РАЗДЕЛ 5. ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ

УДК 004.021

Сейдаметова З. С.

ЗАДАЧА ОПРЕДЕЛЕНИЯ ПОДОБИЯ ДОКУМЕНТОВ

У статті представлені алгоритм визначення відстані між документами, реалізація його модулів на мові програмування Python. Проведена оцінка ефективності роботи алгоритму.

Ключові слова: *відстань між документами, вектор документа, метрика відстані, ефективність алгоритму, Python.*

В статье представлен алгоритм определения расстояния между документами, реализация его модулей на языке программирования Python. Проведена оценка эффективности работы алгоритма.

Ключевые слова: *расстояние между документами, вектор документа, метрика расстояния, эффективность алгоритма, Python.*

We present an algorithm for the document distance problem and it implementation using Python. We provide running time of the algorithm's operations.

Key words: *document distance, document vector, distance metric, running time, Python.*

Постановка проблемы. XXI век характеризуется ростом объема текстовых документов, сосредоточенных в Интернете, в распределенных базах данных, виртуальных библиотеках и т. п. Это влечет необходимость использования алгоритмов быстрого поиска данных, разработке методов информационного поиска и работы с файловыми структурами. В связи с этим становится актуальной и востребованной задача изучения близости или подобия двух документов. Эта проблематика имеет весьма широкое применение – определение схожести двух документов, выявление плагиата/дублирования, поиск в Интернете и базах данных, ранжирование документов в поисковых системах и т. п. Использование таких программ представляется почти обязательным при оценивании степени оригинальности студенческих работ, часто страдающих от плагиата.

Анализ публикаций. В работе Энтони Донсета [1] рассматриваются проблемы извлечения, выбора и анализа последовательности слов в предметно-независимой коллекции текстовых документов на любом естественном языке. Автором представлено описание структуры документа, а также методика эффективной обработки коллекции документов любого размера.

В диссертациях [2; 3] приведены методы оценки тематического подобия различных документов, а также представлены алгоритмы, модели, методы, программное обеспечение автоматической классификации текстовых документов. В диссертационном исследовании [4] представлена методика определения подобия веб-документов с помощью структурно-семантического разбиения документа.

В монографии по интеллектуальному анализу данных (data mining) [5, с. 487–568] приведены основополагающие концепции и алгоритмы кластерного анализа, в том числе для распознавания географических карт, рисунков, определения связности документов.

В статьях [6; 7] рассмотрены математические постановки задач нормализации веб-расстояния и подобия в текстовых документах, а также вопрос информационной близости двух документов. В статье [7] представлены методы использования структурного подобия и метрики в обучении.

В работах [8; 9] описано, каким образом используется теория подобия/схожести документов в поисковой системе Google, а для определения схожести документов введен термин «функция нормализованного расстояния Google» – NGD (Normalized Google Distance). Функция NGD(x, y) измеряет насколько слово x близко к слову y в шкале от 0 до бесконечности и определяется формулой:

$$\mathrm{NGD}(x, y) = \frac{\max\{\log(f(x)), \log((f(y))\} - \log f(x, y)}{\log M - \min\{\log(f(x)), \log((f(y))\}},$$

где f(x) и f(y) – число найденных веб-страниц, где встречаются слова x и y, соответственно, а M – общее число веб-страниц, индексированных Google. Если два слова полностью совпадают, то расстояние равно 0; если они независимы, то – 1. Если же два слова никогда, ни в одном документе при поиске в Google не встречаются, то расстояние стремится к бесконечности.

Цель данной **статьи** – описание модели векторного пространства для определения подобия документов, разработка алгоритма и анализ его эффективности.

Изложение основного материала.

Основные понятия. Пусть D представляет собой некоторый текстовый документ. Под *сло-*

вом будем понимать последовательность буквенно-числовых обозначений, таких как «Достоевский», «2012». Не будем различать заглавные и строчные буквы, следовательно, «Достоевский» и «достоевский» являются одним и тем же словом. Слова оканчиваются не буквенно-числовыми символами, т. е. «информационно-коммуникационный» состоит из двух слов: «информационно» и «коммуникационный».

Мы можем рассматривать частоту распределения D в виде вектора с одной компонентой на возможное слово. Каждая компонента вектора неотрицательное целое число.

Норма этого вектора определяется:

$$\|D\| = \sqrt{D \cdot D} = \sqrt{\sum_w D^2(w)} \cdot$$

Скалярное произведение двух векторов D_1 и D_2 определяется следующим образом:

$$D_1 \cdot D_2 = \sum_w D_1(w) \cdot D_2(w) \cdot$$

Определим *расстояние между двумя документами* как угол между двумя векторами, компонентами которых являются частоты распределения слов в тексте.

Угол θ между двумя векторами D_1 и D_2 определяется: $\theta(D_1, D_2) = \arccos\left(\frac{D_1 \cdot D_2}{\|D_1\| \cdot \|D_2\|}\right)$,

где $\theta(D_1, D_2) \in \left[0; \frac{\pi}{2}\right]$.

Кроме того,

- $\theta(D, D) = 0$ для $\forall$ векторов D;
- $\theta(D_1, D_2) = \frac{\pi}{2}$, если вектора D_1 и D_2 не имеют общих слов и являются ортогональными.

Примеры.

1. Пусть даны два документы D_1 и D_2: D_1 = ’the cat’ и D_2 = ’the dog’. Следовательно, D_1 = {1,0,1}, D_2 = {0,1,1}, а скалярное произведение $D_1 \cdot D_2 = 1$ и $\theta(D_1, D_2) = \arccos\frac{1}{2} = \frac{\pi}{3} \approx 1{,}047$. Следовательно, расстояние между документами D_1 и D_2 составляет 1,047.

2. Пусть даны два документы D_1 и D_2: D_1 = ’the cat the cat’ и D_2 = ’the dog the dog’. Следовательно, D_1 = {2,0,2}, D_2 = {0,2,2}, а скалярное произведение $D_1 \cdot D_2 = 4$ и $\theta(D_1, D_2) = \arccos\frac{1}{2} = \frac{\pi}{3} \approx 1{,}047$. В этом примере расстояние между двумя рассматриваемыми документами такое же, как и в примере 1.

Таким образом, можно сделать вывод, что если провести нормализацию векторов D_1 и D_2 для каждого из представленных выше примеров, то окажется, что эти два примера эквивалентны.

В примерах 1 и 2 при формировании векторов D_1 и D_2 используется теория, представленная в [10].

На рис. 1 представлена модель векторного пространства, имеющая отношение к обоим представленным выше примерам. Векторы OA и OB относятся к примеру 1, а OA_1 и OB_1 – к примеру 2.

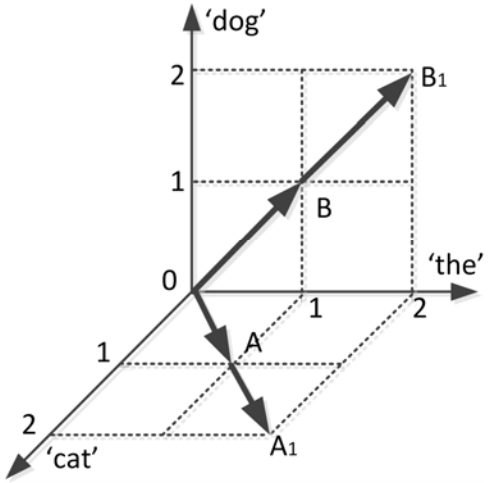


Рис. 1. Модель векторного пространства.

Постановка задачи. *Задача определения расстояния (схожести/подобия) между документами* представляет собой задачу вычисления расстояния между двумя данными текстовыми документами.

Input: два текстовых документа, состоящие из $|w_{D_1}|$ и $|w_{D_2}|$ слов.

Output: расстояние между двумя рассматриваемыми документами, представленное в радианах.

Словесное описание данного алгоритма выглядит следующим образом.

1: Инициализировать список документов
2: Разбить каждый документ на слова
– Для каждой строки в документе:
  - Для каждого символа в строке:
    - Если символ не буквенно-цифровой
      - Присоединить предыдущее слово, если оно есть
      - Взять следующее слово

3: Подсчитать частоты появления слов (определить вектора документов):
a. Отсортировать слова в списке
b. Для каждого слова в списке слов:
  – Если такое же, как предыдущее слово:
    - Увеличить счетчик
  – Иначе:
    - Добавить последнее слово и его счетчик в список
    - Сбросить счетчик на 0

4: Вычислить скалярное произведение:
– Для каждого слова в первом документе (в отсортированном порядке):

- Если оно появляется во втором документе:
  - Перемножить частоты слов
  - Добавить к общей сумме.

Реализация основных модулей алгоритма на языке программирования Python (использована монография [11]) представлена ниже.

Алгоритм 1 (отдельные фрагменты кода на Python).
Определение подобия документов (расстояние между документами)

1: Инициализация документа

```
def read_file(filename):
    read the text file;
    return список строк в тексте файла.
    try:
        fp = open(filename)
        L = fp.readlines()
    except IOError:
        print "Ошибка в чтении файла: ",filename
        sys.exit()
    return L
```

2: Разбить каждый документ на слова

```
def get_words_from_line_list(L):
    разбивка данного списка L строк текста в слова.
    return список всех найденных слов.

    word_list = []
    for line in L:
        words_in_line = get_words_from_string(line)
        word_list.extend(words_in_line)
    return word_list

def get_words_from_string(line):
    return список слов в данном вводе строки,
    конвертирование каждого слова в строчное.

    Input:  line (a string)
    Output: a list of strings
            (каждая строка – последовательность буквенно-цифровых символов)
    word_list = []          # накапливание слов в строчке
    character_list = []     # накапливание символов в слове
    for c in line:
        if c.isalnum():
            character_list.append(c)
        elif len(character_list)>0:
            word = string.join(character_list,"")
            word = string.lower(word)
            word_list.append(word)
            character_list = []
    if len(character_list)>0:
        word = string.join(character_list,"")
        word = string.lower(word)
        word_list.append(word)
   return word_list
```

3: Подсчет частот появления слов (построение векторов документов)

```
def count_frequency(word_list):
    return список данных пар по форме: (word, frequency)
    L = []
    for нового слова в списке слов:
        for entry in L:
            if new_word == entry[0]:
                entry[1] = entry[1] + 1
                break
        else:
            L.append([new_word,1])
    return L
```

```
def insertion_sort(A):
    Сортировка списка A по порядку, без привлечения дополнительной памяти
    (в данном случае используется алгоритм сортировки вставками).
    for j in range(len(A)):
        key = A[j]
        # вставить A[j] отсортированную последовательность A[0..j-1]
        i = j-1
        while i>-1 and A[i]>key:
            A[i+1] = A[i]
            i = i-1
        A[i+1] = key
    return A

def word_frequencies_for_file(filename):
    return отсортированный по алфавиту список пар (word, frequency)
    for данного файла.

    line_list = read_file(filename)
    word_list = get_words_from_line_list(line_list)
    freq_mapping = count_frequency(word_list)
    insertion_sort(freq_mapping)

    print "File",filename,":",
    print len(line_list),"lines,",
    print len(word_list),"words,",
    print len(freq_mapping),"distinct words"

    return freq_mapping
```

4: Вычисление скалярного произведения

```
def inner_product(L1,L2):
    Внутреннее произведение между двумя векторами; векторы представлены как
    отсортированные в алфавитном порядке (word, frequency).
    sum = 0.0
    i = 0
    j = 0
    while i<len(L1) and j<len(L2):
        # L1[i:] & L2[j:] еще не обработаны
        if L1[i][0] == L2[j][0]:
            # оба вектора содержат это слово
            sum += L1[i][1] * L2[j][1]
            i += 1
            j += 1
        elif L1[i][0] < L2[j][0]:
            # слово L1[i][0] имеется в L1, но отсутствует в L2
            i += 1
        else:
            # слово L2[j][0] имеется в L2, но отсутствует в L1
            j += 1
    return sum
```

Оценка времени работы алгоритма. Операция 1 «Инициализация документа» выполняется за время, равное количеству строк в документе или размеру документа, т. е. стоимость этой операции равна $T_1 = O\left(\sum_i |\text{line}_i|\right) = \Theta(w)$.

Стоимость операции 2 «Разбиение каждого документа на слова» равна стоимости предыдущей операции, т. е. ее стоимость составляет $T_2 = O\left(\sum_i |\#\text{words in line}_i|\right) = \Theta(w^2)$.

Операция 3 «Подсчет частот появления слов» выполняется за $T_3 = O(1) + O(w^2) + O\left(\sum_i |\text{word}_i|\right) = \Theta(w^2)$. В этой операции время выполнения пропорционально квадрату количества слов, поскольку в данном алгоритме для сортировки слов используется алгоритм сортировки вставками; использование другого более быстрого алгоритма, например, Merge-sort, позволит выполнить сортировку за O(w1gw).

Операция 4 «Вычисление скалярного произведения» – $T_4 = O(w_1 w_2)$.

Выводы. Представленный алгоритм определения схожести/подобия двух документов работает за квадратичное время, и чтобы его улучшить, необходимо применение других методов, структур данных и другого способа представления исходных массивов данных.

ЛИТЕРАТУРА

1. Doucet A. Advanced Document Description, a Sequential Approach / A. Doucet / Report A-2005-2. – Finland : University of Helsinki, 2005. – 161 p.
2. Добрынин В. Ю. Оценка тематического подобия текстовых документов / В. Ю. Добрынин, В. В. Клюев, И. С. Некрестьянов // Электронные библиотеки: перспективные методы и технологии : II Всерос. науч. конф. – СПб., 2000. – С. 54–62.
3. Шабанов В. И. Модели и методы автоматической классификации текстовых документов : дис. … канд. техн. наук : спец. 05.13.11 / Владислав Игоревич Шабанов. – М., 2003. – 227 с.
4. Косинов Д. И. Разработка математического обеспечения оценки схожести WEB-документов на основе структурно-семантического разбиения : дис. … канд. техн. наук : спец. 05.13.11 / Дмитрий Иванович Косинов. – Воронеж, 2008. – 146 с.
5. Tan P.-N. Introduction to Data Mining / P.-N. Tan, M. Steinbach, V. Kumar. – Boston, MA : Addison-Wesley, 2006. – 769 p.
6. Cilibrasi R. L. Normalized Web Distance and Word Similarity / R. L. Cilibrasi, P. M. B. Vitanyi // arXiv:0905.4039v1 [cs.CL]. – 2009. – 21 p.
7. Wang J. Structural Similarity and Distance in Learning / J. Wang, V. Saligrama, D. A. Castañón // arXiv:1110.5847v1 [stat.ML]. – 2011. – 9 p.
8. Cilibrasi R. L. The Google Similarity Distance / R. L. Cilibrasi, P. M. B. Vitanyi // IEEE Transactions on knowledge and data engineering. – 2007. – V. 19. – № 3. – P. 370–383.
9. Kjos-Hanssen B. Google distance between words / B. Kjos-Hanssen, A. J. Evangelista // arXiv:0901.4180 [cs.CL]. – 2009. – 3 p.
10. Salton G. A vector space model for automatic indexing / G. Salton, A. Wong, C. S. Yang // Communication of the ACM. – 1975. – V. 18. – № 11. – P. 613–620.
11. Miller B. N. Problem Solving with Algorithms and Data Structures using Python / B. N. Miller, D. L. Ranum. – Wilsonville, Oregon : Franklin Beedle&Associates, 2006. – 357 p.

