

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ  
ЧЕРНІГІВСЬКИЙ НАЦІОНАЛЬНИЙ ТЕХНОЛОГІЧНИЙ УНІВЕРСИТЕТ**

# **ВІЛЬНЕ ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ В ОСВІТІ, НАУЦІ ТА БІЗНЕСІ**

**Тези доповідей п'ятої  
Міжнародної науково-практичної конференції**

22-23 травня 2014 р.  
м. Чернігів

**Чернігів 2014**

Друкується за рішенням вченої ради Чернігівського національного технологічного університету (протокол № 6 від 30 червня 2014 року).

**Вільне програмне забезпечення в освіті, науці та бізнесі:** тези доповідей п'ятої Міжнародної науково-практичної конференції. – Чернігів : Черн. нац. технол. ун-т, 2014. – 83 с.

У збірник включені тези доповідей, які були представлені на п'ятій Міжнародній науково-практичній конференції «Вільне програмне забезпечення в освіті, науці та бізнесі». У доповідях розглянуті наукові та методичні питання щодо використання вільного програмного забезпечення під час організації навчального процесу, перевірки знань, вивчення сучасних технологій та їх впровадження в наукову роботу та на виробництві.

**Голова редакційної колегії:**

Казимир В.В., д-р техн. наук, професор

**Редакційна колегія:**

Роговенко А.І.

Стасюк С.С.

## ЗМІСТ

<b>Новачевський С., Ванюков В. МОДЕЛЮВАННЯ ЗМІНИ ПОКАЗНИКІВ СТАНУ ЗДОРОВ'Я ЛЮДИНИ ТА ПРОГНОЗУВАННЯ МОЖЛИВИХ НАСЛІДКІВ.....</b>	<b>5</b>
<b>Власенко Е., Нестеренко С. ОСОБЕННОСТИ ИСПОЛЬЗОВАНИЯ ФРЕЙМВОРКА STARLING ДЛЯ СОЗДАНИЯ ДВУМЕРНЫХ ИГР.....</b>	<b>6</b>
<b>Горб А. ОБЩИЙ ОБЗОР ЯЗЫКОВ ФУНКЦИОНАЛЬНОГО ПРОГРАМ МИРОВАНИЯ.....</b>	<b>7</b>
<b>Дрозд О. СИСТЕМИ ВЕБІНАРІВ З ВІДКРИТИМ КОДОМ У ДИСТАНЦІЙНОМУ НАВЧАННІ.....</b>	<b>9</b>
<b>Жибер О. ТЕНДЕНЦИИ РАЗВИТИЯ ПОИСКОВЫХ СИСТЕМ.....</b>	<b>11</b>
<b>Запривода Л., Нестеренко С. МОДЕЛИРОВАНИЕ И ВИЗУАЛИЗАЦИЯ ЖИДКОСТЕЙ, ОСНОВАННАЯ НА ДИНАМИКЕ ЧАСТИЦ.....</b>	<b>12</b>
<b>Коляда Т., Стасюк С. ІНФОРМАЦІЙНА КОМП'ЮТЕРНА СИСТЕМА КОНТРОЛЮ ТА УПРАВЛІННЯ ДОСТУПОМ ПАТ ФК «АОА ІНВЕСТМЕНТС».....</b>	<b>15</b>
<b>Коробко Д. КОНТЕЙНЕРЫ СЕРВЛЕТОВ И СЕРВЕРА ПРИЛОЖЕНИЙ.....</b>	<b>16</b>
<b>Мельниченко Д. ЛІЦЕНЗІЙНІ АСПЕКТИ ЗАХИСТУ ПРОГРАМНИХ СИСТЕМ НА ОСНОВІ ЯДРА LINUX ВІД НЕСАНКЦІОНОВАНОГО ВТРУЧАННЯ В ЇХ РОБОТУ... </b>	<b>18</b>
<b>Морозов А. РАЗВИТИЕ ФУНКЦИОНАЛЬНОГО ПРОГРАММИРОВАНИЯ НА ПРИМЕРЕ ЯЗЫКА HASKELL.....</b>	<b>20</b>
<b>Мойсєєнко О., Литвинов В. СТАТИСТИЧНИЙ, МУЛЬТІЯЗИЧНИЙ ПЕРЕКЛАД ЗАПИТІВ ПРИ ІНФОРМАЦІЙНОМУ ПОШУКУ.....</b>	<b>21</b>
<b>Нипорко Я., Акименко А. РЕАЛИЗАЦИЯ ОНЛАЙН-ОПРОСОВ В СИСТЕМЕ МАРКЕТИНГОВЫХ ИССЛЕДОВАНИЙ.....</b>	<b>23</b>
<b>Ксєнз С., Рындич Е. СВОБОДНЫЙ МНОГОПЛАТФОРМЕННЫЙ АУДИО-СЕКВЕНСОР ДЛЯ СОЗДАНИЯ МУЗЫКАЛЬНЫХ КОМПОЗИЦИЙ.....</b>	<b>24</b>
<b>Сытый Н., Нестеренко С. БЫСТРОЕ ПРОТОТИПИРОВАНИЕ АЛГОРИТМОВ АНАЛИЗА ВИДЕО С ПОМОЩЬЮ БИБЛИОТЕКИ КОМП'ЮТЕРНОГО ЗРЕНИЯ OPENCV.....</b>	<b>26</b>
<b>Сычова М. ИСПОЛЬЗОВАНИЕ СИСТЕМЫ УПРАВЛЕНИЯ СОДЕРЖИМЫМ JOOMLA ДЛЯ ПОСТРОЕНИЯ WEB-ПОРТАЛА ПРЕДОСТАВЛЕНИЯ IT-КОНСУЛЬТАЦИЙ ....</b>	<b>27</b>
<b>Скитер И., Прелая О., Гуза Т. ОБЗОР ИНСТРУМЕНТОВ РАЗРАБОТКИ МОДЕЛЕЙ CRID-СРЕДЫ.....</b>	<b>28</b>
<b>Верєвко А., Дэвис Д., Тевкун М. ИСПОЛЬЗОВАНИЕ СЕТЕВОГО ФИЛЬТРА NETFILTER/iptables ДЛЯ РАЗРАБОТКИ АКТИВНОГО УЗЛА СЕТИ.....</b>	<b>30</b>
<b>Устименко І. КОМУНІКАЦІЯ В РЕАЛЬНОМУ ЧАСІ. ТЕХНОЛОГІЯ WEBRTC.....</b>	<b>32</b>
<b>Андрущенко Р., Роговенко А. ИСПОЛЬЗОВАНИЕ PROCESS EXPLORER И AUTORUNS ДЛЯ БОРЬБЫ С ВРЕДОНОСНЫМ ПО.....</b>	<b>34</b>
<b>Бальченко І. РОЗРОБКА ГЕОІНФОРМАЦІЙНОЇ СИСТЕМИ ІНФРАСТРУКТУРИ АГРОФІРМИ ЗА ДОПОМОГОЮ GEOTOOLS ТА POSTGIS.....</b>	<b>36</b>
<b>Борщов І., Роговенко А. ОРГАНІЗАЦІЯ БАГАТОРОЗРЯДНИХ ОПЕРАЦІЙНИХ ПРИСТРОЇВ, ЩО ВИКОНУЮТЬ ДОДАВАННЯ ЗА ЗМІННИМ МОДУЛЕМ.....</b>	<b>39</b>
<b>Веремеєнко А., Ульченко Д. ОСОБЕННОСТИ ОПРЕДЕЛЕНИЯ МЕТЕОРОЛОГИЧЕСКИХ УСЛОВИЙ В ЗАКРЫТОМ ПОМЕЩЕНИИ С ПОМОЩЬЮ АППАРАТНОЙ ПЛАТФОРМЫ ARDUINO.....</b>	<b>41</b>
<b>Гаєнко М., Стасюк С. ВОЗМОЖНОСТИ БИБЛИОТЕКИ PTEXT ДЛЯ РАБОТЫ С PDF-ФАЙЛАМИ.....</b>	<b>43</b>
<b>Дорошенко А., Мельниченко Д. РАСПОЗНАВАНИЕ РЕЧИ (SIMON).....</b>	<b>44</b>
<b>Иванова Я., Роговенко А. ИГРОВАЯ ПЛАТФОРМА ARDUINO HACKVISION KIT.....</b>	<b>46</b>

<b>Казимир В., Нікітенко Є., Заровський Р., Єрмак Д., Камко А. ІНФОРМАЦІЙНА СИСТЕМА ЕЛЕКТРОННОГО ГОЛОСУВАННЯ MOBILE-RADA .....</b>	<b>48</b>
<b>Кононова Н. ИСПОЛЬЗОВАНИЕ ИНТЕРАКТИВНОЙ ДОСКИ В ПРОЦЕССЕ ОБУЧЕНИЯ .....</b>	<b>50</b>
<b>Коробко Д., Дрозд О. РАЗРАБОТКА ВЕБ-ПОРТАЛА ДЛЯ ИНТЕГРИРОВАННОЙ ИНФОРМАЦИОННОЙ СИСТЕМЫ УЧЕБНО-ПРОИЗВОДСТВЕННОГО ЦЕНТРА IT-АУТСОРСИНГА .....</b>	<b>53</b>
<b>Легкодух С., Скоп М. ВИКОРИСТАННЯ ФРЕЙМВОРКУ BOOTSTRAP ДЛЯ ШВИДКОГО СТВОРЕННЯ ВЕБ-САЙТІВ .....</b>	<b>55</b>
<b>Михайлюк И. ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ ДЛЯ GPS-МОНИТОРИНГА .....</b>	<b>57</b>
<b>Молофеев А., Красножон А. ИСПОЛЬЗОВАНИЕ АППАРАТНОЙ ПЛАТФОРМЫ С ОТКРЫТОЙ АРХИТЕКТУРОЙ ARDUINO ДЛЯ ПОСТРОЕНИЯ СИСТЕМ УПРАВЛЕНИЯ.....</b>	<b>59</b>
<b>Морозов А., Стасюк С. ПАРАЛЛЕЛЬНЕ ТА КОНКУРЕНТНЕ ПРОГРАМУВАННЯ В ФУНКЦІОНАЛЬНИХ МОВАХ ПРОГРАМУВАННЯ НА ПРИКЛАДІ HASKELL .....</b>	<b>61</b>
<b>Морозова О., Роговенко А. OPENWEATHERMAP – ОТКРЫТЫЕ МЕТЕОРОЛОГИЧЕСКИЕ ТЕХНОЛОГИИ .....</b>	<b>62</b>
<b>Побожая М., Ющенко Ю., Харченко М. ИСПОЛЬЗОВАНИЕ БИБЛИОТЕКИ OPENCV ДЛЯ РЕШЕНИЯ ЗАДАЧ КОМПЬЮТЕРНОГО ЗРЕНИЯ .....</b>	<b>64</b>
<b>Подкур В., Стасюк С. ПРОГРАММА ДЛЯ УПРАВЛЕНИЯ МЕДИЦИНСКОЙ ПРАКТИКОЙ OPENEMR .....</b>	<b>66</b>
<b>Подобед Ю., Стасюк С. ОБРАЗОВАТЕЛЬНЫЙ ПРОЕКТ KDEEDU .....</b>	<b>67</b>
<b>Позній В., Кривоберець С., Зацерковний В. АНАЛІЗ ТЕНДЕНЦІЙ РОЗВИТКУ ГЕОІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ.....</b>	<b>69</b>
<b>Посадська А. ВІЛЬНЕ ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ ДЛЯ МОБІЛЬНИХ ПРИСТРОЇВ НА ПРИКЛАДІ ВІЛЬНОГО ФРЕЙМВОРКУ RAPIDSMS .....</b>	<b>72</b>
<b>Тевкун М., Посадська І. ОГЛЯД ТА ХАРАКТЕРИСТИКА ВІЛЬНОГО ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ДЛЯ ЗАДАЧІ АНАЛІЗУ МЕРЕЖНОГО ТРАФІКУ.....</b>	<b>72</b>
<b>Тітов Є., Роговенко А. ОСНОВНІ МОЖЛИВОСТІ ОБРОБКИ ЗВУКОВИХ ДАНИХ ЗА ДОПОМОГОЮ АУДИОРЕДАКТОРА AUDACITY.....</b>	<b>74</b>
<b>Ткаченко С., Стасюк С. ИСПОЛЬЗОВАНИЕ GOOGLE MAPS API ПРИ РАЗРАБОТКЕ РЕАЛЬНОГО ПРИЛОЖЕНИЯ .....</b>	<b>76</b>
<b>Федоров О., Хижняк А. ПРОЕКТ СИСТЕМИ ЗАХИСТУ ВЕБ-РЕСУРСІВ ВІД DDOS-АТАК.....</b>	<b>79</b>
<b>Шаповал В., Роговенко А. ПРОЕКТИРОВАНИЕ УСТРОЙСТВ НА БАЗЕ ПЛАТФОРМЫ ARDUINO С ПОМОЩЬЮ ОТКРЫТОГО ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ.....</b>	<b>80</b>
<b>Шоломий Ю., Соломаха В. РАЗРАБОТКА ПРИЛОЖЕНИЙ ДЛЯ ANDROID В ECLIPSE IDE.....</b>	<b>82</b>

## МОДЕЛЮВАННЯ ЗМІНИ ПОКАЗНИКІВ СТАНУ ЗДОРОВ'Я ЛЮДИНИ ТА ПРОГНОЗУВАННЯ МОЖЛИВИХ НАСЛІДКІВ

С. Новачевський, В. Ванюков

*Національний технічний університет України «Київський політехнічний інститут»*

У всьому світі існує проблема низького рівня здоров'я, особливо в країнах СНД. Світова статистика смертності внаслідок захворювань серцево-судинної системи в 2012 році перевищила 42%. Україна займає перше місце в Європі за смертністю від серцево-судинних захворювань. За даними ВООЗ тільки в 2011-2012 від серцево-судинних захворювань померло 440 тисяч українців. Європейські науковці пояснюють такі високі цифри відсутністю звички відвідувати лікарів з метою профілактики, відсутністю профоглядів на підприємствах, недовірою до медичних працівників. Варто також відзначити, що сьогодні захворювання серцево-судинної системи є першопричиною смертності у всьому світі в цілому. За останні 20 років, за оцінками експертів, рівень цього захворювання виріс більше ніж на 20%.

У зв'язку з цим виникла ідея популяризації здорового способу життя за допомогою мобільних пристроїв. Було розроблено прототип застосування, за допомогою якого люди мали б змогу проводити контроль способу ведення свого життя та визначати тенденції його впливу на серцево-судинну систему шляхом постійного моніторингу значень запропонованого основного набору показників.

Для аналізу способу життя були визначені наступні, найбільш поширені, звички людини (далі – фактори), які за даними лікарів мають вплив на серцево-судинну систему людини:

- куріння;
- вживання алкоголю;
- заняття фізичними вправами;
- надлишкова вага.

За допомогою системи, що розробляється, користувач має змогу періодично вводити інформацію про кожен фактор щоденно та за період, за який вводиться ця інформація. Також система дає змогу зберігати інформацію про хвороби, які були виявлені у користувача в результаті медичної діагностики. Ця інформація в подальшому може бути використана для розрахунків тенденцій впливу факторів на серцево-судинну систему.

У даній роботі було використано інформацію, що надана експертами медичної галузі та отримана після аналізу літератури. Оцінки обраних факторів розраховуються за формулами, які було визначено на підставі цих даних з урахуванням періоду, інтенсивності та давності. Також для оцінки кожного фактору визначено вагу, яка відображає ступінь шкідливості цього фактору на серцево-судинну систему людини. У результаті було запропоновано інтегральний показник, який враховує зважені оцінки всіх факторів.

Застосування має клієнт-серверну архітектуру та виконує постійну синхронізацію з сервером, який зберігає інформацію у базі даних. Кожен інтервал часу сервер, застосовуючи метод найменших квадратів, розраховує тренд зміни інтегрального показника стану здоров'я (оцінки впливу всіх факторів на здоров'я людини) на основі отриманих даних. Користувач має змогу постійно бачити зміну тренду цього показника у мобільному застосуванні. За допомогою ідентифікації кожного користувача системою забезпечується анонімність даних.

Визначення інтегрального показника, що відображає вплив способу життя на стан здоров'я користувача, можливе без постійної актуалізації даних та за відсутності значень певних факторів, проте (у такому випадку) є можливість отримати результат, який не буде повністю відповідати дійсності. Синхронізація мобільного клієнту з сервером відбувається

за наявності активного з'єднання з мережею Інтернет, інакше (за відсутності підключення) дані зберігаються у мобільній базі даних програми.

Наразі розроблений робочий прототип знаходиться у стадії тестування, реалізуються рішення під конкретні мобільні платформи та вдосконалюється робота серверної частини.

**Список використаних джерел:** 1. *Линник Ю. В.* Метод наименьших квадратов и основы математико-статистической теории обработки наблюдений / Ю. В. Линник. – М. : Государственное издательство физико-математической литературы, 1958. – 336 с.

УДК 004.054

## ОСОБЕННОСТИ ИСПОЛЬЗОВАНИЯ ФРЕЙМВОРКА STARLING ДЛЯ СОЗДАНИЯ ДВУМЕРНЫХ ИГР

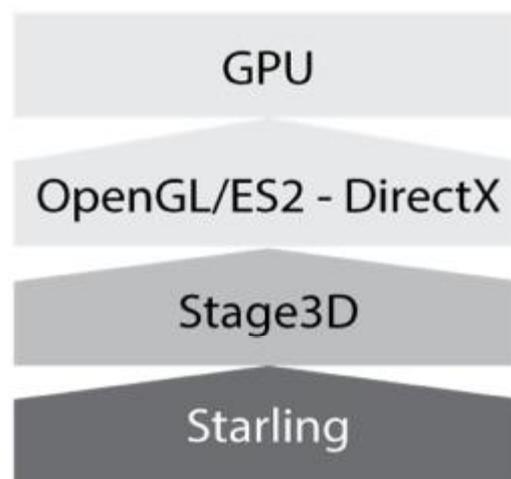
Е. Власенко, С. Нестеренко

*Черниговский национальный технологический университет*

Существует достаточно много средств, облегчающих разработку игр для различных платформ.

Однако, среди них довольно мало средств, позволяющих использовать аппаратное ускорение (речь идет не только о ПК, но и о телефонах, планшетах), и в то же время поддерживать множество различных платформ. Одним из таких средств является Starling Framework.

Starling Framework представляет из себя ActionScript 3 2D-фреймворк, разработанный на основе Stage3D API и распространяющийся бесплатно по лицензии BSD. В свою очередь, Stage3D — это новая технология, созданная компанией Adobe, которая была специально разработана для использования современного аппаратного ускорения видеокарт, работающая на основе OpenGL и DirectX на стационарных устройствах и OpenGL ES2 на мобильных устройствах. Это позволяет значительно повысить производительность графики в играх и уменьшить затраты памяти (рис.).



*Рис. Слой Starling лежит поверх Stage3D*

Starling дает возможность разработчикам использовать все преимущества GPU-ускорения и в то же время позволяет им абстрагироваться от низкоуровневого Stage3D. Так как средства написаны на ActionScript 3, они хорошо подходят Flash-разработчикам. Так же из Flash наследуется иерархическая структура объектов.

Стоит отметить, что данный фреймворк также является достаточно легковесным (80Кб кода), в нем отсутствуют какие-либо внешние зависимости от сторонних библиотек.

Свободное распространение по лицензии BSD Simplified позволяет использовать фреймворк в коммерческих целях.

**Список использованных источников:** 1. *Introducing Starling* / Thibault Imbert, Jasmine Perez – Sebastopol, CA: O'Reilly Media Inc, 2012. 2. *Adobe Flash 11 Stage3D (Molehill) Game Programming Beginner's Guide* / Christer Kaitila – Birmingham, UK: Packt Publishing Ltd., 2011

УДК 004.43

## ОБЩИЙ ОБЗОР ЯЗЫКОВ ФУНКЦИОНАЛЬНОГО ПРОГРАММИРОВАНИЯ

А. Горб

*Чернігівський інститут інформації, бізнесу і права*

Функциональное программирование - парадигма программирования, которая рассматривает программу как вычисление математических функций и избегает состояний и переменных данных. Функциональное программирование отмечает применение функций (в отличие от императивного программирования), которое подчеркивает изменения в состоянии и выполнении последовательностей команд.

Основой функционального программирования является сформулированная Джоном Мак-Карти в 1958г. концепция символьной обработки информации компьютером, которая восходит к идеям Черча, известным как лямбда-исчисление с конца 20-х годов XX века. Выбирая лямбда-исчисление как теоретическую модель, Мак-Карти предложил рассматривать функции как общее базовое понятие, к которому достаточно естественно могут быть сведены все другие понятия, возникающие при программировании.

Среди таких концепций функционального программирования можно выделить:

– функции высших порядков. Это такие функции, которые могут принимать в качестве аргументов и возвращать другие функции. Функции высших порядков позволяют использовать карринг — преобразование функции от пары аргументов в функцию, берущую свои аргументы по одному. Это преобразование получило свое название в честь Х. Карри, создателя комбинаторной логики;

– чистые функции. Чистыми называют функции, которые не имеют побочных эффектов ввода-вывода и памяти (они зависят только от своих параметров и возвращают только свой результат);

– рекурсия. В функциональных языках цикл обычно реализуется в виде рекурсии. Строго говоря, в функциональной парадигме программирования нет такого понятия, как цикл. Рекурсивные функции вызывают сами себя, позволяя операции выполняться снова и снова. Для использования рекурсии может потребоваться большой стек, но этого можно избежать в случае хвостовой рекурсии. Хвостовая рекурсия может быть распознана и оптимизирована компилятором в код, получаемый после компиляции аналогичной итерации в императивном языке программирования.

Рекурсивные функции можно обобщить с помощью функций высших порядков, используя, например, катаморфизм и анаморфизм («свертка» и «развертка»). Функции такого рода играют роль такого понятия, как цикл в императивных языках программирования.

Функциональные языки программирования делят на строгие и нестрогие. К нестрогим языкам относят те, которые поддерживают отложенные вычисления, то есть аргументы функции вычисляются только тогда, когда они действительно понадобятся при вычислении функции. Ярким примером нестрогого языка являются Haskell, F#, Miranda, Clean и т.д. В качестве примера строгого языка можно привести Standard ML.

Чистым функциональным языком можно назвать Haskell. Однако большинство функциональных языков являются гибридными и содержат свойства как функциональных, так и императивных языков. Яркие примеры — языки Scala и Nemerle. В них органично сочетаются характеристики как объектно-ориентированных языков, так и функциональных. Реализована хвостовая рекурсия и её оптимизация. Функция является полноправным объектом, то есть может быть сохранена в переменной, передана в качестве аргумента в другую функцию или возвращена из функции.

Функциональное программирование имеет ряд преимуществ и недостатков.

Среди преимуществ часто выделяют:

– повышение надёжности кода. Повышение надёжности кода происходит за счёт чёткой структуризации и отсутствия необходимости отслеживания побочных эффектов. Любая функция работает только с локальными данными и работает с ними всегда одинаково, независимо от того, как и при каких обстоятельствах она вызывается. Также невозможность мутации данных при пользовании ими в разных местах программы исключает появление труднообнаруживаемых ошибок (таких, например, как случайное присваивание неверного значения глобальной переменной в императивной программе);

– удобство организации модульного тестирования. Поскольку функция в функциональном программировании не может порождать побочные эффекты, менять объекты нельзя как внутри области видимости, так и снаружи (в отличие от императивных программ, где одна функция может установить какую-нибудь внешнюю переменную, считываемую второй функцией). Единственным эффектом от вычисления функции является возвращаемый ей результат, и единственный фактор, оказывающий влияние на результат – это значения аргументов;

– возможности оптимизации при компиляции. Традиционно упоминаемой положительной особенностью функционального программирования является то, что оно позволяет описывать программу в так называемом «декларативном» виде, когда жесткая последовательность выполнения многих операций, необходимых для вычисления результата, в явном виде не задаётся, а формируется автоматически в процессе вычисления функций. Это обстоятельство, а также отсутствие состояний даёт возможность применять к функциональным программам достаточно сложные методы автоматической оптимизации.

– возможности параллелизма. Ещё одним преимуществом функциональных программ является то, что они предоставляют широчайшие возможности для автоматического распараллеливания вычислений. Поскольку отсутствие побочных эффектов гарантировано, в любом вызове функции всегда допустимо параллельное вычисление двух различных параметров – порядок их вычисления не может оказать влияния на результат вызова.

Недостатки функционального программирования вытекают из его особенностей. Отсутствие присваиваний и замена их на порождение новых данных приводят к необходимости постоянного выделения и автоматического освобождения памяти, поэтому в системе исполнения функциональной программы обязательным компонентом становится высокоэффективный сборщик мусора. Нестрогая модель вычислений приводит к непредсказуемому порядку вызова функций, что создает проблемы при вводе-выводе, где порядок выполнения операций важен. Кроме того, очевидно, что функции ввода в своем естественном виде не являются чистыми, поскольку способны возвращать различные значения для одних и тех же аргументов.

**Список использованных источников:** 1. Душкин Р. Функции и функциональный подход / Р. Душкин // Практика функционального программирования. – 2009. – № 1. – С. 17-31. 2. Харрисон Д. Введение в функциональное программирование / Д. Харрисон. – Кембридж, 1997. – 170 с. 3. Функциональный подход к программированию / Учебный Центр безопасности информационных технологий Microsoft Московского инженерно-физического института // Современные языки программирования и NET. – 2003. – № 3.

4. Язык функционального программирования [Электронный ресурс]. – Режим доступа : [ru.wikipedia.org/wiki/](http://ru.wikipedia.org/wiki/).

УДК 37.018.43:004.58

## СИСТЕМИ ВЕБІНАРІВ З ВІДКРИТИМ КОДОМ У ДИСТАНЦІЙНОМУ НАВЧАННІ

О. Дрозд

*Чернігівський національний технологічний університет*

Значну частину інформації людина отримує і передає через спілкування. При цьому, на думку психологів важливу, роль відіграють жести, міміка, інтонації, що дозволяє значно ефективніше сприймати інформацію. Тож не є дивним прагнення людей перенести свій світ спілкування майже у повному обсязі у віртуальний простір.

Дистанційне навчання сьогодні – це вже не просто збірки матеріалів, які треба прочитати, прослухати, вивчити та виконати завдання. Все більше уваги приділяється живому спілкуванню всіх учасників у онлайн-режимі через Інтернет. Можна виділити два основні напрямки побудови зв'язку:

- із залученням спеціальних програмних і апаратних засобів. Такий спосіб використовується у системах відеоконференцзв'язку та телеприсутності і відрізняється високою якістю зв'язку і малими затримками сигналів;

- із використанням стандартного комп'ютерного обладнання і веб-технологій.

Такий спосіб використовують системи відеозв'язку (Skype, ooVoo) та вебінари.

Системи вебінарів активно використовуються у системах дистанційного навчання та соціальних мережах. Традиційними формами використання вебінарів є:

- спілкування і наради, де кожен з учасників бачить і чує один одного, має можливість переглядати спільні документи, проводити обмін документами, бачити віддалений робочий стіл. Часто це некерований режим (без модерації), який потребує більшої інтенсивності використання ресурсів мережі, серверів і робочих станцій;

- лекції, семінари, конференції. У такому режимі кількість учасників може сягати кількох сотень, тому з метою більш ефективного використання ресурсів вебінару усі учасники умовно діляться на три категорії: модератор, який керує вебінаром; лектор, якого усі слухають, бачать і переглядають його документи; слухачі, які сприймають інформацію, можуть вести обмін текстовими повідомленнями і просити слова у модератора. Сам лектор може одночасно бути і модератором.

Користування готовими сервісами вебінарів або якимись їх додатковими можливостями, наприклад, багатоточковою конференцією, досить часто є платними послугами. Популярними рішеннями для організації вебінарів сьогодні називають Webinar.ru, WebEx, GoToMeeting, OnWebinar та інші [1].

Організація або навчальний заклад можуть розгорнути власний сервіс вебінарів на основі систем з відкритим кодом. Найбільш популярними рішеннями у цьому плані є OpenMeetings [2] та BigBlueButton [3], які сьогодні активно підтримуються та розвиваються.

Система OpenMeetings побудована на технологіях Java і XML, використовує медіа сервер з відкритим кодом Red5, протокол RTMP (Real Time Messaging Protocol), сервер документації OpenOffice.org. В якості баз даних можуть використовуватися MySQL, PostgreSQL, Microsoft SQL Server, Oracle, DB2. Передача контенту (аудіо-, відеодані) оснований на технологіях Flash. Передбачений імпорт документів різних форматів (рисунок, текстові документи, електронні таблиці, презентації) та їх конвертація для сумісного перегляду [4].

Важливою характеристикою OpenMeetings є інтеграція з багатьма технологіями: SOAP/REST API, аутентифікація LDAP і MS Active Directory, VoIP на базі сервера Asterisk

та інші. Особливо слід відзначити наявність плагінів для взаємодії з системами керування навчанням (LMS), керування контентом (CMS), керування проектами: Moodle, Sakai, Jira, Joomla, Drupal, Bitrix, Confluence, SugarCRM, Redmine [2]. Наприклад, плагін для Moodle додає у навчальний курс новий ресурс OpenMeetings. При активації цього ресурсу користувач автоматично переходить у кімнату з вебінаром, де може спілкуватись з викладачем та іншими користувачами. Авторизація у цьому випадку здійснюється автоматично.

OpenMeetings може встановлюватися на серверах з операційними системами Linux та Windows. Рекомендовані характеристики сервера: 2x/4x 2 ГГц (32/64 біт) і 4 Гб RAM. Для якісного зв'язку перепускна здатність каналу рекомендується розраховувати, виходячи із 256 Кбіт/с на 1 з'єднання, хоча для передачі тільки аудіо достатньо 64 Кбіт/с.

Вебінари OpenMeetings проводяться у окремих віртуальних кімнатах. Існує три види кімнат: загальні (публічні), групові (для окремо визначеної групи учасників), персональні (створені окремим користувачем). У налаштуваннях кімнат можна визначити режим спілкування (аудіо, відео), можливість модераторів, замовити наявність різних видів ресурсів: віртуальну дошку (Whiteboard), панель файлів, чат, використання VoIP та інше. Таким чином кімнати можуть мати свою індивідуальну спеціалізацію.

Кожен користувач має свій календар, де може планувати вебінари з автоматичною відправкою запрошень іншим учасникам, а також отримувати нагадування про ті події, на які він сам запрошений. Вебінари можуть використовуватися не тільки для спілкування. Функції трансляції робочого столу а також керування віддаленим робочим столом передбачають більш тісну взаємодію учасників. Також в OpenMeetings реалізоване зручне сховище для файлів: окремі набори файлів і папок можуть бути у кожного користувача і спільні в кожній кімнаті.

BigBlueButton є відносно новим проектом, який розпочався у 2007 році, але вже отримав широке визнання. Технологічно він досить схожий з OpenMeetings, проте орієнтований більше на спрощене і ефективне використання. Менші функціональні можливості з організації кімнат, модерування, керування файлами компенсуються значно простішим і інтуїтивно зрозумілим інтерфейсом, що є досить важливим чинником для цієї категорії учасників, у яких немає часу, бажання та підготовки з організації потужних вебінарів, але треба швидко і зручно організувати спілкування [5]. BigBlueButton теж має потужні інтеграційні можливості з різними системами керування та дистанційного навчання, підтримку VoIP на базі серверів Asterisk та FreeSWITCH.

У якості висновку можна спробувати дати відповідь на питання, яку систему (OpenMeetings чи BigBlueButton) краще використовувати у дистанційному навчанні. За технологічними рішеннями, вимогами до обладнання та каналів передачі системи досить схожі. Однозначної порівняльної оцінки параметрів тримання навантаження та якості зв'язку немає, існують тільки окремі стрес-тести та відгуки адміністраторів, на основі яких важко віддати перевагу тій чи іншій системі. Тому на даному етапі розвитку цих систем відповідь на поставлене питання краще шукати у площині тих завдань дистанційного навчання, які повинні бути розв'язані за допомогою вебінарів. Для випадку організації спілкування та навчання в умовах простої модераторів та залученням малопідготовлених учасників більш ефективним було би використання BigBlueButton. У разі, коли вебінари повинні бути не періодичною, а повноцінною компонентою навчального процесу, наслідувати взаємодію викладача та студентів (учнів) із реального життя з усією багатогранністю цього процесу, досить вагомий функціональний набір OpenMeetings стає більш доречним.

