element>
      </xsl:element>
    </xsl:element>
  </xsl:template>
</xsl:stylesheet>
```

Результатом вибірки з БД буде наступний XML-документ з даними:

```
<?xml version="1.0"?>
<models id="-1" oid="101" pid="101" utype="admin" v="1">
  <model id="560" pid="101" utype="admin" v="1">
    <name> Модель рецензування </name>
    <description> Модель процесу рецензування авторефератів, дисертацій та науково-методичної літератури </description>
    <author> Науково-дослідна частина </author>
    <used count="2"/>
  </model>
</models>
```

Подальша робота з отриманими даними відбувається за допомогою набору XSLT-шаблонів, на основі яких будується інтерфейс користувача. Використання XSLT-шаблонів значно впливає на швидкість розроблення інтерфейсу, оскільки виокремлюючи частини інтерфейсу, що найчастіше використовуються, в окремі XSLT-шаблони, а потім, використовуючи їх шляхом комбінування, можна швидко та ефективно створювати однотипні web-сторінки. Наприклад, web-сторінки у великому об'ємі містять різні елементи управління: кнопки, поля вводу, форми, меню, таблиці та ін. Визначивши заздалегідь такі частини інтерфейсу, зникає необхідність розробляти їх знову. Достатньо під час проектування інтерфейсу задавати відповідні XSLT-шаблони його частин. XSLT-парсер, у свою чергу, під час розбору шаблонів web-сторінок буде замінювати виклики шаблонів частин інтерфейсу відповідними елементами.

Для прикладу нижче наведено фрагмент коду, що відповідає за формування посилання на сторінку із виводом списку моделей бізнес-процесів. У ньому атрибут `action` – одна із заздалегідь заданих дій, яку може виконати користувач. У такому випадку він встановлюється у значенні `view`, що означає відображення даних. Атрибут `doctype` визначає тип зв'язку, в цьому випадку це список моделей, відповідно `models`. Атрибут `tmpl` задає шаблон, в якому міститься запит до БД для отримання списку моделей і код формування XML-документа. Атрибут `xslt` визначає XSLT-шаблон, за правилами якого буде сформована web-сторінка з результатами (списком моделей). Якщо цей шаблон буде містити посилання, наприклад, для більш детального перегляду моделі, відповідно він буде містити схожі виклики формування посилань для побудови інших web-сторінок.

```
<xsl:call-template name="design_url">
  <xsl:with-param name="action">view</xsl:with-param>
  <xsl:with-param name="doctype">models</xsl:with-param>
  <xsl:with-param name="tmpl">list</xsl:with-param>
  <xsl:with-param name="xslt">page_models_view.xslt</xsl:with-param>
</xsl:call-template>
```

Запропонований метод дозволяє вбудовувати моделі бізнес-процесів в управляючий web-портал та управляти його роботою. До того ж його використання дозволяє збільшити швидкість розроблення подібних систем за рахунок використання технологій XML, XSL, XSLT.

Висновки і пропозиції. Розглянуто можливості мови XML для програмного опису концептуальної та формалізованої моделей бізнес-процесів.

Враховуючи те, що формалізована модель бізнес-процесу будується із використанням формального апарату управляючих Е-мереж, запропоновано метод конвертації XML-опису концептуальної моделі в XML-формат формалізованої моделі за допомогою введення додаткових визначень (структур) XML для відображення елементів управляючих Е-мереж (позицій, міток, переходів).

Розроблено метод вбудовування програмних моделей бізнес-процесів у контур управління управляючого web-порталу.

Список використаних джерел

1. Бусыгин А. В. Эффективный менеджмент / А. В. Бусыгин. – М. : Финпресс, 2000. – 1056 с.
2. Шабашов В. М. Імітаційне моделювання складних систем : навч. посіб. / В. М. Шабашов, В. Я. Марецький. – Донецьк : РВВ ДНУ, 2000. – 72 с.
3. Казимир В. В. Модельно-орієнтована інформаційна система підтримки управління якістю ВНЗ / В. В. Казимир, М. В. Харченко // Вісник Чернігівського державного технологічного університету. Серія «Технічні науки» : науковий збірник. – 2012. – № 1 (55). – С. 140–147.
4. Казимир В. В. Управляючі веб-портали та їх властивості / В. В. Казимир, М. В. Харченко // Вісник Чернігівського державного технологічного університету. Серія «Технічні науки» : науковий збірник. – 2015. – № 1 (77). – С. 75–82.