УДК 004.658.2

Ильясова Ф. С., Москаленко И. С.

РАЗРАБОТКА АВТОМАТИЗИРОВАННОЙ СИСТЕМЫ «СТУДЕНЧЕСКОЕ ОБЩЕЖИТИЕ»

У статті розглянута і описана розробка автоматизованої системи «Студентський гуртожиток». Також розроблена структура бази даних, інтерфейс, форми, звіти, запити. Детально описаний зв'язок об'єктів бази даних. Проаналізовано роботи відомих вчених, які займалися розробками баз даних.

Ключові слова: *автоматизована система, база даних, система управління базами даних, модель даних, зв'язки між таблицями, таблиця, форма, звіт, структурування, обробка.*

В статье рассмотрена и описана разработка автоматизированной системы «Студенческое общежитие». Также разработана структура базы данных, интерфейса, формы, отчеты, запросы. Подробно описана связь объектов базы данных. Проанализированы работы известных ученых, которые занимались разработками баз данных.

Ключевые слова: *автоматизированная система, база данных, система управления базами данных, модель данных, связи между таблицами, таблица, форма, отчет, структурирование, обработка.*

Development of automatic system «Students hostel» is considered and described in the article. Also, structure of database, interface, forms, reports, queries was developed. A relation of objects in database was described in detail. Works of the famous scientist, whose involved in the development of databases, were analyzed.

Key words: *automated system, database, database management system, data model, relationships between tables, table, form, report, structuring, processing.*

Постановка проблемы. С развитием компьютерных технологий возросла потребность хранения и обмена информацией. Базы данных (БД) позволяют автоматизировать работу пользователя и облегчить его работу [1].

Создание автоматизированной системы «Студенческое общежитие» обуславливает необходимость разработки, которая должна осуществлять поиск просмотра информации о студентах; просмотр, изменение, удаление, редактирование информации о платежах; поиск просмотра информации об отдельных корпусах общежития, которые закреплены за определенным комендантом; добавление, удаление и редактирование данных о студентах, преподавателях, комнатах; вывод информации на печать, экран или в файл.

Анализ литературы. На сегодняшний день имеется несколько крупных компаний, разрабатывающих программное обеспечение для управления, организации хранения данных в БД и работы с БД. Это Oracle (http://www.oracle.com), IBM (www.ibm.com), Microsoft (www.microsoft.com), фирма «1С» (www.1c.ru).

Л. Дж. Эллисон, работая в компании Ampex (1970 г.), разработал базу данных для ЦРУ, которую он назвал «Oracle» [2].

Э. Ф. Кодд в работе «Реляционная модель данных для больших, совместно используемых банков данных» [3] предложил замену иерархической или навигационной структуры простыми таблицами, содержащими строки и столбцы, то есть описал «реляционную» модель представления данных, которая сейчас лежит в основе практически всех баз данных. Также он создал и описал концепцию реляционных баз данных и реляционную алгебру (1970); для проектирования БД предложил аппарат нормализации отношений, работая совместно с Крисом Дейтом, одним из наиболее уважаемых во всем мире экспертов и мыслителей в области технологии баз данных, за что получил премию им. Тюринга (1981).

Д. Энгельбарт заложил основы интерактивного программирования, совместимого с использованием баз данных, видеоконференций, прототипа оконного интерфейса [4].

Цель данной **статьи** – рассмотреть и описать разработку автоматизированной системы «Студенческое общежитие».

Изложение основного материала. *База данных* (БД) – это поименованная совокупность структурированных данных, относящихся к определенной предметной области. Базы данных необходимы для хранения, обработки, редактирования информации введенной пользователем [1].

Система управления базами данных (СУБД) – это такой механизм, который обрабатывает структурированные данные и необходим для записи, поиска, изменения, а также для сортировки и печати информации [1].

Прежде чем создавать базу данных, необходимо выбрать модель данных, наиболее удобную для решения поставленной задачи. С помощью модели данных могут быть представлены объекты предметной области и взаимосвязи между ними. Модели данных, которые поддерживают СУБД, а следовательно, и сами СУБД делят на иерархические; сетевые; реляционные [5].

В основу *иерархической базы данных* положена разветвленная структура с элементами подчиненности [6]. В *сетевой* структуре *(полно связной) базе данных* каждый элемент может быть связан с любым другим элементом [7]. В основу *реляционной базы данных* положена реляционная информационная система. Реляционная структура базы данных ориентирована на организацию данных в виде двумерных таблиц, называемых еще реляционными таблицами [7].

Для создания БД «Студенческое общежитие» была выбрана реляционная модель данных, так как она является наиболее удобной для создания именно такого рода баз данных.

Таким образом, в результате исследования предметной области были определены следующие *входные данные*:

- информация о студентах;
- информация о документах на вселение;
- информация о платежах;
- информация об условиях проживания в комнате;
- информация о корпусах комплекса общежитий.

Выходная информация выводится на экран в специальных формах, упрощающих работу с записями таблиц БД. К выходной информации относится форма с запросами, а также формы, позволяющие осуществить быстрый просмотр по записям: «Студенты», «Приказы», «Ведомость», «Комната», «Корпус». Выходная информация на печать выводится в форме отчетов. На рис. 1 изображена схема взаимодействия субъектов в БД.

Рис. 1. Модель взаимодействия субъектов.

Модель взаимодействия субъектов основана на определенных типах связей между таблицами, представленных в табл. 1.

Таблица 1.

Классификация связей.

Номер связи	Родительская таблица	Дочерняя таблица	Тип связи
1	Комната	Документ_на_вселение	1:M
2	Корпус	Комната	1:M
3	Студенты	Документ_на_вселение	1:M
4	Документ_на_вселение	Ведомость	1:M

Выбор типа связи обоснован определенными причинами.

1. На одну комнату составляется несколько документов на вселение, поэтому выбран тип связи «один ко многим».
2. В одном корпусе находится много комнат, поэтому выбран тип связи «один ко многим».
3. На одного студента может подписываться несколько приказов (по годам), поэтому выбран тип связи «один ко многим».
4. На один приказ может оформляться множество платежей (по месяцам), поэтому выбран тип связи «один ко многим».

На рис. 2 показаны отношения между таблицами.

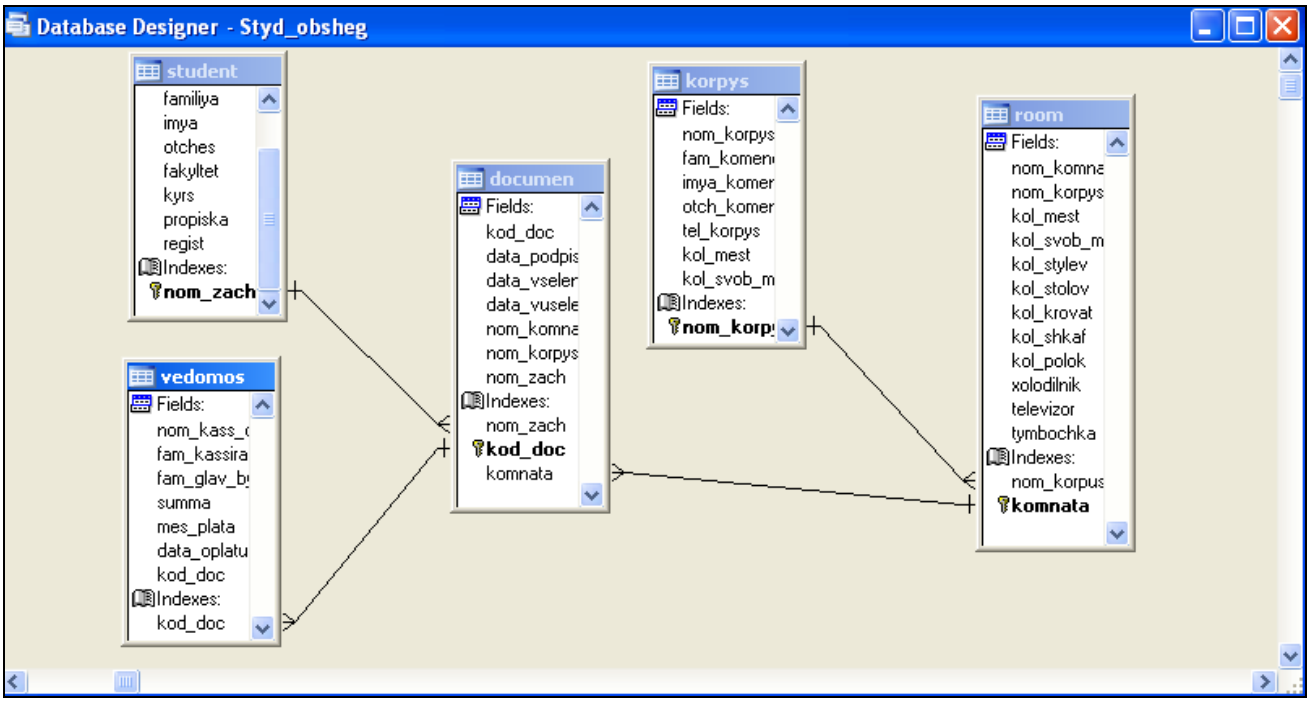


Рис. 2. Отношения между таблицами.

Работа программы основана на диалоге с пользователем через специальные экранные формы. В разработанной БД предусмотрена защита от несанкционированного доступа к данным. С помощью формы «Вход» пользователь вводит пароль и начинает работать с меню (рис. 3).

Рис. 3. Форма «Вход».

Формы используются для просмотра, ввода, редактирования, удаления данных, хранящихся в таблицах, а также для просмотра отчетов, имеющихся в БД.

В разработанной автоматизированной системе «Студенческое общежитие» были созданы следующие формы: «Корпус-комната», «Студент-документ», «Ведомость». Пример формы «Корпус-Комната» показан на рис. 4. Эта форма была создана по связи «один ко многим» таблиц «Корпус» и «Комната», с её помощью пользователь одновременно может просматривать, добавлять, изменять и удалять информацию о корпусах и о комнатах.

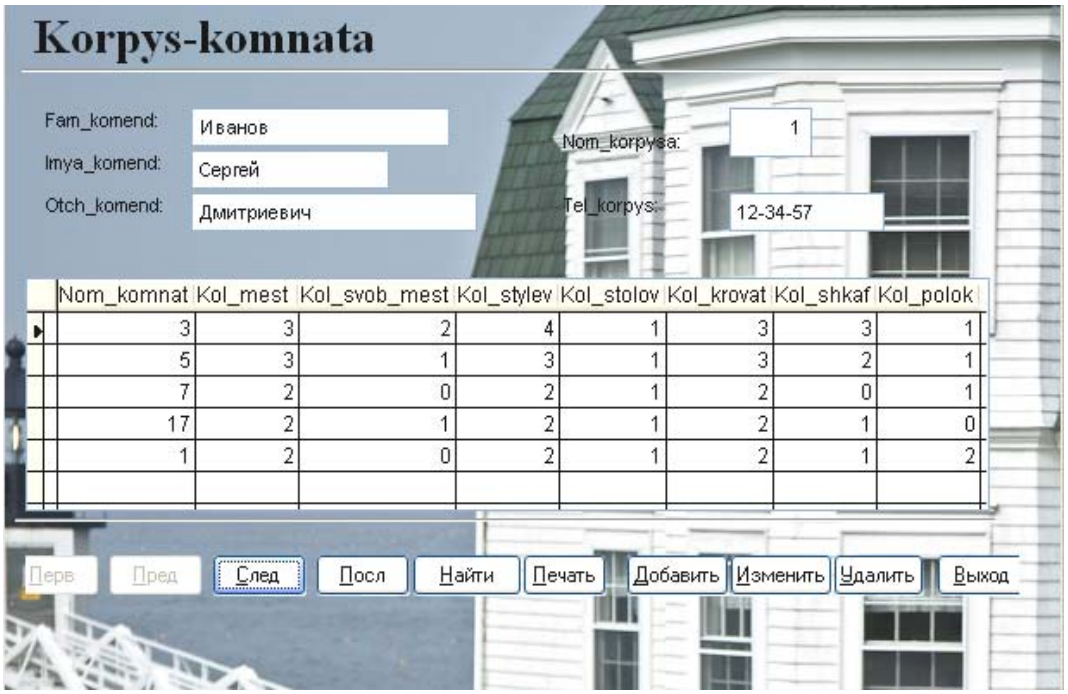


Рис. 4. Форма «Корпус-Комната».

Для удобства поиска необходимой информации в разработанной базе данных были созданы дополнительно две формы вручную: «Поиск информации о корпусе, закрепленном за комендантом по его фамилии» и «Поиск информации о студентах по фамилии» (рис. 5).