**Список використаних джерел:** 1. *Топ 10* рішень для проведення вебінарів [Електронний ресурс]. – Режим доступу : <http://www.webmeetings.ru/news/10786/>. 2. *Офіційний* сайт проекту OpenMeetings [Електронний ресурс]. – Режим доступу : <http://openmeetings.apache.org/>. 3. *Офіційний* сайт проекту BigBlueButton [Електронний

ресурс]. – Режим доступа : <http://bigbluebutton.org/>. 4. Яремчук С. Система видеоконференций OpenMeetings [Электронный ресурс] / Сергей Яремчук // Системный администратор. – 2009. – Вып. № 6 (79). – Режим доступа : <http://samag.ru/archive/article/2034>. 5. Яремчук С. Проект BigBlueButton. Платформа для видеоконференций и дистанционного обучения [Электронный ресурс] / Сергей Яремчук // Системный администратор. – 2012. – Вып. № 5 (114). – Режим доступа : <http://samag.ru/archive/article/2203>.

УДК 339.138:004.7

## ТЕНДЕНЦИИ РАЗВИТИЯ ПОИСКОВЫХ СИСТЕМ

О. Жибер

Поисковые системы предоставляют бесплатный трафик (органическая выдача) для покупателей, что напрямую способствует развитию бизнеса.

В период с 2011 по 2013 год практически полностью изменились принципы ранжирования сайтов в поисковой выдаче.

### **Краткая история:**

**1945 год** – идея создания гипертекста и дальнейшие рассуждения о системе, которая бы систематизировала эти документы (аналог поисковых машин), принадлежит американскому инженеру Ванневару Бушу.

**1960-е** годы – Джерард Сэлтон и его группа в Корнелльском университете разработали «Остроумную систему извлечения информации» (SMART information retrieval system). SMART – аббревиатура от Salton's Magic Automatic Retriever of Text, что переводится как «Волшебный автоматический извлекатель текста Сэлтона». Джерард Сэлтон считается отцом современной поисковой технологии.

**1987-1989** годы – была разработана система Archie, которая индексировала FTP архивы. Названия файлов собирались в документ, по которому и происходил поиск.

**1993 год** – на основе бота World Wide Web Wanderer, автором которого был Метью Греем из Массачусетского технологического института, была создана первая в мире поисковая система для Всемирной сети Wandex; Мартин Костер создаёт Aliweb – одну из первых поисковых систем по WWW. Что бы система искала по Вашим сайтам, их следовало добавлять вручную, и поскольку многие этого не знали, то данная поисковая система не добилась популярности.

**20 апреля 1994 года** – выпущен первый бот WebCrawler, автором которого был Брайан Пинкертон из университета Вашингтон, который полностью индексировал страницы. Так же он предлагал пользователям возможность поиска по любым ключевым словам на любых страницах.

**20 июля 1994 года** – открылась поисковая система Lycos. Изначально открывался с каталогом документов в количестве 54000, который ранжировался, учитывал приблизительные совпадения, окончания, словоформы, что позволило ему к ноябрю 1996 года проиндексировать более 60 млн документов.

**1995 год** – запустилась AltaVista. Благодаря практически неограниченной пропускной способности, возможности формулировать сложные запросы на естественном языке быстро сделали эту поисковую систему очень популярной. Также разработчики AltaVista первыми начали работать с японским, корейским, китайским языками. А в 1996 году – поисковая машина AltaVista запустила морфологическое расширение для русского языка.

**1996 год** запущены первые отечественные поисковые системы – Rambler.ru и Aport.ru, что ознаменовало новый этап развития Рунета.

**1997 год** – переломный момент. Он наступает, когда С. Брин и Л. Пейдж из Стэнфордского университета основали поисковую систему Google, которая выполняла и

при поиске учитывала морфологию, ошибки и опечатки. Результаты, которые получали пользователи, были еще более релевантны запросу.

**23 сентября 1997 года** – анонсирован Yandex, который быстро стал самой популярной у русскоязычных пользователей Интернета системой поиска. Началась конкурентная борьба с Rambler-ом, Aport-ом, в результате которой поисковые системы улучшали свои системы индексации, поиска и выдачи.

#### **ПС Яндекс.**

Яндекс на данный момент является основным поисковиком в России (порядка 70% поискового трафика) и конкурентом для Google в Украине (около 30% поискового трафика). Особенности работы:

Индексация: Создаваемый индекс составляет около 1/3 объема текста (без картинок, tag'ов и пр.), при этом записывается подробный адрес слова — с точностью до позиции в тексте, что потом позволяет искать с учетом близости. В базу входят нормализованные формы слов текста, сами документы не хранятся. Морфологический анализ слов текста происходит одновременно с индексацией, что дает возможность снятия омонимии.

Поиск: Богатый язык запросов поддерживает все логические операции в пределах документа или предложения, позволяет учитывать расстояние между словами, поиск в заданных зонах или с учетом заданных атрибутов документа. Список выданных документов может быть упорядочен по релевантности, дате или другим документным атрибутам. В каждом найденном документе могут быть выделены (подсвечены) найденные слова.

#### **Алгоритмы Google**

Крупнейший поисковик Google сейчас способен найти документы на 191 языке и эффективно обработать информацию основных форматов (Word, PDF, Exel и пр.). Чтобы поисковая выдача была максимально качественной, каждый год Google вносит в свои алгоритмы поиска более полутысячи обновлений. Львиная доля этих изменений достаточно незначительна. Но где-то 5-6 раз в год запускаются нововведения, существенно меняющие выдачу.

**Список использованных источников:** 1. *Режим* доступа : <http://shpargalkablog.ru/2011/02/algorithmy-yandexa.html>. 2. *Режим* доступа : <http://company.yandex.ru/technologies/matrixnet/>. 3. *Режим* доступа : <http://help.yandex.ru/webmaster/>. 4. *Режим* доступа : <http://support.google.com/webmasters/bin/answer.py?hl=ru&answer=35769>. 5. *Режим* доступа : <http://prozhector.ru/publications/vypusk-28-ot-18-aprelya-2013-goda-kak-prodvinut-na-rynke-novyy-brend/tendentsii-razvitiya-poiskovykh-sistem/>. 6. *Режим* доступа : <http://seoeducation.com.ua/seo/algorithms.html>. 7. *Режим* доступа : <http://devaka.ru/articles/algo-2012>.

УДК 004.421

## **МОДЕЛИРОВАНИЕ И ВИЗУАЛИЗАЦИЯ ЖИДКОСТЕЙ, ОСНОВАННАЯ НА ДИНАМИКЕ ЧАСТИЦ**

Л. Запривода, С. Нестеренко

*Черниговский национальный технологический университет*

Симуляция потоков, основанная на гидродинамике сглаженных частиц (ГСЧ smoothed particle hydrodynamics – SPH) – применимый в практике метод представления жидкостей в интерактивных приложениях, таких как виртуальные хирургические тренажеры или компьютерные игры. В последние годы множество различных изданий описывали идеи как ГСЧ, так и ее визуализации. В статье попробуем подробнее

рассмотреть компьютерную реализацию симуляций, а также основные методы визуализации, позволяющие рендеринг множественных преломлений в реальном времени.

Такие нестабильные среды как газы и жидкости – неотъемлемая часть окружающей нас обстановки. Например, процесс наливания молока в стакан. К сожалению, в компьютерной графике реального времени, когда мы традиционно стараемся воспроизвести составляющие нашего мира наиболее реалистично, симулировать такое явление довольно сложно. Расчеты жидкостной динамики – относительно изученная тема, но до сих пор приложения (например, воспроизводящие аэродинамику) требуют улучшения результатов вычислительной скорости и точности воспроизведения. Поэтому симуляции (в большинстве случаев) рассчитываются за кадром, а визуализация в реальном времени если и есть – то используется только для отображения заранее рассчитанных массивов данных.

### ***Математические и физические модели жидкостей.***

В XIX в. Клод Навье и Джордж Стокс создали основы современной динамики неустойчивых сред, сформулировав широко известные уравнения Навье-Стокса [1; 9].

С помощью этих уравнений, описывающих сохранение импульса, а также двух дополнительных уравнений для сохранения массы и энергии можно описать поток жидкости. Для редких сред эти уравнения довольно сложные, но, в основном, они подходят для ньютоновских сред, включая большинство распространенных в природе жидкостей и газов (вода, воздух, и т. д.).

Основные свойства, характеризующие движение жидкостей – масса, плотность, давление и скорость. Однако все жидкости при динамике подчиняются закону Ньютона о сохранении импульса:

$$\mathbf{F} = m\mathbf{a} \quad (1)$$

Раскладывая силы на основные составляющие компоненты, влияющие на динамику жидкостей (давление, плотность, вязкость, скорость), Навье и Стокс составили основные уравнения динамики жидкости. В частном случае, без учета второстепенных составляющих, для несжимаемых неустойчивых сред это уравнение выглядит так:

$$\rho \left( \frac{d\mathbf{v}}{dt} + \mathbf{v} \cdot \nabla \mathbf{v} \right) = -\nabla p + \eta \nabla \cdot \nabla \mathbf{v} + \rho \mathbf{g} \quad (2)$$

Это уравнение само по себе является основой любой гидродинамической симуляции.

В 2003 г. Мюллер, Карипэр и Гросс еще больше разожгли интерес к симуляции жидкостей в реальном времени, предложив тезис, описывающий сравнительно простую модель жидкости, основанную на динамике частиц [5].

Гидродинамика сглаженных частиц – техника, разработанная Джингольдом и Монаганом [4; 6], и (независимо от них – Люси) для симуляции астрофизических газодинамических задач. Как и в других числовых методах решения задач динамики неустойчивых сред, значение физического качества в заданной точке  $A(\mathbf{r})$  должно быть интерполировано из дискретного набора точек. ГСЧ исходит из интегральной интерполяции:

$$A_I(\mathbf{r}) = \int A(\mathbf{r}') W(\mathbf{r} - \mathbf{r}', h) d\mathbf{r}' \quad (3)$$

$W(\mathbf{r}, h)$  – радиальная симметричная сглаживающая функция (также называемая ядром) с длиной сглаживания  $h$  (также называемой радиусом ядра). Можно сказать, что интерполяция использует ядро сглаживания, чтобы распределить количество из данной позиции (точки) по ее окружению.

Применяя математическую технику сглаживающих ядер к уравнению Навье-Стокса, можно сформировать математическую модель основанной на частицах симуляции жидкости, которая была бы достаточно проста для использования ее в реальном времени.

В математической модели каждая частица представляет собой маленькую долю жидкости. Частицы несут свойства массы (которая является константой и в этом случае одинакова для всех частиц), положения и скорости. Из этого можно вычислить все остальные свойства, используя ГСЧ и основные физические уравнения [6; 7].

«Алгоритм обновления» движения сглаженных частиц при расчете действия на них основных сил будет выглядеть следующим образом (очень упрощенное описание):

1. рассчитать плотность в каждой точке положения частиц;
2. рассчитать силы давления, вязкости и цветовое поле (поверхностное натяжение) для каждой частицы;
3. передвинуть частицы с временным промежутком и очистить относящиеся к ним поля;
4. обновить расчет ускорения.

Что касается реализации, частицы (в большинстве случаев) несут с собой только свойства положения и скорости (масса является константой и одна для всех частиц). Это единственная информация, которая меняется от шага к шагу симуляции. Другие данные для каждой частицы (такие, как плотность и силы, на нее действующие) хранятся обычно в отдельных массивах.

Для визуализации частиц можно использовать три техники (рис.): первая напрямую отрисовывает частицы в виде точечных спрайтов, что, в основном, полезно для отладки и настройки их поведения [3]. Вторая, практически полностью основанная на процессорных мощностях, использует алгоритм многоступенчатых кубов для создания треугольной сетки, представляющей собой изоповерхность [8]. Далее эта техника использована для визуализации в качестве теста эффективности многоступенчатых квадратов и треугольной геометрии для представления жидкостей. Последняя и наиболее сложная техника – использование мощностей GPU (графический процессор) [10] для создания участка с определенной пространственной плотностью, имеющего трехмерную текстуру и отрисовка изоповерхности напрямую с использованием метода бегущего луча. Отрисовка методом бегущего луча позволяет визуализацию таких присущих жидкостям эффектов, как множественные преломления и отражения.

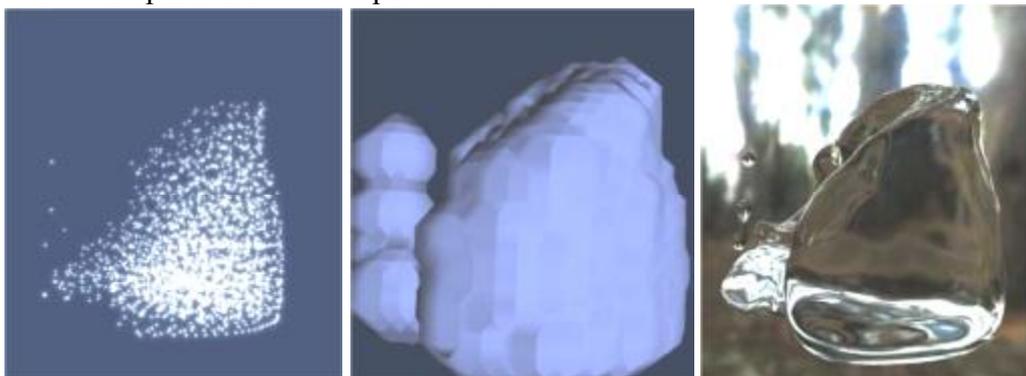


Рис. Примеры трех техник визуализации частиц

**Список использованных источников:** 1. Давыдова М. А. Лекции по гидродинамике / М. А. Давыдова. – М. : ФИЗМАТЛИТ, 2011. – 216 с. 2. Adams B., et al. Adaptively Sampled Particle Fluids // Proceedings of the 2007 SIGGRAPH conference. – 2007. 3. Amada T., et al. Particle-Based Fluid Simulation on GPU // ACM Workshop on General-Purpose Computing on Graphics Processors. – 2004. 4. Gingold and Monaghan. Smoothed particle hydrodynamics: theory and application to non-spherical stars. Royal Astronomical Society, Monthly Notices. 181p. – 1977. 5. Müller M., Charypar D. and Gross M. Particle-Based Fluid Simulation for Interactive Applications // Proceedings of 2003 ACM SIGGRAPH Symposium on Computer Animation. – 2003. 6. Monaghan J.J. Smoothed particle hydrodynamics. Reports on Progress in Physics. – 2005. 7. Müller M., et al. Interaction of Fluids with Deformable Solids. Computer Animation and Virtual Worlds. – 2004, 15. 8. Müller M., Schirm S. and Duthaler S. 2007. Screen Space Meshes. Eurographics/ ACM SIGGRAPH Symposium on Computer Animation. – 2007. 9. wikipedia.org. Navier-Stokes equations/Derivation. [Online] [Цитировано:03.03.2013.]. Режим доступа : [http://en.wikipedia.org/w/index.php?title=Navier-Stokes\\_equations/Derivation&oldid=177609104](http://en.wikipedia.org/w/index.php?title=Navier-Stokes_equations/Derivation&oldid=177609104). 10. NVIDIA Corporation. Environment Mapping Techniques. [Online]. Режим доступа : [http://http.developer.nvidia.com/CgTutorial/cg\\_tutorial\\_chapter07.html](http://http.developer.nvidia.com/CgTutorial/cg_tutorial_chapter07.html).

## **ІНФОРМАЦІЙНА КОМП'ЮТЕРНА СИСТЕМА КОНТРОЛЮ ТА УПРАВЛІННЯ ДОСТУПОМ ПАТ ФК «АОА ІНВЕСТМЕНТС»**

Т. Коляда, С. Стасюк

*Чернігівський національний технологічний університет*

Система контролю та управління доступом (СКУД) являє собою налагоджений програмно-апаратний комплекс, призначений для візуального контролю, запису інформації та передачі її до місця перегляду або зберігання, що дозволяє запобігти несанкційованому доступу до охоронюваного об'єкту ПАТ ФК «АОА ІНВЕСТМЕНТС», а також дозволяє розмежувати доступ до приміщень, зберегти і потім переглянути інформацію про події в системі за певний проміжок часу, вести моніторинг персоналу та облік робочого часу для персоналу.

Метою розробки є проектування інформаційно-комп'ютерної системи, що дозволяє контролювати спроби порушення роботи об'єкта, що охороняється, і спроби несанкційованого заволодіння інформацією та матеріальними цінностями шляхом установки обладнання біометричної ідентифікації на об'єкті.

Предметом – створення програмного продукту для реалізації системи контролю та управління доступом на основі біометричної ідентифікації.

Традиційні методи персональної ідентифікації, засновані на застосуванні паролів або матеріальних носіїв, таких, як пропуск, паспорт, водійське посвідчення, електронний ключ або карта, не завжди відповідають сучасним вимогам безпеки. Пароль можна забути або перехопити, матеріальний носій – скопіювати, втратити або передати іншій особі. Відомо, що людина володіє рядом унікальних ознак, властивих виключно їй – зовнішність, голос, відбитки пальців та ін. Людина сама по собі є "ходячим ідентифікатором".

Рішенням проблеми точної ідентифікації особистості може бути застосування біометричних систем ідентифікації. Зараз ідентифікація особистості – це необхідність, зумовлена широким застосуванням інформаційних технологій у розвинених країнах. Тому біометричний компонент все частіше використовується як елемент сучасних систем контролю доступу, що висувають підвищені вимоги до безпеки.

Біометричні системи і зчитувачі для СКУД є одними з найбільш складних, оскільки призначенням будь-якої СКУД, у тому числі і біометричної, є ідентифікація (аутентифікація) користувачів системи, побудованої на основі зчитування біометричних ознак. Фактично будь-яка біометрична СКУД виробляє звернення задалегідь занесеного в пам'ять системи ідентифікатора з заново введеною біометричною ознакою. Біометричні зчитувачі є сучасними високоточними пристроями ідентифікації і використовуються, в основному, на об'єктах з підвищеними вимогами секретності. Біометрична ідентифікація є однією з найбільш надійних, оскільки людина ідентифікується за ознаками, які неможливо передати іншій особі, на відміну від зовнішніх ідентифікаторів (електронних карт, міток, ключів та ін.)

У рамках поставленого завдання передбачається розробка інформаційно-комп'ютерної системи контролю та управління доступом до охоронюваного об'єкта фірми, яка працює з цінними паперами. Інформація про цінні папери, реквізити клієнтів, котирування векселів, паперів та іншої корисної інформації повинна бути надійно захищена, тому потрібно обмежити можливість проникнення сторонніх осіб і забезпечити контроль за персоналом.

В апаратній підсистемі використовуються біометричні зчитувачі відбитків пальця, управляючі елементи системи – контролери, інтерфейси передачі даних Ethernet та RS-485 і перетворювачі інтерфейсів та периферійні пристрої — відеоспостереження, кнопки виходу, датчики відкриття дверей та інші.

Програмна підсистема складається з таких компонентів:

- веб-сервер Apache Tomcat ( Ліцензія Apache License 2.0);
- фреймворк Vaadin ( Ліцензія Apache License 2.0);
- бібліотеки OpenCV ( Ліцензія BSD);
- сервер баз даних PostgreSQL 9.1 ( Ліцензія BSD);
- середовище розробки Eclipse Eclipse ( Ліцензія Public License);
- операційної системи Ubuntu 13.04 LTS.

Таким чином більшість існуючих систем спостереження створюється на базі розподіленої клієнт-серверної архітектури, яка дозволяє здійснювати контроль на територіально розподілених об'єктах різного масштабу, підтримує різні типи зчитувачів, камер і різних інших пристроїв, володіє підсистемою біометричної ідентифікації, віддаленого доступу до баз даних та захисту даних. Крім того, зазначені системи мають універсальність і масштабованість, у результаті чого система працює як єдине ціле, легко нарощується і підтримує необмежену кількість пристроїв.

Робота виконана методом аналізу прототипів і подальшою розробкою ІКС, що відповідає технічному завданню. Робота може бути використана для розробки більш складних мережевих СКУД, заснованих на декількох видах біометричних зчитувачів. Також розроблена система може бути використана як частина інтегрованої системи безпеки, яка виконує як охоронні функції (охоронна і пожежна сигналізація, управління системою відеоспостереження), так і управління системами життєзабезпечення (електропостачання, освітлення, ліфти та ескалатори, система вентиляції та кондиціонування тощо).

**Список використаних джерел:** 1. *Autosky* – Биокod – Биометрические технологии [Электронный ресурс]. – Режим доступа: URL: <http://autosky.com.ua/content/view/28/51/>. 2. *Биометрическая* идентификация и аутентификация [Электронный ресурс]. – Режим доступа: <http://www.bio-smart.ru/>. 3. *Интегрированные* системы безопасности [Электронный ресурс]. – Режим доступа : <http://datalink.ua/services/sistemu-kontrolya-i-upravlenia-dostupom/>.

УДК 004.738.5:519.682:519.872

## КОНТЕЙНЕРЫ СЕРВЛЕТОВ И СЕРВЕРА ПРИЛОЖЕНИЙ

Д. Коробко

*Черниговский национальный технологический университет*

Сервер приложений – это сервисная программа, которая обеспечивает доступ клиентов к прикладным программам, выполняющимся на сервере.

Клиенты могут взаимодействовать с приложениями через API сервера (Java-клиент <—> контейнер сервлетов <—> сервлет). Большую гибкость и универсальность представляет взаимодействие через сторонние сервисы, в первую очередь, – через веб-сервер.

Понятие сервера приложений традиционно связывают с платформой Java, указывая на то, что сервер Java-приложений представляет реализацию спецификации сервлетов, возможно в виде JSP, и еще некоторые сервисные услуги, в первую очередь – соединение с базой данных.

Но это нечто большее и меньшее одновременно: сервер приложений предоставляет среду, в которой прикладные программы могут работать, независимо от того, что и как именно они делают.

JBoss Application Server (JBoss AS) – Java EE сервер приложений с открытым исходным кодом, разработанный одноимённой компанией. Как и многие открытые

программы, разрабатываемые коммерческими организациями, JBoss Application Server можно свободно загрузить и использовать, однако поддержка и консультации осуществляются за деньги. Достаточно хорошая реализация принципов Java EE делает JBoss Application Server конкурентом для аналогичных проприетарных программных решений, таких как WebSphere или WebLogic.

В качестве контейнера сервлетов JBoss использует Tomcat.

Контейнер сервлетов – один из архитектурных компонентов J2EE, представляющий окружение для выполнения сервлетов. Сервлет — это Java-приложение, выполняющееся на стороне сервера (в отличие от апплета). Контейнер сервлетов может работать как полноценный самостоятельный сервер, но чаще используется (интегрируется) совместно с другим серверным ПО; обеспечивая обмен данными между сервлетом и клиентами, берёт на себя выполнение таких функций, как создание программной среды для функционирующего сервлета, идентификацию и авторизацию клиентов, организацию сессии для каждого из них.

Концепция сервлет-контейнера позволяет создавать как универсальные, так и специализированные серверы приложений (например, для мобильных сервисов).

Примером реализации контейнера сервлетов является Apache Tomcat, который используется в таких серверах приложений, как Apache Geronimo, JBoss, GlassFish, IBM WebSphere Application Server (WAS).

Tomcat (в старых версиях – Catalina) – контейнер сервлетов с открытым исходным кодом, разрабатываемый Apache Software Foundation. Реализует спецификацию сервлетов и спецификацию JavaServer Pages (JSP). Написан на языке Java.

Tomcat позволяет запускать веб-приложения, содержит ряд программ для самоконфигурирования.

Tomcat используется в качестве самостоятельного веб-сервера, в качестве сервера контента в сочетании с веб-сервером Apache HTTP Server, а также в качестве контейнера сервлетов в серверах приложений JBoss и GlassFish.

Разработка и поддержка Tomcat осуществляется фондом Apache Software Foundation и добровольцами. Пользователи имеют свободный доступ к исходным кодам и бинарным файлам Tomcat согласно лицензии Apache License 2.0.

Исторически доминирующий на рынке Tomcat ныне уступает свои позиции. А доля Jetty при этом неуклонно растёт.

Tomcat был эталонной реализацией спецификации сервлетов только до версии 2.4. Начиная с версии 2.5 в качестве такового Sun работает над GlassFish. Он же стал эталонной реализацией для сервлетов версии 3.0 и для JSP, начиная с версии 2.1.

Jetty уделяет немалое внимание спецификациям и добросовестно реализует их. Разработчики Jetty участвовали в создании версий 2.5 и 3.0. Таким образом Jetty не только соответствует всем стандартам, но и активно влияет на них.

Кризис управления и разработки для Tomcat вылились в конечном итоге в:

- потерю статуса эталонной реализации;
- слиянию с проектом GlassFish;
- переходу разработки из режима "Commit then Review" в режим "Review then Commit".

Проект Jetty имеет богатый опыт реагирования на изменяющиеся требования и стандарты. Это привело к платформе которая может быть встроена и развернута практически в любой среде, и которая поддерживается значительным, здоровым и растущим сообществом.

**Список использованных источников:** 1. Учебные материалы для студентов [Электронный ресурс]. – Режим доступа : <http://www.4stud.info/networking/application-server.html>. 2. Блог о шифровании [Электронный ресурс]. – Режим доступа : <http://crypto.pp.ua/2010/06/zapusk-kontejnera-servletov-i-razmeshhenie-proekta/>. 3. Блог веб-

розробчика [Електронний ресурс]. – Режим доступу : <http://jenya-p.ru/home/1-blog/2-jettyvstomcat.html>. 4. *Национальный* центр поддержки и разработки [Електронний ресурс]. – Режим доступу : <http://www.msvsphere.com/products/jboss>. 5. *Новинки* Java-продукта JBoss [Електронний ресурс]. – Режим доступу : [http://www.avim.ru/news/Novinki\\_Java\\_produkta\\_Jboss](http://www.avim.ru/news/Novinki_Java_produkta_Jboss). 6. *JMX*. Мониторинг и управление java-приложениями [Електронний ресурс]. – Режим доступу : <http://www.nestor.minsk.by/kg/2005/34/kg53401.html>. 7. *Форумы* по программированию Vingrad, Фриланс [Електронний ресурс]. – Режим доступу : <http://www.base.vingrad.ru/view/2859-JNDI>. 8. *Инструментирование* в Java. Теория и примеры [Електронний ресурс]. – Режим доступу : <https://docs.google.com/document/d/18rQSVQ9vn5Ntza3MCXRxjDGWvzwa1gV1x3dRXrsKjg/edit#>.

УДК 004.451

## ЛІЦЕНЗІЙНІ АСПЕКТИ ЗАХИСТУ ПРОГРАМНИХ СИСТЕМ НА ОСНОВІ ЯДРА LINUX ВІД НЕСАНКЦІОНОВАНОГО ВТРУЧАННЯ В ЇХ РОБОТУ

Д. Мельниченко

*Чернігівський національний технологічний університет*

Загальновідомо, що ядро Linux являє собою безкоштовне програмне забезпечення з відкритим вихідним текстом. Незважаючи на це існує безліч платних програмних систем, що базуються на безкоштовному ядрі Linux. Не секрет також, що для того, аби захист прикладного програмного забезпечення (ПЗ) від клонування або втручання в його роботу був ефективним, його бажано підтримати апаратними засобами і задіяти якомога раніше в циклі нормальної роботи ПЗ. Це означає, що деякі функції захисту повинні бути реалізовані на рівні ядра ОС, яке (у випадку з Linux) являє собою програму з відкритим вихідним текстом. Наявність відкритого вихідного тексту ядра і відкритого тексту модулів захисту спрощує пошук уразливих місць у цьому захисті.

Як вдіяти, якщо архітектура конкретної програмної системи, що базується на відкритому ядрі Linux, передбачає виконання певних закритих функцій саме на рівні ядра? Чи існує легальна можливість в переданих користувачам системах не відкривати вихідний текст таких фрагментів ядра? У результаті аналізу предметної області розглянутого питання було з'ясовано таке:

1. Ядро Linux поширюється за ліцензією GPLv2.

2. Модуль ядра може не використовувати ліцензію GPL. Чи може модуль ядра Linux бути взагалі не ліцензованим? Відповідь: теоретично так, але наявність ліцензії перевіряється, коли модуль завантажується в ядро.

3. Шифрування бінарного образу програми. Чи можна зашифрувати ядро Linux цілком і в такому вигляді зберігати на зовнішньому пристрої і поширювати без доступу до незашифрованого бінарного образу? Відповідь: так, можна.

Згідно ліцензії GPL її умови поширюються як на вихідний текст, так і на бінарний варіант Програми. Ліцензія наказує поширювати програму (у тому числі і модифіковану) з вихідними текстами. При цьому вихідні тексти визначені як форма програми, що переважно використовується для її модифікації – і це не бінарний файл. Тобто, зашифроване сторонньою програмою бінарне ядро Linux має постачатися зі своїми вихідними текстами – немає проблем. Програми, які шифрують і розшифровують ядро Linux не є ні його частиною, ні виробами, що базуються на ядрі Linux, тому вони можуть не наслідувати ліцензію ядра. Якщо ще вбудувати в систему прив'язку до апаратури та перевірку контрольної суми бінарного ядра, то несанкціонована модифікація і запуск ядра під управлінням відладчика, вочевидь, ускладняться. Це цілком легальна в рамках GPLv2 практика, вона навіть отримала власну назву – Tivoization.

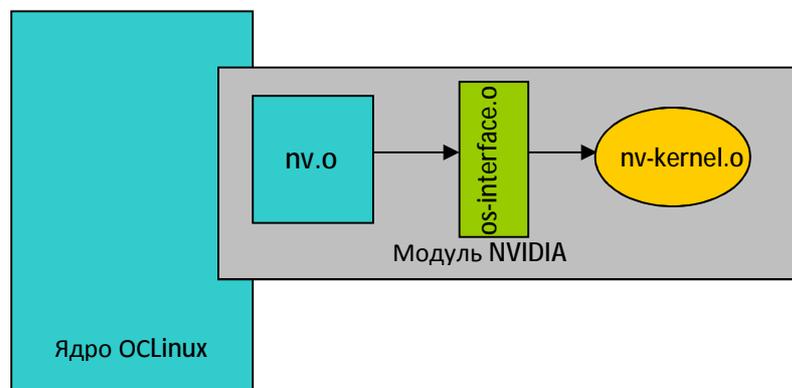
4. Частина модуля ядра може являти собою бінарну програму з закритим вихідним текстом. На сайті GNU у списку частих питань є питання про легальність використання «невільного» драйвера в ядрі Linux: <http://www.gnu.org/licenses/gpl-faq.html> # NonfreeDriverKernelLinux.

Відповідь на це конкретне питання – заперечення такої можливості. Дійсно, до модуля цілком це правило застосовується обґрунтовано – власники авторських прав на тексти ядра не дають особливих дозволів на приєднання закритих модулів до ядра. Але власниками авторських прав на самі модулі ядра даний аспект може бути оформлений по-іншому. Питання ускладнюється ще й тим, що тексти ліцензій GPL і коментарі до них можна тлумачити неоднозначно. Ще одне тлумачення говорить про незаконність БУДЬ-ЯКОЇ комбінації non-free + GPL:

<http://www.gnu.org/licenses/gpl-faq.html> # GPLWrapper, хоча в цьому ж документі дозволяється комбінація GPL + non-free (Lib).

**Приклад** (прецедент) драйвери NVIDIA Linux Display Drivers

Драйвер NVIDIA 319.23 Certified date: 2013.05.23 для ядра Linux має наступну структуру (рис.):



*Рис. Структура модуля графічного драйвера NVIDIA*

Головний програмний файл `nv.c` являє собою вихідний текст модуля ядра Linux, який реєструє два драйвери пристроїв (PCI і символьний). Модуль NVIDIA оголошує про свою приватну ліцензію: `MODULE_LICENSE ("NVIDIA")` та `MODULE_INFO (supported, "external")`.

Функції драйверів, таким чином, мають відкритий код, але можуть викликати інші функції з бінарної частини модуля ядра без необхідності відкрити її вихідний текст.

Бінарна частина модуля ядра Linux представлена у вигляді об'єктного файлу `nv-kernel.o`. У таблиці імен цього файлу всі внутрішні об'єкти мають імена виду «`_nv000142gm`».

Код бінарної частини модуля може викликати сервіс ядра Linux тільки через спеціальну інтерфейсну частину, що міститься у файлі `os-interface.c`. У свою чергу, ядро не має можливості безпосередньо звернутися до вмісту бінарної частини модуля NVIDIA.

Отже, вже більше 10 років існує (не єдиний) прецедент, який означає легальну можливість не відкривати вихідний текст принаймні частини модуля ядра ОС Linux.