5. *Способы* оптимизации бизнес-процессов для автоматизации управления компанией / Л. В. Кузнецова, А. В. Николаев, О. И. Максимов, А. Е. Глухов // Технические науки. Информатика, вычислительная техника. – 2008. – № 3. – С. 73–81.

6. *ISO 23081-1:2006*. Information and documentation – Records management processes – Metadata for records – Part 1: Principles.

7. *Либерти Дж.* Создание документов XML для WEB : учеб. пособие / Дж. Либерти, М. Крейли. – М. : Вильямс, 2000. – 256 с.

8. *Казимир В. В.* Моделювання робочих процедур в системі підтримки управління якістю / В. В. Казимир, М. В. Харченко // Математичне та імітаційне моделювання систем. МОДС'2011 : пошта науково-практична конференція з міжнародною участю : тези доповідей (27 червня–30 червня). – Чернігів, 2011. – С. 438–442.

9. *Казимир В. В.* Модельно-ориентированное управление интеллектуальными производствами и системами : дис. ... д-ра техн. наук : спец. 05.13.06 / Казимир Владимир Викторович. – К., 2005. – 328 с.

10. *Казимир В. В.* Метод проектирования Internet-ориентированных информационно-управляющих систем на основе стандарта XML / В. В. Казимир, В. М. Мирошниченко, А. В. Пастухов // Проблемы программирования. – 2002. – № 1–2. Специальный выпуск. – С. 326–332.

УДК 004.42:004.75

С.Д. Точилин, канд. физ.-мат. наук

Запорожский национальный технический университет, г. Запорожье, Украина

АНАЛИЗ ПРОИЗВОДИТЕЛЬНОСТИ SOAP И RESTFUL JAVA WEB-СЛУЖБ ПОИСКА В ДАННЫХ СУБД POSTGRESQL

С.Д. Точилін, канд. фіз.-мат. наук

Запорізький національний технічний університет, м. Запоріжжя, Україна

АНАЛІЗ ПРОДУКТИВНОСТІ SOAP ТА RESTFUL JAVA WEB-СЛУЖБ ПОШУКУ В ДАНИХ СКБД POSTGRESQL

Sergey Tochilin, PhD in Physico-mathematical Sciences

Zaporozhye National Technical University, Zaporozhye, Ukraine

PERFORMANCE ANALYSIS OF SOAP AND RESTFUL JAVA WEB SERVICES FOR DATA SEARCH IN POSTGRESQL DBMS

Разработаны SOAP и RESTful Web-службы поиска в данных СУБД, а также программа-тестер для определения их производительности. С помощью созданных программ получены данные о производительности Web-служб при обработке различных объемов информации, хранящейся в СУБД PostgreSQL. Анализ данных тестирования был выполнен на основе математической аппроксимации. Полученные результаты свидетельствуют как о существенном влиянии технологии разработки службы на ее производительность, так и о целесообразности использования RESTful Web-служб при создании SOA-приложений для работы с СУБД PostgreSQL.

Ключевые слова: производительность, SOAP, RESTful, JAX-WS, JAX-RS, Web-служба, PostgreSQL.

Розроблені SOAP та RESTful Web-служби пошуку в даних СКБД, а також програма-тестер визначення їх продуктивності. За допомогою створених програм отримані дані про продуктивність Web-служб під час оброблення різних обсягів інформації, яка зберігається в СКБД PostgreSQL. Аналіз даних тестування був виконаний на основі математичної апроксимації. Отримані результати свідчать як про істотний вплив технології розроблення служби на її продуктивність, так і про доцільність використання RESTful Web-служб під час створення SOA-додатків для роботи з СКБД PostgreSQL.

Ключові слова: продуктивність, SOAP, RESTful, JAX-WS, JAX-RS, Web-служба, PostgreSQL.

SOAP and RESTful Web services for data search in the DBMS, as well as, a test program for determination of their performance had been developed. By means of created software applications obtained Web services performance data, while processing different amount of information, saving in PostgreSQL DBMS. Data analysis of test was executed on the basis of mathematical approximation. The results indicate a significant influence of service technology on its performance, as well as the feasibility of using RESTful Web Services by developing SOA applications for PostgreSQL DBMS.

Key words: performance, SOAP, RESTful, JAX-WS, JAX-RS, Web service, PostgreSQL.

Введение. В настоящее время интенсивно развиваются технологии создания различного рода распределенных приложений, в частности, особое внимание уделяется программной реализации и оптимизации работы Web-служб.