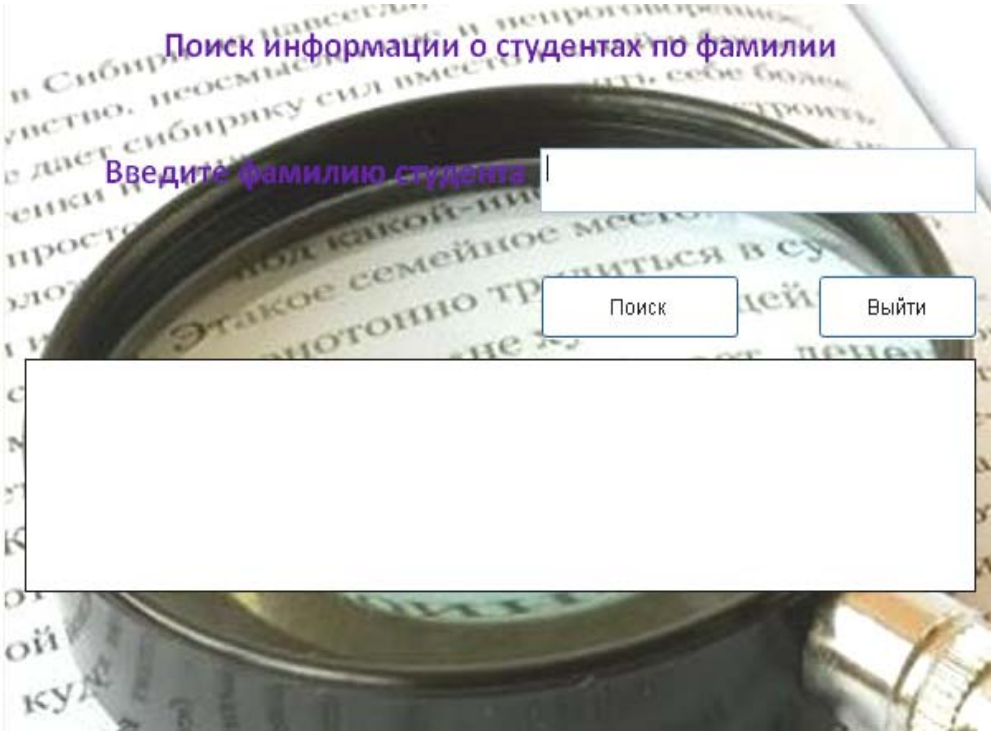
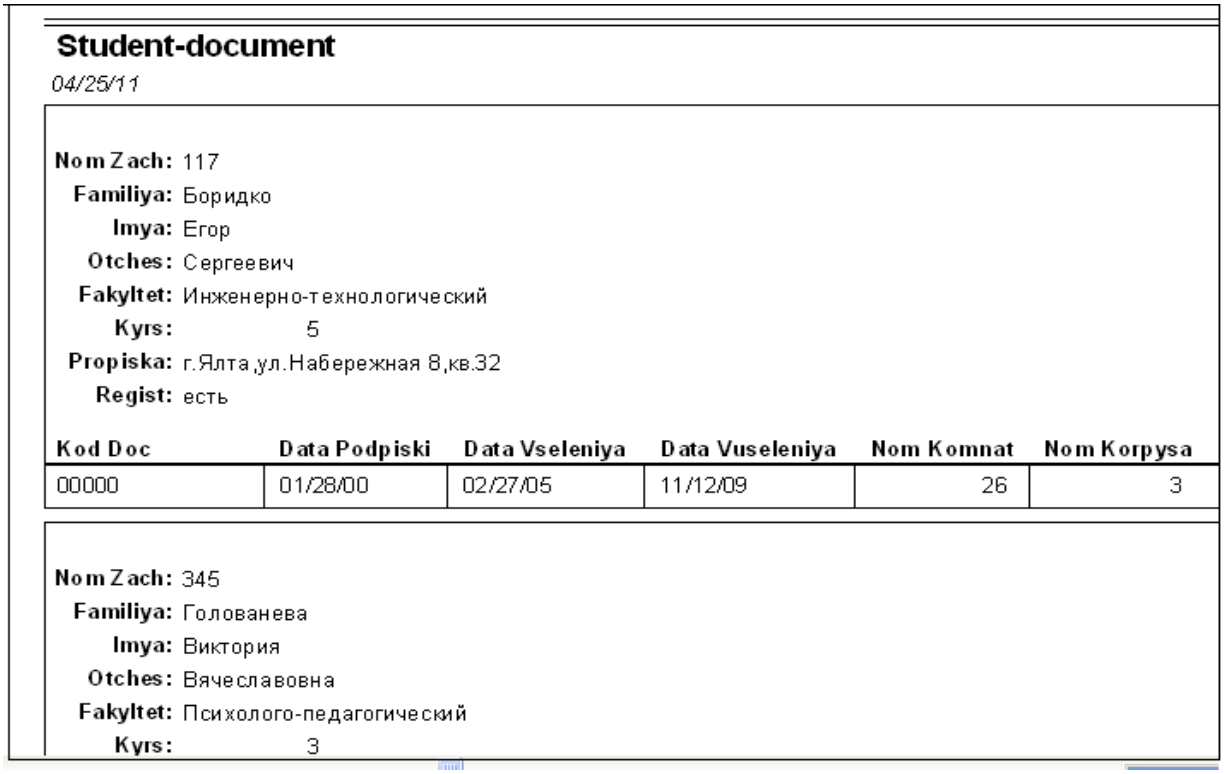


Рис. 5. Форма «Поиск информации о студентах по фамилии».

Администратор базы данных с легкостью может просмотреть информацию об определенном студенте, введя его фамилию и нажав кнопку «Поиск».

В процессе разработки БД «Студенческое общежитие» были созданы следующие отчеты: «Студент-Документ», «Ведомость», «Корпус-Комната». Отчеты очень удобны для вывода на печать, экран информации, содержащейся в базе данных.

На рис. 6 представлен отчет «Студент-Документ», который содержит всю информацию о студентах, то есть их инициалы, в каком корпусе и какой комнате проживает каждый из них, а также подписанных ими документах на вселение. Преимущества этого отчета в том, что одновременно просматривается информация о студенте и о его документе на вселение: код документа, когда подписан. Отчет соответствует информации из таблиц «Студент» и «Документ».



Student-document

04/25/11

Nom Zach: 117
Familiya: Боридко
Imya: Егор
Otches: Сергеевич
Fakyltet: Инженерно-технологический
Kyrs: 5
Propiska: г.Ялта,ул.Набережная 8,кв.32
Regist: есть

Kod Doc	Data Podpiski	Data Vseleniya	Data Vuseleniya	Nom Komnat	Nom Korpysa
00000	01/28/00	02/27/05	11/12/09	26	3

Nom Zach: 345
Familiya: Голованева
Imya: Виктория
Otches: Вячеславовна
Fakyltet: Психолого-педагогический
Kyrs: 3

Рис. 6. Отчет «Студент-документ».

Выводы. Разработанная автоматизированная система «Студенческое общежитие» предназначена для решения задач организации деятельности комплекса общежитий. Основное назначение спроектированной базы данных – предоставление основной информации о проживающих в общежитиях студентах, об условиях проживания, а также предоставление перечня приказов на вселение и ведомостей об оплате в простой и удобной форме. Созданная БД поможет сократить затраты времени на обработку этих данных, поможет снизить количество ошибок при регистрации студентов, а также позволит автоматизировать рабочее место коменданта.

Основными возможностями спроектированной базы данных являются

1) два уровня доступа к базе данных: Администратор базы данных; Гость (человек, имеющий право на просмотр некоторой информации);
2) структурирование и обработка данных, а также вывод интересующей информации на экран и печать.

ЛИТЕРАТУРА

1. Хомоненко А. Д. Базы данных : учеб. для вузов / А. Д. Хомоненко, В. М. Цыганков, М. Г. Мальцев. – СПб. : Корона-Век, 2010. – 736 с.
2. Свободная энциклопедия «Википедия» / Ларри Эллисон [Электронный ресурс]. – Режим доступа : http://en.wikipedia.org/wiki/Larry_Ellison.
3. Codd E. F. A Relational Model of Data for Large Shared Data Banks / E. F. Codd // Communications of the ACM. – 1970. – Vol. 13. – № 6. – 387 p.
4. Выдающиеся программисты и математики-программисты [Электронный ресурс]. – Режим доступа : http://www.chernyshov.com/PROGRAMMERS/GrosProg.htm.
5. Каратыгин С. А. Visual FoxPro 7 : учеб. пособие / А. С. Каратыгин, Ф. А. Тихонов, Н. Л. Тихонова. – М. : ЗАО «Издательство «БИНОМ», 2000. – 784 с.
6. Карпова Т. С. Базы данных : модели, разработка, реализация / Т. С. Карпова. – СПб. : Питер, 2002. – 304 с.
7. Райордан Р. Основы реляционных баз данных / Р. Райордан ; [пер. с англ.]. – М. : Издательско-торговый дом «Русская Редакция», 2001. – 384 с.

УДК 004.658.2

Шкарбан Ф. В., Мустафаєва Е. І.

ПРОЕКТУВАННЯ БАЗИ ДАНИХ ОРГАНІЗАЦІЇ З РЕМОНТУ АВТОМОБІЛІВ

У статті проаналізовані поняття і структура систем управління базами даних, зокрема Visual FoxPro. Розглянута як приклад база даних організації з ремонту автомобілів.

Ключові слова: *системи управління базами даних, бази даних, сутність, атрибут, зв'язки.*

В статье проанализированы понятия и структура систем управления базами данных, в частности Visual FoxPro. Рассмотрена в качестве примера база данных организации по ремонту автомобилей.

Ключевые слова: *системы управления базами данных, базы данных, сущность, атрибут, связи.*

Notions and structure of database management system, particularly Visual FoxPro is analyzed in the article. As the example, database of the car repairing organization is observed.

Key words: *database system management, database, attribute, connection.*

Постановка проблеми. Сучасне суспільство прагне керувати конкурентоспроможністю даних, використовуючи системи управління базами даних (СУБД), які стали невід'ємним компонентом повсякденного життя. Управління системами баз даних перетворилося із спеціалізованого комп'ютерного додатка до центрального компонента сучасного комп'ютерного середовища. В результаті, знання про системи управління базами даних стали невід'ємною частиною освіти в галузі комп'ютерних наук та інформатики.

Аналіз публікацій. Так як потреби і задачі суспільства стають все більш і більш вимогливими та складними, то використання СУБД направлено на управління даними тієї чи іншої організації і набувають критично важливого значення для загального успіху, яке полягає в забезпеченні інформаційної здатності конкретного напряму. Вибір СУБД для реалізації певного завдання – це досить складний процес, який вимагає навичок, знань і уваги професіонала в цій галузі. Неправильний вибір СУБД може заблокувати процес реалізації даної технології, а заміна СУБД може коштувати надто дорого.

СУБД відноситься до дуже складної частини програмного забезпечення, яку важко зрозуміти у всій повноті і є життєво важливою для визначення критеріїв, які можуть бути використані адміністратором БД в процесі їх відбору. Багато практиків-розробників БД і авторів-теоретиків [1] в своїх роботах відзначають, що вибір критеріїв використання СУБД полягає в первинному аналізі організаційних потреб і необхідних функцій СУБД. Практики-розробники виділяють ряд факторів вибору СУБД для реалізації задачі [2]:

- вимоги до застосування (це обмеження, які вводяться в базу даних в додатку);
- функції та інструменти СУБД (вбудований набір інструментів, що допомагає полегшити завдання розробки додатків);
- моделі СУБД (ієрархічні, мережеві, реляційні, об'єктно-орієнтовані);
- переносимість СУБД (платформи, системи та мови програмування);
- вимоги СУБД до обладнання (мінімальна швидкість процесора, обсяг оперативної пам'яті, дискового простору і так далі);
- вартість СУБД (витрати на придбання та технічне обслуговування, експлуатаційні витрати, вартість ліцензій, витрати на установку, вартість навчання та перетворення витрат);
- збереження цілісності даних (захист даних у базі даних від несанкціонованого доступу, зміни або знищення даних).

Таким чином, системи управління БД – це спеціальні програми, які призначені для структурування інформації, розміщення її в таблицях і маніпулювання даними [3]. З поняттям СУБД тісно пов'язане поняття БД. База даних – це інформаційна модель, що дозволяє впорядковано зберігати дані про групу об'єктів, що володіють однаковим набором властивостей [4].

Мета статті – розкрити та проаналізувати основні поняття систем управління БД, зокрема СУБД Visual FoxPro, а також на прикладі баз даних з ремонту автомобілів обґрунтувати етапи проектування даної системи.

Виклад основного матеріалу. До найбільш поширених типів СУБД відносяться Visual FoxPro, MSSQLServer, Oracle, Informix, Sybase, DB2, MSAccess і т. д.

Visual FoxPro (VFP) – об'єктно-орієнтована і процедурна мова програмування систем управління реляційними базами даних, розроблена корпорацією Microsoft [5]. Visual FoxPro – абсолютно нова програма, що дозволяє зробити те, що в деяких СУБД дається з величезними труднощами або просто недоступне. Всього існує шість версій: VFP 3.0, 5.0, 6.0, 7.0, 8.0, 9.0. Версія 3.0 є першою версією. Остання версія – Visual FoxPro 9.0, пакет оновлень 2 (SP2). У табл. 1 представлена коротка характеристика версій Visual FoxPro, що описує сумісність з Windows, розмір і дату EXE-файла, а також назва DLL-файлу.

Таблиця 1.

Порівняльна характеристика версій Visual FoxPro.

Версія	Windows 3.x	Windows 95	Windows 98	Windows XP	Windows Vista	Windows 7	Розмір EXE-файла, kb	Дата EXE-файла	Назва DLL-файла
VFP 3.0	сумісний	сумісний	сумісний	сумісний	в режимі сумісності	сумісний	4,374	16.12.95	VFP300R.ESL
VFP 5.0	несуміс-ний	сумісний	сумісний	сумісний	сумісний	сумісний	4,065	24.01.97	VFP500R.DLL
VFP 6.0	несуміс-ний	сумісний	сумісний	сумісний	сумісний	сумісний	4,091	18.08.00	VFP6R.DLL
VFP 7.0	несуміс-ний	runtime	сумісний	сумісний	сумісний	сумісний	4,260	04.01.02	VFP7R.DLL
VFP 8.0	несуміс-ний	сумісний	runtime	сумісний	сумісний	сумісний	5,236	25.09.03	VFP8R.DLL
VFP 9.0	несуміс-ний	несуміс-ний	runtime	сумісний	сумісний	сумісний	5,620	13.12.04	VFP9R.DLL

У Visual FoxPro реалізовані всі атрибути реляційної СУБД. В Visual FoxPro поняття БД розглядається як сукупність зв'язаних таблиць, інформація про які зберігається в словнику даних.

У БД розглянутої СУБД визначаються умови її цілісності за допомогою первинних і зовнішніх ключів таблиць. Visual FoxPro володіє високою швидкістю в обслуговуванні баз даних. Використовуючи стандарт ODBC і SQL-запити, Visual FoxPro дозволяє працювати з даними в форматі СУБД Access, Paradox, dBase і т. д., з серверами баз даних – Microsoft SQL Server, Oracle та ін.

Visual FoxPro дозволяє створювати бази даних в інтерактивному режимі за допомогою конструктора бази даних, зокрема:

- створювати і модифікувати таблиці, збережені процедури, представлення даних;
- додавати створені раніше таблиці;
- визначати для таблиць індекси;
- встановлювати відносини між таблицями, які будуть підтримуватися при створенні форм і звітів.

Як приклад, розглянемо процес проектування і роботи бази даних організації з ремонту автомобілів (система послуг з ремонту автомобілів), реалізованої за допомогою СУБД Visual FoxPro. Дана база даних містить всі необхідні відомості про клієнтів, їх адреси, транспорт, який підлягає ремонту, дату отримання та видачі тощо. Доступ до цієї бази даних має досить велика кількість співробітників даного підприємства. База даних також містить методи і засоби, що дозволяють кожному зі співробітників керувати тільки тими даними, які входять в його певні права. У результаті взаємодії даних з методами утворюється інформація, яку вони використовують і на підставі якої роблять введення і редагування даних. Отже, розроблена система послуг з ремонту автомобілів являє собою базу даних організації, яка займається ремонтом автомобілів.

Для проектування бази даних необхідно визначити наступне.

1. *Правила*, за якими формуються об'єкти в ER-моделі і заповнюються таблиці:

- всі клієнти мають код, адресу, телефон, паспортні дані;
- при виборі несправності необхідно вказати компоненти, необхідні для ремонту;
- коли укладається договір, вказується дата ремонту, вид несправності, вартість ремонту;
- вартість ремонту залежить від виду несправності та запчастин.

2. *Сутності*:

- автомобілі, що знаходяться у веденні організації;
- співробітники (управитель і обслуговуючий склад);
- клієнти;
- додаткові дані.