Якщо у кого-небудь виникне необхідність створення гібрида ПЗ з ліцензією GPL і бібліотеки з невилною ліцензією, то потрібно визначити, чи підходить бібліотека під визначення «системна бібліотека». Для системних бібліотек існують винятки в GPLv2 і GPLv3, тому ніяких дій по «легалізації» такого гібрида вживати не потрібно. В інших випадках все вирішується додаванням власного виключення до ліцензії, причому, щоб зробити це, потрібно бути власником авторських прав на ПЗ GPL. Деталі й приклад можна подивитися на сайті GNU: <http://www.gnu.org/licenses/gpl-faq.html> # GPLIncompatibleLibs.

## Висновки

1. Відкритість вихідних текстів ПЗ не на всі сто відсотків є його корисною характеристикою.
2. Існує легальна можливість створення комбінацій з «вільного» і «невільного» ПЗ.

**Список використаних джерел:** 1. *Вихідні* тексти ядра Linux [Електронний ресурс]. – Режим доступу : <http://www.kernel.org>. 2. *Вихідні* тексти драйверів NVIDIA [Електронний ресурс]. – Режим доступу : <http://www.nvidia.com/object/linux-display-ia32-319.23-driver.html>. 3. *Ответы* на вопросы о лицензиях GNU [Электронный ресурс]. – Режим доступа : <http://www.gnu.org/licenses/gpl-faq.ru.html>.

УДК 004.432.42

## РАЗВИТИЕ ФУНКЦИОНАЛЬНОГО ПРОГРАММИРОВАНИЯ НА ПРИМЕРЕ ЯЗЫКА HASKELL

А. Морозов

*Черниговский национальный технологический университет*

Начиная с момента появления программирования в классическом его понимании и до сегодняшнего дня, разработчики стремятся с каждым годом выйти на более абстрактный уровень, упростить рутинную работу. Это также очевидно, как и появление в своё время языка Си, так как практически невозможно проводить разработку крупных, семантически нагруженных проектов без повышения уровня абстракции. Ценой такого роста является скорость работы, изменение требований к навыкам сотрудников, а также специфика применения того или иного средства разработки.

Одной из наиболее интересных теорий в плане исчислений, в будущем связанной с программированием, является теория лямбда – исчисления, разработанная ещё в 20-х годах XX столетия Алонзо Чёрчем. Это формальная система, которая была создана для исследования и определения вычислимости. Система была удобна в первую очередь, тем, что основывалась на двух операциях: аппликации и абстракции. Некоторое время спустя, во время активного развития программирования появился язык LISP, основным преимуществом которого было то, что с его помощью можно было довольно быстро обрабатывать списки, символьные цепочки. Язык был очень выгоден для разработки специализированных приложений. Но основное преимущество языка, давшее толчок функциональному программированию в принципе, состояло в том, что язык придерживался довольно интересных концепций программирования, в том числе и лямбда-исчисления.

Так как технические средства не позволяли реализовать задуманные ранее идеи, то (как и для любой другой идеи) требовалось многократное повышение производительности или же специфическая архитектура, позволяющая выполнить необходимые идеи. В 1973 году началась разработка так называемой LISP-машины, вычислительной машины, оптимизированной для выполнения программ на Lisp. Данная машина позволила в некоторой степени развить лямбда-исчисление, а также поспособствовала развитию ряда других технологий, таких, как сборка мусора, лазерная печать, графический интерфейс, растровая графика высокого разрешения, рендеринг. Исследования на тот момент проводились фирмой Хегох. Самым важным достижением данного исследования является то, что нестандартные подходы также могут приносить огромную практическую пользу. Стоит заметить, что данное событие не могло не повлиять на программирование в целом.

В 1987 году группой исследователей было решено создать чистый функциональный язык программирования на открытой основе. В 1990 году этот язык программирования был стандартизирован и получил название на честь математика Хаскелла Карри, – Haskell.

Отличительной чертой этого языка является то, что он ориентировался как на производственное программирование, так и на научную деятельность.

За время существования программирования было придумано достаточное количество различных механизмов для упрощения работы. Часть из них было описано математически уже после их появления, даже такие элементы как рекурсия, рекурсивные типы данных, callback-функции. Одной из приоритетных целей создания Haskell было то, что уже знакомые элементы, механизмы для работы должны описываться несколько иначе, исходя из семантики самой парадигмы языка, незначительные механизмы не должны быть определены монолитно, они должны из чего-либо исходить или как-либо быть организованными.

Haskell является довольно большим языком, всё (за исключением ядра компилятора) написано на самом Haskell. На Haskell нет каких-либо ограничений на программирование в том или ином стиле, но это может привести к запутанности и низкой производительности. Существует достаточно много рекомендаций по поводу стиля программирования на Haskell, которые помогают более грамотно спроектировать программу. Так, одним из самых распространённых и рекомендуемых стилей программирования является программирование «сверху-вниз». Такой подход практически противоположен обычному стилю программирования: обычно описываются части программы, позже соединяются в одну единую программу, здесь же написание программы начинается с самой верхней функции, программирование начинается с неформального описания, а далее уже описываются детали и недостающие части программы. Написать большую программу начисто (с «нуля») практически невозможно, поэтому в Haskell существует специальный тип `undefined`, который позволяет периодически проверять программу на работоспособность. Данный подход увеличивает уровень абстракции, а также помогает избежать лишнего объема работы.

Haskell является «ленивым языком». Вся «ленивость» языка выражается в ленивых вычислениях. Как известно, существует всего три типа вычислений: `call-by-value`, `call-by-name`, `call-by-need`. Последний из которых является ленивым порядком вычисления. Принцип такого вычисления довольно прост, вычисления производятся только тогда, когда они потребуются, соответственно, некоторые вычисления могут оказаться совершенно ненужными. Помимо ленивости вычислений в принципе, также существует ряд механизмов Haskell, которые придерживаются такого принципа. Например, мемоизация, то есть вычисление функции с одним набором данных единожды.

Таким образом Haskell представляет функциональную парадигму программирования, взяв при этом лучшее из предыдущих функциональных языков. При этом является довольно мощным средством для разработки как научных, так и производственных программ.

**Список использованных источников:** 1. Холomorphic А. Учебник по Haskell / Антон Холomorphic. – М., 2012. 2. *Практика* функционального программирования / Дмитрий Астапов, Роман Душкин, Сергей Зефилов, Евгений Кирпичёв, Алексей Отт, Дэн Пипони. – М., 2009. 3. *Lipovača Miran // Learn Haskell for great good // No Starch Press, San Francisco CA, 2011.* 4. *Лямбда-исчисление* [Электронный ресурс]. – Режим доступа : <http://ru.wikipedia.org/wiki/Лямбда-исчисление>.

УДК 519.768

## СТАТИСТИЧНИЙ, МУЛЬТІЯЗИЧНИЙ ПЕРЕКЛАД ЗАПИТІВ ПРИ ІНФОРМАЦІЙНОМУ ПОШУКУ

О. Мойсеенко, В. Литвинов

*Чернігівський національний технологічний університет*

Розглядається можливість покращення релевантності видачі результатів інформаційного пошуку на запити користувача, враховуючи мультіязичність вхідних даних розроблюваної системи обробки високошвидкісних потоків текстових даних.

Основною проблемою при частковому перекладі (на рівні запитів), є складність виявлення тематичних зв'язків між змістом пошукових запитів та змістом текстових документів. Виною тому є причини розрізненого представлення у схожих текстах на різних мовах однієї конкретної ситуації чи події. Даючи запит, людина формулює його, керуючись лише своїми представленнями про зміст необхідного документа. Розповсюджені лексичні засоби серед інформаційно-пошукових систем будуються на списках ключових слів. Виражений таким чином семантичний зміст документа обмежується цими списками для різних предметних галузей. Порівнюючи документи, представлені на різних мовах, але з однаковим тематичним змістом, їх схожість виявиться лише у разі співпадіння понять ключових слів для списків на цих мовах.

Одним із варіантів вирішення проблеми є розширення лексичного складу запитів та списків ключових слів (або пар ключових слів). Також (поки що в теорії) видається можливим залучення алгоритмів статистичного перекладу. Даний тип машинного перекладу широко використовується у великих комерційних організаціях, оскільки потребує великих потужностей для обробки та зберігання мовних пар документів для кожної мови окремо. Ця технологія передбачає існування відкритих онлайн-сервісів – це Google translate та Яндекс-перекладач. Нині для роботи подібних систем необхідна наявність великих баз паралельних текстів, де зберігаються словосполучення (N-грами) та їх переклади (рис.).

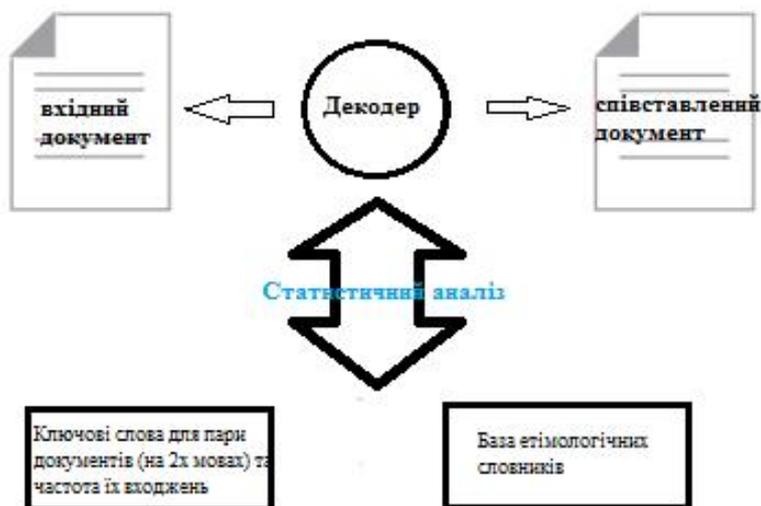


Рис. Блок-схема видачі запитів

Звуживши задачу з повнотекстового перекладу до перекладу лише запитів та видачі результатів на мові оригіналу, можливо суттєво знизити затрати розрахункових потужностей, реалізуючи алгоритми статистичного пошуку. Керуючись правилами спільного походження більшості слів для переліку мов, що входять до однієї мовної групи (слов'янські, романські та ін. мови), видається можливим реалізація програмного алгоритму для більш детального виокремлення семантичного навантаження ключових слів запитів та документів на різних мовах. Планується реалізація даного підходу для розроблюваної системи обробки високошвидкісних потоків текстових даних, залучивши до роботи електронні бази етимологічних словників для різних мов (словники, що містять інформацію про фонетичні та семантичні зміни окремих слів та морфем конкретної мови).

**Список використаних джерел:** 1. Апресян Ю. Д. Идеи и методы современной структурной лингвистики / Ю. Д. Апресян. – М. : Просвещение, 1966. 2. Иванюк В. Ю. Избирательность сочетания смыслов и возможные способы ее статистического выражения / В. Ю. Иванюк, В. В. Левицкий // Квантитативная лингвистика и автоматический анализ текстов. – 1990. – № 6(16). 3. Бобылёв В. Н. Краткий этимологический словарь научно-технических терминов / В. Н. Бобылёв. – М. : Логос, 2004. – 99 с.

## РЕАЛИЗАЦИЯ ОНЛАЙН-ОПРОСОВ В СИСТЕМЕ МАРКЕТИНГОВЫХ ИССЛЕДОВАНИЙ

Я. Нипорко, А. Акименко

*Черниговский национальный технологический университет*

С развитием информационных систем и возрастающим влиянием интернет-технологий наступил новый этап в сфере социологических исследований. Сбор информации с помощью онлайн опроса имеет больше преимуществ в сравнении с традиционными методами. Во-первых, интернет-опрос позволяет охватить значительные географические территории. Во-вторых, результаты можно получить за более короткое время, к тому же, подобный способ изучения общественного мнения значительно снижает трудовые и финансовые затраты.

В мировой практике широкое распространение получили различные программные системы для работы с информацией, это – Bellview WEB, Pulsar, Marketing Analytic и другие. Можно перечислять большое количество достоинств этих систем, но маркетинговым компаниям, работающим в условиях неактивного рынка респондентов, экономически не целесообразно приобретать дорогостоящие технологии, поэтому чаще всего используются САТІ и САРІ (телефонные и холл-исследования) опросы.

В сложившихся условиях одной из основных задач является разработка наиболее функционального, то есть с большой степенью адаптации, программного обеспечения онлайн-сбора информации и ее обработки. При этом система должна быть недорогой и простой в использовании.

При проектировании компьютерной системы онлайн-исследований необходимо учесть следующие функциональные требования:

- возможность работы с информацией различных типов: аудиозаписи, изображения, комментарии – ответы на открытые вопросы;
- наличие групп пользователей, которые, исходя из своих ролей, будут иметь определенный доступ к информации (для прохождения опроса достаточно зарегистрироваться);
- наличие механизмов работы с базой данных по контролю информации. Для менеджера структура содержимого баз данных должна позволять запустить контроль основных показателей. Соответственно требуется наличие возможности выполнять операции над совокупностями объектов информации, учитывая их взаимосвязи;
- наличие нескольких состояний информации («жизненного цикла»). Жизненный цикл информации состоит из нескольких этапов, каждый из которых определяет уровень доступности материала для различных категорий пользователей.

Разрабатываемая ИКС должна быть кроссплатформенной, реализация её осуществляется посредством языка PHP, и предполагает наличие Web-сервера Apache и СУБД MySQL.

Веб-приложение состоит из трёх основных компонентов. Клиентская часть веб-приложения – это графический интерфейс. Серверная часть – это программа или скрипт на сервере для обработки запросов пользователя. Серверная часть программируется на PHP. При каждом переходе пользователя по ссылке браузер отправляет запрос к серверу. Сервер обрабатывает этот запрос, вызывая некоторый PHP-скрипт, который формирует веб-страничку, описанную языком HTML, и отправляет клиенту по сети. Браузер тут же отображает полученный результат в виде очередной веб-страницы.

База данных располагается на сервере. Серверная часть веб-приложения обращается к базе данных, извлекая данные, которые необходимы для формирования страницы, запрашиваемой пользователем.

Их взаимодействия описывается следующим образом:

- браузер через Интернет отправляет HTTP-запросы веб-серверу;
- веб-сервер вызывает PHP-скрипт;
- PHP-скрипт обращается к базе данных;
- в результате PHP-скрипт возвращает клиенту веб-страницу, которую и отображает браузер.

В результате разработки концептуальной модели предметной области были выделены следующие сущности системы: пользователь, роль, анкета, опрос, вопрос, ответ и результат. Проанализировав сущность-связи, определены таблицы базы данных: Users, UserRoles, Roles, Form, Common, Question, Answer, Survey, Results. Построены диаграммы активности, которые показывают, как реагирует компьютерная система на различные этапы работы с информацией.

Разработку анкет можно осуществить с помощью конструктора или же для несложных проектов создавать опросники, все зависит от количества проектов, одновременно запущенных в поле. Подобного рода опрос позволит осуществлять контроль полноты данных, логическую увязку отдельных вопросов, использование случайной выборки респондентов и расчет ошибки репрезентативности непосредственно в процессе сбора информации посредством сравнения средних контролируемых показателей генеральной и выборочной совокупности. Этап контроля осуществляется с помощью JavaScript. Для подтверждения информации о покупках и потреблении товаров и продуктов питания респондентам онлайн-панели будут предложены сканеры штрих-кода.

Внедрение информационно компьютерной системы маркетинговых исследований позволит избежать субъективного влияния интервьюера, а также сэкономить около 50 грн. на каждом заполненном опроснике.

**Список использованных источников:** 1. *Визуальные Web-конструкторы для сбора и обработки информации* [Электронный ресурс]. – Режим доступа : <http://itc.ua/articles/>. 2. *Гольцман В. MySQL 5.0. Библиотека программиста* [Электронный ресурс] / В. Гольцман. – Режим доступа : <http://www.rumvi.com/products/ebook/mysql-5-0-библиотека-программиста/66cdefb7e9914c42bda94e85e/preview/preview.html>. 3. *Принцип работы сканера штрих-кодов* [Электронный ресурс]. – Режим доступа : <http://we.easyelectronics.ru/howwork/skaner-shtrihkoda-kak-on-rabotaet.html>. 4. *PHP и базы данных* [Электронный ресурс]. – Режим доступа : <http://www.php.su/phpdb/>.  
УДК 78.02

## СВОБОДНЫЙ МНОГОПЛАТФОРМЕННЫЙ АУДИО-СЕКВЕНСОР ДЛЯ СОЗДАНИЯ МУЗЫКАЛЬНЫХ КОМПОЗИЦИЙ

С. Ксёنز, Е. Рындич

*Черниговский национальный технологический университет*

Audacity – свободный многоплатформенный аудиоредактор звуковых файлов, ориентированный на работу с несколькими дорожками. Программа была выпущена и распространяется на условиях GNU General Public License[1].

Работает под управлением операционных систем: Microsoft Windows, Linux, Mac OS X и других.

Редактор Audacity обеспечивает выполнение следующих функций[2]:

- импорт и экспорт файлов WAV, MP3 (с использованием кодировщика LAME MP3), Vorbis, FLAC и других форматов;
- запись с микрофона, линейного входа и других источников;
- запись с одновременным прослушиванием имеющихся дорожек;
- запись до 16 каналов одновременно (необходима многоканальная звуковая карта);

- создание эффектов и расширения как в комплекте поставки, так и устанавливаемые отдельно (LADSPA либо на функциональном языке Nyquist);
- индикаторизация уровня записи и воспроизведения;
- изменение темпа с сохранением высоты тона;
- изменение высоты тона с сохранением темпа;
- удаление шума по образцу;
- спектральный анализ с использованием преобразования Фурье с различными формами окна;
- воспроизведение множества дорожек одновременно (без поддержки многоканального звука – при воспроизведении используются только два канала, в которые микшируются все дорожки);
- сведение дорожек с разными качественными характеристиками с автоматическим преобразованием к заданным характеристикам проекта в режиме реального времени;
- сохранение результатов во множество форматов, обеспечиваемых библиотекой libsndfile.

Audacity активно пользуется сторонними библиотеками, распространяемыми под разными лицензиями:

- графический интерфейс основан на библиотеке wxWidgets;
- звуковой движок использует PortAudio (альтернативная поддержка RtAudio так и не стала востребованной);
- опциональная поддержка MP3 (чтение и запись) реализована за счёт кодировщиков libmad и LAME;
- с помощью библиотек FFmpeg реализована поддержка чтения/записи AC3, AAC, WMA, AVI, MKV и других (*только начиная с версии 1.3.6*).

Заметная часть активных разработчиков Audacity живёт в США, поэтому они стараются не нарушать действующие в их стране законы. Это накладывает некоторые ограничения на функциональность программы, а именно[3]:

- отсутствует поддержка ASIO в Windows из-за несовместимой с GPL лицензией средств разработки;
- из-за (опять же) несовместимых с GPL лицензией средств разработки поддержка VST отсутствует по умолчанию, но может быть подключена за счёт отдельно скачиваемого дополнительного расширения, оригинальный интерфейс расширений VST поддерживается, начиная с версии 1.3.8;
- из-за требования компании Thomson Multimedia выплачивать сборы за средства кодирования MP3 – функция сохранения файлов в этом формате работает только при наличии отдельно скачанной библиотеки lame.

Программа переведена на многие языки мира, в том числе на русский (с 2002 года). В комплекте с программой поставляется электронная документация, русская версия которой доступна, начиная с 2004 года.

Данная программа представляет класс программного обеспечения, которое позволяет записывать, редактировать и воспроизводить цифровой звук. Обладает большим набором функций. Приведенная программа может быть усовершенствована, а именно введены следующие функции и возможности: улучшение графического интерфейса, добавление возможности работы с форматами ALAC и FLAC, а также с плагинами DXi и поддержка драйвера ASIO. Данное направление является перспективным.

**Список использованных источников:** 1. Audacity® [Электронный ресурс]. – Режим доступа : <http://audacity.sourceforge.net/>. 2. Category:Tutorial [Электронный ресурс]. – Режим доступа : <http://wiki.audacityteam.org/wiki/Category:Tutorial>. 3. Audacity [Электронный ресурс]. – Режим доступа : <http://ru.wikipedia.org/wiki/Audacity>.

## БЫСТРОЕ ПРОТОТИПИРОВАНИЕ АЛГОРИТМОВ АНАЛИЗА ВИДЕО С ПОМОЩЬЮ БИБЛИОТЕКИ КОМПЬЮТЕРНОГО ЗРЕНИЯ OPENCV

Н. Сытый, С. Нестеренко

*Черниговский национальный технологический университет*

Open Computer Vision — библиотека компьютерного зрения с открытым исходным кодом, предоставляющая набор типов данных и численных алгоритмов для обработки изображений алгоритмами компьютерного зрения. Библиотека была разработана компанией Intel для утверждения общего стандартного интерфейса компьютерного зрения для приложений в этой области. OpenCV широко используется как в коммерческих, так и в научных разработках, а также фактически является лидером среди различных открытых проектов компьютерного зрения, используемых в образовательной сфере. Упомянув использование OpenCV в обучении, нельзя не упомянуть наличие большого количества качественной и постоянно обновляемой документации.

Библиотека состоит из небольших модулей, разделенных по функциональному признаку:

- **opencv\_core** – ядро: базовые структуры, вычисления (математические функции, генерация псевдослучайных чисел, DFT, DCT, ввод/вывод в XML);
- **opencv\_imgproc** – обработка изображений (фильтры, преобразования и т. д.);
- **opencv\_highgui** – простой UI, загрузка/сохранение изображений и видео;
- **opencv\_ml** – методы и модели машинного обучения (SVM, деревья принятия решений);
- **opencv\_features2d** – различные дескрипторы (SURF);
- **opencv\_video** – анализ движения и отслеживание объектов (оптический поток, шаблоны движения, устранение фона);
- **opencv\_objdetect** – детектирование объектов на изображении (вейвлеты Хаара, HOG);
- **opencv\_calib3d** – калибровка камеры, поиск стереосоответствия и элементы обработки трехмерных данных;
- **opencv\_flann** – библиотека быстрого поиска ближайших соседей (FLANN);
- **opencv\_contrib** – сопутствующий код, еще не готовый для применения;
- **opencv\_legacy** – устаревший код, сохраненный ради обратной совместимости;
- **opencv\_gpu** – ускорение некоторых функций OpenCV за счет CUDA (NVIDIA);

На рис. изображена структура библиотеки OpenCV и взаимодействие между отдельными её модулями.

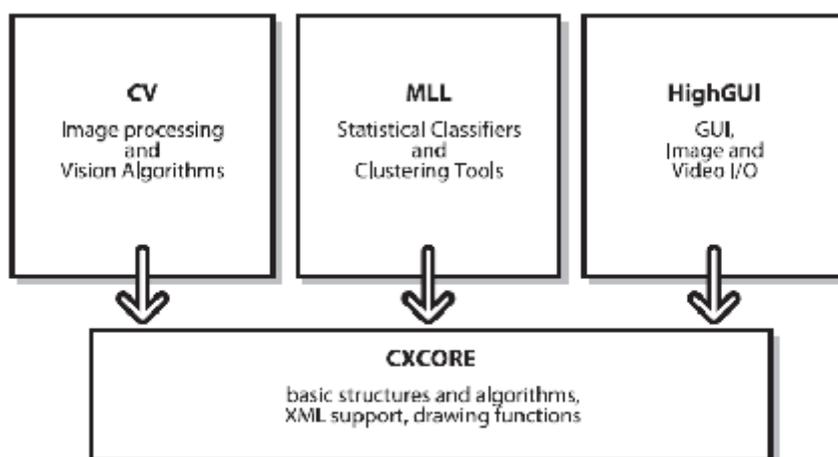


Рис. Структура библиотеки OpenCV

Библиотека OpenCV имеет обширный набор базовых структур и алгоритмов данных, механизмов работы с видеофайлами и изображениями. В ней также имеется набор базовых примитивов для создания пользовательского интерфейса. Всё это способно помочь исследователю быстро проверить гипотезу об использовании какого-либо алгоритма для анализа видеопоследовательностей. В случае подтверждения этой гипотезы исследователь может заняться вопросами улучшения производительности алгоритма, в частности, с использованием технологии CUDA. В частности простой алгоритм подсчета количества людей в кадре был реализован автором менее чем за один рабочий день.

**Список использованных источников:** 1. *Learning OpenCV* / Bradsky G., Kaehler A. O'Reilly, 2008. 2. *Документация* библиотеки компьютерного зрения OpenCV [Электронный ресурс]. – Режим доступа : <http://docs.opencv.org/>.

УДК 004.71

## **ИСПОЛЬЗОВАНИЕ СИСТЕМЫ УПРАВЛЕНИЯ СОДЕРЖИМЫМ JOOMLA ДЛЯ ПОСТРОЕНИЯ WEB-ПОРТАЛА ПРЕДОСТАВЛЕНИЯ IT-КОНСУЛЬТАЦИЙ**

М. Сычова

*Чернігівський національний технологічний університет*

Информационный веб-ресурс предоставления компьютерных услуг – это веб-ресурс, предоставляющий информацию о программных и аппаратных составляющих компьютерной техники. Данный веб-ресурс создан с учетом того, что пользователи не имеют больших познаний в компьютерной технике и мало осведомлены в ее терминах. Все статьи подобраны с учетом самых распространенных проблем и самых задаваемых вопросов о компьютерной технике.

Интерфейс данного веб-ресурса был разработан максимально простым и понятным для более удобного и быстрого поиска нужной информации.

Для навигации по разделам, категориям и статьям ИВР компьютерных услуг создано меню, располагающееся в верху сайта, в которой каждый подпункт меню соответствует отдельной категории или разделу со статьями аналогичной тематики. Также в данном веб-ресурсе имеется специальный раздел – “Вопрос - Ответ”, отвечающий за вывод всех задаваемых вопросов пользователями с ответами на них. Вопрос в этом разделе может задать любой пользователь ИВР компьютерных услуг, но ответ оставит только администратор веб-ресурса.

В данном веб-ресурсе имеется поисковая система, которая способствует быстрому переходу к интересующей информации без перехода по пунктам меню. Поисковая система сравнивает введенные ключевые слова со словами из баз данных и выводит перечень статей, где встречаются эти ключевые слова. Поисковая система ИВР компьютерных услуг является модулем CMS Joomla.

Joomla – система управления содержимым (CMS), написанная на языках PHP и JavaScript, использующая в качестве хранилища базы данных СУБД MySQL или другие индустриально-стандартные реляционные СУБД. Является свободным программным обеспечением, распространяемым под лицензией GNU GPL.

CMS Joomla включает в себя различные инструменты для разработки веб-сайта. Основной особенностью системы является минимальный набор инструментов при начальной установке, который можно дополнить по мере необходимости. Joomla позволяет отображать интерфейс пользователя и административной части на любом языке. При установке стандартный язык – английский, но его очень легко можно сменить. Необходимо просто скачать нужный языковой пакет и установить его.

Функциональность CMS Joomla легко можно увеличить установкой новых модулей и компонентов. Так же есть система шаблонов, с помощью которой очень просто сменить

внешний вид сайта, а так как система шаблонов имеет открытый код, то, зная CSS и HTML, вы можете создать свой уникальный дизайн.

К преимуществам системы можно отнести и то, что все компоненты, модули, плагины и шаблоны можно написать самому, разместить их в структурированном каталоге расширений или отредактировать существующее расширение по своему усмотрению.

Большим плюсом Joomla является возможность редактирования мета-данных не только сайта, но каждой страницы отдельно, а также стандартная возможность быстро включать ЧПУ-ссылки, что позволяет поисковым системам после переноса на хостинг проиндексировать сайт.

Сейчас Joomla активно развивается, разработчики постоянно выпускают обновления и новые версии с доработкой и усовершенствованием, что расширяет функциональные возможности данной CMS.

УДК 004.75:004.414.23

## ОБЗОР ИНСТРУМЕНТОВ РАЗРАБОТКИ МОДЕЛЕЙ GRID-СРЕДЫ

И. Скитер, О. Прелая, Т. Гуза

*Черниговский национальный технологический университет*

На сегодняшний день grid-технологии активно развиваются и применяются для решения вычислительных задач большой размерности. Построение моделей распределенной grid-среды широко используется при исследовании вопросов планирования и управления распределенными ресурсами, прогнозировании времени выполнения задачи в grid, проверки корректности работы grid-приложения и др. В работе приведены результаты исследования открытых инструментов построения моделей grid-среды.

### **1 Требования к системам моделирования распределенной grid-среды**

С целью обозначения критериев исследования были определены следующие требования к системам моделирования grid:

- 1) поддержка моделей grid-задач разнотипного типа;
- 2) поддержка различных архитектур grid-среды, в т.ч. гетерогенной;
- 3) возможность моделирования QoS параметров ресурсов grid-сети;
- 4) наличие высокоуровневых примитивов построения моделей;
- 5) возможность взаимодействия с внешними системами, например, создание модели, обменивающейся данными с реальными компонентами grid-среды.
- 6) расширяемость;
- 7) кроссплатформенность;
- 8) использование стандартизированных и открытых технологий;
- 9) документированный API.

### **2 MicroGrid**

Инструмент позволяет производить симуляции с использованием grid-приложений без адаптации, в случае, если они основаны на использовании Globus Toolkit [1]. MicroGrid предоставляет возможность автоматической генерации структуры grid-сети, а также симуляцию сетевых устройств и протоколов, включая OSPF (Open Shortest Path First) и BGP (Border Gateway Protocol), что позволяет оценить эффективность алгоритма планирования в условиях, приближенных к реальным.

MicroGrid предоставляет промежуточному ПО доступ к виртуальным grid-ресурсам. Таким образом промежуточное ПО и приложение не требуют адаптации для проведения моделирования.

Виртуальный grid, который MicroGrid предоставляет для выполнения пользовательского приложения, может иметь как однородную, так и гетерогенную

структуру, что определяется соответствующей конфигурацией. Это позволяет моделировать различные среды исполнения grid-приложений и отслеживать зависимость эффективности алгоритмов планирования от степени гетерогенности среды исполнения.

Недостатком инструмента является жесткая привязанность к Globus Toolkit и тот факт, что проект не развивается с 2004 года. Таким образом многие современные стандарты grid-вычислений не поддерживаются MicroGrid.

### **3 WorkflowSim**

В WorkflowSim в подготовке и выполнении потока работ [2] принимают участие несколько слоев системы.

Модель системы управления потоками [3] состоит из:

- 1) Workflow Mapper для преобразования абстрактного потока работ в множество задач, привязанных к конкретному вычислительному ресурсу;
- 2) Workflow Engine для обработки зависимостей между подзадачами;
- 3) Workflow Scheduler, отвечающий за размещение подзадач на ресурсах.

Другие компоненты включают модуль кластеризации, который объединяет небольшие задачи в одну, трассировщик выполнения, который отслеживает процесс выполнения задачи и модуль разделения заданий, который отвечает за преобразование одного потока работ в множество задач типа «поток работ».

### **4 SimGrid**

SimGrid — открытый фреймворк для симуляции распределенных вычислений [4], включая grid-среды, облачные вычисления и приложения MPI. Основой данного фреймворка является сущность сообщения, и сам процесс симуляции описывается как последовательность передач сообщений от одного примитива к другому. Такой способ взаимодействия объектов выбран из исторических соображений, поскольку SimGrid изначально был нацелен на симуляцию среды MPI. Это позволяет запускать MPI-приложения в среде моделирования SimGrid без необходимости их модификации.

Инструмент поддерживает виртуализацию, планирование потоков работ и моделирование различных характеристик сети. Также предоставляется инструментарий для формальной верификации программ, исполняемых в среде SimGrid.

Недостатком инструмента является низкоуровневый и специфичный API, ориентированный на стандарт MPI. Фреймворк требует перекомпиляции для каждой платформы, в том числе и перекомпиляции модели, что увеличивает время разработки и проверки модели.

### **5 GridSim**

GridSim является объектно-ориентированным фреймворком [5] для создания моделей grid-сетей, использует JVM в качестве платформы, поэтому является переносимым и не требует повторной компиляции моделей. API фреймворка предоставляет следующие абстракции – grid-ресурсы, задачи, планировщики, и специфические для grid-среды характеристики. Фреймворк является расширяемым и предоставляет API для реализации собственных абстракций grid-среды, алгоритмов планирования и др.

GridSim позволяет моделировать различные характеристики сети – пропускную способность, загруженность, алгоритмы маршрутизации пакетов. Предоставляются инструменты определения зависимости характеристик grid-ресурсов, таких, как пропускная способность сети, загруженность процессоров, время суток, дни недели и т.д.