3. *Властивості* (атрибути) сутностей для кожної окремо:

- <u>Договір має наступні атрибути</u>:
  - Державний номер автомобіля;
  - Марку;
  - Вид (вантажний, легковий і т. п.);
  - Вид несправності (необхідність ремонту);
  - Співробітник (людина, яка ремонтувала цей транспорт);
- <u>Співробітники:</u>
  - ПІБ;
  - Професія (водій, зварювальник, збирач і тому подібне);
  - Приналежність (тобто в якому гаражі або відділі працює);

- Клієнт:
  - ПІБ;
  - Телефон клієнта;
  - Адреса;
  - Паспортні дані.

Для побудови даталогічної моделі даних, тобто моделі, що відбиває логічні взаємозв'язки між елементами даних, їх змісту та фізичної організації, необхідно розглянути зв'язок між атрибутами сутностей (рис. 1).

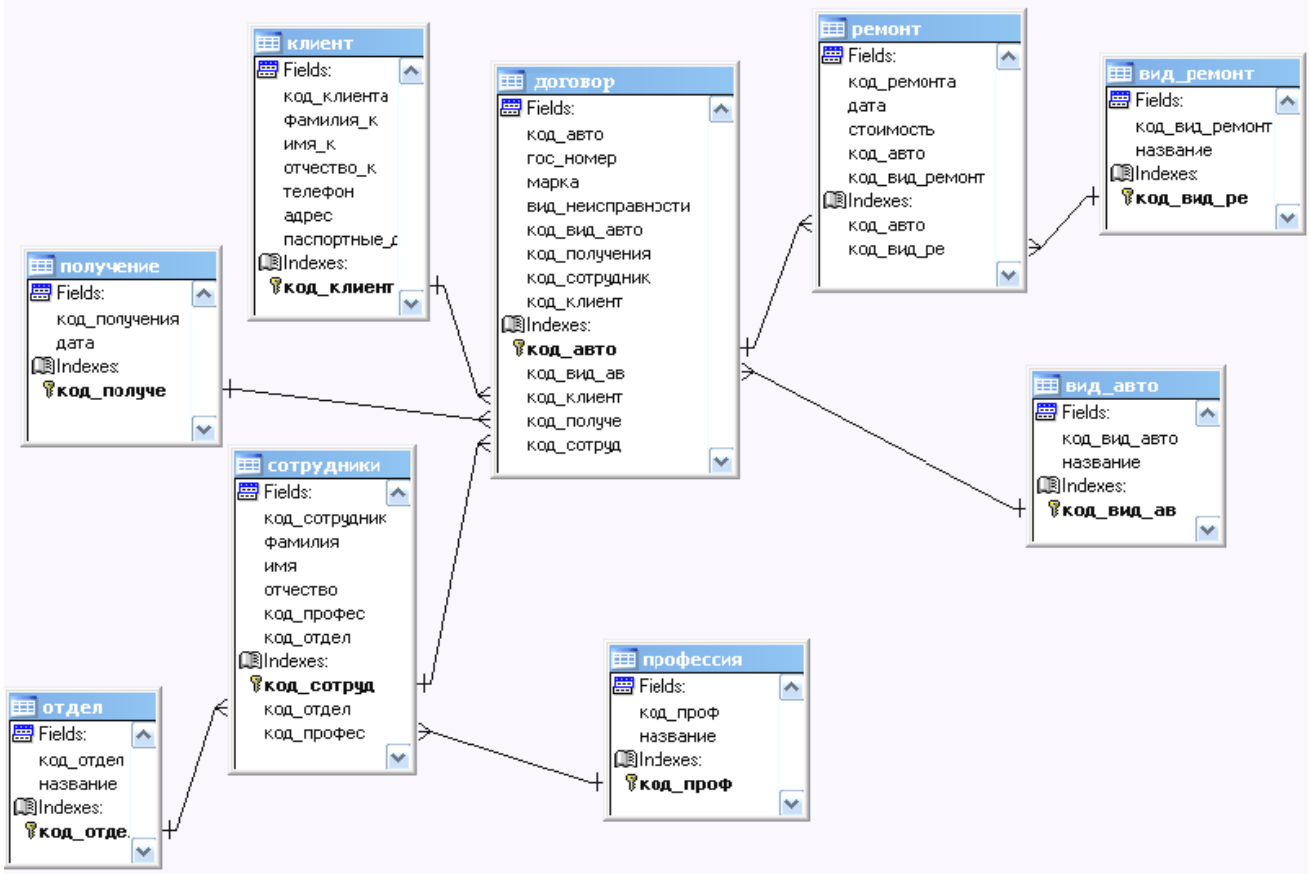


Рис. 1. Даталогічна модель системи послуг по ремонту автомобілів.

У побудованій моделі визначено вид відносин «один до багатьох». Це означає, що один екземпляр однієї сутності пов'язаний з безліччю екземплярів іншої сутності. Наприклад, одна професія автослюсар може відповідати кільком працівникам. Тобто, у відділі можуть працювати кілька співробітників з кваліфікацією автослюсар. Кожна таблиця містить інформацію певної сутності. Наприклад, у таблиці «Клієнти» зберігається вся інформація про клієнтів – прізвище, ім'я, по батькові, телефон, адреса та паспортні дані (рис. 2).

Фио	Адрес	Номер_тел	Серия_паспорта	Идентификационный_код
Федоров Сергей Иванович	г.Симферополь	959379992	ЕТ017899	3336804898
Мелешко Сергей Константинович	г.Симферополь	952255776	ЕТ087945	9856489465
Бахича Халил Алиевич	г.Симферополь	950000000	ЕТ017801	9856489411
Константинов Ваня Иванович	г.Белогорск	950000010	ЕТ017802	9856489412
Бахича Салдин Алиевич	г.Белогорск	950000020	ЕТ017810	9856489413
Асанов Салидин Сейранович	г.Симферополь	950000030	ЕТ017809	9856489414
Петров Пётр Афанасьевич	г.Симферополь	950000040	ЕТ017808	9856489415
Паламарчук Виктор Егорович	г.Симферополь	950000050	ЕТ017807	9856489416
Асанов Сервер Шевхиевич	г.Севастополь	950000060	ЕТ017862	9856489417
Киндрук Сергей Афанасьевич	г.Симферополь	950000070	ЕТ017805	9856489418
Гаврилюк Артур Анатольевич	г.Белогорск	950000080	ЕТ017804	1599465000

Рис. 2. Таблиця «Клієнти».

У представленій базі даних для перегляду, введення та редагування даних, що зберігаються в таблицях, використовуються форми. Форми є більш наочним засобом представлення інформації і дозволяють працювати не з однією, а з кількома пов'язаними таблицями, що, в свою чергу, також збільшує наочність і, найголовніше, зручність (рис. 3).

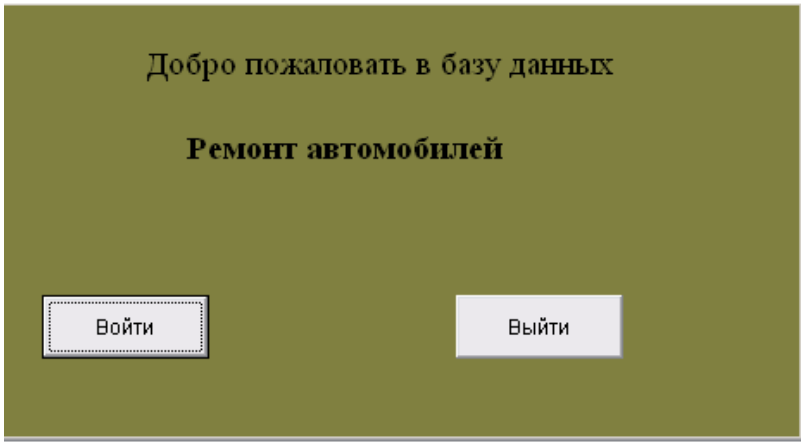


Рис. 3. Головна форма бази даних.

Система послуг з ремонту автомобілів містить звіти, які представляють собою форматоване подання даних, що виводиться на екран, принтер або у файл. Звіти представлені в табличному вигляді або у вільній формі. У звітах відображаються необхідні дані, які повинні бути зрозумілі і доступні як адміністратору, так і клієнтам (рис. 4).

Клиенты

Фио	Серия Паспорта	Адрес	Номер Тел
Федоров Сергей Иванович	ET017899	г.Симферополь	959,379,992
Мелешко Сергей Константинович	ET087945	г.Симферополь	952,255,776
Бахича Халил Алиевич	ET017801	г.Симферополь	950,000,000
Константинов Ваня Иванович	ET017802	г.Белогорск	950,000,010
Бахича Салдин Алиевич	ET017810	г.Белогорск	950,000,020

Рис. 4. Звіт «Клієнти».

Таким чином, робота з клієнтами, облік, домовленість, отримання і видача автомобілів в організації по ремонту автомобілів, реєстрація, закріплення робітників за кожною машиною, аналіз несправностей та інше займає тривалий час. Цей процес роботи можна спростити. Досягти цієї мети можливо за допомогою мінімізації кількості операцій, виконуваних вручну і автоматизуючи в першу чергу ті, які оперують великою кількістю чисел, робота з якими і представляє найбільшу складність.

Одним з найбільш підходящих способів для автоматизації будь-якого процесу, що вимагає роботу з великою кількістю даних, є використання системи управління базою даних, яка виробляє всілякий облік та сортування даних.

ЛІТЕРАТУРА

1. Türker C. Semantic Integrity Support in SQL-99 and Commercial (Object-Relational) Database Management Systems / Can Türker, Michael Gertz // The VLDB Journal. – 2000. – № 10(4). – P. 241–269.
2. Akadia Information Technology: 2005, Oracle Tips and Tricks of the Week Part 4 [Електронний ресурс]. – Режим доступу : http://www.akadia.com/services/ora_important_part_4.html.
3. Кузнецов С. Д. Базы данных. Модели и языки / С. Д. Кузнецов. – М. : Бином-Пресс, 2008. – 720 с.
4. Хомоненко А. Д. Базы данных : учебник для высших учебных заведений / А. Д. Хомоненко, В. М. Цыганков, М. Г. Мальцев ; [под ред. проф. А. Д. Хомоненко]. – [изд. 6-е]. – М. ; СПб. : Бином-Пресс ; КОРОНА-век, 2007. – 736 с.
5. Дунаев В. В. Базы данных. Язык SQL / В. В. Дунаев. – [2-е изд.]. – СПб. : БХВ-Петербург, 2007. – 320 с.

УДК 004.855.5

Крылов В. С.

КОМПЬЮТЕРНОЕ ЗРЕНИЕ: ИННОВАЦИОННЫЙ ВИРТУАЛЬНЫЙ СТЕНД

У статті описаний віртуальний лабораторний стенд з програмами комп'ютерного зору, що дає можливість закріплювати теоретичні знання та набувати практичні навички в цьому інноваційному напрямку.

Ключові слова: *віртуальний стенд, комп'ютерний зір.*

В статье описан виртуальный лабораторный стенд с программами компьютерного зрения, который дает возможность закреплять теоретические знания и приобретать практические навыки в этом инновационном направлении.

Ключевые слова: *виртуальный лабораторный стенд, компьютерное зрение.*

This paper describes a virtual lab bench with the programs of computer vision, which makes it possible to consolidate the theoretical knowledge and acquire practical skills in this innovative area.

Key words: *virtual stand, computer vision.*

Постановка проблемы. Виртуальные лабораторные стенды (ВЛС) широко используются в подготовке и проведении испытаний различных технических устройств, в том числе и сложных информационных систем. Они являются важной инновационной составляющей в проведении экспериментов с различными моделями устройств, процессов в первую очередь в области естественнонаучных, инженерно-технических и технологических дисциплин.

Главная задача ВЛС – добиться баланса между познавательным пониманием процессов работы различных устройств и приобретения практических навыков их управления. Для достижения такого баланса необходимо инновационное методическое решение организации работы с виртуальным стендом, которое обеспечит сквозную связь системообразующих компонентов в изучаемом устройстве, информационной системе. Таким образом, при проектировании ВЛС при всех прочих требованиях необходимо отбирать подходы интегрирующего, междисциплинарного и сквозного характера, чтобы получить необходимый баланс между теоретическими знаниями и практическими моделями их применения, а также установить сквозную связь различных дисциплин в методике подготовки работы со стендом.

Анализ литературы. В настоящее время разработано множество вариантов виртуальных лабораторных стендов, основанных на различных принципах, идеологиях, технологиях и призванных решать различные практические задачи разработки программ и методик испытаний различных устройств, информационных систем [1–3].

В работе А. В. Дьяченко определены наиболее общие положения, на основе которых должны строиться ВЛС исходя из задач, решаемых с его помощью:

- экспериментальные исследования теоретических положений, лежащие в основе практического применения в технических устройствах, программно-аппаратных комплексах;
- экспериментальные исследования различных режимов, эмулируемых с помощью программ реальных устройств или процессов [1].

Цель статьи состоит в определении инновационного методического решения организации сквозной связи системообразующих в области анализа информационных процессов, внедрения программно аппаратных решений внедрения информационных систем для опытной проверки разрабатываемых программ и методик испытаний.

Изложение основного материала. ВЛС являются важной инновационной составляющей в анализе различных процессов, в первую очередь изучения теоретических основ естественнонаучных инженерно-технических и технологических областей знаний [1–3].

Эффективный виртуальный стенд целесообразно организационно составлять из отдельных виртуальных лабораторных стендов. Обычно ВЛС представляет собой специальное программное обеспечение, которое основано на математической модели, описывающей процессы, происходящие в эмулируемой лабораторной установке; визуального отображения элементов, связей и состояния лабораторной установки и элементов управления ею [3]. В отличие от таких стендов, ВЛС в области информатики представляют не только примеры, но и программное обеспечение, с помощью которого решаются те или иные задачи соответствующей области науки или практики.

Виртуальный стенд при соответствующем отборе составляющих его ВЛС может стать важным звеном в изучении сквозной связи различных компонентов, составляющих изучаемое уст-

ройство, информационную систему. Решающее значение в этом случае играет выбор области и направления, решаемых программным обеспечением входящих в состав виртуальных лабораторных стендов задач. ВЛС должен представлять собой решение задачи междисциплинарного направления, то есть направления, которое интегрирует в себе различные методы и подходы различных дисциплин. Таким направлением является «компьютерное зрение», которое представляет общий набор методов и устройств, позволяющих компьютеру «видеть» с помощью цифровых устройств ввода/вывода, и связано с вычислительной техникой и оптикой [4].

Компьютерное зрение – теория и технологии создания систем, которые могут «видеть». Научная дисциплина, которая относится к теории и технологиям разработки и создания систем, извлекающих информацию из изображений. Видеоданные могут быть представлены множеством форм, таких как последовательность кадров видео, изображения от различных камер, для получения стереоскопической картины, трехмерные данные от медицинского сканера [4].

Как технологическая дисциплина компьютерное зрение использует теории и модели для создания таких систем, как

- системы управления технологическими линиями различного назначения, промышленные роботы, роботы как бытовая техника, автономные транспортные средства;
- системы видеонаблюдения и видеорегистрации;
- системы организации информации, например, индексация баз данных изображений;
- системы моделирования объектов или окружающей среды, такие как системы изображения для медицинских целей, топографическое моделирование и т. д.;
- системы взаимодействия, например человек–машина.

Область компьютерного зрения является инновационной, стремительно развивающейся в самых разнообразных направлениях. Несмотря на то, что исследования распознавания изображений проводились в сороковые годы прошлого века, особенно активно они стали проводиться в семидесятые годы. Именно в это время стали доступны широкому кругу исследователей вычислительные системы, способные обрабатывать в реальном времени большие массивы данных, в том числе и связанные с изображениями.

В настоящее время рост активности в области компьютерного зрения также обусловлен тем, что произошел технологический скачок как в технологиях ввода изображений, так и их обработки. Еще вчера исключительно научные разработки стали воплощаться в устройства и системы решающие различные не только технические, технологические задачи, но и те, которые связаны исключительно с организацией быта, например компоненты системы «умный дом» [5].

Доступность технических средств ввода и обработки изображений обуславливает направленность усилий в области компьютерного зрения в разработке специализированных алгоритмов и программ анализа изображений. Появившиеся десять лет назад библиотеки программ для работы с компьютерным зрением в последнее время активно совершенствуются, регулярно появляются новые версии работы с все большим классом устройств, учитывающие новейшие достижения микропроцессорных устройств.

Open CV библиотека с открытым исходным кодом программ компьютерного зрения. Она содержит более 500 функций, ориентированных на выполнение программ компьютерного зрения в реальном времени. Библиотека Open CV сопровождается множеством примеров, иллюстрирующих работу реализованных в функциях алгоритмов [6].

Перед Open CV изначально ставились следующие цели:

- исследование машинного зрения, разработка и оптимизация кода;
- распространение информации по машинному зрению, разработка общей инфраструктуры, на которой могли бы основываться разработчики-программисты, код должен быть удобочитаемым и передаваемым;
- приложения должны быть переносимыми, оптимизируемыми, код которых не обязательно должен быть открытым, приложения могут создаваться в коммерческих целях.

Учебный процесс предполагает использование качественных, отлаженных и работающих в полном соответствии с заявленной спецификацией исходных данных программ. То есть таких, которые прошли соответствующее тестирование и опытную эксплуатацию.

Программы, которые можно использовать в качестве ядра виртуального стенда для лабораторного практикума, широко представлены на сайте «The code project» [7]. Этот сайт является открытой площадкой для сообщества разработчиков программного обеспечения со всего мира. С ним на постоянной основе сотрудничают такие компании, как Microsoft, Sun, Oracle и др.

При соблюдении сформулированных требований к ВЛС, например, виртуальный лабораторный стенд с ядром, представленным в [8], «Алгоритмы распознавания движения» в дисциплине «История информационно-компьютерных технологий» обеспечивает демонстрацию того,

как работают современные программы автоматического выделения движущихся объектов. Тот же стенд в дисциплине «Объектно-ориентированное программирование» уже выступает как пример реализации алгоритмов анализа изображений в парадигме программирования объектно-ориентированного программирования. Кроме того, стенд обеспечивает как демонстрацию, так и возможность получения практических навыков в работе с библиотеками программ компьютерного зрения, в первую очередь библиотеки с открытым кодом Open CV.

Выводы. Инновационная составляющая виртуального лабораторного стенда состоит не только в том, что возможно знакомиться и приобретать практические навыки в инновационном направлении «компьютерное зрение», но и в том, что стенд обеспечивает сквозную связь различных компонентов как единого целого. Это дает возможность детальной разработки и проверки программ и методик испытаний разных вариантов сборки изучаемой системы из различных компонентов.

ЛИТЕРАТУРА

1. Построение информационных систем непрерывного образования на основе Интернет-технологий [Электронный ресурс] / [А. В. Дьяченко, В. Г. Манжула, А. Э. Попов и др.]. – Режим доступа : http://www.rae.ru/monographs/98.
2. Звягинцева А. В. Лабораторные программные средства для изучения процессов развития сложных систем «живой организм – окружающая среда» / А. В. Звягинцева, Е. А. Климова, А. А. Цымбалова // Наукові праці Донецького національного технічного університету. Серія : Проблеми моделювання та автоматизації проектування. – 2011. – № 9(179). – С. 213–224.
3. Зюбин В. Е. Использование виртуальных лабораторных стендов для обучения программированию в области задач промышленной автоматизации / В. Е. Зюбин // Приборы и системы. – 2009. – № 2. – С. 29–33.
4. Форсайт Д. А. Компьютерное зрение: современный подход / Д. А. Форсайт, Ж. Понс ; [пер. с англ.]. – М. : Вильямс, 2004. – 928 с.
5. Устелемова М. С. Основы построения системы «умный дом» [Электронный ресурс] / М. С. Устелемова. – Режим доступа : http://www.intuit.ru/department/expert/bsmarthouse.
6. Open CV [Электронный ресурс]. – Режим доступа : http://opencv.itseez.com.
7. The code project [Электронный ресурс]. – Режим доступа : http://www.codeproject.com.
8. Kirillov A. Motion Detection Algorithms [Электронный ресурс] / A. Kirillov. – Режим доступа : http://www.codeproject.com/Articles/10248/Motion-Detection-Algorithms.

УДК 004.75

Аблялимова Э. И., Кадыров М. Р., Манжос Л. А.

ТЕХНИЧЕСКИЕ АСПЕКТЫ ВИРТУАЛИЗАЦИИ IT-ИНФРАСТРУКТУРЫ

У статті представлена організація IT-інфраструктури навчального закладу до та після віртуалізації. Розглянуто технології апаратної віртуалізації. Приводяться технічні аспекти віртуалізації.

Ключові слова: *віртуалізація, апаратна віртуалізація, IT-інфраструктура, VMware, Intel VT, AMD-V.*

В статье представлена организация IT-инфраструктуры учебного заведения до и после виртуализации. Рассмотрены технологии аппаратной виртуализации. Приводятся технические аспекты виртуализации.

Ключевые слова: *виртуализация, аппаратная виртуализация, IT-инфраструктура, VMware, Intel VT, AMD-V.*

The organization of IT-infrastructure of the educational institutions before and after virtualization presented in the paper. Hardware virtualization technology is considered. The technical aspects of virtualization are proposed.

Key words: *virtualization, hardware virtualization, IT-infrastructure, VMware, Intel VT, AMD-V.*

Постановка проблемы. В современное время инфраструктура серверов становится сложнее: объемы компьютерных систем увеличиваются быстрыми темпами и на организацию данных сетей, обслуживание и управление серверами тратится больше времени и средств. Возникают вопросы о приобретении большего количества компьютеров, сетевого оборудования, серверов и прочего.

Помимо этого, растут и вычислительные мощности современных процессоров. При этом в учебных заведениях они практически не используются в полном объеме. По сведениям, представленным в статье [1], работающие приложения в IT-инфраструктуре компаний создают небольшую нагрузку на отдельно взятый компьютер – в среднем, процессор загружен лишь на 5–15%.

Технология виртуализации становится важной стратегией, позволяющей реализовать огромный потенциал, управлять ресурсами учреждения либо предприятия на современном этапе развития информационных технологий, увеличить ресурсы, минимизировать вложения и затраты на инфраструктуру.

Анализ последних исследований и публикаций. С 2004 года Университет штата Северная Каролина, используя технологии IBM, предлагает студентам и преподавателям образовательные услуги и усовершенствованные IT-сервисы в рамках виртуальной вычислительной лаборатории – Virtual Computing Laboratory (VCL) [2]. В статье сотрудников IBM [3] отмечено, что «VCL обслуживает свыше 40000 пользователей, предоставляет более 11 млн. часов процессорного времени в год для высокопроизводительных вычислений, предоставляет широкий спектр от программных пакетов и приложений для работы студентов до специально сконфигурированных компьютеров и поддержки исследователей».

Техническую сложность реализации возможных решений виртуализации и зависимость от аппаратной платформы, а также возможности применения аппаратной виртуализации для решения задач информационной безопасности описал в своей статье Д. В. Силаков [4].

Даниэль и Нельсон Руст в своей книге [5] рассматривают программную и аппаратную виртуализации, преимущества каждого вида, а также проблемы, решаемые виртуализацией. Они предлагают пятиступенчатый план развертывания виртуализации, а также поэтапно описывают настройку виртуальной вычислительной среды, дают подробную информацию о технологиях от Citrix, Microsoft и VMware.

В книге «Практические решения для виртуализации» [6] авторы демонстрируют способы управления всеми аспектами виртуализации, в том числе виртуальными машинами, виртуальными файловыми системами, виртуальными решениями для хранения данных и кластеризации, рассматривая для каждой технологии конкретную среду. Также авторы предлагают подробное описание виртуальных инфраструктур с использованием VMware Server и VMware ESX.

Установке, настройке, управлению и мониторингу виртуальной средой с помощью VMware посвящена и книга [7], в которой описано также обслуживание виртуализации, возможные неполадки и способы их устранения.

Цель статьи – анализ и разработка решений для современной IT-инфраструктуры учебного заведения на примере РВУЗ «Крымский инженерно-педагогический университет» и изучение технических аспектов ее виртуализации.

Изложение основного материала. Современная IT-инфраструктура любого предприятия (учебного заведения, крупной IT-фирмы и т. д.) зачастую представляет собой крупную в плане точек подключения локальную сеть, имеет достаточно сложную степень построения. То есть, как правило, такая сеть имеет несколько подсетей, и основывается не только на коммутаторах второго уровня (таких как D-Link 3200 или подобные ему), но и на маршрутизаторах третьего уровня, а также, по мере необходимости, могут использоваться межсетевые экраны седьмого уровня (в качестве примера, D-Link-800 или его старший брат D-Link-1600).

Сеть может выполнять разнообразные функции, например:

- организацию различных сервисов (например, почтового);
- использование общих сетевых периферийных устройств (принтер, сканер, сетевую мультимедийную установку для презентаций);
- файлообменник (общий доступ к файлам и различным ресурсам);
- возможность удаленного сетевого администрирования.

Наряду с вышеперечисленными функциями на крупных предприятиях локальная вычислительная сеть может использоваться для следующих возможностей:

1) доступа к сети Интернет через сервер авторизации;
2) доступа к серверам специального программного обеспечения (Microsoft Lync Server, MS SharePoint, Парус Бухгалтерия и т. д.).

На рис. 1 приведена структура локальной вычислительной сети на примере РВУЗ «Крымский инженерно-педагогический университет» до ее реорганизации.

Рассмотрим подробнее эту IT-инфраструктуру.

Сеть имеет общую протяженность порядка 8 км, охватывая при этом 4 корпуса, несколько лабораторий и научную библиотеку. Каждая компьютерная лаборатория имеет специализированное программное обеспечение, направленное на решение задач групп дисциплин. Например, есть лаборатории с программой 1С: Предприятие 8.0, DelCam и т. п. Каждая из этих программ представляет собой сервер (сервер лицензий или сервер программного обеспечения). Научная библиотека использует серверную программу Ирбис, в состав которой входят Ирбис.Каталогизатор, Ирбис.Администратор и пр. Помимо этого, плазменные телевизоры в разных корпусах университета объединены в медиа-сервер KIPU-TV и вещают потоковое видео, носящее информационный характер.

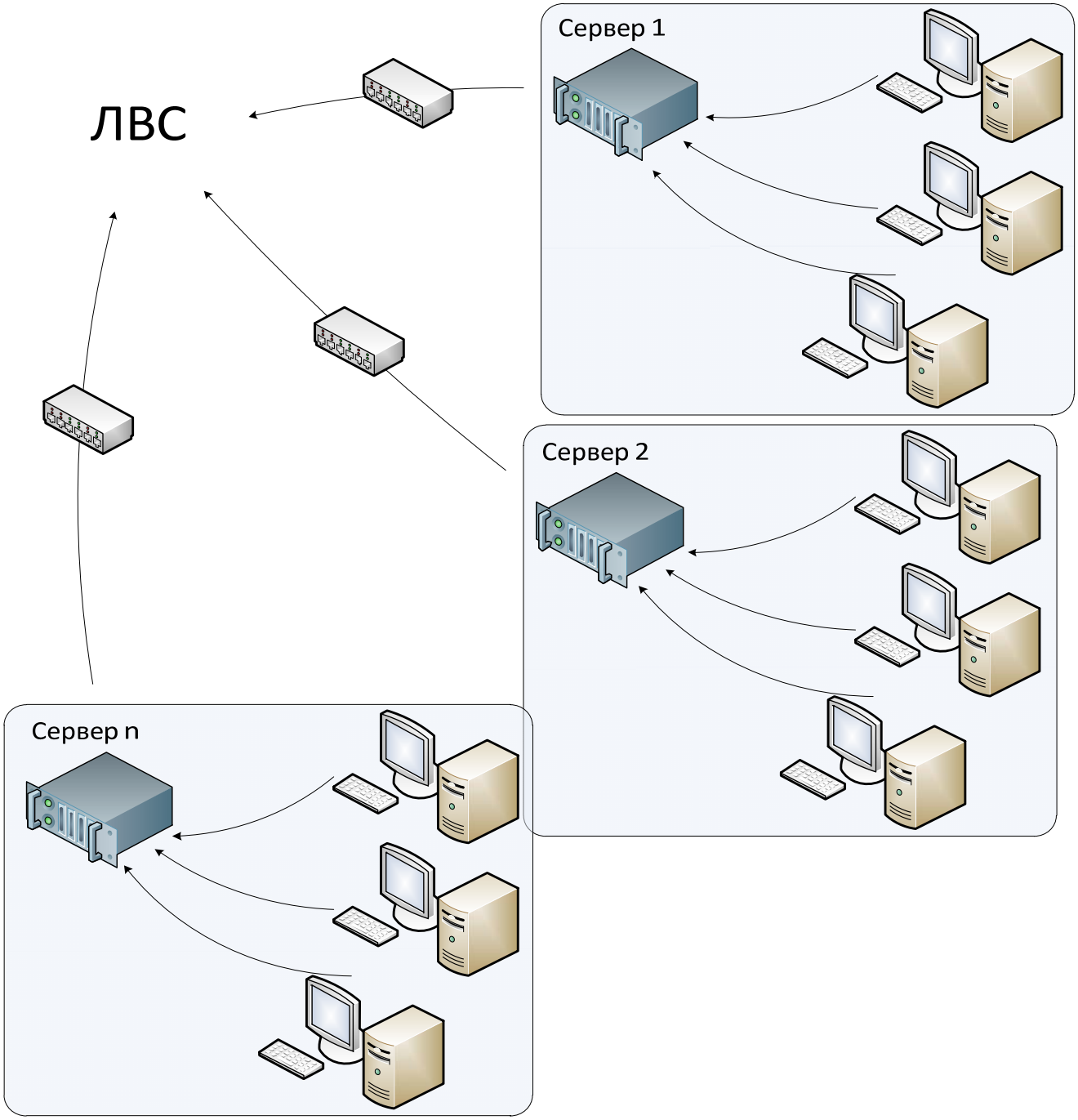


Рис. 1. Первоначальная организация инфраструктуры РВУЗ «КИПУ».