GridSim не предоставляет инструментов моделирования grid-задач, характеризующихся сложной структурой, которые могут быть представлены в виде набора последовательно-параллельных блоков с определенной схемой синхронизации вычислений. Модели планирования представлены локальными планировщиками grid-ресурсов, инструменты разработки модели метапланировщика не предоставляются.

Результаты сравнительного анализа рассмотренных систем приведены в таблице 1.

Сравнение инструментов моделирования

Название	Расширяемость	Поддержка workflow	Взаимодействие с внешними системами	Высокоуровневые примитивы grid-среды	API
MicroGrid	-	-	+	-	-
WorkflowSim	+	+	-	+	-
SimGrid	+	+	+	+	+
GridSim	+	-	+	+	+

Таким образом, в данной статье представлены результаты исследования существующих систем моделирования grid-среды. Из рассмотренных систем как наиболее полнофункциональные и активно развивающиеся можно выделить инструменты SimGrid и GridSim. Указанные инструменты предоставляют гибкий API и используют в своей реализации открытые технологии и стандарты.

SimGrid требует проектировать модель в терминах MPI, что усложняет описание модели. GridSim работает на платформе Java, что позволяет использовать этот инструмент без перекомпиляции на различных платформах. Кроме того, архитектура фреймворка GridSim является масштабируемой, что позволяет выполнять крупномасштабное моделирование grid-сети с использованием нескольких вычислительных узлов.

Авторами предлагается расширение архитектуры GridSim инструментами моделирования планирования разных типов задач в grid-среде.

**Список использованных источников:** 1. Robert H. Lasseter, Paolo Piagi, Microgrid: A Conceptual Solution, SC'04 Aachen, Germany, 2004. 2. Alberto Forti. DAG Scheduling for grid computing systems. Ph.D. Thesis, University of Udine – Italy, Department of Mathematics and Computer Science, 2005-2006. 3. Chen, W., and E. Deelman, WorkflowSim: A Toolkit for Simulating Scientific Workflows in Distributed Environments // The 8th IEEE International Conference on eScience, Chicago, 2012. 4. Henri Casanova, Arnaud Legrand, Martin Quinson. SimGrid: a Generic Framework for Large-Scale Distributed Experiments. 10th IEEE International Conference on Computer Modeling and Simulation - EUROSIM / UKSIM 2008, Cambridge : United Kingdom (2008). 5. Rajkumar Buyya and Manzur Murshed, GridSim: A Toolkit for the Modeling and Simulation of Distributed Resource Management and Scheduling for Grid Computing, The Journal of Concurrency and Computation: Practice and Experience (CCPE), Volume 14, Issue 13-15, Wiley Press, Nov.-Dec., 2002.

УДК 681.3.07

## ИСПОЛЬЗОВАНИЕ СЕТЕВОГО ФИЛЬТРА NETFILTER/LPTABLES ДЛЯ РАЗРАБОТКИ АКТИВНОГО УЗЛА СЕТИ

А. Верёвко, Д. Дэвис, М. Тевкун

*Черниговский национальный технологический университет,  
Университет Глиндор*

Файервол – это комплекс программных или аппаратных средств, предназначенный выполнять задачи контроля и фильтрации проходящего через него сетевого трафика [1].

В зависимости от размещения и функциональных возможностей файерволы можно разделить на:

- традиционный фаервол — программное обеспечение (или часть операционной системы) на шлюзе (сервере, передающем трафик между сетями) или аппаратное обеспечение, контролирующее входящие и исходящие пакеты данных между сетями.

- персональный фаервол — программное обеспечение, установленное на пользовательском компьютере и предназначенное для блокирования несанкционированного доступа к этому компьютеру.

Также существует гибридный вариант, когда традиционный фаервол блокирует доступ к собственным ресурсам.

Из типичных функциональных возможностей у фаерволов можно выделить [2]:

- фильтрация доступа к публичным службам;
- защита от получения конфиденциальных данных с внутренней сети;
- защита от внедрения ложных данных во внутреннюю сеть через уязвимые службы;
- контроль доступа к компьютерам во внутренней сети;
- уведомление о попытках несанкционированного доступа к компьютерам в сети или самому фаерволу.

Практически все современные сетевые операционные системы имеют свои реализации фаерволов. Операционная система GNU/Linux – не исключение, и на все ОС на основе ядра Linux версии выше 2.4 имеется встроенный фаерволл «netfilter», а также утилита для управления фаерволлом «iptables» (или «ip6tables» для Ipv6) [3].

В netfilter/iptables применяется идея списка правил, состоящих из критериев и действия, которое выполняется, если пакет соответствует критериям. Основной работы netfilter/iptables является концепция создания цепочек правил, а также переход пакетов между цепочками.

Netfilter выполняет анализ пакетов данных и фильтрацию согласно заданным цепочкам правил. Стандартная поставка netfilter не имеет возможности проводить сложный анализ передаваемых данных, однако может быть расширена за счет подключения дополнительных модулей, в том числе от сторонних разработчиков.

Под активным узлом сети подразумевается коммутатор или маршрутизатор, который может выполнять обработку данных на этапе их передачи по сети. [4] Для разработки активного узла сети для подсчета статистики посещения веб-ресурсов предполагается разработка match модуля для Netfilter, в который будут передаваться пакеты для анализа. Данный модуль будет разрешать пересылку пакета в сеть, а его копию передавать на анализ.

Для указания match модуля в правиле, задаваемом в Iptables, для него необходимо добавить расширение, иначе Iptables не будет знать, какой модуль вызывать при обработке правила.

Анализ пакетов данных будет выполняться в приложении, запущенном как демон (работающее в фоновом режиме без взаимодействия с пользователем), которое будет получать пакет, классифицировать его по содержимому и считывать данные.

Match модуль для netfilter будет реализован как модуль ядра. Соответственно данный модуль будет и работать в адресном пространстве ядра. Приложение для анализа и логирования пакетов будет выполнено в виде демона и запускаться в пространстве пользователя. Следовательно для передачи данных от модуля ядра к демону целесообразно использовать файл устройства. Пакеты данных могут иметь различный размер, по этой причине файл устройства должен быть символьным.

После анализа данные будут записываться в базу данных MySQL. Расчёт и отображение статистических данных будет выполняться в клиентском веб-приложении.

С учетом перечисленных выше частей узла активной сети для статистического анализа посещения социальных сетей была разработана архитектура сети, представленная на рис.

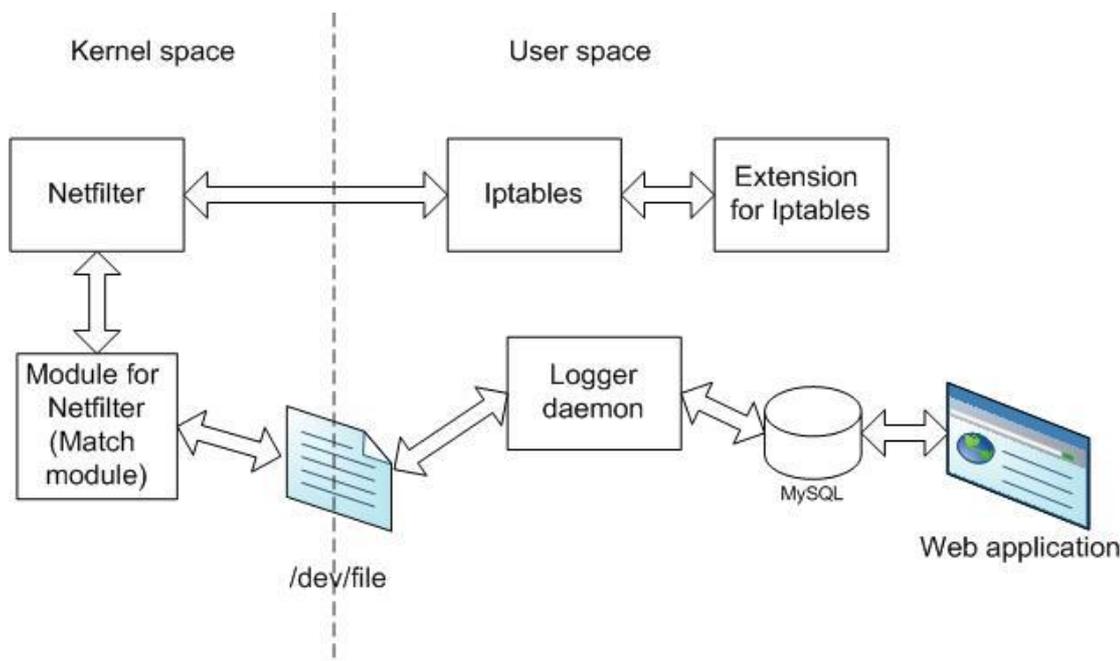


Рис. Архитектура активного узла сети

В полученной системе каждый пакет, передаваемый по протоколу http, будет перехватываться файерволом netfilter и передаваться в match модуль для анализа. Match модуль будет создавать копию пакета, оригинал пакета передавать дальше в сеть, а копию будет записывать в символьный файл устройства. Logger daemon будет считывать пакет с символьного файла устройства, считывать с него необходимые данные и записывать данные в базу данных MySQL. Клиентское веб приложение (при запросе данных) будет выполнять запрос к базе данных, обработку полученных данных и выдачу их пользователю.

**Список использованных источников:** 1. Terry Escamilla, *Intrusion Detection: Network Security Beyond the Firewall*, Wiley (1 edition), Sep. 1998. 2. Richard Deal, *Cisco Router Firewall Security*, Cisco Press, Aug. 2004. 3. *Iptables/netfilter* website. [Online]. Available: <http://netfilter.org/>. 4. Calvert K., ed. *Architectural Framework for Active Networks*. AN draft, AN Architecture Working Group, 1998.

УДК 316.774

## КОМУНІКАЦІЯ В РЕАЛЬНОМУ ЧАСІ. ТЕХНОЛОГІЯ WEBRTC

І. Устименко

*Чернігівський інститут інформації, бізнесу і права*

На початку ХХ століття світ інтернет-технологій почав стрімко розвиватись. Спочатку існував звичайний http сервер, сторінки були не динамічні, у 2004 році з'являється технологія Ajax, яка дозволяє асинхронно здійснювати зв'язок між клієнтом та сервером. Орієнтовно 2008 року з'явилась технологія Web Sockets, яка реалізувала також асинхронний зв'язок, але при цьому клієнт міг здійснювати запити незалежно від часу. У 2012 році з'являється технологія WebRTC (Real-Time Communications – комунікація в реальному часі), розроблена компанією Google. Ця технологія твердить, що ми можемо використовувати клієнта для передачі потокової інформації, такої як файли аудіо- та відеозображення. На практиці це використовується для організації відео-конференцій, співбесід, дзвінків тощо.

Нині технологія WebRTC активно використовується у просторах Інтернету, для прикладу можна навести сайт <http://webcamtoy.com>, який використовує цю технологію. Значними перевагами цієї технології є те, що вона не потребує авторизації, не використовує плагіни, не потребує завантаження додаткового програмного забезпечення, також вона є мульти-платформенною (підтримують всі браузери, але браузер Internet Explorer підтримує тільки з застосуванням технології Chrome Frame). Все це свідчить, що ця технологія є не тільки успішною, але й виступає головним конкурентом такого програмного забезпечення, як Skype. Технологія WebRTC застосовується і в мобільних пристроях, що дає змогу здійснювати дзвінки на комп'ютер, але цю функцію підтримують поки що лише мобільні пристрої з операційною системою Android. Важливим фактором є те, що WebRTC API (*application programming interface* – інтерфейс прикладного програмування) доступний як C++ бібліотека, що дає змогу писати не тільки інтернет-додатки, а й програмне забезпечення з використанням WebKitGTK та QT.

Основними особливостями WebRTC є : доступ до Веб-камери та мікрофону, реалізація відео- та аудіодзвінків, передача даних від клієнта до клієнта, відеоконференції та показ екрану (робочого столу). При здійсненні аудіо- та відеодзвінків відбувається шифрування з'єднання та контроль кодеків, так як відео, яке ми отримуємо від камери, буває різного формату. У WebRTC використовуються два аудіокодеки, створені в GIPS, а також відеоформат VP8 (WebM). На відміну від широко використовуваного стандарту H.264, розповсюдженого на пропрієтарних умовах, вихідний код VP8 відкритий. Передача поточкових даних між браузерами або іншими підтримуваними її додатками відбувається за технологією Peer-to-peer. WebRTC підтримує передачу даних до 4ГБ таким чином, що ми можемо написати кроссплатформне рішення щодо здійснення відеодзвінків з мобільних пристроїв (Android) на персональні комп'ютери, знаючи тільки JavaScript і HTML5. Для того, щоб показати хоча б мінімальні можливості цієї технології, створимо файл з розширенням "html" та збережемо його на сервері. Вміст файлу повинен мати вигляд:

```
<!DOCTYPE HTML>
<html>
<head>
</head>
<body>
<script>
  window.onload = function(){
    navigator.webkitGetUserMedia({ video:true },getStream, noStream);
  };
  function getStream(stream){
    var url = webkitURL.createObjectURL(stream);
    video.src = url;
  }
  function noStream(faild){
  }
</script>
<video id="video" autoplay="autoplay" width="500"></video>
</body>
</html>
```

Після запуску цього файлу за допомогою браузера Google Chrome з'явиться діалогове вікно, в якому браузер запитав дозвіл на використання Web-камери. Після надання дозволу на використання камери в браузері з'явиться вікно розміром 500x500 пікселів, у якому ми побачимо зображення, що передається на камеру.

Можливо, що WebRTC повністю замінить Skype, оскільки має вагомні переваги, але корпорація Microsoft оголосила що в майбутньому буде використовувати цю технологію у

програмному забезпеченні Skype. Існує підтримка технології WebRTC у браузерях, починаючи з таких версій: Google Chrome з 17 версії, Opera з 12 версії і Firefox з 18 версії. Для інших браузерів використовують плагін WebRTC4All. На початку 2013 року було здійснено перший відеодзвінок між браузером Google Chrome і Firefox. WebRTC – сильний інструмент для створення власних програм голосового та відеозв'язку, розпізнавання та редагування.

**Список використаних джерел:** 1. *Режим* доступу : <http://ru.wikipedia.org/wiki/WebRTC>. 2. *Google* open source WebRTC for open video/audio chat [Електронний ресурс]. – Режим доступу : <http://www.h-online.com/open/news/item/Google-open-source-WebRTC-for-open-video-audio-chat-1253848.html>. 3. *WebRTC* FAQ [Електронний ресурс]. – Режим доступу : <https://sites.google.com/site/webrtc/faq>.

УДК 004.4

## ИСПОЛЬЗОВАНИЕ PROCESS EXPLORER И AUTORUNS ДЛЯ БОРЬБЫ С ВРЕДНОСНЫМ ПО

Р. Андрущенко, А. Роговенко

*Черниговский национальный технологический университет*

ОС Windows на данный момент – наиболее популярная ОС в мире. Ввиду своей популярности это одна из самых уязвимых систем, так как существует огромное количество вредоносных программ, которые могут вывести систему из строя. Причем чаще всего такие программы наносят больший вред не столько своей деятельностью, сколько теми ошибками, которые в них содержатся. Зачастую возможностей системы для предотвращения потерь данных и борьбы с вредоносным ПО недостаточно и требуется установка дополнительного ПО, антивирусов в частности. Например, в Windows практически нет инструментов для работы с процессами. А диспетчер задач, к сожалению, имеет очень скудные функции для слежки и управления процессами.

Process Explorer – программа от компании Sysinternals для наблюдения в реальном времени за действиями различных процессов и управления ими в среде операционной системы Windows.

Process Explorer ныне – самая популярная утилита среди продуктов Sysinternals. Выводимая Process Explorer информация значительно подробнее, чем у диспетчера задач Windows. Утилита не требует установки. Достаточно распаковать архив и запустить файл prosexp.exe.

Основные возможности:

- 1) Иерархическое отображение процессов.
- 2) Возможность идентификации системных процессов (например, является ли процесс svchost.exe системным или «левым»).
- 3) Изменяемый диапазон измерений загрузки CPU и графические индикаторы.
- 4) Возможность закрытия древа процессов.
- 5) Возможность в реальном режиме времени менять приоритет и то, на каком ядре процессора будет выполняться тот или иной процесс.
- 6) Возможность проверки сертификата файла процесса.
- 7) Возможность заменять системный диспетчер задач по тем же горячим клавишам.
- 8) Управление службами системы (остановить, приостановить, возобновить, перезапустить).
- 9) Функции принудительного завершения процессов (Kill), приостановки (Suspend) и продолжения выполнения (Resume).

10) Управление потоками, т.е. динамически внедряемыми в основной процесс программными модулями (DLL).

11) В любой момент времени принудительно выполнять дампы памяти (Minidump или Full Dump) с сохранением в выбранный файл.

Иерархический характер дерева процессов способствует визуальному восприятию родительски-дочерних отношений каждого активного процесса.

Главное окно состоит из двух панелей. В верхней панели отображается дерево процессов и информация о них. Нижняя панель дает информацию обо всех DLL, загруженных выделенным в верхней панели процессом, открытых им файлах, папках, разделах и ключах реестра. При выборе уровня System дерева процессов в нижней панели можно получить информацию обо всех загруженных драйверах системы, их описание, версию, путь исполняемого файла, адрес в оперативной памяти, размер. Кроме того, можно проверить цифровую подпись, а также просмотреть строковые значения в самом исполняемом файле или в оперативной памяти.

Также в программу встроена функция поиска процессов, которые используют те или иные файлы и папки, модули, библиотеки в системе.

Process Explorer предоставляет удобный инструмент, с помощью которого очень просто определить то, каким процессом открыто определённое окно. Утилитой Process Explorer также можно заменить стандартный Диспетчер задач Windows. После такой замены по нажатию комбинации клавиш Ctrl+Shift+Esc будет открываться Process Explorer.

Process Explorer может вывести скрытое окно приложения на экран. Воспользовавшись пунктом контекстного меню Bring to Front, можно получить доступ к такому окну.

Также Process Explorer позволяет просмотреть сетевые соединения каждого процесса. Если, например, такая программа, как «Блокнот», использует сетевое соединение, то это явный признак, что в системе есть вредоносное ПО.

Однако наибольшие возможности открываются при использовании Process Explorer в связки с другими программами, такими, как, например, Autoruns, PsExec от той же фирмы Sysinternals.

Autoruns – программа для получения подробнейшей информации обо всех процессах, запускающихся автоматически, в среде операционной системы Windows. Программа также не требует инсталляции в системе и имеет консольный вариант.

Autoruns на данный момент является одним из самых информативных инструментов для отслеживания точек автоматического запуска процессов в Windows, в том числе, скрытых или необычных, часто используемых вирусами и другим вредоносным программным обеспечением (malware). Поиск и устранение внедрившегося в среду Windows вредоносного программного обеспечения – одно из основных направлений использования Autoruns.

Кроме работы в среде активной операционной, возможно использование утилиты для анализа точек автозапуска другой ОС, системный каталог которой и каталог с профилем пользователя можно выбрать с помощью главного меню.

Возможности программы для исследования и нейтрализации элементов автозапуска позволяют легко справиться с внедрившимся в систему вредоносным ПО. Любой вирус, лишенный возможности автоматического запуска, становится совершенно безвредным. При возникновении сомнений в отношении какого-либо элемента автозапуска, приведенного в списке выходных данных, Autoruns желательно провести подробное его исследование, используя следующие приемы:

1) Проанализировать описание, сведения об издателе, наличие и достоверность цифровой подписи. Проверить точку автостарта программы в реестре или каталоге файловой системы.

2) Можно сразу получить дополнительные сведения по результатам поиска в Интернете.

3) Если имеется сохраненный журнал предыдущих сессий – можно сравнить текущие данные с сохраненными.

4) Отправить файл на онлайн-проверку VirusTotal.com. Если файл является вредоносным, с большой долей вероятности, служба VirusTotal этот факт подтвердит.

5) Для подробного анализа активности подозрительного процесса можно использовать тот же Process Explorer. Можно воспользоваться прямым вызовом утилиты через пункт контекстного меню для выбранного элемента автозапуска.

В качестве примера работы с программами Autoruns и Process Explorer можно привести пример поиска и устранения вредоносного ПО в ОС Windows. Возьмем зараженную флэш-карту вирусом Trojan.Win32.Lebag.akl. Его особенностью является то, что предположительно из-за ошибок в его исходном коде он постоянно производит чтение/запись с флэш-карты. Поэтому ее нельзя безопасно извлечь из компьютера, что может привести к нежелательным потерям данных или вообще вывести из строя устройство хранения данных.

Сделав предварительный снимок списка автозапуска с помощью Autoruns, можно сразу же определить местоположение вредоносной программы на компьютере. По местоположению можно произвести поиск в Process Explorer и обнаружить процесс, который запускается исполняемым файлом. Как оказалось, процесс маскируется под браузер, установленный в системе по умолчанию, поэтому вредоносную программу нельзя было обнаружить штатными средствами Windows. Далее можно приостановить процесс, закрыть все файлы, которые он использует и безопасно извлечь флэш-карту из компьютера. После этого производим удаление исполняемых файлов и удаляем в Autoruns все строчки, связанные с этими исполняемыми файлами. Таким образом был произведен пример выполнения операции обезвреживания вредоносного ПО с помощью утилит Autoruns и Process Explorer.

В итоге можно сделать вывод, что использование Autoruns и Process Explorer позволяет осуществлять всеобъемный контроль над процессами. Утилиты предоставляют удобный инструмент для анализа процессов, поиска причин неполадок, неисправностей системы и их устранения. Данные утилиты являются отличной альтернативой системным средствам и гораздо безопаснее встроенных инструментов, так как не требуют от пользователя непосредственной работы с системным реестром.

**Список использованных источников:** 1. *Режим* доступа : <http://www.ixbt.com/soft/sysinternals-procexp.shtml>. 2. *Компьютерный практикум [Электронный ресурс]. – Режим* доступа : <http://ab57.ru/procexp.html>; <http://ab57.ru/autoruns.html>. 3. *Режим* доступа : <http://habrahabr.ru/post/108536/>. 4. *Режим* доступа : [http://ru.wikipedia.org/wiki/Process\\_Explorer](http://ru.wikipedia.org/wiki/Process_Explorer) 5. *Режим* доступа : [http://blogs.technet.com/b/mark\\_russinovich/archive/2012/09/07/3518691.aspx](http://blogs.technet.com/b/mark_russinovich/archive/2012/09/07/3518691.aspx); <http://technet.microsoft.com/ru-RU/sysinternals/bb96390>.

УДК 681.518:004.925.8

## **РОЗРОБКА ГЕОІНФОРМАЦІЙНОЇ СИСТЕМИ ІНФРАСТРУКТУРИ АГРОФІРМИ ЗА ДОПОМОГОЮ GEOTOOLS ТА POSTGIS**

І. Бальченко

*Чернігівський національний технологічний університет*

Передові дослідники агроінженерної науки і виробництва відзначають, що основними напрямками досліджень у найближчій перспективі будуть автоматизація та роботизація управління сільськогосподарськими та виробничими процесами, а також розробка інтегрованого комп'ютерного забезпечення для техніко-економічного обґрунтування господарської діяльності з урахуванням місцевих умов [1].

У роботі [2] авторами була запропонована концепція побудови автоматизованої системи управління сільськогосподарським підприємством (АСУ СХП). Така система управління має специфічні особливості, серед яких розподіленість об'єкта управління, що змушує враховувати просторові чинники для забезпечення функцій управління [3].

У зв'язку з цим виникла необхідність у побудові геоінформаційної системи (ГІС) інфраструктури агрофірми, що забезпечує ведення нормативної інформації по об'єктах інфраструктури, таких як: відділення, дороги, сільськогосподарські угіддя, сховища для врожаю, склади матеріальних ресурсів, машинні парки, ремонтні майстерні, – а також має можливість розраховувати раціональні шляхи, що з'єднують різні пари об'єктів.

Безумовно, створення такої ГІС можливе за допомогою такого пропрієтарного програмного забезпечення (ПЗ), як ENVI, MapInfo, ARC/INFO тощо. Однак ГІС потребує розширення функціоналу для вирішення інших задач АСУ СХП, які передбачають створення іншого ПЗ та інтеграцію його з ГІС. Тому для побудови ГІС було використано ПЗ з відкритим вихідним кодом: GeoTools, PostGIS, PgRouting (рис. 1).

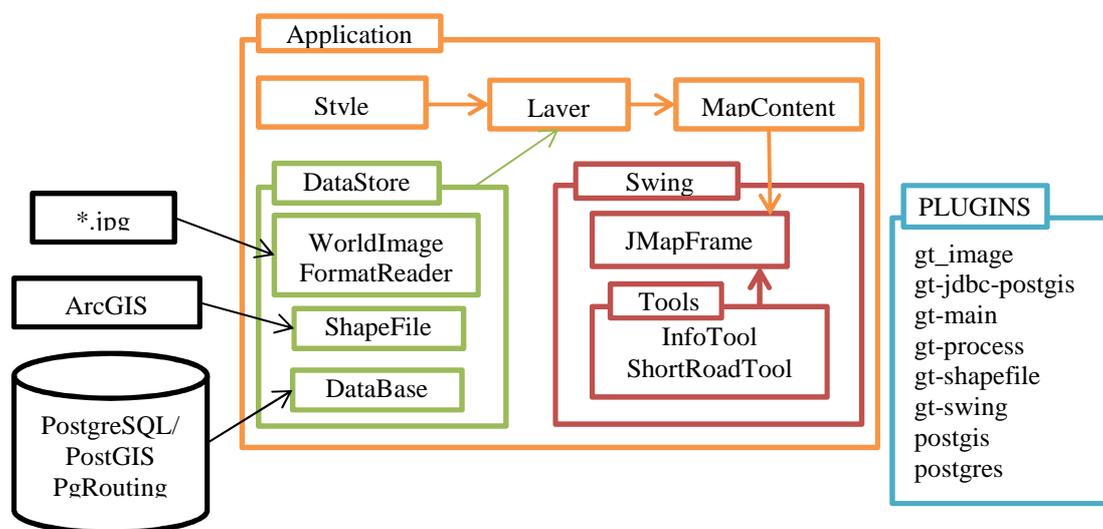


Рис. 1. Архітектура ГІС інфраструктури агрофірми

GeoTools є відкритим вихідним кодом бібліотеки Java, яка забезпечує інструменти для маніпулювання геопросторовими даними для створення ГІС. Бібліотека GeoTools випускається під ліцензією Lesser General Public License (LGPL) і реалізує специфікацію Open Geospatial Consortium (OGC) [4]. При створенні ГІС використовувались об'єкти:

- класів для створення стилів відображення даних (Style);
- класів для завантаження растрових зображень як підкладки (WorldImageFormat, GridReaderLayer);
- класів для завантаження просторових об'єктів з PostGIS / PostgreSQL (DataStore, FeatureSource);
- класів для створення шарів відповідних даних (Layer, MapContent).

Варто зазначити, що вміст пакету swing (InfoTool, JMapFrame) модифікувався під потреби розробки, у тому числі для русифікації об'єктів інтерфейсу. Для реалізації візуального відображення пошуку найкоротшого шляху було створено клас ShortRoadTool. Крім того, було виявлено, що GeoTools має в своєму складі плагіни для вирішення подібних задач, створені різними розробниками, але за єдиним інтерфейсом. Політика GeoTools вимагає використання, що призводить до конфліктів, тому варто використовувати лише один з плагінів, потрібний для вирішення окремої задачі.

PostGIS – розширення об'єктно-реляційної СУБД PostgreSQL з відкритим вихідним кодом, призначене для зберігання та обробки геопросторових даних. PostGIS випущений під ліцензією GNU General Public License (GPL) [5]. При створенні ГІС використовувались:

- спосіб задання координат об'єктів (типи geometry чи geography);
- функції по роботі з відповідними типами;
- функції збереження геопросторових об'єктів із \*.shp в БД для взаємодії з пропріетарним ПО (рис. 2).

PgRouting – розширення геопросторової бази даних PostGIS, що забезпечує геопросторові функції маршрутизації. PgRouting доступний під ліцензією GPLv2 і підтримується постійно зростаючим числом приватних осіб, компаній і організацій [6].

Використовувались наступні функції маршрутизації:

- знаходження координат найближчих геопросторових об'єктів в якості початку і кінця маршруту пошуку;
- знаходження найкоротшого шляху, використовуючи алгоритм пошуку найкоротшого шляху за алгоритмом Дейкстри.

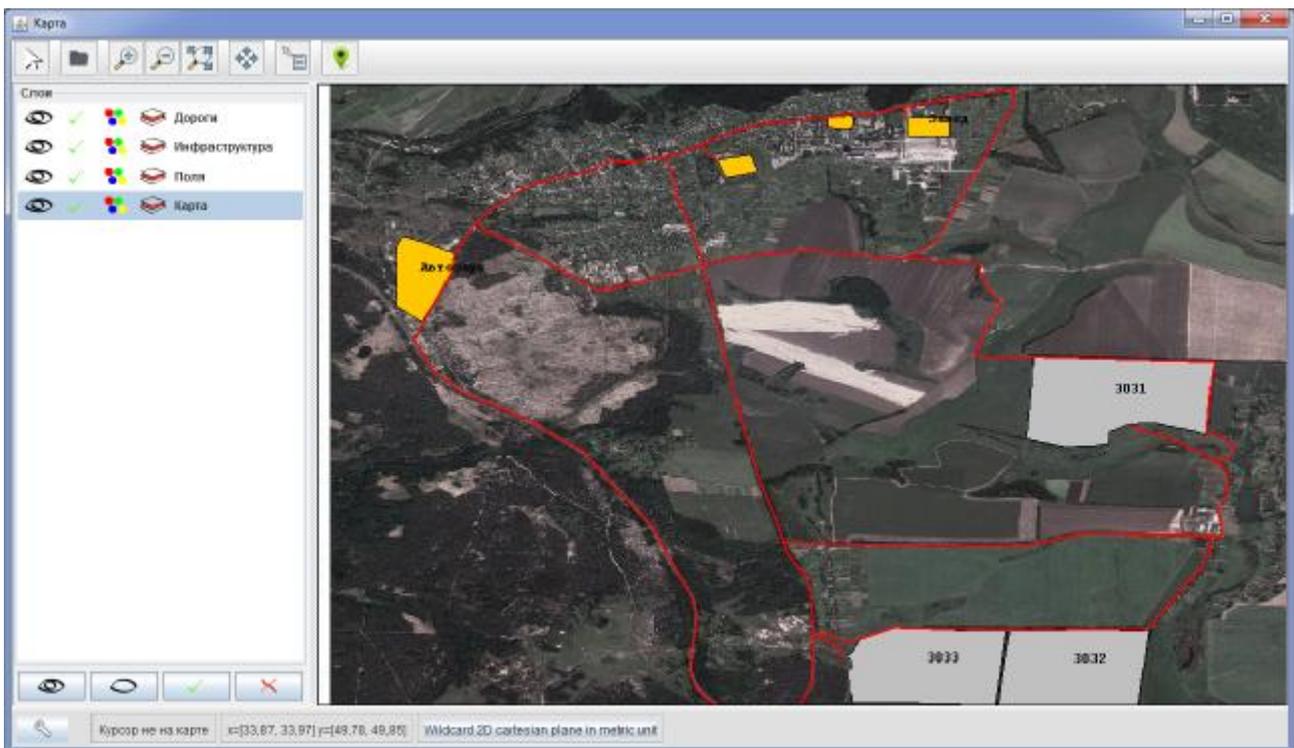


Рис. 2. Інтерфейс ГІС інфраструктури агрофірми

Приведені концепції розкривають основні особливості застосування відкритого ПЗ для побудови власної ГІС і показують шляхи інтеграції такої ГІС з пропріетарним ПЗ та іншим ПЗ для вирішення завдання побудови АСУ СХП.