Перед лабораторией, обслуживающей эти серверы, встала непростая задача оптимизации компьютерных мощностей, а также решение задач безопасности. Очевидно, что в каждой лаборатории ставить по одному серверу (в данном случае речь идет не только о программной части, но и об аппаратной) не является рациональным решением. Собрать все такие сервера, порядка восьмидесяти физических машин, в одной лаборатории тоже не целесообразно. К тому же каждая такая физическая машина использует не 100% своих возможностей, а максимум 5–15%.

На схеме, представленной на рис. 1, в качестве серверов (№ 1, № 2, …, № *n*) могут быть следующие: медиа-сервер; сервер авторизованного доступа к сети Интернет; сервер 1.С:Предприятие; сервер Научной библиотеки «Ирбис»; почтовый сервер.

Рассмотрим другой вариант организации инфраструктуры (рис. 2), который предусматривает замену всех машин одной. В этом случае актуальным и эффективным является внедрение понятия виртуализации IT-инфраструктуры учебного заведения. На сегодняшний день разные компании предлагают различные решения виртуализации, однако наиболее надежным и получившим признание во всем мире является WMvare ESX [8].

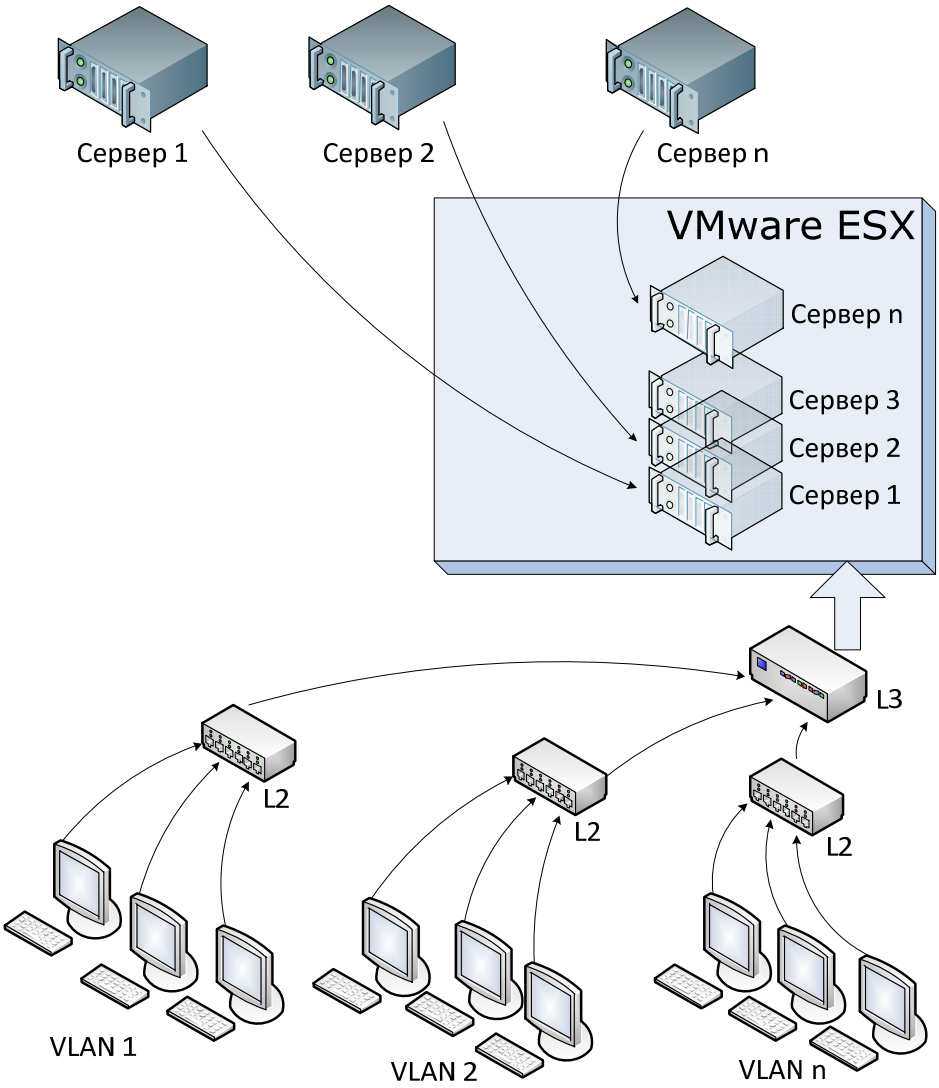


Рис. 2. Современная организация IT-инфраструктуры РВУЗ «КИПУ».

В данном случае, как представлено на рис. 2, на одной физической машине разворачивается программное обеспечение для виртуализации ESX, предлагаемое VMware [8]. Оно устанавливается непосредственно на физическом сервере, позволяет разделить его на несколько виртуальных машин, которые могут работать одновременно, распределяя физические ресурсы основного сервера под определенные задачи (в нашем случае – разворачивание виртуального сервера).

Рассмотрим технические аспекты виртуализации.

1. ***Подбор оборудования.*** Оборудование должно поддерживать возможность аппаратной виртуализации. Если оборудование этого поддерживать не будет, виртуализация возможна, но качество такой виртуализации заметно ниже. Поддержку аппаратной виртуализации имеют процессоры Intel [9–10] и AMD [11]. У компании Intel это Intel Virtualization Technology (Intel VT), а у AMD – AMD Virtualization (AMD-V). Перечень процессоров данных компаний, поддерживающих аппаратную виртуализацию или технологии Intel VT и AMD-V можно просмотреть, соответственно, в [12] и [13].

Для реализации виртуализации в РВУЗ «КИПУ» был использован сервер с процессором компании AMD – Opteron, 16 ГБ оперативной памяти, двумя сетевыми контроллерами 1 ГБ/с, четырьмя жесткими дисками по 250 ГБ. В качестве платформы для виртуализации серверов была выбрана VMware Server.

2. ***Локальные вычислительные сети.*** Локальная сеть должна быть структурированной, с высокой степенью защиты от внутренних и внешних угроз (это обеспечивают межсетевые экраны седьмого уровня). Каждый виртуальный сервер в итоге будет смаршрутизирован при помощи коммутаторов третьего уровня L3. То есть, учебная лаборатория и виртуальный сервер, с которым она работает, должны быть (не является обязательным требованием, но дает возможность увеличить продуктивность) в одной виртуальной сети, так называемой VLAN.

3. *Высококвалифицированный персонал.* Естественно, оплата труда такого персонала требует больших затрат. Однако несколько таких специалистов смогут заменить целые отделы, которые ведут дела по администрированию нескольких лабораторий.

4. *Экономичность.* Наращивание мощности может происходить путем обновления сервера, что является более эффективным и экономичным по сравнению с заменой каждой рабочей станции. При обновлении следует учитывать следующий ряд параметров: процессор, память, жесткий диск, сетевую карту.

Приведенный список параметров может изменяться в зависимости от поставленных задач.

Таким образом, в результате реорганизации текущей сетевой корпоративной структуры сформирована качественно новая инфраструктура. При организации подобной виртуализации есть возможность быстро делать backup каждого виртуального сервера по расписанию, что позволяет восстанавливать работоспособность в считанные минуты в случае сбоев.

Аппаратная виртуализация представляет собой развивающуюся технологию, у которой есть все возможности стать доминирующей, особенно для серверных платформ, потому что она имеет потенциал, чтобы способствовать консолидации нескольких рабочих нагрузок на одном физическом сервере без дополнительного программного обеспечения.

ЛИТЕРАТУРА

1. Самойленко А. Что такое виртуализация и виртуальные машины [Электронный ресурс] / А. Самойленко // Виртуализация для бизнеса. – 2012. – Режим доступа : http://www.vmworld.ru/что-такое-виртуализация.
2. Virtual Computing Lab [Электронный ресурс] // North Carolina State University. – 2011. – Режим доступа : URL:http://vcl.ncsu.edu.
3. Analytics, student labs and research collaboration available through IBM SmartCloud [Электронный ресурс] // New IBM Cloud Services to Address Education Challenges. – 2012. – Режим доступа : http://www-03.ibm.com/press/us/en/pressrelease/ 34642.wss.
4. Силаков Д. В. Использование аппаратной виртуализации в контексте информационной безопасности / Д. В. Силаков ; [под ред. В. П. Иванникова] // Труды Института системного программирования РАН. – 2011. – Т. 20. – С. 25–35.
5. Ruest D. Virtualization, A Beginner's Guide / Danielle Ruest, Nelson Ruest. – McGraw-Hill Osborne Media, 2009 – 464 p.
6. Hess K. Practical Virtualization Solutions: Virtualization from the Trenches / Kenneth Hess, Amy Newman. – Prentice Hall, 2009. – 336 p.
7. McCain C. Mastering VMware Infrastructure 3 / Chris McCain, Rawlinson Rivera. – Wiley Publishing, Inc., 2008. – 576 p.
8. VMware [Электронный ресурс] // Company home page. – 2012. – Режим доступа : http://www.vmware.com.
9. Hardware-Assisted Virtualization Technology [Электронный ресурс] // Intel Corporation official website. – 2012. – Режим доступа : http://www.intel.com/content/ www/us/en/virtualization/virtualization-technology/hardware-assist-virtualization-technology.html.
10. Intel® Virtualization Technology List [Электронный ресурс] // Intel Corporation official website. – 2012. – Режим доступа : http://ark.intel.com/VTList.aspx.
11. AMD Virtualization (AMD-V^{TM}) Technology [Электронный ресурс] // AMD Corporate Website. – 2012. – Режим доступа : http://sites.amd.com/ru/business/ it-solutions/virtualization/Pages/amd-v.aspx.
12. Поддержка виртуализации процессорами Intel [Электронный ресурс] // Виртуализация VMware. – 2011. – Режим доступа: http://www.vsphere5.ru/ doku.php?id=technical-info:intel-vt-support.
13. Поддержка виртуализации процессорами AMD [Электронный ресурс] // Виртуализация VMware. – 2011. – Режим доступа : http://www.vsphere5.ru/ doku.php?id=amd-v-support.

УДК 530.1

Кубовская Т. Н., Темненко В. А.

ПРОЦЕДУРА ЧИСЛЕННОГО МОДЕЛИРОВАНИЯ ВОЛН ЯНГА-МИЛЛСА

Представлена постановка задачі про плоску тріплетну хвилю Янга-Міллса, як завдання класичної математичної фізики. Детально викладено спосіб забезпечення кіральної визначеності рішення при чисельному рішенні рівнянь Янга-Міллса.

Ключові слова: *тріплет Янга-Міллса, рівняння Янга-Міллса, плоска хвиля Янга-Міллса, киральність.*

Представлена постановка задачи о плоской триплетной волне Янга-Миллса как задачи классической математической физики. Детально изложен способ обеспечения киральной определенности решения при численном решении уравнений Янга-Миллса.

Ключевые слова: *триплет Янга-Миллса, уравнения Янга-Миллса, плоская волна Янга-Миллса, киральность.*

We described formulation of Yang-Mills theory's plane wave triplet problem as a classical problem of the mathematical physics. We presented in detail the way of providing determined chirality solutions for the numerical solution of the Yang-Mills theory.

Key words: *Yang-Mills' triplet, Yang-Mills' equation, Yang-Mills' plane wave, chirality.*

Постановка проблемы. Поля Янга-Миллса появились в теоретической физике в середине двадцатого века [1; 2]. В настоящее время эти поля составляют важную часть так называемой Стандартной модели (например, [3]).

Лагранжиан Стандартной модели содержит три сектора – один синглетный (максвелловский) и два янг-миллсовских, триплетный и октуплетный, – описывая единообразно электромагнитное, слабое и сильное взаимодействие элементарных частиц. Каждый сектор лагранжиана Стандартной модели содержит три слагаемых, каждое из которых отображает, соответственно, вклад токов, вклад свободных (безтоковых) полей и вклад взаимодействия токов и полей. Лагранжиан свободных полей описывает три типа волн – синглетную максвелловскую волну и два типа волн Янга-Миллса – триплетную и октуплетную. Эти волны в отсутствие токов не взаимодействуют друг с другом и могут быть рассмотрены независимо.

Анализ литературы. Теория синглетных максвелловских волн – предмет изучения классической электродинамики. Эта теория представлена во многих превосходных учебниках и монографиям, например, в классическом учебнике Л. Д. Ландау и Е. М. Лифшица [4].

Теория классических триплетных и октуплетных волн Янга-Миллса, несомненно, заслуживает столь же тщательного и детального изложения как специальный раздел математической физики. В виду нелинейности полевых уравнений Янга-Миллса эта теория более сложна, чем теория электромагнитных волн. Насколько известно авторам, в литературе отсутствует даже описание процедур численного моделирования простейшего объекта теории волн Янга-Миллса – плоской волны.

Цель статьи – описание процедур численного моделирования триплетной волны Янга-Миллса.

Изложение основного материала.

1. ***Лагранжиан и полевые уравнения.*** Базовой переменной теории триплетных волн Янга-Миллса является тройка четырехмерных векторов-потенциалов $\overset{a}{W}_\mu$, …. Надбуквенные индексы a, b, c, … принимают значения 1, 2, 3, нумеруя Янга-Миллсовские (далее – YM) компоненты триплета. Греческие индексы μ, ν, … принимают значения 0, 1, 2, 3, нумеруя лоренцовские компоненты 4-вектора потенциала; индекс 0 соответствует временной компоненте, остальные индексы нумеруют пространственные компоненты векторов. Тройку YM-потенциалов можно трактовать как один объект – вектор в трехмерном YM-пространстве, с обычной трехмерной векторной алгеброй, позволяющей использовать безиндексную запись YM-векторов, а также скалярное и векторное произведение YM-векторов. Скалярное произведение YM-векторов A и B обозначается обычным образом $\mathbf{A}\cdot\mathbf{B} = \overset{a}{A}\overset{a}{B}$ (по повторяющемуся YM-индексу суммирование от 1 до 3). Векторное произведение YM-векторов A и B задается с помощью символа Леви-Чивитта $\overset{abc}{\varepsilon}$: если C = A×B, то $\overset{a}{C} = \overset{abc}{\varepsilon}\overset{b}{A}\overset{c}{B}$.

Полевой тензор триплетной теории имеет вид [3]:

$$W_{\mu\nu} = \partial_\mu W_\nu - \partial_\nu W_\mu + p_T W_\mu \times W_\nu \qquad (1)$$

или, при явном указании YM-индексов:

$$\overset{a}{W}_{\mu\nu} = \partial_\mu \overset{a}{W}_\nu - \partial_\nu \overset{a}{W}_\mu + p_T \overset{abc}{\varepsilon} \overset{b}{W}_\mu \overset{c}{W}_\nu . \qquad (2)$$

Все переменные теории могут считаться безразмерными. Для измерения потенциалов $\overset{a}{W}_\mu$ используется некоторая стандартная единица заряда (например, заряд электрона) и некоторая фундаментальная (но не конкретизируемая) единица длины. Скорость света принята за единицу.