**Список використаних джерел:** 1. Адамчук В. В. Приоритетні напрямки агроінженерних досліджень / В. В. Адамчук, М. І. Грицишин // Механізація та електрифікація сільського господарства. Вип. 97. Т.1. – Глеваха, 2013. – С. 14-23. 2. Бальченко І. В. Особливості побудови автоматизованої системи управління сільськогосподарським підприємством / І. В. Бальченко, В. В. Литвинов, В. П. Клименко // Математичні машини і системи. – 2013. – № 4. – С. 82-94. 3. Бальченко І. В. Особливості інформаційних технологій управління сільськогосподарським підприємством / І. В. Бальченко, В. В. Литвинов, В. П. Клименко // Вісник Чернігівського державного технологічного університету. – Чернігів : Черніг. держ. технол. ун-т, 2013. – № 3 (67). – С. 211-219. 4. GeoTools The Open Source Java GIS Toolkit [Електронний ресурс]. – Режим доступу : <http://geotools.org>. 5. PgRouting project [Електронний ресурс]. – Режим доступу :

УДК 004.315.2

## ОРГАНІЗАЦІЯ БАГАТОРОЗРЯДНИХ ОПЕРАЦІЙНИХ ПРИСТРОЇВ, ЩО ВИКОНУЮТЬ ДОДАВАННЯ ЗА ЗМІННИМ МОДУЛЕМ

І. Борщов, А. Роговенко

*Чернігівський національний технологічний університет*

До основних галузей сучасних інформаційних технологій, в яких застосовуються модульні операції, належать криптографія, завадостійке кодування і цифрова обробка сигналів. У криптографії особливо важливо забезпечити можливість швидкого виконання модульних операцій для ключів великої розрядності (від 128 біт), тому що відповідно до теорії криптостійкості Клода Шеннона криптографічна стійкість шифру безпосередньо залежить від довжини ключа[1]. Прикладом застосування операції зведення в ступінь по модулю є поширений алгоритм шифрування RSA, який використовує її для шифрування відкритим ключем порції вихідного тексту, рівної розміру ключа. За мірками порівняння з іншими алгоритмами шифрування, RSA володіє високою криптостійкістю, але низькою продуктивністю [2], через це його застосування обмежене шифруванням тестових даних, web-сторінок і цифрових підписів. Для шифрування аудіо/відео потоків його продуктивності недостатньо, в результаті чого для шифрування таких даних, зазвичай, використовують сеансовий ключ RSA в сукупності з менш криптостійкими алгоритмами. Найповільнішою операцією алгоритму є програмно реалізована операція модульного зведення у ступінь через те, що на процесорі вона виконується послідовно для порцій ключа, рівних розміру машинного слова.

Розроблюваний операційний пристрій має виконувати операцію обчислення залишку від ділення суми операндів  $A$  і  $B$  на модуль  $P$ :

$$R = (A + B) \bmod P, \quad A, B, P, R \in \mathbb{N}^{2^n} \quad (1)$$

Ця операція називається *складанням за модулем* або просто *модульним складанням*. При цьому операнди, модуль і результат є натуральними числами розрядності  $n$ .

На основі даної операції можна також виконати ще дві операції: множення за модулем і зведення в ступінь за модулем:

$$R = (A * B) \bmod P, \quad R = (A^B) \bmod P \quad (2)$$

Модульні операції, здебільшого, необхідні у програмах на алгоритмічній мові, що виконуються на процесорі. Тому доцільною буде реалізація операційного пристрою в якості співпроцесора. Перший спосіб реалізації співпроцесора – це реалізація його у вигляді окремої мікросхеми, яка є зовнішньою до цього процесору [3]. Недоліком такого підходу є відсутність сумісності між процесорами, знижена надійність системи, а також підвищена трудомісткість розробки, пов'язана зі складністю розробки друкованої плати та налагодження системи.

Другий спосіб, обраний у нашій роботі, полягає в реалізації співпроцесора на одному кристалі з процесором у рамках програмованої логічної інтегральної схеми. В якості платформи для реалізації було обрано ПЛІС Xilinx сімейства Spartan6 та процесорне софт-ядро Microblaze. Для підключення співпроцесора до софт-ядра передбачено спеціальний односпрямований інтерфейс FSL (прискорений симплексний лінк), призначений для миттєвого обміну даними з співпроцесором. Загальна периферійна шина (така як PLB або AXI) не підходить для підключення модульного співпроцесора: вона може бути зайнята іншими периферійними пристроями, що призведе до недопустимої затримки для даних співпроцесора.

Реалізація операційного пристрою була виконана двома методами: з використанням стандартного оператора `mod` мови `vhdl` і з використанням каскаду конструктивних модулів [4].

Для оцінки кількості ресурсів був виконаний синтез операційного пристрою для десяти значень розрядностей з кроком в 16 біт. Як наслідок над результатом була виконана однофакторна парна регресія (рис. 1, *зліва*). Перша розглянута характеристика – кількість LUT. LUT являє собою 4-и бітну таблицю, відповідно кожному рядку якої ставиться один біт вихідної функції (для сімейства Spartan 6 оцінка виконується в 6-и бітних LUT, які складаються з 4-и бітних). Для оцінки кінцевого обсягу ресурсів вимірюється число слайсів, регресійні моделі якого наведені на рис. 1, *справа*. Слайс складається з декількох LUT-ів, тригерів та мультиплексорів. З графіка видно, що процент використання останніх двох видів ресурсів несуттєво впливає на загальний об'єм ПЛІС, тобто основний об'єм займає комбінаційна логіка.

Як видно з графіків, для операції `mod` на VHDL була отримана нелінійна ступенева залежність, а для операції, виконуваної на ККМ – лінійна залежність. Коефіцієнт детермінації обраних моделей близький до одиниці, що свідчить про правильність обраних регресійних моделей. Помилка апроксимації складає менше 7%. Це свідчить про можливість застосування даних моделей для оцінки ресурсів, які будуть використовувати операційні пристрої різних розрядностей.

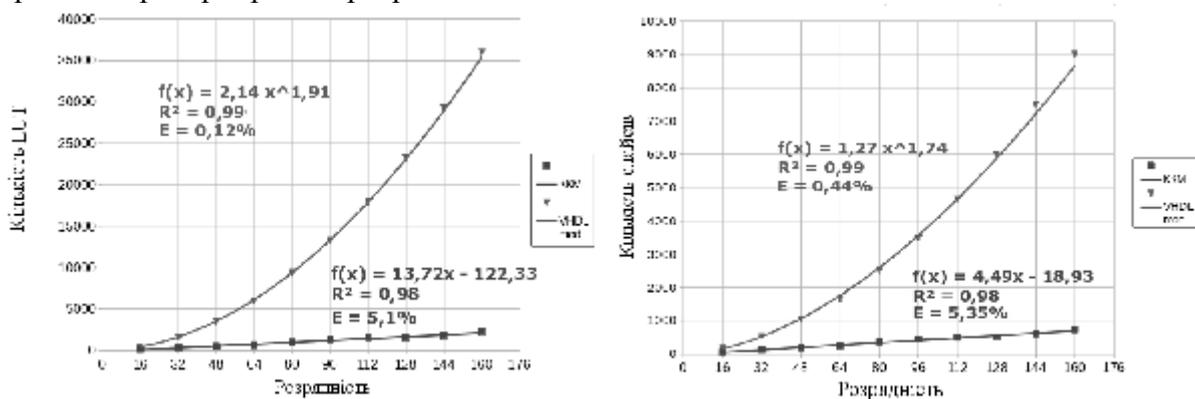


Рис. 1. Залежність числа LUT (зліва) та числа слайсів (справа) від розрядності

Ще однією важливою характеристикою для операційного пристрою є продуктивність. Оскільки продуктивність є кількістю правильно виконаних операцій за одиницю часу, для її визначення можна вимірювати час виконання однієї операції і знаходити величину, зворотну до нього. Затримка обчислення викликана часом перемикання логічних елементів у ПЛІС, яка для випадкових вхідних операндів є випадковою величиною. Для оцінки продуктивності була згенерована тестова вибірка, причому, значення операндів розбиті на три класи, по три варіанти в кожному класі. Класи відповідають різним співвідношенням суми та модуля (менше, більше та рівні), в кожному класі обирається випадкове значення, просте число, та число, що на одиницю менше ступеня двійки. Вимірювання часу виконання операції проводилося при моделюванні з урахуванням тимчасових затримок. На кожній точці виконувався автоматизований контроль правильності результату. За результатами вимірювання дев'яти значень тимчасової затримки було взято максимальне. Його залежність від розрядності відображено на рис. 2.

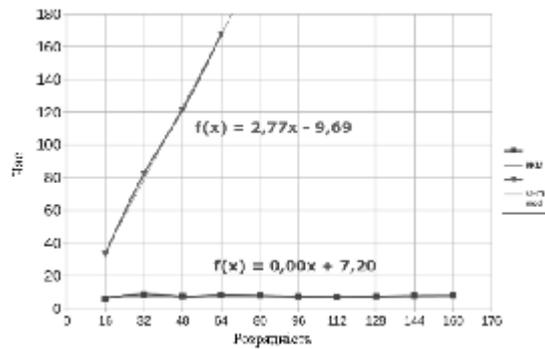


Рис. 2. Залежність часу (в нс) виконання операції від розрядності

Коефіцієнти детермінації і помилок у цьому випадку не розглядалися, тому що затримка має великий розкид, викликаний тим, що дані були обрані випадковим чином. Як видно з графіка, функція тимчасової затримки для ККМ є константою і не залежить від розрядності.

Таким чином, застосування ККМ значно скорочує апаратні витрати на реалізацію модульного операційного пристрою (для 256 біт кількість слайсів зменшується в 17 разів). Продуктивність операційного пристрою, побудованого на базі ККМ, не залежить від розрядності і складає біля 140 MIPS. При цьому для 256-розрядного пристрою вона на два порядки перевищує продуктивність операції mod мови VHDL, яка складає біля 1,4 MIPS. Слід зазначити, що наведені значення швидкодії справедливі лише для безпосередньо операційного пристрою, при урахуванні часу завантаження операндів та вивантаження результату через інтерфейс співпроцесора сумарна швидкодія буде менша.

**Список використаних джерел:** 1. Шеннон К. Теория связи в секретных системах : пер. С. Карпова / К. Шеннон // Работы по теории информации и кибернетике. – М. : ИЛ, 1963. – С. 243-322. 2. B. Padmavathi, S. Ranjitha Kumari – A Survey on Performance Analysis of DES, AES and RSA Algorithm along with LSB Substitution Technique – IJSR, Volume 2 Issue 4, April 2013. – P. 170-174. 3. Nicolas Guillermin – A coprocessor for secure and high speed modular arithmetic – DGA Information Superiority, Bruz, France, IRMAR, Universite Rennes 1, France – 2011. 4. Тарасенко В. П. Створення параметричних ядер (softcores) для виконання операцій в скінчених полях / В. П. Тарасенко, О. К. Тесленко, А. І. Роговенко // Радіоелектронні і компютерні системи. – 2008. – № 6. – С. 261-264.

УДК 004.3

## ОСОБЕННОСТИ ОПРЕДЕЛЕНИЯ МЕТЕОРОЛОГИЧЕСКИХ УСЛОВИЙ В ЗАКРЫТОМ ПОМЕЩЕНИИ С ПОМОЩЬЮ АППАРАТНОЙ ПЛАТФОРМЫ ARDUINO

А. Веремеенко, Д. Ульченко

Черниговский национальный технологический университет

Метеорологические условия на рабочем месте в производственных помещениях и на открытых рабочих площадках определяются температурой воздуха, относительной влажностью, скоростью движения воздуха, барометрическим давлением и интенсивностью теплового излучения от нагретых поверхностей. Совокупность этих параметров, характерных для конкретного производственного участка, называется производственным микроклиматом.

Ныне популярным есть использование домашних метеостанций для определения микроклимата в закрытом помещении.

На рынке производителями предоставляется большое разнообразие устройств для определения микроклимата в закрытом помещении и на рабочем месте. Не всегда перечень

функциональных возможностей устраивает пользователя. Есть возможность разработать системы, которые устраивают пользователей по функционалу и имеют небольшую себестоимость.

Устройство для определения метеорологических условий можно сконструировать на базе Open Hardware технологий, а именно Arduino. Arduino – аппаратная вычислительная платформа, основными компонентами которой являются простая плата ввода-вывода и среда разработки на языке Processing/Wiring.

Микроклимат закрытого помещения определяется основными физическими параметрами: температурой, влажностью и скоростью движения воздуха, температурой окружающих поверхностей. Для примера возьмем простейшую метеостанцию, функционал которой ограничивается снятием показаний нескольких датчиков температуры и влажности и выводом их на дисплей. Схема подключения – рис.

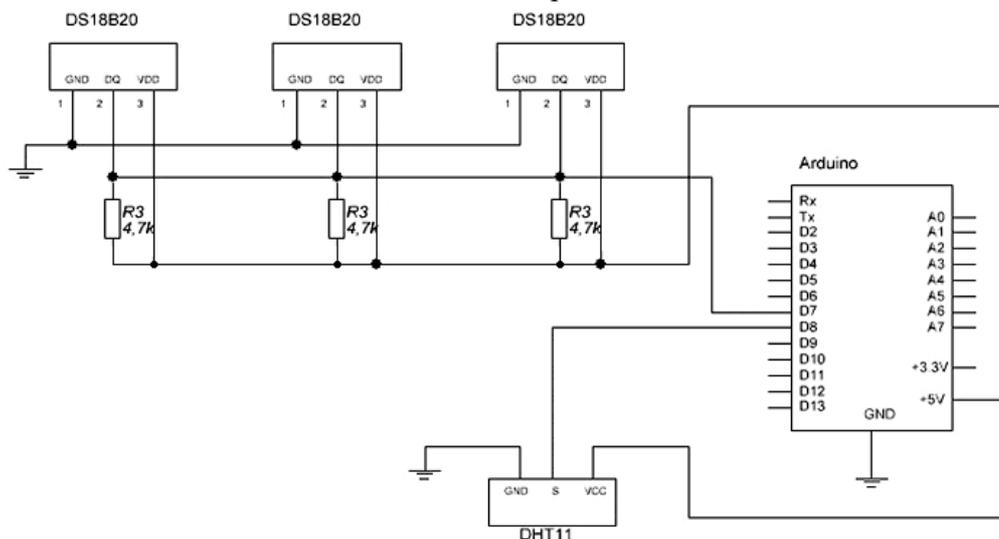


Рис. Схема подключения датчиков к Arduino

*Температура окружающей среды* – это основной погодно-климатический фактор, который влияет на жизнедеятельность организма человека. Температура на рабочем месте не должна выходить с заданного диапазона (для офиса – 20-29 градусов). Для измерения температуры используем три цифровых термометра DS18B20, которые должны находиться в разных частях дома и один за окном. Все DS18B20 подключаются через шину 1-Wire. 1-Wire – двунаправленная шина связи для устройств с низкоскоростной передачей данных (обычно 15,4 Кбит/с, максимум 125 Кбит/с в режиме overdrive), в которой данные передаются по цепи питания (то есть всего используются два провода – один для заземления, а второй – для питания и данных; в некоторых случаях, как с DS18B20, используют и отдельный провод питания). Для устройств с шиной 1-Wire в открытом доступе есть библиотека для Arduino. Таким образом все три термометра подключаются к одному выводу на Arduino, что позволяет использовать оставшиеся выводы для подключения другой периферии. Термометры получают адреса, начиная с 0, доступ к каждому термометру можно получить при помощи функций с библиотеки OneWire.

*Влажность воздуха.* Водяные пары всегда в том или ином количестве содержатся в воздухе, увлажняя его. Приняты следующие понятия при оценке влажности. Относительная влажность — отношение в % абсолютной влажности к максимальной влажности при данных температурных условиях. Относительная влажность принята как показатель в санитарных нормах. Для рабочего места относительная влажность должна находиться в пределах 40-60%. Для измерения относительной влажности используем датчик температуры и влажности DHT11, работа с которым производится при помощи готовых библиотек для Arduino, которые находятся в открытом доступе.

Вывод всех данных производим в командную строку через COM порт, используя функции самого Arduino. Все библиотеки, необходимые для работы устройств, можно найти на официальном сайте Arduino.

**Список использованных источников:** 1. *Жидецький В. Ц.* Основи охорони праці / В. Ц. Жидецький. – Л. : Афіша, 2005. – 349 с. 2. ГОСТ 12.0.005-84 ССБТ. Метрологическое обеспечение в области безопасности труда. Основные положения. – Введ. в действие 01.07.1985. 3. *Bionic Arduino* – Introduction to Microcontrollers with [Электронный ресурс]. – Режим доступа : <http://todbot.com/blog/bionicarduino>.

УДК 004.63

## ВОЗМОЖНОСТИ БИБЛИОТЕКИ ITEXT ДЛЯ РАБОТЫ С PDF-ФАЙЛАМИ

М. Гаценко, С. Стасюк

*Черниговский национальный технологический университет*

Во многих приложениях требуется динамически создавать PDF-документы. Это могут быть доставляемые по электронной почте отчеты для клиентов банка или отдельные главы книг, приобретаемые в формате PDF. Для реализации этой задачи существует несколько программных решений.

Одним из них является сводобно доступная Java библиотека iText, размещенная на <http://itextpdf.com/>. Данная библиотека доступна по лицензии AGPL.

Кроме PDF, библиотека iText поддерживает документы в форматах HTML, RTF и XML. Структура iText позволяет создавать любой из вышеупомянутых типов документов с помощью одного и того же кода.

Помимо создания файлов у этой библиотеки есть такие возможности:

- использование PDF в web, а именно передача их браузеру;
- создание файлов динамически из XML и баз данных;
- использование интерактивных возможностей PDF;
- добавление ссылок, номеров страниц, "водяных знаков";
- разрезание, соединение и другое манипулирование страницами PDF-документов;
- добавление цифровой подписи к файлу PDF.

Основным классом для создания PDF-документов является `com.itextpdf.text.Document`. Это первый класс, экземпляр которого нужно создать. Когда документ создан, потребуется редактор для его заполнения. Это класс `com.itextpdf.text.pdf.PdfWriter`. Также существуют классы для создания параграфов, списков, таблиц, изображений, ссылок и класс, который содержит все спецификации шрифтов, такие как начертание, размер, стиль и цвет.

Кроме создания файла с нуля есть возможность конвертации xml-файлов в PDF с помощью `XMLWorker`. Можно также конвертировать html файлы со стилями, описанными в css, с помощью `HTMLWorker`.

Часто itext применяется в случаях, когда необходимо реализовать печать документа на основе данных, введенных пользователем. При этом используется готовый бланк с отведенными ячейками. `Class PdfFormField` позволяет использовать разные типы полей ввода.

Библиотека itext имеет ряд возможностей для работы с графикой. Например, существует возможность напрямую передать `Graphics2D` объект в itext и созданный pdf файл будет иметь такое же качество, как и масштабируемая векторная графика.

Одним из важнейших свойств деловых документов является его подлинность, так как он может содержать важную информацию о правах и обязанностях сторон. В настоящее время к цифровым документам применяются электронные подписи, которые гарантируют:

- целостность документа - документ не был изменен в рабочем процессе;
- подлинность документа - автором документа является действительно тот, за кого себя выдает;
- неотказуемость - автор не может отказаться от авторства.

Если речь идет о PDF-документах, то библиотека iText может стать удобным решением для организации их защиты с помощью электронной цифровой подписи.

В iText электронные подписи строятся на основе алгоритмов асимметричного шифрования. Данная библиотека поддерживает 3 алгоритма:

- RSA (криптографический алгоритм с открытым ключом, основывающийся на вычислительной сложности задачи факторизации больших целых чисел);
- DSA (алгоритм с использованием открытого ключа для создания электронной подписи, но не для шифрования, в отличие от RSA);
- ECDSA (алгоритм с открытым ключом для создания цифровой подписи, аналогичный, по своему строению, DSA, но определённый, в отличие от него, не над полем целых чисел, а в группе точек эллиптической кривой).

С помощью библиотеки iText можно наложить подпись на основании сертификата открытого ключа. Для этого нужно получить экземпляр класса PdfSignatureAppearance, отвечающий за свойства подписи. Например, меняя значение свойства certificationLevel данного класса, можно разрешить или запретить возможность многократного подписывания документа. Установив свойство image класса PdfSignatureAppearance, к подписи добавляется рисунок, а установив новое значение для свойства runDirection, направление текста может стать обратным.

Есть возможность добавить к существующему PDF-файлу специальное поле для цифровой подписи, для чего используется метод createSignature класса PdfFormField.

Для непосредственного добавления подписи к файлу используется метод signDetached класса MakeSignature.

Библиотека iText является удобным инструментом, который позволяет как создавать новые PDF-документы с нуля, так и редактировать существующие. Она предоставляет удобный интерфейс для создания даже самых структурно сложных документов. Более того, данная библиотека позволяет защитить с помощью электронной цифровой подписи PDF-файл, что является важным преимуществом в современном мире, где доля электронного документооборота становится все больше.

УДК 004.934

## РАСПОЗНАВАНИЕ РЕЧИ (SIMON)

А. Дорошенко, Д. Мельниченко

*Черниговский национальный технологический университет*

### **Признаки классификации систем распознавания речи:**

- размер словаря (чем больше размер словаря, который заложен в систему распознавания, тем больше частота ошибок при распознавании слов системой);
- дикторозависимость или дикторонезависимость системы.
- раздельная или слитная речь;
- назначение (назначение системы определяет требуемый уровень абстракции, на котором будет происходить распознавание произнесенной речи).

### **Различия методов распознавания речи:**

1. По типу структурной единицы (при анализе речи в качестве базовой единицы могут быть выбраны отдельные слова или части произнесенных слов).
2. По выделению признаков (для представления речевого сигнала из него требуется выделить какие-либо параметры, адекватно представляющие этот сигнал для распознавания).
3. По механизму функционирования (вероятностно-сетевой подход, и подход, основанный на решении обратной задачи синтеза звука).

### **Структура стандартной системы распознавания речи:**

1. Необработанная речь.
2. Анализ сигнала.

3. Речевые кадры.
4. Акустические модели.
5. Акустический анализ.
6. Последовательность слов.

### **Sphinx**

1. Sphinx ([cmusphinx.sourceforge.net](http://cmusphinx.sourceforge.net)).
2. Julius (Альтернатива Sphinx японского происхождения). По качеству распознавания Julius не уступает Sphinx, однако имеет два изъяна.
3. Отличная графическая оболочка для Julius - программа Simon, написанная с использованием библиотеки Qt4.!

### **SIMON**

1. Simon – система распознавания речи, позволяющая организовать управление десктопом при помощи голосовых команд, без необходимости использования клавиатуры и мыши (open source speech recognition solution).
2. В составе проекта развиваются инструменты для создания голосовых и акустических моделей, распознавания речи и организации управления голосом.

### **Какие возможности предоставляет SIMON?**

- Можно слушать музыку, смотреть лайд-шоу, телевизор или видео или слушать радио только с помощью таких слов, как "вправо", "влево", "вверх", "вниз", "в порядке", "остановка" и т.д.
- Можно каждый день читать газеты, без проблем заходить в Интернет, просто называя адреса сайтов и цифры.
- Можно диктовать адрес твоей почты и отвечать на все письма (с помощью базовых вопросов).
- Можно общаться в скайпе.
- Руководить мышкой и считать на калькуляторе.

### **Процесс распознавания**

- Процесс распознавания в Simon построен на статических алгоритмах – для начала поддержки нового языка достаточно сформировать акустическую и языковую модели, которые в совокупности составляют речевую модель.
- Если речевая модель уже создана, для начала распознавания команд достаточно пятиминутной тренировки для адаптации системы к особенностям речи конкретного пользователя.

### **Новости**

1. В апреле этого года вышел новый релиз свободной системы распознавания речи SIMON 0.4.
2. Проведена большая работа по улучшению юзабилити. Интерфейс стал более удобен и понятен, как для новичков, так и для давно знакомых с системой пользователей. Проведена реорганизация основного окна управления Simon, наиболее важные опции вынесены на основной базовый экран;
3. В апреле этого года вышел новый релиз свободной системы распознавания речи SIMON 0.4.
4. Проведена большая работа по улучшению юзабилити. Интерфейс стал более удобен и понятен, как для новичков, так и для давно знакомых с системой пользователей. Проведена реорганизация основного окна управления Simon, наиболее важные опции вынесены на основной базовый экран.

**Список использованных источников:** 1. Релиз свободной системы распознавания речи Simon 0.4 [Электронный ресурс]. – Режим доступа : <http://www.opennet.ru/opennews/art.shtml?num=35731>. 2. Распознавание речи. Часть 1. Классификация систем распознавания речи [Электронный ресурс]. – Режим доступа : <http://habrahabr.ru/post/64572/>. 3. Simon listens [Электронный ресурс]. – Режим доступа : <http://simon-listens.org>. 4. Julius [Электронный ресурс]. – Режим доступа : [http://en.wikipedia.org/wiki/Julius\\_\(software\)](http://en.wikipedia.org/wiki/Julius_(software)). 5. Sphinx [Электронный ресурс]. – Режим доступа : <http://sphinxsearch.com>.

## ИГРОВАЯ ПЛАТФОРМА ARDUINO HACKVISION KIT

Я. Иванова, А. Роговенко

*Черниговский национальный технологический университет*

Ардуино и Ардуино-совместимые платы спроектированы таким образом, чтобы их можно было при необходимости расширять, добавляя в схему устройства новые компоненты. Эти платы расширений подключаются к Ардуино посредством установленных на них штыревых разъёмов.

Существует множество разных по функциональности плат расширения – от простейших, предназначенных для макетирования (прототипирования), до сложных. С помощью различных плат расширения может быть создано огромное количество устройств разного назначения. В данном докладе будет представлено описание одной из плат расширения – игровой платформы Arduino Hackvision Kit.

Особенностью Arduino Hackvision Kit является то, что это простая игровая платформа на базе микроконтроллера ATmega328. Основана на технологии Arduino, так что есть возможность написать свои собственные игры и загружать их, используя Arduino IDE.

Цель Hackvision состоит в том, чтобы позволить пользователю создавать в ретро-стиле аркадные игры, которые могут быть воспроизведены на монитор или телевизор с аналогового видеовхода. Хотя разрешение экрана только 128 на 96 пикселей, этого достаточно, чтобы получить некоторые интересные действия.

Микроконтроллер предварительно загружен в двух играх, так что вы можете начать играть, как только строительство завершено. Однако вам понадобится кабель 5V, если вы хотите загрузить новые игры, поскольку плата не имеет интерфейса USB.

Все, что нужно, это USB в TTL последовательный кабель или адаптер. Плата уже обладает "встроенным геймпадом", набором интерфейсов к разным внешним джойстикам и умеет формировать видеоизображения формата PAL или NTSC. Подключается непосредственно к телевизору с помощью стандартного подключения RCA.

Сборка является быстрой и относительно простой, инструкции есть в Интернете, им легко следовать. Все необходимое включено в комплект платы.

Файлы, хранящиеся в памяти EEPROM, сохраняются даже при отключенном питании. Комплект может быть собран за 30-45 мин. Полностью собранные и проверенные устройства также доступны.

Для более интересной работы с Hackvision есть возможность присоединить свои собственные ручные элементы управления, такие как paddles, Nunchuks и т. д. Дополнительные аксессуары доступны в интернет-магазинах, в том числе 9В адаптеры, RCA кабели (используются для видеосигнала и стереофонического звука. Жёлтый – для видеосигнала, белый – для монофонического сигнала или левого канала стереофонического двухканального аудиосигнала, красный — для правого канала стереофонического двухканального аудиосигнала.), комплекты контроллеров paddle (игровой контроллер с круглым колесом и одной или более кнопками, где колесо обычно используется для управления движением объекта игрока вдоль одной оси видеоскрена. Контроллер вращается по дуге (обычно около 330 градусов); он имеет остановку на каждом конце), USB – серийные адаптеры и т.д.

Технические характеристики:

- ATmega 328 микроконтроллер;
- Монохромное композитное видео;
- Разрешение – 136x96 пикселей (Вы можете управлять этим в собственных играх);
- Требуется 9В источника питания.

Предустановленные игры: Space Invaders и Pong. Поддержка контроллеров: Wii Nunchuk, SuperNES. Программное обеспечение библиотеки для разработки игр и поддержка контроллера.

Рассмотрим более подробное подключение выводов платы.

1) Подключение контроллеров Paddle к плате Arduino Hackvision Kit. Первым шагом является подключение к разъему, отмеченному как "PADDLE A" или "PADDLE B" 4-х контактного разъема. Его необходимо удерживать на месте так, чтобы он был расположен вертикально и стабильно. Далее необходимо перевернуть плату и припаять эти четыре контакта. Следующим шагом является подключение провода к разъему на плате. Провод для кнопки идет на цифровой вход контактом, обозначенным D0 для PADDLE A, и D1 для PADDLE B. Аналоговый выход идет в аналоговый вход, отмеченный A3 для PADDLE A и A2 для PADDLE B.

2) Выводы, используемые встроенным кнопочным контроллером, разделены, что дает возможность использовать их для другого типа контроллера, который вы можете изобрести.

3) Программировать и загружать свои собственные игры на ваш Hackvision можно помощью Arduino IDE. Разъем на верхней части платы позволяет подключить USB для TTL последовательного кабеля или адаптера.

4) Настройка Hackvision для стандарта PAL TV, используемого в Европе, Африке, Азии, Южной Америке: с помощью пайки провода переключки на PAL контакты Hackvision может использовать видео-тайминги для стандарта PAL.

Программирование и загрузка своих программ также происходит довольно просто. Недостатком Hackvision является то, что не хватает документации по программированию данного устройства, хотя есть некоторые очень хорошие реализации игры в сети Интернет.

Для управления видеовыходом нужно скачать и установить Hackvision – версии Arduino – TVout библиотеку. Обратите внимание, что эта библиотека немного отличается от общей библиотеки Arduino – TVout с относительным определением функций и параметров. Чтобы использовать прилагаемые кнопки проще, есть также библиотека-контроллеров. Общий подход в том, что вы установите код игры в вашем Arduino скетче в папку и установите библиотеки TVout и контроллеры в Arduino – папки библиотек.

Библиотека TVout используется для рисования на экране. Текст, линии, отдельные пиксели и т.д. можно сделать на экране, используя простые примитивы рисования. Библиотека полностью документирована на сайте TVout Google Code. Также можно посмотреть на Hackvision прошивки предустановленной игры, чтобы увидеть, как библиотека используется для рисования на экране.

Для инициализации TVout необходимо указать разрешение и вероятность использования стандарта NTSC (в США, Канаде, Японии, некоторых частях Южной Америки) или стандарта PAL (Европа, Австралия, Африка, Азия, некоторые части Южной Америки). Arduino – цифровой вывод 12 соединен с PAL переключки на плате Hackvision и может быть использован для инициализации TVout правильно в настройках из вашей игровой эскизы.

Использование библиотеки-контроллеры. Библиотека-контроллеры обеспечивают поддержку контроллеров-кнопки, Wii Nunchuk, весла и SuperNES контроллеров. реализуются классом ButtonController. Каждая кнопка имеет метод, чтобы проверить, нажата ли она.

Paddle контроллеры, которые имеют кнопку и потенциометр, могут быть легко добавлены к вашему Hackvision. Библиотека-контроллеры включает в себя класс Paddle, что делает его удобным для чтения входного сигнала от контроллера.

*Советы и рекомендации для разработки игр для экономии памяти.*

Память, которая хранит то, что рисуется на экране ("буфера кадра"), потребляет много SRAM-памяти. Не путайте SRAM с флэш-памятью, где хранится ваша программа. Есть 32К флэш-памяти для хранения вашего игрового кода, и это довольно много. Но есть

только 2К из SRAM в чип ATmega328 и с разрешением 128x96 точек. TVout использует 1,5К при этом. Предустановленные игры на Hackvision используют немного более высокое разрешение (136x96 пикселей), так что это потребляет 1632 байт.

Для чего нам нужно SRAM? Для хранения переменных, которые вы объявляете в вашей игре, в стеке программы (где хранятся локальные переменные функции), числовых констант и строковых констант. Если вы не следите за использованием памяти, они могут использовать всю память.

1) Не используйте высокое разрешение. Лучше всего использовать разрешение с 128x96 или 136x96.

2) Используйте только самый большой тип данных для переменных.

3) Не позволяйте вашему стеку получать вызовы слишком глубоко. То есть, не пишите код, который вызывает 10 или 20 функций.

4) Текст, который имеет тип данных String, объявляйте во флэш-памяти.

5) С помощью специального модификатора Arduino PROGMEM вы можете хранить константы во флэш-памяти.

*Сохранение счета в играх в EEPROM.*

Микроконтроллер ATmega328 имеет 1К EEPROM, который сохранит данные даже при отключенном электропитании. Предустановленная игра Space Invaders использует EEPROM для хранения и извлечения достижений в игре. Каждый файл содержит записи о 10 наивысших баллах, и каждая запись состоит из 5 байт команды: 3 байта для инициалов игроков, и 2 байта для счета. Метод enterHighScore () в EEPROM нужен для того, чтобы увидеть, если ваш счет в конце игры достаточно высок, чтобы позволить вам ввести свои инициалы и хранить его. Метод enterInitials () позволяет вводить инициалы игроков. Метод displayHighScores () показывает их на экране.

Игровая платформа Hackvision имеет большой потенциал и является интересным продолжением Arduino. Это еще один способ для отправки данных на экраны мониторов и телевизоров, и возможность играть в самостоятельно написанные ретро-игры.

УДК 004.312

## **ІНФОРМАЦІЙНА СИСТЕМА ЕЛЕКТРОННОГО ГОЛОСУВАННЯ MOBILE-RADA**

**В. Казимир, Є. Нікітенко, Р. Заровський, Д. Єрмак, А. Камко**  
*Чернігівський національний технологічний університет*

Метою створення системи електронного голосування є автоматизація процесу проведення сесій рад народних депутатів за допомогою планшетних комп'ютерів, що значно полегшує процес голосування, підрахунку голосів та оприлюднення результатів голосування у сесійній залі.

Система електронного голосування на основі бездротової мережі представляє програмне забезпечення для автоматизації процесу проведення сесій рад народних депутатів за допомогою планшетних комп'ютерів. Ця система призначена для вирішення таких задач:

- автоматизація процесу голосування депутатів під час проведення сесій;
- фіксація всіх автоматизованих процедурних дій, їх відображення та документування;

- формування електронного архіву процедурних дій.

Система забезпечує виконання таких функцій:

- реєстрацію та ідентифікацію депутатів;
- проведення голосування;
- проведення поіменного голосування;

- відображення ходу і результатів реєстрації й голосування на інформаційних табло, моніторах президії та дисплеях пристроїв голосування;
- збереження протоколу процедурних дій у базі даних сервера у реальному часі;
- проведення автоматичної діагностики зв'язку головного сервера з клієнтськими пристроями;
- підтримка функції резервного сервера.

Особливості системи електронного голосування:

- дискретно-подійна система;
- підтримка великої кількості з'єднань;
- система розроблена за допомогою мови програмування JAVA з використанням технологій: ORM (JPA), EJB, Rest Services (JAX-RS), HTTP та ін.;
- реалізація системи: серверна частина – платформонезалежна, клієнтська – ОС планшетних комп'ютерів.

У процесі розробки системи голосування було використано таке вільне програмне забезпечення з відкритим кодом:

- серверна ОС CentOS [1];
- сервер застосувань Glassfish 3.1;
- клієнтська ОС Android 4.0.3 [2].

Для того, щоб можна було голосувати, необхідно авторизуватися в системі та залишатися в ній. Як тільки почнеться голосування, клієнтська програма сама відобразить вікно для голосування та відправить результат на сервер в автоматичному режимі. Далі з'явиться вікно з результатами голосування. Після цього процес голосування для поточного питання закінчується і клієнтська програма переходить у режим очікування наступного опитування.

Клієнтська програма також може відображати порядок денний поточної сесії. Для його перегляду необхідно авторизуватися та натиснути на кнопку «Порядок денний» , після цього з'явиться список поточних питань. Для того, щоб відкрити документи для питання, яке нас цікавить, необхідно натиснути на це питання. Після цього клієнтська програма перейде у вікно вибору документів, в якому буде відображено список документів стосовно цього питання. Варто зазначити, що самі документи завантажуються відразу після авторизації з локального файлового сервера, тому їх достовірність гарантована.

Також система підтримує функцію блокування роботи планшетного комп'ютера, яка передбачає блокування клієнтської програми без розірвання зв'язку з сервером. Для активування цієї функції необхідно натиснути кнопку «Блокування» , після цього клієнтська програма перейде в режим очікування та відобразить вікно для вводу пароля. Для того, щоб розблокувати програму, необхідно ввести пароль та натиснути на кнопку «Ок». Після цього програма переходить в режим очікування голосування.

Для зміни паролю користувача необхідно натиснути клавішу «Зміна паролю» . Далі необхідно заповнити всі запропоновані поля («Логін», «Старий пароль», «Новий пароль», «Повторіть новий пароль») та натиснути на кнопку «Змінити пароль».

У свою чергу серверне програмне забезпечення розділяється на дві програми: серверна програма та клієнтська програма для сервера.

Клієнтська програма для сервера дозволяє управляти процесом голосування, редагувати порядок денний, використовуючи для цього панель інструментів або випадаючий список меню. За допомогою панелі інструментів можна додавати, видаляти та редагувати питання порядку денного. За допомогою випадаючого списку можна вибирати питання для редагування та видалення. Також дана програма виконує формування зображення на інформаційних табло. За допомогою інструментів вікна можна вивести інформацію про результати голосування. Клієнтська програма для сервера має функцію

відображення таблиці, в якій знаходяться внутрішні параметри планшетних комп'ютерів, до яких належить:

- логін та прізвище, ім'я, по батькові користувача, який працює з планшетним комп'ютером;
- стан заряду акумулятора;
- ір-адреса;
- статус у мережі (у мережі, не в мережі);
- статус у системі (авторизований, неавторизований).

**Список використаних джерел:** 1. *Red Hat Enterprise Linux 4.5.0. System Administration Guide* [Електронний ресурс]. – Режим доступу : <http://www.opencontent.org/openpub/>. 2. *Android 4.3.* [Електронний ресурс]. – Режим доступу: <http://www.android.com/>.

УДК 004.582

## ИСПОЛЬЗОВАНИЕ ИНТЕРАКТИВНОЙ ДОСКИ В ПРОЦЕССЕ ОБУЧЕНИЯ

Н. Кононова

*Славутичский филиал Национального технического университета Украины «КПИ»*

Сегодня общеобразовательная практика большинства развитых стран формируется с учетом следующих приоритетов: гуманизации, индивидуализации, интенсификации, профилизации и ранней специализации, реализация которых на практике предполагает трансформацию учебного процесса, его перевод на субъект-объектную основу [1].

Одним из важных средств повышения качества учебного процесса, успешного усвоения знаний детьми, формирования у них умений и навыков являются информационные средства обучения, которые позволяют не только реализовать принципы наглядности в обучении, но и деятельностный метод в обучении.

Грамотное использование возможностей инновационных средств обучения в образовательном процессе способствует активизации познавательной деятельности и повышению качественной успеваемости школьников; достижению целей обучения с помощью современных электронных учебных материалов, предназначенных для использования на уроках в общеобразовательной школе; развитию навыков самообразования и самоконтроля в учащихся; снижению у них дидактических затруднений, повышению активности и инициативности, а также уровня комфортности обучения на уроке; развитию информационного мышления школьников; формированию информационно-коммуникационной компетенции; приобретению навыков работы на персональном компьютере [2].

Вопросы применения интерактивных методов и средств обучения на уроках в общеобразовательной школе исследовали многие ученые-педагоги: И.Р. Высоцкий, Б.С. Гершунский, А.П. Ершов, Э.И. Кузнецов, А.М. Коротков, В.М. Монахов, Е.С. Полат, И.В. Роберт, А.Ю. Уваров, О.К. Филатов, Н.И. Захарова, Г.К. Селевко, А.А. Ушаков, А.А. Веряев, Т.Н. Лось, И.В. Гиршин и др. [3].

Применение интерактивной доски в процессе обучения позволяет реализовать как совершенно новые, так и годами отработанные педагогические приемы, поскольку соединяет бесспорные преимущества компьютера с достоинствами обычной школьной доски.

Интерактивность – это одно из слагаемых компьютерных технологий. По данным исследований психологов люди запоминают только 20% того, что видят, 30% того, что слышат, 50% того, что видят и слышат и 80% того, что они видят, слышат и делают одновременно. На основе этих данных можно сказать, что применение интерактивных технологий может во много раз улучшить качество знаний учащихся [4].

Программно-аппаратный комплект "Интерактивная доска" – это современное мультимедиа-средство, которое, обладая всеми качествами традиционной школьной доски, имеет более широкие возможности графического комментирования экранных изображений; позволяет контролировать и производить мониторинг работы всех школьников группы одновременно; естественным образом (за счет увеличения потока предъявляемой информации) увеличить учебную нагрузку учащегося; обеспечить эргономичность обучения; создавать новые мотивационные предпосылки к обучению; вести обучение, построенное на диалоге.

Интерактивная доска – это поверхность, на которую проецируется экран монитора, она является чувствительной и позволяет использовать специальную ручку, чтобы рисовать, писать на самой доске и управлять работой компьютера, подключенного к ней, использовать такие средства и ресурсы, как презентации, текстовые программы, электронные программы учебного назначения, изображения, видеофайлы и др.

Существует два основных типа интерактивных досок: твердые и мягкие. Твердые интерактивные доски имеют намагниченную поверхность за экраном и предполагают использование специальных ручек. Мягкие интерактивные доски имеют упругую мембрану, по поверхности которой можно писать и рисовать не только специальной ручкой, но и пальцем.

Используемое программное обеспечение для интерактивной доски (SMART Board Software) включает следующие инструменты: записную книжку (SMART Notebook), средство видеозаписи (SMART Recorder), видеоплеер (SMART Video Player), дополнительные (маркерные) инструменты (Floating Tools), виртуальную клавиатуру (SMART Keyboard). Все эти инструменты могут быть использованы как отдельно, так и в совокупности в зависимости от решаемых учебных задач.

*Записная книжка* представляет собой графический редактор, позволяющий создавать документы собственного формата и включать в себя текст, графические объекты, как созданные в других Windows программах, так и с помощью соответствующих инструментов.

*Средство видеозаписи* позволяет записать в видеофайл формата AVI все манипуляции, производимые в данный момент на доске, а затем воспроизвести его с помощью видеоплеера (SMART Player) или любого другого подобного программного средства.

*Дополнительные (маркерные) инструменты* используются для создания разного рода пометок на всей площади экрана монитора независимо от используемого текущего приложения. Все пометки, делаемые преподавателем, например, в презентации PowerPoint, могут быть сохранены.

*Виртуальная клавиатура* используется для управления компьютером, когда преподаватель находится непосредственно около доски, т.е. дублирует стандартную клавиатуру компьютера [5].

Проектируя будущий урок с интерактивной доской, учитель должен продумать последовательность технологических операций, формы и способы подачи информации на большой экран [7].

Виды воздействия на обучение при использовании интерактивной доски в процессе обучения (табл.) [6].

Таблица

*Виды воздействия интерактивной доски на обучение*

Инструмент интерактивной доски	Воздействие на обучение
Цвет	Разнообразие цветов, доступных на интерактивной доске, позволяет преподавателям выделять важные области и привлекать внимание к ней, связывать общие идеи или показывать их отличие и демонстрировать ход размышления. Примером может быть работа с географической картой или схемой пищеварительной системы организма

Записи на экране	Возможность делать записи позволяет добавлять информацию, вопросы и идеи к тексту, диаграммам или изображениям на экране. Все примечания можно сохранить, еще раз просмотреть или распечатать
Аудио- и видео-вложения	Значительно усиливают подачу материала. Также можно захватывать видеоизображения и отображать их статично, чтобы иметь возможность обсуждать и добавлять к нему записи
Drag & drop	Помогает учащимся группировать идеи, определять достоинства и недостатки, сходства и различия, подписывать карты, рисунки, схемы и многое другое
Выделение отдельных частей экрана	Тест, схему или рисунок на интерактивной доске можно выделить. Это позволяет преподавателям и ученикам фокусироваться на отдельных аспектах темы. Часть экрана можно скрыть и показать его, когда будет нужно. Программное обеспечение для интерактивных досок включает фигуры, которые могут помочь учащимся сконцентрироваться на определенной области экрана. Используя инструмент «прожектор», можно выделить определенные участки экрана и сфокусировать внимание на них
Вырезать и вставить	Объекты можно вырезать и стирать с экрана, копировать и вставлять, действия - отменять или возвращать. Это придает учащимся больше уверенности - они знают, что всегда могут вернуться на шаг назад или изменить что-нибудь
Страницы	Страницы можно листать вперед и назад, демонстрируя определенные темы занятия или повторяя то, что некоторые из учеников не очень поняли. Страницы можно просматривать в любом порядке, а рисунки и тексты перетаскивать с одной страницы на другую
Разделение экрана	Преподаватель может разделить изображение с экрана компьютера и показать его на разных досках. Это может пригодиться при тщательном исследовании предмета
Поворот объекта	Позволяет перемещать объекты, показывая симметрию, углы и отражения
Соединение с электронным микроскопом	Позволяет рассматривать и исследовать микроскопические изображения

На занятии с интерактивной доской вы можете использовать все материалы, доступные на компьютере: таблицы, схемы, анимацию, аудио- и видеозаписи. Разнообразие ресурсов предоставляет широкий выбор подходов к обучению. Использование интерактивной доски на уроке помогает решить такие задачи, как мобилизация психической активности учащихся, введение новизны в учебный процесс, повышение интереса к уроку, увеличение возможности произвольного запоминания материала, выделение в материале главного и его систематизация.

**Список использованных источников:** 1. Соловьева А. Р. Интерактивность в условиях системного подхода к обучению как дидактическое средство достижения его целей : автореф. дис. ... канд. пед. наук / А. Р. Соловьева. – Астана, 2008. 2. Саржанова А. Н. Использование информационно-коммуникативных технологий в начальных классах / А. Н. Саржанова, В. Г. Пустовалова // Начальная школа Казахстан. – 2011. – № 1. 3. Акпаева А. Б. Методика формирования математических понятий у младших школьников : автореф. дис. ... канд. пед. наук / А. Б. Акпаева. – Алматы, 2000. 4. Каримова Я. Г. Инновационные методы преподавания с использованием интерактивной доски и флипчартов как средств мотивации учащихся / Я. Г. Каримова // Творческая

педагогика. – 2011. – № 3. **5.** Смагина И.Н. Использование современных технических средств и их программного обеспечения в учебном процессе / И. Н. Смагина ; Ливенский филиал Госуниверситета-УПНК. – М., 2010. **6.** Кабулова Г. С. Использование интерактивной доски на уроках в начальной школе / Г. С. Кабулова, И. И. Ефимова, С. В. Тороян // Школьные технологии. – 2011. – № 9. **7.** Лось Т. Н. Интерактивная доска на уроках русского языка / Т. Н. Лось // Творческая педагогика. – 2011. – № 3.

УДК 004.9

## **РАЗРАБОТКА ВЕБ-ПОРТАЛА ДЛЯ ИНТЕГРИРОВАННОЙ ИНФОРМАЦИОННОЙ СИСТЕМЫ УЧЕБНО-ПРОИЗВОДСТВЕННОГО ЦЕНТРА ИТ-АУТСОРСИНГА**

Д. Коробко, О. Дрозд

*Черниговский национальный технологический университет*

Черниговский национальный технологический университет является участником проекта Tempus «Инновационная гибридная стратегия ИТ-аутсорсингового партнерства с предприятиями». Координатором проекта от Европейского Союза представлен Технический университет Вильдау (Германия). Координатором проекта от Украины является Харьковский национальный университет радиоэлектроники.

Общей целью проекта является развитие сотрудничества между университетами и предприятиями в Украине с целью повышения ИТ-образования для обеспечения рынка ИТ-аутсорсинговых услуг профессионалами высокой квалификации [1].

Чтобы координировать работу и обеспечить участников проекта Tempus взаимосогласованной, достоверной, точной и своевременной информацией, необходимой для высококачественного планирования, учета, контроля и организации всех основных видов деятельности (бизнес-процессов) Учебно-производственного Центра Аутсорсинга (OSTPC) и его филиалов, возникла необходимость разработки веб-портала. Архитектура данного портала построена на основе CMS (системы управления контентом) Joomla с открытым исходным кодом, а также множества расширений (компонентов, модулей, плагинов) для решения нестандартных задач согласно требованиям проекта Tempus. Для проведения вебинаров используется сервер Apache OpenMeetings с открытым исходным кодом.

Система управления контентом (англ. Content Management System, CMS) – это программный комплекс, используемый для управления содержанием сайта или информационной системой с неструктурированным контентом [2].

Каркас системы управления контентом (англ. Content Management Framework, CMF) – это программный комплекс, используемый для построения CMS с требуемой функциональностью [3].

Joomla – система управления содержимым (CMS), написанная на языках PHP и JavaScript, использующая в качестве хранилища базы данных СУБД MySQL или другие индустриально-стандартные реляционные СУБД. Данная CMS является свободным программным обеспечением, распространяемым под лицензией GNU GPL [4].

Apache OpenMeetings – сервер для проведения аудио- и видеоконференций. Сервер поддерживает ряд дополнительных функций, таких как обмен сообщениями, совместное редактирование документов, создание записи – протокола мероприятия, совместное планирование выполнения задач, трансляция вывода запущенных приложений, проведение голосований и опросов. Исходный код OpenMeetings написан на языке Java. Интерактивный веб-интерфейс построен с использованием фреймворка OpenLaszlo [5].

Joomla позволяет создавать самые разные сайты, при этом сильно сокращая объем разработки и, в конечном счете, повышая качество продукта. Истинные преимущества Joomla заключаются в большей, по сравнению с другими CMS, гибкости и расширяемости, а также в контентно-ориентированном подходе к организации содержания сайта.

#### Преимущества CMS Joomla:

- функциональность можно увеличивать с помощью дополнительных расширений (компонентов, модулей и плагинов);
- имеется модуль безопасности для многоуровневой аутентификации пользователей и администраторов (используется собственный алгоритм аутентификации и «ведения» сессий);
- система шаблонов позволяет легко изменять внешний вид сайта или создать свой уникальный;
- предусмотрены настраиваемые схемы расположения модулей, включая левый, правый, центральный и любое другое произвольное положения блока. При желании содержимое модуля можно включить в содержимое материала;
- начиная с версии 1.6 встроена многоязычность;
- начиная с версии 2.5 расширена поддержка баз данных. Реализована поддержка Microsoft SQL Server, а с версии 3.0 – PostgreSQL.

#### Этапы работы системы:

- пользователь осуществляет какое-либо действие на портале через веб-браузер, который посылает некоторый запрос веб-серверу;
- веб-сервер определяет, кому предназначен запрос и передает его ядру разработанной системы веб-портала;
- ядро системы проверяет поступивший запрос на корректность и выполнение требований безопасности;
- если вышеуказанные требования соблюдены, начинается сбор запрошенной информации ядром системы, в роли которой могут быть:
  - 1) некоторые поля из базы данных;
  - 2) графические и мультимедиа файлы;
  - 3) выполнение внешних функций (программ, модулей).
- ядро создает конечную HTML-страницу и отправляет ее веб-браузеру пользователя.

#### Функциональность разработанного веб-портала (webinars.stu.cn.ua):

- предоставляет регистрацию и аутентификацию пользователей;
- обеспечивает возможность создания онлайн-мероприятий в календаре событий;
- разработанный компонент OpenMeetings позволяет создавать, удалять и редактировать настройки комнат для вебинаров в Apache OpenMeetings;
- реализует возможность отправки уведомлений на e-mail соответствующим участникам онлайн-мероприятий с указанной ссылкой на комнату для вебинаров;
- предоставляет возможность создания ленты новостей на сайте и размещения объявлений о соответствующих онлайн-мероприятиях;
- дает возможность пользователям оставить комментарий в книге отзывов, который предварительно будет проверен администратором сайта и, в случае корректности отзыва, опубликован;
- реализует проверки, которые запрещают создание онлайн-мероприятий в одной и той же комнате Openmeetings в одно и то же время;
- реализует плагин, который запрещает доступ к комнате вебинаров пользователей, которые не получили приглашение на участие в онлайн-мероприятии.

В целом можно сделать вывод, что система управления контентом Joomla является на сегодня одним из наиболее эффективных средств реализации достаточно сложных и функционально насыщенных веб-ресурсов.

Разработка данного веб-портала webinars.stu.cn.ua является актуальной, поскольку решает требования реального проекта в современной сфере науки и образования. Отдельные компоненты системы могут использоваться в различных проектах, построенных

на основі CMS Joomla 2.5. Система веб-портала може використовуватися в якості автономного ресурса для підтримки дистанційного навчання.

**Список використаних джерел:** 1. *Проект INSITOR* [Електронний ресурс]. – Режим доступу : <http://stu.cn.ua/staticpages/INSITOR/>. 2. *Режим доступу* : <http://ru.wikipedia.org/wiki/CMS>. 3. *Режим доступу* : <http://ru.wikipedia.org/wiki/CMF>. 4. *Режим доступу* : <http://ru.wikipedia.org/wiki/Joomla!> 5. *Режим доступу* : <http://uk.wikipedia.org/wiki/OpenMeetings>.

УДК 517.957

## **ВИКОРИСТАННЯ ФРЕЙМВОРКУ BOOTSTRAP ДЛЯ ШВИДКОГО СТВОРЕННЯ ВЕБ-САЙТІВ**

С. Легкодух, М. Скоп

*Чернігівський національний технологічний університет*

У сучасному світі ми часто зустрічаємо слово «сайт». Наше життя стало неможливим без нього. Будь-яку інформацію можна знайти буквально за декілька хвилин. Сайти створюються розробниками і в їхню основу закладено немало коштів і часу. Працю розробника сайтів можна зробити не такою складною за допомогою фреймворку Bootstrap. Частіше за все програмісти при створенні сайту мають небагато ресурсів. Аналізуючи всі можливі методи, можна звернутися до безкоштовних сервісів швидкого проектування сайтів, одним із яких є фреймворк Bootstrap.

Фреймворк Bootstrap – набір об'єднаних компонентів для спрощення роботи розробника сайтів, швидкої розробки та виключення максимально можливої кількості помилок (проблеми з різними версіями браузерів і т.д.). Розповсюджується по ліцензії MIT. Це ліцензія вільного ПО розроблена Масачусетським технологічним університетом. В ній сказано, що користувачам дозволяється безкоштовно використовувати дане програмне забезпечення та супутню документацію, щоб мати справу з ПО без обмежень, у тому числі, без обмежень права на використання, копіювання, зміну, публікацію та розповсюдження копії. Програмне забезпечення надається «як є», без будь-яких гарантій, автори або власники авторських прав не відповідають за будь-які претензії, збитки або будь-яку іншу відповідальність.

Попри те, що у Bootstrap'а великий набір можливостей, далеко не всі, кому він міг би знадобитися, можуть ним скористатися, часто через те, що просто не вміють користуватися технологією. По суті Bootstrap має в собі готові стилі для оформлення елементів html, у тому числі кнопок, таблиць, форм, елементів навігації, списків, картинок і відео, а також базові елементи на javascript, такі як слайдери, випадаючі списки, вкладені меню, навігація і т.д. Цей фреймворк дозволяє створити базову структуру і стилізувати на базовому рівні більшість елементів. Для тих, хто хоче створювати прості сайти, досить просто знати html і що робить кожний із класів Bootstrap. Велика перевага в тому, що структура сайту на Bootstrap основана на 12-ти колонній grid системі. Дизайн сайту повинен бути розкреслений на 12 вертикальних полос, кожна із яких традиційно займає 60px. За фактом, в css можливі 4 варіанти розміру колонок, залежно від того, який сайт необхідно отримати – фіксований, гумовий, або адаптивний. Файл bootstrap.css відповідає за розмітку сторінки під монітори комп'ютера. Тут містяться стилі для фіксованих і гумових блоків. Файл bootstrap – responsive.css дозволяє зробити верстку адаптивного сайту, тут також є можливість створення гумових і фіксованих блоків. Для того, щоб почати роботу, достатньо завантажити файли Bootstrap на офіційному сайті <http://twitter.github.com/bootstrap/index.html>, створити html – сторінку, імпортувати туди css і javascript файли фреймворка. Як і бібліотеки в різних мовах програмування, як Java чи C++, бібліотеки

фреймворку, які зазвичай мають вигляд зовнішнього файлу у форматі css і js, під'єднуються до проекту прописування декількох рядків html коду.

Bootstrap був спочатку розроблений з використанням LESS. LESS – це динамічна мова стильової розмітки. Це дозволило зробити розробку CSS більш швидкою і простою. Один з творців Bootstrap опублікував про це короткий пост у своєму блозі. Ось його основні думки:

- Bootstrap компілюється с LESS в 6 разів швидше, ніж з Sass;
- Less написаний на JavaScript, що набагато простіше порівняно з Ruby, написаному на Sass.
- Less – це щось більше, ми хочемо відчувати, що самі пишемо CSS і налаштовуємо в Bootstrap абсолютно всі можливості.

Як розширення CSS, LESS включає в себе змінні, функції для повторно використовуваних шматків коду, найпростіші математичні операції, вкладення, і навіть функції кольору.

Головне в роботі з нашим фреймворком – розібратися зі структурою побудови блоків. Вона буде приблизно такою :

```
<div class="container">
  <div class="row">
    <div class="span7"> </ div >
    <div class="span2"> </ div >
  </ div >
</ div >
```

де container – це загальний блок, row – один ряд блоків, а span\* – самі блоки, що займають кількість колонок, відповідно зазначеної після span цифри. Так як колонок всього 12, класів Span\* теж 12, span1 дорівнює ширині однієї колонки, тобто 60px, Span2 дорівнює двом колонкам, тобто 140px (60px +60 px +20 px (відступ зліва)), Span3 = 220px (60px +60 px +60 px +20 px +20 px) і т.д. до span12 - 940px (60 \* 12 +20 \* 11).

Блоки можна вкладати один в одного, але не можна забувати про те, що кожен новий ряд потрібно обгорнути в div Row :

```
<div class="row">
  <div class="span8">
    <div class="row">
      <div class="span4"> </ div >
      <div class="span4"> </ div >
    </ div >
  </ div >
  <div class="span4"> </ div >
</ div >
```

Так як кількість колонок 12, сума цифр усередині одного ряду (першого рівня) не може перевищувати цього числа. Наприклад, в одному div.row можуть бути span10 і span2 (10 +2 = 12) або span6 і span4 (6 +4 = 10, відповідно відстань в 2 колонки (140px) залишиться вільною) і т.д. Якщо вкласти в div.span8 внутрішні блоки, то сума колонок не може перевищувати вже не 12, а 8ми. Наприклад, ви не можете вкласти в Span5 (рівний 380px) елементи з класами Span4 (300px) і Span3 (220px), оскільки вони просто не помістяться в батьківському блоці.

Відступи задаються за допомогою додавання класів Offset1 - Offset11. Додавання цього класу диву ставить йому відступ зліва, що дорівнює кількості колонок, які передують цьому диву. Тобто якщо блок, який необхідно відсунути, відокремлюють від попереднього блоку або від початку контейнера 4 колонки, то цьому блоку необхідно прописати class = "offset4 ". Ніде не потрібно обнуляти обтікання - Twitter Bootstrap це вже передбачив. Фреймворк Bootstrap використовує найновіші технології верстки, найостанніші розробки. Тому необхідно звернути увагу на те, що в head документа обов'язково має бути зазначений

доктайп html5 - <!DOCTYPE html >. Twitter Bootstrap підтримується Internet Explorer, починаючи з 7-ї версії. Хоча іноді й тут бувають промахи, але їх легко виправити. Також необхідно підключити файл html5.js, що забезпечує підтримку html5 браузерами Internet Explorer. Це основне, що потрібно натиснути перед тим, як почати верстку з Twitter Bootstrap.

На даний момент існує багато різновидів фреймворків, які мають непоганий функціонал. Це фреймворк Foundation, html5 boilerplate, plusstrap. Всім цим фреймворкам Bootstrap створює дійсно гарну конкуренцію. Існує також інша версія Bootstrap, розроблена Російськими програмістами, яка має назву Bootstrap.PHP. Створити меню, що випадає або навігаційну панель простіше на Bootstrap.PHP, ніж на "чистому" Bootstrap. І не тільки їх, чим складніше компонент, тим більше явно, що це перевага. Так як скрізь працює автодоповнення, швидше за все, навіть не доведеться заглядати в документацію, щоб зрозуміти, що і як зробити.

На кого бібліотека орієнтована?

1) Перш за все на розробників, які створюють свої проекти самостійно. Навряд чи таку бібліотеку використовуватимуть у великій команді, де верстальник не знає php.

2) На не надто досвідчених розробників, оскільки якщо давно працюєш з Bootstrap, то можливо швидше і звучніше робити все, як раніше.

3) На тих, кому зручніше працювати з бібліотеками, де документація в кодї російською мовою.

Основні пункти, на які робився акцент при розробці – це простота. Ідея була в тому, щоб розробнику не довелося більше заглядати в документацію після того, як він з нею в перший раз ознайомився.

Сайт проекту <http://allking.ru/bootstrap.php/>

Чим Bootstrap.PHP краще "чистого" Bootstrap?

1. Кожен програміст знає, що потрібно уникати повторення коду. Також хочеться уникати повторення html в шаблонах. Коли на основі одного і того ж компонента Bootstrap створюється кілька різних шаблонів, повторення неминуче. Bootstrap.PHP допомагає цього уникнути.

2. Bootstrap.PHP дозволяє створювати шаблони швидше, його можна використовувати прямо в шаблонах, зазвичай, якщо є готові шаблони на php.

Отже, за допомогою Bootstrap можна зробити розроблення сайтів значно ефективнішим, а саме: зменшити час на проектування сайту, конструювати сайти з великим функціоналом, використовуючи великий набір стилів для навігації, у тому числі кнопок, таблиць, форм, картинок і відео. Використовуючи Bootstrap.PHP, можна писати сайти навіть недосвідченому програмісту з базовими знаннями верстання інтернет-сторінок.

УДК 004.67

## ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ ДЛЯ GPS-МОНИТОРИНГА

И. Михайлюк

*Черниговский национальный технологический университет*

С каждым годом GPS-мониторинг автотранспорта получает все большее распространение в мире [1]. Во многих развитых странах (таких, как государства Евросоюза, США, Украина, Россия, Канада, Япония) его внедрение входит в перечень обязательных требований к компаниям, осуществляющим деятельность в сфере ЖКХ, железнодорожных и грузопассажирских перевозок, сельском хозяйстве, добыче полезных ископаемых и т. д.

Современные системы GPS-мониторинга транспорта позволяют в режиме реального времени отслеживать местонахождение наблюдаемого объекта. Для этого, он оборудуется

трекером – устройством, которое по данным, полученным из сигналов GPS-спутников, определяет свое местоположение и с помощью GSM и Интернет передает эти данные на сервер. В свою очередь, сервер хранит данные мониторинга и предоставляет к ним доступ. Благодаря этому диспетчер или автоматическая система может контролировать маршрут движения объекта наблюдения, следить за пробегом, скоростным режимом, остановками и стоянками, а также по данным датчиков следить за режимами работы двигателя, состоянием груза. В качестве трекера может быть использован смартфон с GPS-приемником и установленным соответствующим ПО.

Ныне для GPS-мониторинга можно воспользоваться несколькими вариантами.

Первый – бесплатные сервера GPS-мониторинга, такие как Live GPS Tracks [2], Freetrack [3], GPS-Trace Orange [4] и другие. Недостатком этого решения есть то, что данные мониторинга хранятся короткий промежуток времени и нет никаких гарантий их конфиденциальности и сохранности.

Второй – купить услуги какого-нибудь сервиса или серверное ПО GPS-мониторинга, например, Wialon[5], Vidis[6], GPS-Ukraine [7]. Таких решений очень много. Недостатком решения есть то, что за него нужно заплатить.

Третий вариант – установить собственный сервер GPS-мониторинга на базе свободного ПО. Примерами такого решения могут быть ПО OpenGTS [8] или Traccar [9]. Недостатки такого решения – это потребность в отдельном сервере и администраторе.

OpenGTS («Open GPS Tracking System») — первое open-source решение, разработанное специально для мониторинга транспортных средств в web-браузерах, используя данные с GPS-приемников. На сегодняшний день OpenGTS введено в эксплуатацию в 110 странах по всему миру [8]. OpenGTS распространяется под лицензией Apache Software License 2.

OpenGTS поддерживает сбор и хранение данных GPS-мониторинга и включает в себя следующие функции:

- Web-аутентификация. Каждая учетная запись может поддерживать несколько пользователей, и каждый пользователь имеет свой собственный пароль пользователя и контролируемый доступ к разделам, относящимся к их учетной записи.

- Поддержка большого количества GPS-трекеров.

- Настраиваемый вид web-интерфейса.

- Работа с несколькими картографическими сервисами: OpenLayers, Google Maps, Microsoft Virtual Earth и Mapstraction.

- Настраиваемые отчеты.

- Настраиваемые геозоны направления.

- Кроссплатформенность.

- Поддержка языков: голландский, английский, французский, немецкий, греческий, венгерский, итальянский, португальский, румынский, русский, словацкий, испанский, сербский и турецкий.

Также на базе OpenGTS создан коммерческий продукт – "GTS Enterprise", функционал которого еще шире, в основном, за счет большого количества поддерживаемых устройств.

OpenGTS полностью написан на Java. OpenGTS использует такие технологии, как Apache Tomcat для развертывания web-службы, MySQL для хранилища данных. Таким образом OpenGTS будет работать на любой системе, которая поддерживает эти технологии. На рис. показана архитектура системы.