Символ ∂_μ в (1) и (2) означает производную по пространственно-временной координате x^μ. Постоянная триплетной теории p_T, фигурирующая в записи тензора поля важна только при рассмотрении двухсекторных физических процессов, развивающихся одновременно в синглетном и триплетном секторе физики. Если же рассматриваются свободные YM-волны, расположенные в YM-секторе физики, то можно положить, что $p_T = 1$, поскольку параметр p_T можно устранить из (1) с помощью тривиального перемасштабирования потенциала W_μ.

Лагранжиан свободного триплетного поля имеет вид [3]:

$$L = -\frac{1}{16\pi} W_{\mu\nu} \cdot W^{\mu\nu} . \qquad (3)$$

В выражении (3) подразумевается суммирование по повторяющимся лоренцевским индексам с учетом обычного изменения знака при опускании/поднимании индексов, соответствующих пространственным компонентам 4-вектора x^μ [4].

Минимизация функционала «действие», содержащего лагранжиан (3), проинтегрированный по некоторой пространственновременной области, дает следующие полевые уравнения триплетной теории Янга-Миллса [3]:

$$\partial_\mu W^{\mu\nu} + W_\mu \times W^{\mu\nu} = 0. \qquad (4)$$

Исключая из уравнений Янга-Миллса тензор $W^{\mu\nu}$ с помощью (2), можем представить уравнение Янга-Миллса в виде системы нелинейных волновых уравнений:

$$-\Box W^\nu + W_\mu \times \left(2\partial^\mu W^\nu - \partial^\nu W^\mu + W^\mu \times W^\nu\right) =$$
$$= \partial^\nu \ell + W^\nu \times \ell. \qquad (5)$$

В волновых уравнения (5) $\Box$ – оператор Д'Аламбера:

$$\Box = -\partial_\mu \partial^\mu,$$

а ℓ – 4-дивергенция YM-потенциала:

$$\ell = \partial_\mu W^\mu.$$

На потенциалы W_μ может быть наложено одно произвольное калибровочное условие [3]. Удобно положить, что дивергенция ℓ тождественно равна нулю. Легко показать, что это условие совместимо с полевыми уравнениями (5).

При такой калибровке волна Янга-Миллса описывается следующими уравнениями

$$\Box \mathbf{W}^\nu = \mathbf{W}_\mu \times \left(2\partial^\mu \mathbf{W}^\nu - \partial^\nu \mathbf{W}^\mu + \mathbf{W}^\mu \times \mathbf{W}^\nu\right);$$
$$\partial_\mu \mathbf{W}^\mu = 0. \qquad (6)$$

2. *Плоская волна Янга-Миллса.* Простейшие решения полевых уравнений (6) можно искать в форме плоских волн с волновым 4-вектором k^μ, полагая, что двенадцать искомых функций $\overset{a}{W}_\mu$ зависят только от одного аргумента – фазы волны φ:

$$\varphi = k_\mu x^\mu.$$

Для плоской YM-волны полевые уравнения (6) превращаются в систему обыкновенных дифференциальных уравнений:

$$k_\mu k^\mu W_\nu'' + 2k_\mu W^\mu \times W_\nu' - k_\nu W^\mu \times W_\mu' + W_\mu \times$$
$$\times \left(W^\mu \times W_\nu\right) = 0;$$
$$k^\mu W_\mu' = 0. \qquad (7)$$

В уравнениях (7) штрих означает производную по волновой фазе φ.

Уравнения плоской волны (7) обладают очевидной масштабной инвариантностью по отношению к волновому вектору k^μ.

Пусть k^μ – времениподобный вектор, т. е.

$$k^\mu k_\mu = \varepsilon^2 > 0. \qquad (8)$$

Произведем масштабное преобразование потенциалов W^μ, координат x^μ и волнового вектора k^μ:

$$\mathbf{W}^\mu \Rightarrow \varepsilon \mathbf{W}^\mu;\ x^\mu \Rightarrow \frac{1}{\varepsilon} x^\mu;\ k^\mu \Rightarrow \varepsilon k^\mu. \qquad (9)$$

Преобразование (9) не меняет фазы волны и не меняет вида уравнений (7). Оно позволяет считать волновой вектор нормированным на единицу ($\varepsilon = 1$ в уравнении (8)).

Из условия $k^\mu W_\mu' = 0$, содержащегося в (7), следует, что свертка $k^\mu W_\mu$ не зависит от фазы волны:

$$k^\mu W_\mu = q = \text{const}.$$

Мы будем полагать, что YM-вектор q тождественно равен нулю. Отличие q от нуля означало бы существование некоторого постоянного YM-поля, внешнего по отношению к волне, на фоне которого распространяется волна.

С учетом изложенного, задача о плоской волне принимает вид:

$$W_\nu'' - k_\nu W^\mu \times W_\mu' + W_\mu \times \left(W^\mu \times W_\nu\right) = 0 \quad \text{(a)}$$
$$k^\mu W_\mu = 0 \quad \text{(b)} \qquad (10)$$
$$k^\mu k_\mu = 1 \quad \text{(c)}.$$

Свертка уравнения (10 a) с волновым вектором k^μ с учетом (10 b) и (10 c) дает следующий результат:

$$W^\mu \times W_\mu' = 0, \qquad (11)$$

что позволяет упростить запись волнового уравнения (10 a):

$$\mathbf{W}_\nu'' + \mathbf{W}_\mu \times \left(\mathbf{W}^\mu \times \mathbf{W}_\nu\right) = 0, \qquad (12)$$

или, в индексной записи

$$\overset{a}{W}{}_\nu'' + \overset{ab}{I}\, \overset{b}{W}_\nu = 0, \qquad (13)$$

где $\overset{ab}{I} = \overset{a}{W}_\mu \overset{b}{W}{}^\mu - \overset{ab}{\delta}\, \overset{c}{W}_\mu \overset{c}{W}{}^\mu$ – есть симметричный YM-тензор второго ранга ($\overset{ab}{\delta}$ – символ Кронекера); тензор $\overset{ab}{I}$ уместно назвать тензором инерции Янга-Миллса.

Уравнения первого порядка (11) можно рассматривать в качестве неголономных условий, наложенных на динамическую систему (13). Достаточно потребовать, чтобы эти условия выполнялись при одном значении фазы, т. е. являлись ограничениями, наложенными на начальные условия – в силу волновых уравнений (13) они будут выполнены при всех значениях фазы.

3. *Плоская волна Янга-Миллса в собственной системе отсчета.* Обозначим временную компоненту потенциала $\overset{a}{W}{}^0$ через $\overset{a}{T}$, а три пространственные компоненты $\overset{a}{W}{}^\mu$ через $\overset{a}{U}$: $\overset{a}{W}{}^\mu = \left\{\overset{a}{T}; \overset{a}{U}\right\}$.

Векторы $\overset{a}{U}$ образуют тройку обычных трехмерных векторов.

Задача о плоской волне Янга-Миллса проще всего решается в собственной системе волны, в которой волновой вектор k^{μ} имеет вид:

$$k^{\mu} = \{1,0,0,0\}. \qquad (14)$$

В этой системе отсчета все три $\overset{a}{T} = 0$ в силу (10 b). Фаза волны в этой системе совпадает с собственным временем *t* в этой системе. Следовательно, все ненулевые компоненты потенциала $\overset{a}{U}$ в этой системе не зависят от пространственных переменных, а зависят только от времени: волна «вспыхивает» и «гаснет» одновременно по всему бесконечному трехмерному пространству.

Наша глубоко укорененная лоренцовская интуиция, несомненно, протестует против этой картины. Однако здесь нет нарушения лоренц-инвариантности физики. Плоская волна Янга-Миллса с фиксированным волновым вектором k^{μ} является идеализированным математическим объектом, охватывающим все пространство и требующим для своего создания бесконечно большой энергии; реальная YM-волна, локализованная в конечной пространственной области и несущая конечную полную энергию, является волновым пакетом, т. е. суперпозицией плоских волн с различными волновыми векторами – чем компактней пространственная локализация волнового пакета, тем шире дисперсия в пространстве волновых векторов.

Понятие «собственная система отсчета» имеет смысл только для плоской волны с фиксированным волновым вектором. Для волнового пакета это понятие лишено смысла. Однако математика волновых пакетов для нелинейных волн Янга-Миллса представляется нам в данный момент практически «нетрактабельной». Мы ограничиваемся здесь одиночной волной с фиксированным волновым вектором.

В собственной системе отсчета уравнения (13) принимают следующий вид:

$$\overset{a}{U}{}^{\bullet\bullet} + \overset{ab}{I}\,\overset{b}{U} = 0, \qquad (15)$$

где

$$\overset{ab}{I} = \overset{ab}{\delta}\,U^2 - \overset{a}{U}\cdot\overset{b}{U}, \qquad (16)$$

$$U^2 = \sum_a \overset{a}{U}{}^2 .$$

Точка в (15) означает производную по собственному времени волны.

Неголономные связи (11) приобретают при этом следующий вид:

$$\overset{abc}{\varepsilon}\,\overset{b}{U}\cdot\overset{c}{U}{}^{\bullet} = 0. \qquad (17)$$

Система уравнений (15) имеет интеграл энергии:

$$K + U = E, \qquad (18)$$

где «кинетическая» энергия: $K = \frac{1}{2}\sum_a \left(\overset{a}{U}{}^{\bullet}\right)^2$, а «потенциальная» энергия:

$$U = \frac{1}{2}\left(\left(\overset{1}{\mathbf{U}}\times\overset{2}{\mathbf{U}}\right)^2 + \left(\overset{2}{\mathbf{U}}\times\overset{3}{\mathbf{U}}\right)^2 + \left(\overset{3}{\mathbf{U}}\times\overset{1}{\mathbf{U}}\right)^2\right). \qquad (19)$$

Задача является масштабно-инвариантной по отношению к величине энергии E. Масштабное преобразование: $\overset{a}{U} \Rightarrow E^{1/4}\,\overset{a}{U}$; $t \Rightarrow E^{-1/4}\cdot t$, не меняя вида уравнений (15), (16), позволяет положить в (18) E = 1, т. е. нормировать энергию на единицу. Фактически, это не вся энергия волны (она бесконечна), а величина, пропорциональная плотности энергии волны в одиночном объеме.

4. *Проблема киральности.* Уравнения Янга-Миллса (15) или (13) являются кирально-симметричными, т. е. не различают правого и левого. Они допускают построение кирально-симметричных решений. Эти решения носят характер хаотических колебаний с непредсказуемыми всплесками амплитуды осцилляций по некоторым переменным с одновременным угасанием амплитуды осцилляций по другим переменным. Поскольку система консервативна, она не имеет регулярного или странного аттрактора. Траектория решения постепенно заполняет в координатном пространстве энергетически доступную область. Возможно, решения системы (15) являются одним из самых красивых и впечатляющих математических объектов современной теории динамических систем. Сами по себе зависимости переменных $\overset{a}{U}$ от времени t вряд ли имеет смысл изучать: они подвержены так называемому «эффекту бабочки», т. е. чрезвычайно чувствительны к выбору схемы дискретизации задачи и выбору шага интегрирования. Интерес представляет статистическая обработка решений, т. е. изучение плотности заполнения в девятимерном конфигурационном пространстве.

Однако исходные переменные триплетной теории Янга-Миллса не обладают киральной симметрией. Сам факт присутствия в теории янг-миллсовского векторного произведения означает, что мы не имеем права произвольно перенумеровывать векторы в YM-тройке: эта тройка должна иметь фиксированную ориентацию. В частности, для рассматриваемой задачи о плоской волне это означает, что ориентация тройки векторов $\overset{a}{U}$ должна совпадать с выбранной ориентацией пространственной системы координат. Следить за ориентацией удобно по знаку смешанного скалярно-векторного произведения трех векторов $\overset{a}{U}$. Назовем это произведение «киральным детерминантом» CD:

$$CD\left(\overset{1}{\mathbf{U}},\overset{2}{\mathbf{U}},\overset{3}{\mathbf{U}}\right) = \left(\overset{1}{\mathbf{U}}\times\overset{2}{\mathbf{U}}\right)\cdot\overset{3}{\mathbf{U}} = \overset{1}{\mathbf{U}}\cdot\left(\overset{2}{\mathbf{U}}\times\overset{3}{\mathbf{U}}\right).$$

Условие киральной определенности (chirality condition) имеет вид:

$$\mathrm{CD}\left(\overset{1}{\mathrm{U}},\overset{2}{\mathrm{U}},\overset{3}{\mathrm{U}}\right)\geq 0. \qquad (20)$$

Этому условию должны быть подчинены начальные условия. Кроме того, оно должно соблюдаться в любой момент времени. Непрерывные гладкие решения уравнений Янга-Миллса (15) не подчиняются условию киральной определенности (20): величина CD осциллирует, меняя знак. Выполнить условие (20) можно только жертвуя непрерывностью либо гладкостью решений.

Способ, основанный на потере непрерывности решения, состоит в перестановке векторов $\overset{1}{\mathrm{U}}$ и $\overset{2}{\mathrm{U}}$ в те моменты времени, когда CD = 0. Компонента $\overset{3}{\mathrm{U}}$ не должна затрагиваться этой перестановкой по следующим соображениям. Третья YM-компонента YM-тока связана с максвелловским током. Иными словами, третья компонента YM-векторов изначально выделена и «неравноправна» с компонентами номер 1 и номер 2. Компоненты 1 и 2 равноправны и различаются только условием киральной определенности (20). При нарушении этого условия следует просто переставить местами эти векторы, и условие восстановится. Назовем это «жесткой кирализацией» решения. «Жесткая кирализация» порождает разрывные решения, столь же хаотичные, как и непрерывные и гладкие кирально-симметричные решения киральносимметричных уравнений (15). Собственно, это те же самые киральносимметричные решения, но «переинтерпретированные» за счет спорадических перестановок 1↔2 в YM-индексах.

Однако подобная «жесткая кирализация», – шокирующий, но законный способ обеспечения киральной определенности решения, – приемлемая в триплетном секторе физики, оказывается недопустимой для октуплетного сектора. Для обеспечения киральной определенности восьми векторов октуплетной волны (такая волна является классической моделью квантового объекта – глюона) необходимо делать перестановку векторов одновременно в нескольких тройках векторов (структура этих троек определяется уже не символом Леви-Чивитта, а структурными константами группы SU(3)), что оказывается невозможным.

Поэтому обратимся к рассмотрению другого способа обеспечения условия киральной определенности (20), который основан на использовании непрерывных, но негладких решений.

Этот способ, который мы назовем «мягкой кирализацией», основан на механической интерпретации условия (20) как односторонней, неудерживающей голономной связи, наложенной на решение.

Такая связь, заданная неравенством, возникает, например, в задаче о плоском движении математического маятника на нерастяжимой нити. Если нить натянута, то точечный маятник совершает колебания, двигаясь по дуге окружности радиуса, равного длине нити. Если же нить не натянута, точечный маятник, как свободная материальная точка, движется по дуге параболы в однородном поле тяжести. В моменты, когда нить натягивается, радиальная компонента скорости маятника скачком меняет знак, азимутальная компонента скорости остается прежней.