На домашнем сайте проекта можно скачать последние версии исходников проекта и техническую документацию.

Traccar – система с открытым исходным кодом для GPS мониторинга различных объектов. На данный момент сервер поддерживает более 30 различных протоколов. Версии, начиная с 1.0, включают встроенный HTTP-сервер и веб-интерфейс для отслеживания устройств.

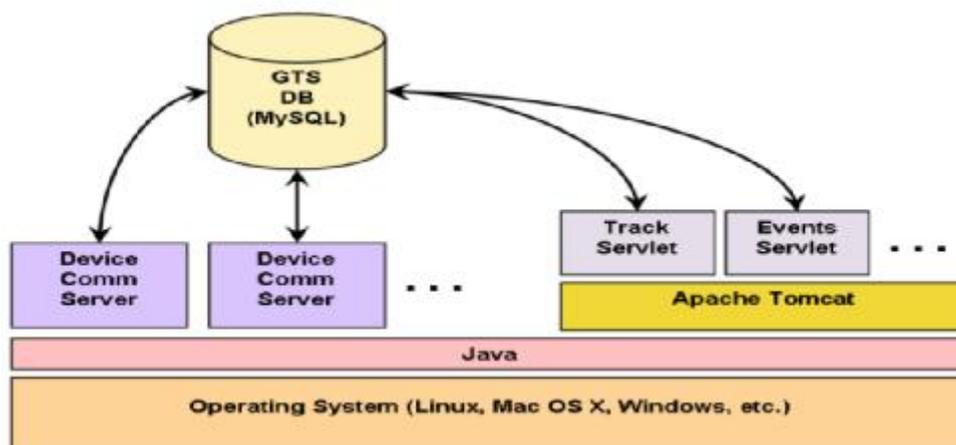


Рис. Программная архитектура OpenGTS

Одной из основных миссий проекта является поддержка максимального количества устройств слежения.

Трассар написан на языке Java, это дает возможность запускать продукт практически на любой ОС. На данный момент существуют пакеты установки для Windows, Mac и Linux. Исходники доступны в репозиториях. Также на сайте [9] доступны инструкции по установке и настройке. Трассар – программное обеспечение, распространяемое под лицензией Apache License 2.0.

В качестве базы данных используется встраиваемая H2 Database, интерфейс создан на ExtJS и Google Maps API, HTTP-запросы обрабатывает Jetty контейнер. Есть возможность интеграции с OpenGTS-системами.

Подводя итоги, можно сказать, что существует достаточно функциональное свободное программное обеспечение для создания систем GPS-мониторинга.

**Список использованных источников:** 1. *GPS-мониторинг* транспорта [Электронный ресурс]. – Режим доступ а: <http://glonass-system.ru/gps-monitoring-transporta/>. 2. *GPS/ГЛОНАСС-мониторинг* [Электронный ресурс]. – Режим доступа : <http://livegpstracks.com/>. 3. *FreeTrack* [Электронный ресурс]. – Режим доступа : <http://freetrack.com.ua/>. 4. *GPS-Trace Orange* [Электронный ресурс]. – Режим доступа : <http://gps-trace.com/>. 5. *Gurtam Wialon* [Электронный ресурс]. – Режим доступа : <http://gurtam.com/>. 6. *Системы gps-мониторинга транспорта и контроля топлива* [Электронный ресурс]. – Режим доступа : <http://www.gps-monitoring.com.ua/>. 7. *GPS-UKRAINE* [Электронный ресурс]. – Режим доступа : <http://gps-ukraine.com.ua/>. 8. *Open GPS Tracking System* [Электронный ресурс]. – Режим доступа : <http://opengts.org/>. 9. *Traccar open source GPS tracking system* [Электронный ресурс]. – Режим доступа : <http://www.traccar.org/>.

УДК 004.3

## ИСПОЛЬЗОВАНИЕ АППАРАТНОЙ ПЛАТФОРМЫ С ОТКРЫТОЙ АРХИТЕКТУРОЙ ARDUINO ДЛЯ ПОСТРОЕНИЯ СИСТЕМ УПРАВЛЕНИЯ

А. Молофеев, А. Красножон

*Черниговский национальный технологический университет*

ARDUINO представляет собой среду разработки под одноименный модуль, содержащий в себе микроконтроллер. Данный модуль с легкостью подключается к персональному компьютеру с помощью последовательного интерфейса ЮЗБ и имеет множество входов и выходов. Они делятся на аналоговые и цифровые. Питается модуль как от ЮЗБ, так и от других внешних источников в диапазоне от 0 до +5В. ARDUINO

может работать с множеством периферийных устройств. Практически под каждое из этих устройств есть библиотеки. Сама среда разработки – это простое в использовании приложение, в котором пишут так называемые скетчи. Скетчи представляют собой последовательность команд, написанных на языке, похожем на Си. В приложении – простой интерфейс, который содержит стандартное меню и несколько кнопок, нажимая на которые можно создать новый скетч, скомпилировать текущий и кнопка программирования модуля.

“Умный” дом – это современное жилое помещение, в котором применяются новые достижения науки и техники для упрощения жизни человека. Под «умным» домом следует понимать систему, которая обеспечивает комфорт (в том числе и безопасность), и ресурсосбережение для всех пользователей. В простейшем случае она должна уметь распознавать конкретные ситуации, происходящие в доме, и соответствующим образом на них реагировать. В наше время эта система очень актуальна, так как сейчас многие сферы науки продвигаются в сторону энергосберегающих систем.

“Умный” дом предназначен для того, чтобы упростить жизнь его жителям. В таком доме есть автоматическая система безопасности, система управления некоторыми приборами и предметами. Основные функции такого дома: обеспечивать автоматическое освещение в помещениях, управление жалюзи или шторами, оповещение хозяина о взломе, предотвращение несчастных случаев (утечка природного газа и т. п.), информация жителей о погодных условиях в помещении и за его пределами.

Чтобы реализовать систему “умный” дом, понадобятся такие ключевые элементы: система управления (представляет собой микроконтроллер, с помощью которого будет все управляться, он будет принимать данные с сенсоров и передавать команды системе реагирования), разного рода сенсоры (периферийные устройства, которые будут подключены непосредственно к системе управления и сообщать ее о происходящих событиях) и система реагирования (данная система будет реагировать на сигналы, которые будут приходить на нее с системы управления и выполнять поставленные задачи).

Реализовать вышеперечисленные функции с помощью ARDUINO не очень сложно. Автоматическое освещение в помещениях можно реализовать в помощью подключения к ARDUINO реле- и фоторезисторов, а все это (в свою очередь) подключить к системе освещения (к выключателям). В ночное время, когда хозяева спят, освещение будет отключаться, а когда хозяева не спят, то работать в нормальном режиме. Управление жалюзи или шторами будет реализовано при помощи серводвигателей (двигатели, которые прокручиваются на заданное количество градусов) и фоторезисторов. Когда на улице будет светить яркое солнце, то жалюзи будут закрываться и наоборот. Оповещение хозяина о взломе будет проводиться с помощью отправки ему на мобильный телефон текстового сообщения с соответствующим текстом. Для этого можно подключить к ардуино GSM-модуль и датчики взлома, которые необходимо установить на окна и входную дверь.

На примере функции предотвращения несчастного случая рассмотрим утечку природного газа. Для того, чтобы это предотвратить, нужно установить на кухне или в помещении, где рядом есть газовые трубы, датчик, который реагирует на газ. К ARDUINO еще нужно подсоединить достаточной мощности серводвигатель, который будет перекрывать газовый кран.

Информировать жителей о погодных условиях можно с помощью подключения к ардуино датчиков температуры и влажности воздуха, а выводить информацию можно на ЖКИ-мониторчики или на персональный компьютер.

Особенностью данного подхода для решения поставленной задачи является простота реализации. Все, что должно обеспечивать работоспособность системы, можно написать в одном скетче, даже не владея сверхзнаниями в программировании. Если на модуле не будет достаточно выходов, их можно расширить с помощью специальных устройств.

Все, что необходимо для воплощения данного проекта в жизнь, стоит не очень дорого. Основа проекта модуль ARDUINO стоит 30-35 \$. Вся остальная периферия или

входить в комплект поставки ARDUINO или нужно покупать отдельно. То, что придется докупить отдельно, стоит 10-15 \$. Среда разработки под данный модуль является бесплатным приложением. Все вместе будет стоить около 45 \$.

Существует много аналогов ARDUINO. Даже сам ARDUINO имеет несколько разновидностей. Можно найти аналоги намного дешевле оригинала, но под них может не быть качественного софта или не будет поддержки некоторой периферии и библиотек под них.

**Список использованных источников:** 1. Режим доступа : <http://www.wikipedia.org/>.  
2. Режим доступа : <http://www.aliexpres.com/>.

УДК 004.415.2

## ПАРАЛЕЛЬНЕ ТА КОНКУРЕНТНЕ ПРОГРАМУВАННЯ В ФУНКЦІОНАЛЬНИХ МОВАХ ПРОГРАМУВАННЯ НА ПРИКЛАДІ HASKELL

А. Морозов, С. Стасюк

*Чернігівський національний технологічний університет*

Паралельність сама по собі як загальне поняття є невід'ємною частиною роботи будь-якої обчислювальної системи, оскільки дозволяє швидше обробляти дані, розмежувати роботу вводу-виводу та обчислювальних операцій.

Якщо розглядати програмну паралельність (та конкурентність) більш детально, то слід зазначити, що загальні принципи є досить універсальними і майже не підлягають деталізації, а шляхи реалізації – досить різні. Таким чином, поняття багатопоточності на даний момент все ще залишається актуальним, оскільки все ще досить вигідно розподіляти задачі на одній системі. Будуть це стандартні `unix`-ові `pthread` чи `java.lang.Thread` – різниця зовсім невелика у використанні. Основною проблемою конкурентного програмування є нарощення абстракції та синхронізація цих самих потоків. Але нарощуючи кількість потоків, рано чи пізно постає питання доцільності такої технології, такого виконання, тому що при високому рівні конкурентності для вирішення складних задач можуть виникати проблеми з продуктивністю і тоді слід звертати увагу не на просте використання потоків, а вже паралельних систем та розподілення задач на різні виокремлені системи.

Використовуючи паралельність у чистому вигляді, простіше розділити задач, та отримати потрібну продуктивність при вирішенні задач. Але і при паралельності є свої мінуси – простий обладнання, неефективне використання обладнання. З цього можна зробити висновок, що, використовуючи той чи інший тип паралельного програмування, складається наступна ситуація: розбиття на потоки не є достатньо ефективним методом для вирішення нашої задачі, розпаралелювання на системи є занадто “широким” вирішенням питання, а більшість задач знаходяться у цьому невеликому проміжку між потоками та розпаралелюванням. Така ситуація складається, незважаючи на складність розпаралелювання самої задачі.

Паралельне програмування в Haskell кардинально не міняє саму сутність паралелізму, але уточнює деякі поняття, робить ремарки і більш ідеологічно, як і належить мові високого рівня, відноситься до процесу розпаралелювання. Таким чином все паралельне програмування в Haskell розбито на дві основні категорії: конкурентне програмування і паралелізм. Конкурентне програмування – це такий тип паралельного програмування, при якому одиниці паралелізму взаємодіють один з одним за допомогою будь-яких значень. Паралелізм – тип паралельного програмування, при якому одиниці паралелізму є чистими і не взаємодіють. Як можна здогадатися – конкурентне більш направлене на роботу з потоками і з однією системою. Але це не завжди так, оскільки Haskell не обмежує нас роботою з однією машиною. Так само і з паралелізмом, межа роботи з кількістю систем не є чіткою, все залежить лише від задачі, яку треба вирішити.

Достаточно цікавою тема паралельного програмування на Haskell є тільки тому, що це чиста функціональна мова програмування. Найпростіший приклад того, над чим слід задуматись – це те, що в цій мові немає змінних і все є функцією. Це добре для паралелізму, але достатньо складно для конкурентності.

**Список використаних джерел:** 1. *Concurrent Haskell* [Електронний ресурс]. – Режим доступу : [http://www.haskell.org/ghc/docs/7.0.4/html/users\\_guide/lang-parallel.html#concurrent-haskell](http://www.haskell.org/ghc/docs/7.0.4/html/users_guide/lang-parallel.html#concurrent-haskell). 2. *Using SMP parallelism* [Електронний ресурс]. – Режим доступу : [http://www.haskell.org/ghc/docs/7.0.4/html/users\\_guide/using-smp.html](http://www.haskell.org/ghc/docs/7.0.4/html/users_guide/using-smp.html). 3. *GHC/Data Parallel Haskell* [Електронний ресурс]. – Режим доступу : [http://www.haskell.org/haskellwiki/GHC/Data\\_Parallel\\_Haskell](http://www.haskell.org/haskellwiki/GHC/Data_Parallel_Haskell).

УДК 551.50

## OPENWEATHERMAP – ОТКРЫТЫЕ МЕТЕОРОЛОГИЧЕСКИЕ ТЕХНОЛОГИИ

О. Морозова, А. Роговенко

*Черниговский национальный технологический университет*

Погодой всегда занимались и занимаются изолированные от внешнего мира институты, лаборатории и крупные государственные организации, объединенные в World Meteorological Organization. Как рождаются прогнозы погоды, всегда оставалось загадкой. Создатели OpenWeatherMap решили, что такая жизненно важная тема как погода, должна быть доступна всем.

Идея свободной и доступной информации о погоде привела к тому, что был создан и предоставлен всем разработчикам приложений бесплатный API для получения разнообразных данных о погоде:

- интерактивная карта с данными о текущей погоде;
  - прогноз на неделю в городе;
  - исторические данные в 120 000 городах мира;
  - данные от 40 000 метеостанций по всему миру, получаемые практически в режиме online. (Задержка от секунд до часа);
  - много различных web-карт, включая карты облаков, осадков, ветра, температуры и т.п.
- Принцип работы OpenWeatherMap изображен на рис. 1.

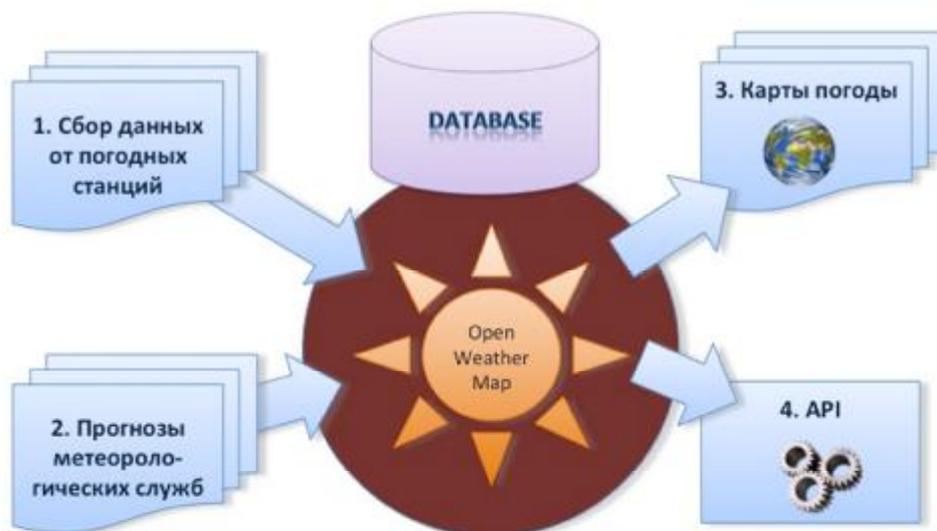


Рис. 1. Концепция OpenWeatherMap

Сервис OpenWeatherMap получает данные от профессиональных и частных погодных станций. Кроме погодных станций, сервис OWM собирает уже обработанные данные о прогнозе погоды.

OWM объединяет данные разномасштабных прогнозов – от усредненных и глобальных до локальных и более точных соответственно. В результате web-карты OpenWeatherMap работают очень удобно – для крупного масштаба используются глобальные прогнозы, а в процессе увеличения загружаются все более и более детальные данные (рис. 2).

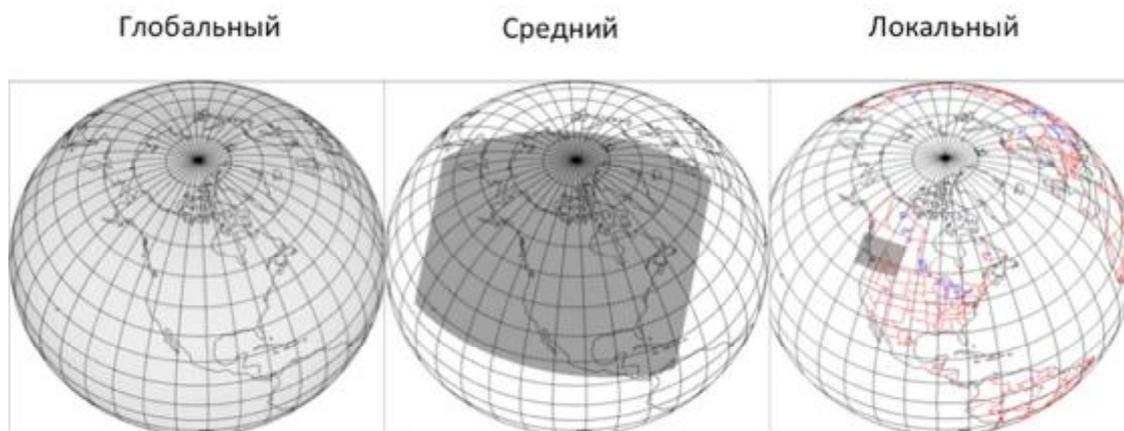


Рис. 2. Виды прогнозов

Сервис объединяет точные локальные модели вместе с глобальными. После этого он ассимилирует сверхоперативные данные от погодных станций. И уже на основе всех этих данных и строятся интерактивные карты (карты облаков, давления, температуры, осадков, ветра), в которых степень детализации данных зависит от масштабирования карты.

Архитектура OpenWeatherMap изображена на рис. 3. Используются только Open Source software, среди которых Nginx, Apache, PHP, Tilecache, OpenLayers, Leaflet, Mapnik, PostGIS, Memcache, MongoDB, Gearman, MySQL, Python and more.

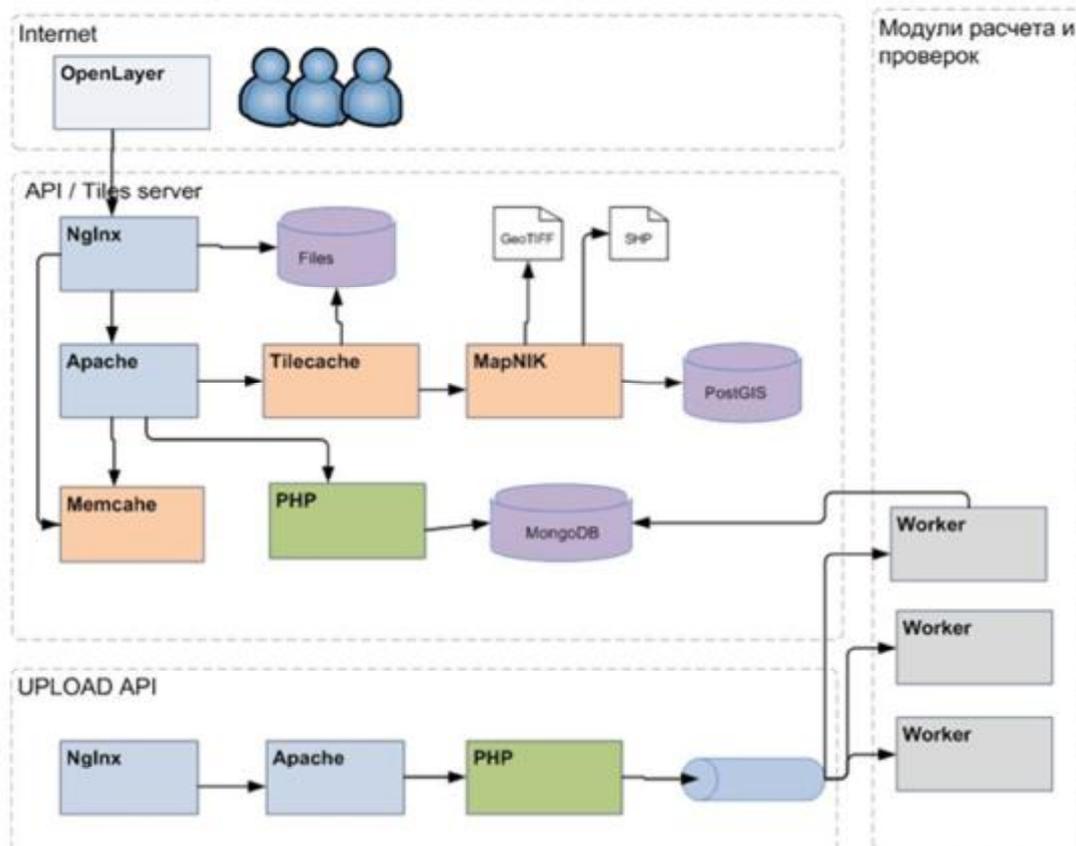


Рис. 3. Архитектура OpenWeatherMap

Сервис OpenWeatherMap предоставляет бесплатный API ко всем данным о погоде, к их истории, прогнозам и всему многообразию погодных карт. API есть двух видов – JSON для получения данных и Tile / WMS для картографии.

Спектр применения API бесконечно широк. Это мобильные приложения для всех платформ, разнообразные web-сайты, которые могут использовать API для отображения текущей погоды, различных погодных графиков, виджетов и т. д., системы “умного” дома.

Таким образом, развитие проекта OpenWeatherMap направлено на: социальность; добавление внешних расчетных модулей; добавление погодных станций; разработку специализированных метеорологических приложений.

УДК 004.457

## **ИСПОЛЬЗОВАНИЕ БИБЛИОТЕКИ OPENCV ДЛЯ РЕШЕНИЯ ЗАДАЧ КОМПЬЮТЕРНОГО ЗРЕНИЯ**

М. Побожая, Ю. Ющенко, М. Харченко

*Черниговский национальный технологический университет*

Открытые библиотеки становятся все более и более популярной категорией многократного используемого кода. Общая значимость их «открытости» состоит в том, что исходный код доступен для каждого желающего. Относительно включения открытого кода в собственные приложения существует ряд формальных определений и правовых норм, но здесь важно помнить, что открытые программные продукты позволяют любому читать их исходный код.

Все мы рано или поздно сталкиваемся с бесплатными и OpenSource компонентами, и библиотеками, средами, редакторами. Как показывает практика, иногда они даже лучше некоторых платных, а иногда просто не имеют аналогов. Примером такой библиотеки есть OpenCV.

С точки зрения науки, OpenCV – это библиотека, которая до 1-й версии разрабатывалась в Центре разработки программного обеспечения Intel. OpenCV написана на языке высокого уровня (C/C++) и содержит алгоритмы для: интерпретации изображений, калибровки камеры по эталону, устранение оптических искажений, определение сходства, анализ перемещения объекта, определение формы объекта и слежение за объектом, 3D-реконструкция, сегментация объекта, распознавание жестов и т.д. Эта библиотека очень популярна за счёт своей открытости и возможности бесплатно ее использовать как в учебных, так и коммерческих целях.

Фактически OpenCV – это набор типов данных, функций и классов для обработки изображений алгоритмами компьютерного зрения.

Основные модули библиотеки (рис.):

- 1) *sxcore* – ядро содержит базовые структуры данных и алгоритмы:
  - базовые операции над многомерными числовыми массивами;
  - матричная алгебра, математические функции, генераторы случайных чисел;
  - запись/восстановление структур данных в/из XML;
  - базовые функции 2D графики.
- 2) *CV* – модуль обработки изображений и компьютерного зрения:
  - базовые операции над изображениями (фильтрация, геометрические преобразования, преобразование цветовых пространств и т. д.);
  - анализ изображений (выбор отличительных признаков, морфология, поиск контуров, гистограммы);
  - анализ движения, слежение за объектами;
  - обнаружение объектов, в частности лиц;
  - калибровка камер, элементы восстановления пространственной структуры.

3) Highgui – модуль для ввода/вывода изображений и видео, создания пользовательского интерфейса:

- захват видео с камер и видеофайлов, чтение/запись статических изображений;
- функции для организации простого UI (все демо-приложения используют HighGUI).

4) Cvaux – экспериментальные и устаревшие функции:

- пространств. зрение: стереокалибрация, самокалибрация;
- поиск стерео-соответствия, клики в графах;
- нахождение и описание черт лица.

5) CvCam – позволяет осуществлять захват видео с цифровых видео-камер (поддержка прекращена и в последних версиях этот модуль отсутствует).

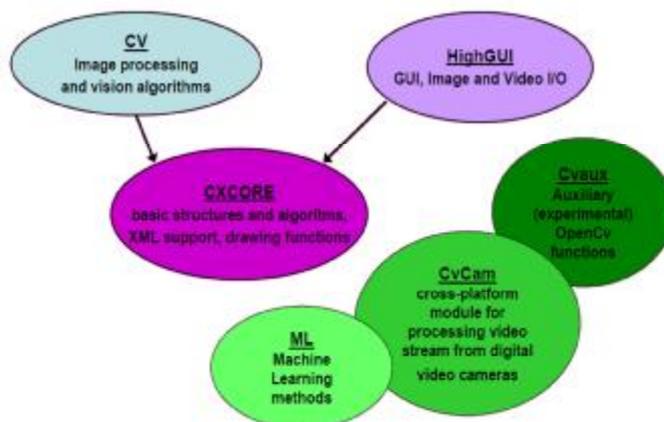


Рис. Основные модули библиотеки OpenCV

Использование этой библиотеки на сегодняшний день является актуальным. Она может решать ряд современных задач, таких как:

– распознать номер на машине – для библиотеки OpenCV это тривиальная задача. С использованием двух библиотек (Qt и OpenCV) не трудно сделать программу для распознавания номеров на авто. При большом желании можно добавить лазерный дальномер и выходить на улицу – выписывать штрафы;

– автоматическое распознавание дорожных знаков на видео в режиме реального времени. При разработке может использоваться простая камера. OpenCV предоставляет довольно простой способ работы с web-камерами (модуль HighGui);

– система слежения и распознавания лиц на Arduino – веб-камера, установленная на поворотном механизме подключена к компьютеру под управлением операционной системы Windows и с установленным программным обеспечением OpenCV. Если программа обнаруживает в поле зрения веб-камеры лицо, то вычисляется центр лица. Координаты X и Y передаются в контроллер Arduino, который подключен к компьютеру по USB. В свою очередь, контроллер Arduino по принятым командам управляет двумя сервомоторами: по координате X и по координате Y, т.о. обеспечивается следящая система;

– рисование на изображениях – очень часто нам будет необходимо рисовать или писать что-либо на изображениях взятых откуда-либо. OpenCV предоставляет нам такую возможность, обладая функциями способными рисовать линии, квадраты, круги и т.д.;

– детектор движения – часто используется для управления светом в доме;

– робот-автомобиль – он может автономно двигаться за человеком на медленной скорости (около 5-7 км/ч) и запоминать маршрут.

Именно последнее привлекло наше внимание. Возможно потому, что его можно назвать заданием будущего. Именно такие машины мы видим почти в каждом фантастическом фильме. Конечно, рассматриваемый нами пример робота-автомобиля далек от идеала, но для нас все же главное использование при его разработке библиотеки OpenCV.

Данный проект вообще был разработан для участия в соревновании “Робокросс-2013”.

Если говорить в целом о разработке авто, то в качестве операционной системы используется Ubuntu, среда разработки – QtCreator, язык – C++, библиотека для gui – Qt, получение и обработка изображений – OpenCV, обработка данных лазерного сканера – PCL, а также Boost для работы с файлами и сетью. В результате получилось кроссплатформенное решение. Однако всё же используется Ubuntu, так как система более предсказуема, в фоне случайно не начинает работать антивирус или индексация файлов, тормозя все остальные процессы. Да и общая производительность несколько выше.

Создатели данного проекта предпочитают электронику, созданную собственными руками.

Механика представляет собой набор актуаторов, которые приводят в движение органы управления автомобиля. При разработке механики главным требованием было сделать легкосъёмную конструкцию, чтобы можно было быстро всё разобрать, и машина превращалась в обычную Газель.

В частности, библиотека OpenCV используется в данном проекте для распознавания меток, за которыми должен следовать автомобиль.

В общем робот-автомобиль – сложная система. Создание каждого компонента – это вызов, который должны принять и преодолеть разработчики, особенно в условиях нашей страны.

В заключении хотелось бы заметить, что использование библиотеки OpenCV не имеет определенных рамок и с ее помощью можно решать огромное количество различных задач. И, следовательно, вывод такой: использование бесплатных библиотек не только облегчит Вам поиск необходимых ресурсов, но и расширит Ваши возможности в проектировании.

УДК 004.41:61

## **ПРОГРАММА ДЛЯ УПРАВЛЕНИЯ МЕДИЦИНСКОЙ ПРАКТИКОЙ OPENEMR**

В. Подкур, С. Стасюк

*Черниговский национальный технологический университет*

Медицинская информационная система осуществляет управление поликлиникой, больницей, аптекой, медицинским центром и позволяет организовать работу в виде единого механизма всех работников по медицинскому учреждению. Медицинские программы позволяют сотрудникам организации на регистратуре открывать электронные медицинские карты, вести учет медицинских книжек и записывать пациентов на прием. Медицинский учет обеспечивает кассирам возможность видеть за своим рабочим местом тех, кого записали на услугу, и принимать оплату от граждан.

“Электронная история болезни” может хранить данные долгие годы и по ней значительно легче искать информацию, чем на бумажных носителях. Управление в медицине позволяет заполнять автоматически программным способом медицинские справки, журналы учета и формы обязательной медицинской отчетности. Управление поликлиникой, клиникой или больницей осуществляется программно рабочего места. За счет этого медицинский учет становится невероятно легким. Медицинская программа требуется для управления лабораторией, центром реабилитации, диагностическим центром, роддомом, диспансером, госпиталем, государственной больницей или частным реабилитационным, оздоровительным комплексом.

OpenEMR является открытым исходным кодом в медицинской практике управления (приложения EHR ЭМИ PMS) с полностью интегрированной электронной медицинской

документацией, осуществляет планирование, электронный биллинг, интернационализацию, бесплатную поддержку и многое другое (рис.).



*Рис. Внешний вид OpenEMR*

OpenEMR – это LAMP – тип программного обеспечения, использующего Веб – сервис Apache, базу данных MySQL. Написана на языке программирования Python, клиент-серверное взаимодействие реализовано посредством протокола XML-RPC. В качестве системы управления базами данных для серверной части используется PostgreSQL. Является кроссплатформенной программой.

Таким образом возможности OpenEMR широки в медицинской практике и позволяют осуществлять: ONC Complete Ambulatory EHR Certified; демографию пациентов; планирование; электронные медицинские записи; рекомендации; бухгалтерию; клинические правила принятия решений; портал пациента; отчетность; безопасность.

УДК 377.1

## ОБРАЗОВАТЕЛЬНЫЙ ПРОЕКТ KDEEDU

Ю. Подобед, С. Стасюк

*Черниговский национальный технологический университет*

При переводе образовательного процесса на платформу свободно распространяемого программного обеспечения неизбежно возникает ряд проблем. Одной из главных проблем (по мнению педагогов) является проблема наличия образовательных компьютерных программ.

**KDE Education Project** – пакет образовательных компьютерных программ. Является частью проекта среды рабочего стола KDE, включают в себя набор обучающих программ по различным учебным дисциплинам (математике, химии, астрономии, языкам и др.), клавиатурный тренажер, простой и функциональный редактор для составления тестов и средства для обучения программированию школьников. Распространяется согласно GNU General Public License.

KDE Education Project фокусирует своё внимание на разработке программ для детей в возрасте от 3 до 18 лет. Кроме того, в пакет входят программы для преподавателей, помогающие планировать уроки.

Следует отметить, что практически все программы имеют очень простой и интуитивно понятный интерфейс. Освоение работы с этими программами не потребует длительного обучения педагогов, что очень важно на переходном этапе.

### Список программ

**BlinKen.** BlinKen – KDE-версия игры «Саймон сказал», входящая в пакет образовательных программ KDE Edutainment Project. Распространяется согласно GNU General Public License. Blinken основана на электронной игре, выпущенной в 1978 году, которая проверяет способность игрока запоминать определенные последовательности постоянно увеличивающейся длины. Основой интерфейса являются четыре разноцветные кнопки, каждая из которых имеет свой собственный? Кнопки загораются случайным образом, создавая последовательность, которую должен повторить игрок. Если игрок успешно выполнил задание и правильно запомнил последовательность, он переходит на следующий шаг, где ему предлагается идентичная последовательность с одним дополнительным элементом. Если игрок ошибается, игра останавливается, и надо начинать сначала. Цель игры – набрать как можно больше очков, каждый новый элемент последовательности дает больше очков.