Опираясь на эту механическую аналогию, мы сформулируем следующий способ «кирализации» решения: в те моменты времени, когда $\mathrm{CD}\left(\overset{1}{\mathrm{U}},\overset{2}{\mathrm{U}},\overset{3}{\mathrm{U}}\right)=0$, знаки всех векторов скоростей $\overset{a}{\mathrm{U}}$ меняются на противоположные. Этот способ «мягкой кирализации» порождает непрерывные, но негладкие кирально определенные решения: в моменты «обнуления» кирального определителя CD скорости терпят разрыв.

Этот способ «мягкой кирализации» дословно переносится в октуплетный сектор: при «обнулении» хотя бы одного из множества киральных определителей меняются знаки скоростей изменения всех восьми векторов октуплета.

Заметим, что в задаче о движении математического маятника можно избавиться от решений с разрывом скорости, вводя более сложную модель связи – например, учитывая упругую растяжимость нити, на которой подвешен маятник. В задачах Янг-Миллсовской физики такая возможность отсутствует: у нас нет никакой более точной физики за пределами физики Янга-Миллса. Мы должны смириться с негладкостью кирально-определенных YM-решений.

В общем случае CD обращается в нуль, когда три YM-вектора $\overset{a}{\mathrm{U}}$ оказываются в одной плоскости. Это «плоскость отражения», – ударяясь о нее, все три вектора упруго отскакивают, обеспечивая киральную определенность решения.

При произвольных начальных условиях возникают две такие «плоскости отражения», и точки, изображающие концы трех YM-векторов, совершают периодическое движение между этими двумя плоскостями.

Отметим, что обнаруженная нами при численном решении периодичность решения была для авторов неожиданным результатом и подлинным подарком Янг-Миллсовской физики – на

фоне многих сотен часов вычислений, затраченных на изучение невероятно хаотичных киральносимметричных решений. В численном моделировании киральносимметричных решений принимали участие в 2010–2011 гг. Валерий Брагин и Станислав Лебёдкин; все вычисления выполнялись на персональных компьютерах, но по объему вычислений, необходимых для сколь-нибудь полного исследования YM-хаоса в киральносимметричной задаче – это суперкомпьютерная задача.

Период колебаний киральноопределенного триплетного решения, как показывают вычисления, зависит от начальных условий, однако изучение этой зависимости в 14-тимерном пространстве начальных условий пока представляется нам неразрешимой задачей. Размерность пространства начальных условий – это размерность фазового пространства триплетной YM-волны (18) минус количество ограничений, налагаемых на начальные условия. Таких ограничений четыре – условие нормировки энергии на единицу и три неголономных условия (17).

Контроль за точностью вычислений можно осуществлять, контролируя точность выполнения этих четырех условий в процессе счета.

Выводы. Описанная процедура численного моделирования триплетной волны Янга-Миллса позволяет провести широкомасштабные численные эксперименты по изучению этих волн. При небольшом обобщении этой процедуры можно численно исследовать плоскую волну Янга-Миллса в октуплетном секторе физики.

ЛИТЕРАТУРА

1. Yang C. N. Conservation of isotopic spin and isotopic gauge invariance / C. N. Yang, R. L. Mills // Phys. Rev. – 1954. – V. 96. – № 1. – P. 191–195.
2. Straumann N. On Pauli's invention of non-abelian Kaluza-Klein theory in 1953 / N. Straumann // arXiv:gr-qc/0012054v1. – 2000. – 4 p.
3. Хуанг К. Кварки, лептоны и калибровочные поля / К. Хуанг ; [пер. с англ.]. – М. : Мир, 1985. – 382 с.
4. Ландау Л. Д. Теоретическая физика : учебное пособие : в 10-ти томах / Л. Д. Ландау, Е. М. Лифшиц. – [7-е изд., испр.]. – М. : Наука ; Гл. ред. физ.-мат. лит., 1988. – Т. II. Теория поля. – 512 с.

НАШИ АВТОРЫ

1. **Абдулгазис Азиз Умерович** – кандидат технических наук, доцент кафедры эксплуатации и ремонта автомобилей Республиканского высшего учебного заведения «Крымский инженерно-педагогический университет», г. Симферополь
2. **Абдулгазис Умер Абдуллаевич** – доктор технических наук, профессор, заведующий кафедрой эксплуатации и ремонта автомобилей, декан инженерно-технологического факультета Республиканского высшего учебного заведения «Крымский инженерно-педагогический университет», г. Симферополь
3. **Абильтарова Эльвиза Нуриевна** – кандидат педагогических наук, старший преподаватель кафедры охраны труда Республиканского высшего учебного заведения «Крымский инженерно-педагогический университет», г. Симферополь.
4. **Абитова Шазие Юсуфовна** – кандидат сельскохозяйственных наук, доцент, доцент кафедры охраны труда Республиканского высшего учебного заведения «Крымский инженерно-педагогический университет», г. Симферополь
5. **Аблялимова Эльзара Изетовна** – аспирант кафедры информационно-компьютерных технологий Республиканского высшего учебного заведения «Крымский инженерно-педагогический университет», г. Симферополь
6. **Абхаирова Сусана Велишаевна** – преподаватель кафедры технологического образования Республиканского высшего учебного заведения «Крымский инженерно-педагогический университет», г. Симферополь
7. **Аметов Исмаил Энверович** – кандидат химических наук, преподаватель кафедры технологического образования Республиканского высшего учебного заведения «Крымский инженерно-педагогический университет», г. Симферополь
8. **Бабицкий Леонид Федорович** – доктор технических наук, профессор, заведующий кафедрой механизации, энергетики и технического сервиса Южного филиала Национального университета биоресурсов и природопользования Украины «Крымский агротехнологический университет», г. Симферополь
9. **Бекиров Расим Нафеевич** – кандидат технических наук, доцент, заведующий кафедрой охраны труда в машиностроении Республиканского высшего учебного заведения «Крымский инженерно-педагогический университет», г. Симферополь.
10. **Богуцкий Борис Владимирович** – студент третьего курса специальности «Технология машиностроения» Севастопольского национального технического университета, г. Севастополь
11. **Богуцкий Владимир Борисович** – старший преподаватель кафедры технологии машиностроения Севастопольского национального технического университета, г. Севастополь
12. **Гордеева Элеонора Сергеевна** – старший преподаватель кафедры технологии машиностроения Севастопольского национального технического университета, г. Севастополь
13. **Гусев Виталий Александрович** – эксперт сектора пожарно-технических исследований отдела взрывотехнических и пожарно-технических исследований Научно-исследовательского экспертно-криминалистического центра при ГУМВД Украины в АР Крым
14. **Ена Владимир Дмитриевич** – кандидат технических наук, доцент кафедры сельскохозяйственной техники Южного филиала Национального университета биоресурсов и природопользования Украины «Крымский агротехнологический университет», г. Симферополь
15. **Ильясова Фатиме Серверовна** – преподаватель кафедры информационно-компьютерных технологий Республиканского высшего учебного заведения «Крымский инженерно-педагогический университет», г. Симферополь
16. **Кадыров Мемет Рустамович** – аспирант кафедры информационно-компьютерных технологий Республиканского высшего учебного заведения «Крымский инженерно-педагогический университет», г. Симферополь
17. **Канареев Феликс Николаевич** – кандидат технических наук, доцент кафедры технологии машиностроения Севастопольского национального технического университета, г. Севастополь
18. **Клец Дмитрий Михайлович** – аспирант кафедры технологии машиностроения и ремонта машин Харьковского национального автомобильно-дорожного университета, г. Харьков
19. **Кормилец Александр Владимирович** – магистрант кафедры технологии машиностроения и металлорежущих станков Национального технического университета «Харьковский политехнический институт» г. Харьков

20. **Крылов Владимир Сергеевич** – кандидат биологических наук, доцент, старший преподаватель кафедры информационно-компьютерных технологий Республиканского высшего учебного заведения «Крымский инженерно-педагогический университет», г. Симферополь
21. **Кубовская Татьяна Николаевна** – магистрант кафедры прикладной математики Таврического национального университета им. В. И. Вернадского, г. Симферополь
22. **Кувшинов Андрей Алексеевич** – кандидат технических наук, доцент, доцент кафедры механизации, энергетики и технического сервиса Южного филиала Национального университета биоресурсов и природопользования Украины «Крымский агротехнологический университет», г. Симферополь
23. **Куклин Владимир Алексеевич** – ассистент кафедры механизации, энергетики и технического сервиса Южного филиала Национального университета биоресурсов и природопользования Украины «Крымский агротехнологический университет», г. Симферополь
24. **Манжос Леонид Александрович** – преподаватель кафедры информационно-компьютерных технологий Республиканского высшего учебного заведения «Крымский инженерно-педагогический университет», г. Симферополь
25. **Мевлют Шевхи Тевабильевич** – кандидат физико-математических наук, доцент, доцент кафедры сварочного производства Республиканского высшего учебного заведения «Крымский инженерно-педагогический университет», г. Симферополь
26. **Москалевич Вадим Юрьевич** – кандидат технических наук, ассистент кафедры механизации, энергетики и технического сервиса Южного филиала Национального университета биоресурсов и природопользования Украины «Крымский агротехнологический университет», г. Симферополь
27. **Москаленко Ирина Сергеевна** – студентка третьего курса специальности «Информатика» Республиканского высшего учебного заведения «Крымский инженерно-педагогический университет», г. Симферополь
28. **Мустафаева Эльзара Изетовна** – студентка третьего курса специальности «Информатика» Республиканского высшего учебного заведения «Крымский инженерно-педагогический университет», г. Симферополь
29. **Новиков Павел Анатольевич** – старший преподаватель кафедры технологии машиностроения Севастопольского национального технического университета, г. Севастополь
30. **Оболонский Виктор Васильевич** – директор ЧП «Семья», г. Симферополь
31. **Овчаренко Федор Александрович** – аспирант кафедры сельскохозяйственной техники Южного филиала Национального университета биоресурсов и природопользования Украины «Крымский агротехнологический университет», г. Симферополь
32. **Подригало Михаил Абович** – доктор технических наук, профессор, заведующий кафедрой технологии машиностроения и ремонта машин Харьковского национального автомобильно-дорожного университета, г. Харьков
33. **Подригало Надежда Михайловна** – кандидат технических наук, доцент кафедры инженерной и компьютерной графики Харьковского национального автомобильно-дорожного университета, г. Харьков
34. **Резинкина Галина Павловна** – старший преподаватель кафедры технологии машиностроения Севастопольского национального технического университета, г. Севастополь
35. **Сейдаметова Зарема Сейдалиевна** – доктор педагогических наук, профессор, заведующая кафедрой информационно-компьютерных технологий Республиканского высшего учебного заведения «Крымский инженерно-педагогический университет», г. Симферополь
36. **Соболевский Иван Витальевич** – кандидат технических наук, доцент кафедры механизации, энергетики и технического сервиса Южного филиала Национального аграрного университета биоресурсов и природопользования Украины «Крымский агротехнологический университет», г. Симферополь
37. **Сулейманов Эрнест Сейдаметович** – преподаватель кафедры эксплуатации и ремонта автомобилей Республиканского высшего учебного заведения «Крымский инженерно-педагогический университет», г. Симферополь
38. **Тараховский Алексей Юрьевич** – кандидат технических наук, доцент кафедры технологии машиностроения Севастопольского национального технического университета, г. Севастополь
39. **Темненко Валерий Анатольевич** – кандидат физико-математических наук, доцент, доцент кафедры прикладной математики Таврического национального университета им. В. И. Вернадского, г. Симферополь

40. **Умеров Эрвин Джеватович** – старший лаборант кафедры эксплуатации и ремонта автомобилей Республиканского высшего ученого заведения «Крымский инженерно-педагогический университет», г. Симферополь
41. **Усеинов Бекир Казимович** – кандидат технических наук, доцент, доцент кафедры технологии сварочного производства Республиканского высшего ученого заведения «Крымский инженерно-педагогический университет», г. Симферополь
42. **Феватов Сададин Асанович** – преподаватель кафедры эксплуатации и ремонта автомобилей Республиканского высшего ученого заведения «Крымский инженерно-педагогический университет», г. Симферополь
43. **Феденюк Дмитрий Владимирович** – инженер кафедры технологии машиностроения и металлорежущих станков Национального технического университета «Харьковский политехнический институт», г. Харьков
44. **Хабрат Николай Иванович** – доцент кафедры эксплуатации и ремонта автомобилей Республиканского высшего ученого заведения «Крымский инженерно-педагогический университет», г. Симферополь
45. **Халилов Вадим** – кандидат технических наук, доцент, доцент кафедры эксплуатации и ремонта автомобилей Республиканского высшего ученого заведения «Крымский инженерно-педагогический университет», г. Симферополь
46. **Чеботарь Сергей Васильевич** – кандидат технических наук, старший преподаватель кафедры охраны труда Республиканского высшего учебного заведения «Крымский инженерно-педагогический университет», г. Симферополь
47. **Шабанов Николай Петрович** – кандидат технических наук, доцент, доцент кафедры сельскохозяйственной техники Южного филиала Национального университета биоресурсов и природопользования Украины «Крымский агротехнологический университет», г. Симферополь
48. **Шелковой Александр Николаевич** – доктор технических наук, профессор, профессор кафедры технологии машиностроения и металлорежущих станков Национального технического университета «Харьковский политехнический институт» г. Харьков
49. **Шкарбан Фатима Витальевна** – старший преподаватель кафедры информационно-компьютерных технологий Республиканского высшего учебного заведения «Крымский инженерно-педагогический университет», г. Симферополь
50. **Шрон Борис Леонидович** – студент четвертого курса специальности «Технология машиностроения» Севастопольского национального технического университета, г. Севастополь
51. **Шрон Леонид Борисович** – кандидат технических наук, доцент, доцент кафедры технологии машиностроения Севастопольского национального технического университета, г. Севастополь
52. **Эреджепов Марлен Керимович** – старший преподаватель кафедры эксплуатации и ремонта автомобилей Республиканского высшего ученого заведения «Крымский инженерно-педагогический университет», г. Симферополь
53. **Ягьяев Эльмар Энверович** – кандидат технических наук, доцент кафедры эксплуатации и ремонта автомобилей Республиканского высшего ученого заведения «Крымский инженерно-педагогический университет», г. Симферополь

НАУКОВЕ ВИДАННЯ

ВЧЕНІ ЗАПИСКИ КРИМСЬКОГО ІНЖЕНЕРНО-ПЕДАГОГІЧНОГО УНІВЕРСИТЕТУ

Випуск 35

Технічні науки

(*Мовою оригіналу*)

Головний редактор Якубов Ф. Я.
Заступник головного редактору Абдулгазіс У. А.
Відповідальний за випуск Фазилова А. Р.
Коректура та верстка Халілаєва С. Н., Ібрагімова Е. Е.

Підписано до друку 25.06.2012 р. Формат $60\times84^{1}/_{8}$.
Папір офсетний. Гарнітура Times New Roman.
Обл.-вид. друк. арк. 13,7. Об'єм 16,75 друк. арк.
Тираж 100 прим.

Підготовлено до друку та віддруковано
у редакційно-видавничому відділі Науково-інформаційного центру
Республіканського вищого навчального закладу «Кримський інженерно-педагогічний університет»
95015, м. Сімферополь, вул. Севастопольська, пров. Учбовий, 8