**KGeography.** KGeography – образовательное программное обеспечение, входящее в пакет образовательных программ KDE Edutainment Project. Распространяется на условиях GNU General Public License. KGeography обучает географии. Оно дает знания о политическом делении определенных стран и регионов (регионы, столицы регионов и их флаги). Основной сайт программы – <http://kgeography.berlios.de/>. По следующему адресу <http://kgeography.berlios.de/maps.php> вы сможете найти карты стран, которые могут отсутствовать в установленном у вас варианте.

**KTtouch.** KTouch – клавиатурный тренажер, входящий в пакет образовательных программ KDE Edutainment Project. Распространяется согласно GNU General Public License. KTouch обучает методу слепой печати. Возможности программы:

- Поддержка различных раскладок клавиатур.
- Возможность добавления новой раскладки.
- Редактирование упражнений.
- Детальная статистика по результатам выполняемых упражнений.
- Подсчет скорости и качества (аккуратности) печати.

**KmPlot.** KmPlot – графопостроитель алгебраических функций для интегрированной среды KDE. Он имеет встроенный мощный интерпретатор. Предназначен для построения разных функций одновременно и комбинирования их выражения для построения новых функций.

**KTurtle.** KTurtle – это образовательная программная оболочка для изучения языка программирования на языке Logo, которая позволяет программировать максимально легко и просто. Оболочка содержит визуального исполнителя, что позволяет сразу видеть результат выполнения программы. KTurtle может использоваться для обучения основам программирования детей как младшего, так и среднего школьного возраста. Имеется возможность писать команды как на английском, так и на русском языке.

KTurtle обладает замечательными особенностями, которые позволяют начать программировать легко и непринуждённо:

1. Встроенный интерпретатор Logo устраняет необходимость скачивать и устанавливать дополнительные программы.

2. Выполнение можно замедлить и остановить в любое время

3. Имеется мощный редактор команд Logo с подсветкой синтаксиса, нумерацией строк и многим другим.

4. Холст с результатами работы программы может быть сохранен как изображение или распечатан.

5. Холст имеет функцию переброса Черепашки на другой край, когда она достигнет первого.

6. Контекстная подсказка по всем командам Logo, которая вызывается простым нажатием F2.

7. Все Logoкоманды могут быть переведены на любой язык.

8. Имеется диалог с сообщениями об ошибках, облегчающий процесс отладки.

**Раздел «Языки».** Раздел включает в себя программы, способствующие развитию мышления и словарного запаса обучающихся на родном и иностранных языках.

- KAnagram – генератор анаграмм; программа переставляет буквы в слове, которое берет из заранее составленных словарей.

- KHangMan – известная игра в виселицу; предлагается отгадать слово способом “буква за буквой”, на отгадывание дается 10 попыток, слова берутся из заранее составленных словарей.

- KLettres – программа для изучения алфавита; помогает выучить алфавит иностранного языка и произношение простейших слогов (буквосочетаний).

- KVerbos – программа для изучения форм глаголов испанского языка; программа выводит глагол и его время, пользователь вводит формы глагола, затем программа исправляет ошибки пользователя.

- KVocTrain, KWordQuiz – программы, предназначенные для изучения новых иностранных слов; работа программы основана на известной методике двусторонних карточек.

- KLatin – программа, помогающая изучить латинский язык; тренировка лексики, грамматики, изучение форм глаголов.

- Kiten – японо-английский и англо-японский словарь.

**Список использованных источников:** 1. *KGeography* [Electronic Resource]. – Mode of access : <http://edu.kde.org/kgeography/>. 2. *KGeography/Making Maps* [Electronic Resource]. – Mode of access : [https://userbase.kde.org/KGeography/Making Maps](https://userbase.kde.org/KGeography/Making_Maps). 3. *Interactive Periodic Table* [Electronic Resource]. – Mode of access : <http://touchspin.touchspin.com/DisplayTable.php>. 4. *Обзор* некоторых образовательных программ в ОС Linux на примере KdeEdu и Gcompris (Обзор образовательных пакетов) : учебное пособие / [Беккерман Е. Н., Жексенаев А. Г., Ковригина Е. В. и др.] ; под ред. Ю. П. Немчиновой. – М., 2008. – 103 с.

УДК 528:001 + 681.51

## **АНАЛІЗ ТЕНДЕНЦІЙ РОЗВИТКУ ГЕОІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ**

В. Позній<sup>1</sup>, С. Кривоберець<sup>1</sup>, В. Зацерковний<sup>2</sup>

*Національний авіаційний університет, м. Київ, Україна*

*Чернігівський державний інститут економіки і управління, м. Чернігів, Україна*

Сьогодні у світі розроблені і використовуються понад 600 різноманітних геоінформаційних програмних продуктів в управлінських структурах, сільському господарстві, державних і комерційних підприємствах, силових структурах, кадастрах, наукових та освітніх установах тощо, які володіють різними функціональними можливостями. За оцінками консорціуму компаній Data+ і Esri CIS, аналітичної компанії Daratech світовий ринок ГІТ за останні 15 років щорічно зростає у середньому на 10 – 15% (рис.1) [1].



Рис.1. Динаміка зростання ринку геоінформаційних продуктів у світі

На думку аналітиків, попит на геоінформаційні і геопросторові продукти стимулює потребу в просторовій інформації. З появою численних географічних сайтів, таких як Google Earth, і навігаційних систем підприємства і користувачі зрозуміли, яку користь приносять територіальні дані, і додатки з підтримкою ГІС одержали широке поширення.

Основною причиною зацікавленості до ГІС фахівці вважають зміни, що відбулись на ринку геоінформаційних продуктів. З одного боку, геоінформаційні продукти стали більш прозорими для користувачів, тобто стало зрозуміло, навіщо вони потрібні, як їх впроваджувати і як застосовувати. З іншого боку, завдяки реалізації масштабних державних проєктів з'явилося більше доступних просторових даних, які можна використовувати в інших додатках [4].

Геоінформаційні додатки стали сьогодні у світі інтегральним ресурсом для виконання різних функцій урядів і органів влади, в тому числі у сфері суспільних робіт (комунально-експлуатаційні служби, фінанси, забезпечення громадської безпеки, економічного розвитку тощо). Так, наприклад:

- 77% респондентів використовують цю технологію для прив'язки і перегляду даних аерофотознімання;
- 70% використовують її для підтримки управління даними про нерухому власність і в оподатковуванні;
- 57% використовують засоби ГІС для забезпечення публічного доступу до інформації;
- 41% використовують ГІС при розподілі капіталовкладень, інвестицій, для розвитку інфраструктури муніципалітетів і планування територій тощо;
- 38% використовують для підтримки процедури видачі дозволів, при розробці і реалізації заходів по реагуванню на надзвичайні ситуації;
- 33% використовують для спільного використання з САПР;
- 28% використовують при підтримці діяльності правоохоронних органів [3].

Незважаючи на велике розмаїття програмного забезпечення (ПЗ), що пропонується ринком геопродуктів, на сьогодні не існує стандартизованої ГІС, яка б враховувала всі потреби користувача. Інформація, що накопичується, дуже часто не може бути порівняною, є різномірною та розмежованою; в деяких регіонах використовуються несертифіковані програмні засоби, багато інформації дублюється. Крім того, ліцензійне ПЗ не є дешевим, що аж ніяк не сприяє його широкому впровадженню.

Створення повноцінної ГІС (як свідчить досвід) можливе тільки на основі програмного забезпечення, спеціально спроектованого для задач управління територіями. Тобто повинен бути сформований комплекс програмних засобів, який би забезпечував ефективне збереження великих об'ємів просторових даних, організацію віддаленого

доступу до централізованого або розподіленого сховища даних у режимі клієнт-сервер, набір клієнтських додатків різної функціональності, зручний настроюваний інтерфейс користувача, можливість створення спеціалізованих додатків тощо.

Перспективним у цьому напрямку є впровадження відкритих настільних ГІС – Open Source Initiative (OSI). Основними особливостями відкритого програмного забезпечення (ПЗ) згідно з визначенням організації Open Source Initiative є вільне поширення, доступний вихідний код, дозвіл на модифікацію цього вихідного коду [3].

Сьогодні вільне ПЗ стало символом інновацій і прогресу. Вільне використання, зміна і розповсюдження ПЗ і його вихідних кодів гарантоване підтримкою вільного обміну ідеями між користувачами і розробниками. Для ПЗ ГІС це означає послідовний міжнародний розвиток з орієнтуванням на відповіді запитам користувачів за короткий час.

Хронологія створення відкритого ПЗ ГІС представлена на рис.2 [2].

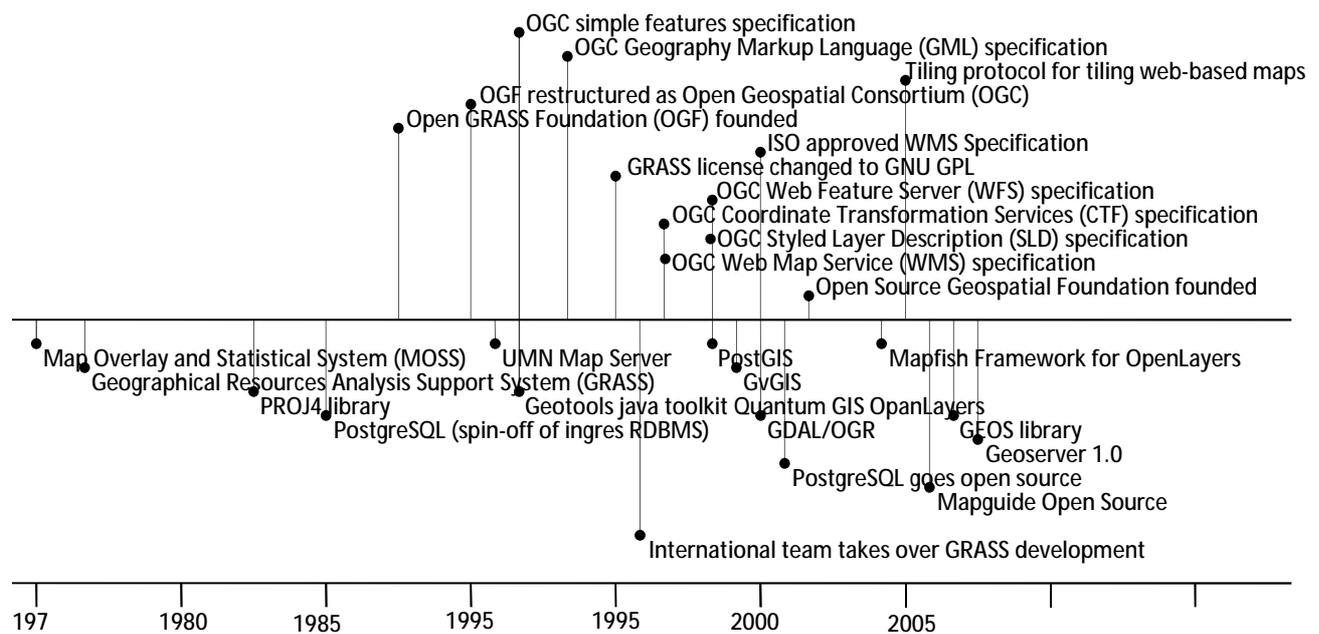


Рис. 2. Хронологія розвитку відкритого ПЗ ГІС

Відкриті ГІС, як і відкрите ПЗ, відбивають сучасну тенденцію зменшення залежності користувача програмного забезпечення від розробника (так званий *vendor lock-in*). Очевидно, що ця проблема відносна, оскільки тривалий досвід використання ПЗ (відкритого або закритого) тим або іншим чином приводить до вибудовування навколо нього технологічної лінійки, ціна переходу з якої може виявитися більшою, ніж освоєння нового ПЗ. Однак в умовах відкритості користувачу відкритого ПЗ ГІС гарантована можливість внесення необхідних йому змін самостійно [5].

Відкриті ГІС не ставлять під загрозу існування інших, пропрієтарних програмних продуктів, проте забезпечують кращі умови конкуренції. Вигода у використанні такого ПЗ, з одного боку, особливо очевидна для невеликих, некомерційних і громадських об'єднань і компаній, для дослідницьких, державних й інших організацій з великою кількістю філіалів, де достатньо обмеженої функціональності. З іншого боку, відкриті ГІС являють собою новий інструмент конкурентної боротьби для компаній, основний прибуток яких отримується не від продажу ПЗ (наприклад, компаній-інтеграторів).

Використання відкритих ГІС здатне різко зменшити видатки і підсилити конкуренцію. Незважаючи на це, недоліки, притаманні відкритому ПЗ ГІС, на даний момент перешкоджають негайному його впровадженню в організаціях в якості основного ПЗ ГІС.

**Список використаних джерел:** 1. Daratech, Inc. [Електронний ресурс] / режим доступу: <http://www.directionsmag.com/pressreleases/gisgeospatial-industry-worldwide-growth->

slows-to-1-in-2009/119103. 2. Open Geospatial Consortium [Електронний ресурс] / I. Daratech. – Режим доступу: <http://www.opengeospatial.org/ogc/>. 3. Open Source Initiative [Електронний ресурс] / Режим доступу: <http://opensource.org/>. 4. Esri [Електронний ресурс]. – Режим доступу: <http://www.esri.com/>. 5. GeoTools [Електронний ресурс] / Режим доступу: [http://geoportal.e-reg36.com/index.php?id=59&tx\\_ttnews%5Btt\\_news%5D=143&cHash=1](http://geoportal.e-reg36.com/index.php?id=59&tx_ttnews%5Btt_news%5D=143&cHash=1).

УДК 004.457:621.396

## **ВІЛЬНЕ ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ ДЛЯ МОБІЛЬНИХ ПРИСТРОЇВ НА ПРИКЛАДІ ВІЛЬНОГО ФРЕЙМВОРКУ RAPIDSMS**

А. Посадська

*Чернігівський державний інститут економіки та управління*

Сьогодні в середовищі користувачів і розробників спостерігається значне зростання популярності вільного програмного забезпечення (ПЗ) з відкритим вихідним кодом у сфері логістики. Його популярність зумовлена безкоштовністю та можливістю збереження даних у відкритих форматах.

В Україні вільне ПЗ для логістики досі не так широко відомо порівняно з комерційним програмним забезпеченням, таким як Oracle, ArcGIS.

Особливості, що перераховані вище, роблять актуальною і перспективною розробку фреймворку RAPIDSMS, яка створена на основі вільного програмного забезпечення.

Як завжди на шляху успішності компаній стоять проблеми підвищення продуктивності інтегрованих транспортних систем, отримання якісної продукції та інформації на ієрархічних рівнях, суттєве зниження сукупних затрат. Саме тому застосування логістичних інформаційних систем є актуальним.

**Список використаних джерел:** 1. *Hribernik K. A. Dynamics in Logistics: Second International Conference, LDIC 2009, Bremen, Germany, August 2009 Proceedings / K. A. Hribernik, C. Hans, and K.-D. Thoben // The Application of the EPCglobal Framework Architecture to Autonomous Control in Logistics. – 2009. – P. 365-385.* 2. *Нагіна Е. К. Информационная логистика. Теория и практика / Е. К. Нагіна, В. А. Ищенко. – Воронеж : ИПЦ ВГУ, 2007. – 87 с.* 3. *ArcGIS Resources [Електронний ресурс]. – Режим доступу : <http://resources.arcgis.com/en/communities/>.*

УДК 004.451:004.7

## **ОГЛЯД ТА ХАРАКТЕРИСТИКА ВІЛЬНОГО ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ДЛЯ ЗАДАЧІ АНАЛІЗУ МЕРЕЖНОГО ТРАФІКУ**

М. Тевкун, І. Посадська

*Чернігівський національний технологічний університет*

Аналізатор трафіку – це програма або програмно-апаратний комплекс, що виконує функцію аналізу (з попереднім перехватом або без нього) мережного трафіку, який призначено іншим мережним пристроям.

Аналізатори та сніффери використовуються для визначення, локалізації та виправлення помилок, обслуговування мереж (бездротових, локальних, розподілених тощо). Інтегровані декодери протоколів та експертні системи сніфферів дозволяють швидко визначити та виправити різноманітні мережні проблеми, мінімізуючи при цьому вплив на продуктивність і роботу мережі. Окрім цього виконується функція ведення детальних статистичних даних щодо мережної діяльності, що в подальшому можуть бути

використані системними адміністраторами. Використання сніфферів дає можливість передивлятися усі пакети, що передаються по всім мережним з'єднанням та інтерфейсам. Сучасні сніфтери забезпечують представлення детальної інформації через зручні та інтуїтивно зрозумілі графічні інтерфейси.

Аналіз мережного трафіку за допомогою сніфферу використовується для вирішення таких задач, як:

- визначення паразитного, вірусного або кільцевого трафіку, що збільшує завантаження мережного обладнання та каналів зв'язку;
- перехват будь-якого незахищеного (а у випадках несанкціонованого доступу і захищеного) трафіку з метою отримання з нього інформації;
- локалізація помилок мережі або помилку конфігурації мережних агентів.

Оскільки в класичних сніфферах проводиться лише базовий аналіз трафіку з використанням лише найпростіших засобів автоматизації, то використання сніфферів доцільно лише для невеликого об'єму трафіку.

У цій роботі розглядаються чотири найбільш популярні сніфтери, що є вільним програмним забезпеченням.

**Wireshark** (раніше – Ethereal) – програма-аналізатор трафіку, що використовується в комп'ютерних мережах Ethernet та деяких інших. Має графічний інтерфейс користувача. Утиліта Wireshark має інтегровані декодери найрізноманітніших мережних протоколів, що дозволяє розібрати пакет та відобразити значення кожного поля протоколів будь-якого рівня. Обмеженням роботи цієї утиліти є можливість захвату даних лише тих мереж, що підтримується бібліотекою pcap, так як саме ця бібліотека використовується для моніторингу трафіку. У той же час Wireshark підтримує можливість роботи з великою кількістю форматів вхідних даних, що дає можливість працювати із файлами, які були сконфігуровані іншими системами захвату, що значно розширює його функціональні можливості.

**Iris Network Traffic Analyzer** є програмним забезпеченням, яке окрім стандартних функцій збору, фільтрації та пошуку пакетів, побудови звітів тощо, має функцію реконструювання даних. Аналізатор трафіку Iris Network Traffic Analyzer дає можливість детально реконструювати сеанси роботи користувачів із різноманітними web-ресурсами. Однією з функцій цієї програми є можливість імітації відправки паролів для доступу до захищених веб-серверів за допомогою cookies. Унікальна технологія реконструювання даних, що реалізована в модулі дешифрування, трансформує сотні зібраних двійкових пакетів в звичну око користувача форму (електронні листи, web-сторінки, повідомлення ICQ тощо). Утиліта аналізатору eEye Iris дозволяє проглядати незашифровані повідомлення web-пошти та програм негайного обміну повідомленнями, розширюючи при цьому можливості існуючих засобів моніторингу та обліку. Аналізатор пакетів eEye Iris також дозволяє зафіксувати різноманітні деталі вторгнення, такі як дата, час, використані порти, IP-адреси і DNS-імена комп'ютерів порушника та жертви тощо.

**Ethernet Internet traffic Statistic** - це програма, що показує кількість отриманих та переданих даних (всього або за останню сесію), а також швидкість підключення. Для більшої зручності отримані дані відображаються на графіку в режимі реального часу. Програма має допоміжну утиліту, що виконує функцію контролю за станом мережної активності, а саме – показує кількість отриманих та переданих даних та рахує статистику за різні інтервали часу.

**CommTraffic** – це мережна утиліта для збору, обробки та відображення статистики мережного трафіку для модемного (dial-up) або виділеного з'єднання. У випадку моніторингу сегменту локальної мережі аналізатор CommTraffic показує дані щодо трафіку, обробленого кожним комп'ютером у мережі. Перевагою цієї програми є зручний та зрозумілий інтерфейс, простий у налаштуванні, що відображає статистику у вигляді графіків.

Далі представлена таблиця порівняння деяких характеристик розглянутих програм (табл.).

Таблиця

*Порівняльна характеристика аналізаторів трафіку*

Параметри	Wireshark	Iris Network Traffic Analyzer	Ethernet Internet traffic Statistic	CommTraffic
Наявність графіка швидкості	+	+	–	–
Наявність графіка трафіку	–	–	+	+
Експорт файлів	+	–	–	–
Мінімальний крок отримання даних	0,001 с	1 с	1 с	1 с
Можливість зміни мінімального кроку	+	+	–	–

Із представленої таблиці видно, що кожна з розглянутих програм має свої переваги та недоліки і вибір тієї чи іншої програми залежить від поставлених задач.

**Список використаних джерел:** **1.** Chankhantod A., Danzig P.B. and Neerdaels C. “A Hierarchical Internet Object Cache”, *Proceedings*, USENIX. 1996. **2.** Mark Ward. “How the web went world wide”, Technology Correspondent, BBC News [Online]. Available: <http://news.bbc.co.uk>. **3.** Wireshark website [Online]. Available: <http://www.wireshark.org>. **4.** Iris Network Traffic Analyzer website [Online]. Available: <http://iris-network-traffic-analyzer.en.softonic.com>. **5.** CommTraffic website [Online]. Available: <http://commtraffic-pro.en.softonic.com>.

УДК 004.4:004.9

## ОСНОВНІ МОЖЛИВОСТІ ОБРОБКИ ЗВУКОВИХ ДАНИХ ЗА ДОПОМОГОЮ АУДІОРЕДАКТОРА AUDACITY

Є. Тітов, А. Роговенко

*Чернігівський національний технологічний університет*

Audacity – вільний багатоплатформений редактор звукових файлів з відкритим серцевим кодом, орієнтований на роботу з декількома доріжками. Дозволяє виконувати такі функції, як редагування, аудіозапис, оцифрування звуку, зміна параметрів файлу, мікшування треків і застосування ефектів.

Програма була випущена і розповсюджується на умовах GNU GPL.

Редактор Audacity забезпечує виконання наступних функцій:

- імпорт та експорт файлів WAV, MP3 (з використанням кодувальника LAME MP3), Ogg Vorbis, FLAC та інших форматів;
- запис з мікрофону, лінійного входу та інших джерел;
- запис з одночасним прослуховуванням наявних доріжок;
- запис до 16 каналів одночасно (необхідна наявність багатоканальної звукової карти);
- ефекти і розширення як у комплекті постачання, так і встановлювані окремо (LADSPA або на функціональній мові Nyquist);
- індикатори рівня запису і відтворення;
- зміна темпу зі збереженням висоти тону;
- зміна висоти тону зі збереженням темпу;
- вилучення шуму згідно попередньо створеної моделі;
- відтворення великої кількості доріжок одночасно;

- зведення доріжок з різними якісними характеристиками з автоматичним перетворенням до характеристик проекту в режимі реального часу;
- запис результатів зведення у файли різних форматів, підтримуваних бібліотекою libsndfile.

В Audacity використані бібліотеки, які поширюються під різними ліцензіями:

- графічний інтерфейс базується на бібліотеці wxWidgets;
- звуковий двигун використовує PortAudio;
- опціональна підтримка формату MP3 реалізована за рахунок бібліотек libmad та lame.

Що стосується підтримки пропріетарних технологій, то в цьому випадку накладено ряд обмежень:

- підтримка ASIO у Windows відсутня через несумісну з GPL ліцензію засобів розробки;
- через (знову ж таки) несумісну з GPL ліцензію засобів розробки, підтримка VST відсутня за умовчанням, але включається з вантаженням додаткового розширення, оригінальний інтерфейс розширень VST не підтримується;
- через вимоги Thomson Multimedia виплачувати збори за засоби кодування MP3, функція збереження файлів у цьому форматі працює тільки за наявності окремо встановленої бібліотеки lame.

Набір вбудованих плагінів та інструментів Audacity дозволяють виконувати найбільш часто необхідні операції над звуковими даними. Типовими процедурами є: вирізання небажаної частини аудіо, копіювання, вставка, створення фрагменту тиші, зменшення рівня шуму відповідно до заданої моделі, нормалізація рівня сигналу, плавне зростання та плавне спадання рівня гучності сигналу на початку доріжки та в кінці відповідно, вирівнювання амплітудно-частотної характеристики бажаної зони файлу та ін.

Із перелічених вище дій найактуальнішою для масового використання є операція зменшення рівня шуму. Це, перш за все, пов'язано з тим, що для нормальної якості запису, наприклад, мовного сигналу необхідно мати далеко не бюджетні мікрофон та додаткові зовнішні пристрої, які забезпечують низький рівень фонового шуму. Все ж таки в більшості випадків запис ведеться вбудованим або зовнішнім мікрофонами, що мають не дуже гарні показники щодо якості.

Щоб виконати описану вище обробку, необхідно, перш за все, знайти ділянку, яка б найкраще слугувала для побудови моделі шуму. Коли програма створила відповідний профіль, то його можна застосовувати до бажаних частин або всього файлу.

Також із інших досить-таки важливих можливостей застосування вбудованих ефектів є нормалізація рівня сигналу, тобто збільшення значення миттєвих амплітуд на бажаній ділянці, що (в свою чергу) призведе до збільшення гучності звуку відповідної зони. При виконанні вказаних маніпуляцій потрібно бути обережним, тому що в деяких випадках може мати місце такий ефект спотворення сигналу, як кліппінг (у разі занадто великого значення рівня нормалізації).

Стосовно засобів аналізу сигналів, то Audacity має можливість побудови амплітудно-частотної характеристики (АЧХ, спектр амплітуд) за допомогою застосування швидкого перетворення Фур'є, підтримуючи різні форми вікон. Дана функція буде особливо корисною тим, хто займається вивченням впливу складових коливань на досліджуваний сигнал.

Audacity має мікшер, що дозволяє зводити декілька доріжок в одну, регулюючи параметри впливу кожної на результуючу. Після досягнення необхідних властивостей кінцевий результат можна отримати за допомогою такої операції, як рендерінг.

Основні переваги аудіоредактора Audacity такі: він є безкоштовним, має інтуїтивно зрозумілий інтерфейс, який дозволяє без особливих зусиль розібратися навіть не дуже досвідченій в обробці звуку людині, задля реалізації своїх потреб. Саме тому Audacity і став таким популярним серед користувачів з усього світу.

**Список використаних джерел:** 1. *Audacity* (audio editor) [Електронний ресурс]. – Режим доступу : [http://en.wikipedia.org/wiki/Audacity\\_\(audio\\_editor\)](http://en.wikipedia.org/wiki/Audacity_(audio_editor)). 2. *Audacity* official website [Електронний ресурс]. – Режим доступу : <http://audacity.sourceforge.net/>. 3. *Audacity* Manual Contents [Електронний ресурс]. – Режим доступу : <http://manual.audacityteam.org/o/>. 4. *Горюнов В. А.* Обработка и монтаж аудиозаписей с использованием Audacity (ПО для обработки и монтажа аудиозаписей) : учебное пособие / В. А. Горюнов, А. Н. Стась. – М., 2008. – 40 с.

УДК 004.4

## ИСПОЛЬЗОВАНИЕ GOOGLE MAPS API ПРИ РАЗРАБОТКЕ РЕАЛЬНОГО ПРИЛОЖЕНИЯ

С. Ткаченко, С. Стасюк

*Черниговский национальный технологический университет*

В наши дни люди часто путешествуют. Во время своих поездок они пытаются посетить как можно больше интересных мест, но иногда бывает так, что рядовой турист не знает, где это место. Раньше эта проблема решалась на уровне контакта с местными жителями. Но как можно общаться с местными, если, например, не знать их язык?

С развитием картографии и GPS-навигации процесс поиска нужных мест стал невероятно простым. Достаточно ввести интересующее вас место и мобильное устройство или компьютер выдаст вам нужную информацию. А что делать, если хочется самостоятельно как-то настроить карту, поставить отметку, запомнить место, где вы побывали? В этом случае вам поможет Google Maps API. Вызвав несколько библиотечных функций, вы можете с легкостью создать уникальное приложение. Почти вся работа за вас уже реализована корпорацией Google.

Что же требуется для работы с Google Maps API? Прежде всего определиться, какое приложение вы будете делать. Если мобильное (например, под Android), то несомненно необходимо использовать библиотеку для Java. Если вы работаете с web-сайтом, то подходит JavaScript. Рассмотрим API для JavaScript.

Далее предоставлен пример, который реализует изначальную установку 2-х маркеров, находит расстояние по прямой между этими маркерами и кратчайший путь, нарисованный полилинией. Помимо этого добавлена возможность ставить свои маркеры на карте по клику.

*Инициализация:*

```
<script src="http://maps.googleapis.com/maps/api/js?sensor=false&libraries=geometry"></script><script src="https://maps.googleapis.com/maps/api/js?v=3.exp&sensor=false&libraries=places"></script><script>function initializeMap() {var place = new google.maps.LatLng(51.5025, 31.2703);var mapOptions = {zoom: 12,center: place};var map = new google.maps.Map(document.getElementById('map-canvas'), mapOptions);addMarkers(map,place);addMarkerOnClick(map);}google.maps.event.addListener(window, 'load', initializeMap);</script>
```

...

```
<div id="map-canvas"></div>
```

Прежде всего нужно подключить библиотеки с API и произвести инициализацию карты. Для инициализации я выбрал случайную точку в Чернигове и использовал 12-кратный зум. Затем добавил на карту маркеры и обработчик добавления новых маркеров по клику. В самом конце производится добавление карты на web-страничку. Сама карта размещается с помощью любого тега (обычно div). Для этого нужно указать тегу id="map-canvas".

*Реализация:*

```
function addMarkers(map,place){
  var markerA = new google.maps.Marker({
    position: place, map: map,
    title: 'Случайная точка '
  });
  var markerBPlace = new google.maps.LatLng(51.505, 31.335);
  getPath(map,place,markerBPlace);
  var markerB = new google.maps.Marker({
    position: markerBPlace, map: map,
    title: 'ЧНТУ '
  });
  var contentDistance =
google.maps.geometry.spherical.computeDistanceBetween(place,markerBPlace);
  var contentString = "Расстояние между А и В: ";
  contentString+=Math.round(contentDistance);
  contentString+=" м.";
  var infowindow = new google.maps.InfoWindow({
    content: contentString
  });
  google.maps.event.addListener(markerB, 'click', function() {
    infowindow.open(map,markerB);
  });
}

function getPath(map,placeA,placeB) {
  var directionsService = new google.maps.DirectionsService();
  var directionsDisplay = new google.maps.DirectionsRenderer({
    'map': map, polylineOptions: {
      strokeColor: '#660033',
      strokeOpacity: 0.8
    }
  });
  var request = {
    origin: placeA, destination: placeB,
    travelMode: google.maps.TravelMode.DRIVING
  };
  directionsService.route(request, function (response, status) {
    if (status == google.maps.DirectionsStatus.OK) {
      directionsDisplay.setDirections(response);
    }
  });
}
```

```

function addMarkerOnClick(map){
  google.maps.event.addListener(map, 'click', function (event){
    placeMarker(map,event.latLng);
  });
}
function placeMarker(map,location){
  var newMarker = new google.maps.Marker({
    position: location,
    map: map
  });
  addMarkerInfo(map,newMarker);
}

var searchStringResult;
function addMarkerInfo(map,newMarker){
  var nearlyPlaces = new google.maps.places.PlacesService(map);
  var searchRequest = {
    location: newMarker.getPosition(), radius:200
  };
  nearlyPlaces.search(searchRequest, searchCallback);
  var infowindow = new google.maps.InfoWindow({
    content: searchStringResult
  });
  google.maps.event.addListener(newMarker, 'click', function() {
    infowindow.open(map,newMarker);
  });
}
function searchCallback(results, status) {
  if (status == google.maps.places.PlacesServiceStatus.OK) {
    var placesNames="Места: \n";
    for (var i = 0; i < results.length; i++) {
      placesNames+=results[i].name;
      placesNames+="; \n ";
    }
  }
  searchStringResult=placesNames;
}

```

Рассмотрим функцию добавления маркеров на карту. Один из маркеров будет находиться в ранее выбранной случайной точке, которую мы установили центром карты (это маркер А). Второй маркер будет указывать на ЧНТУ (маркер В). Основными параметрами для создания маркеров является их позиция и карта, на которой их нужно отобразить. Для маркера В добавим информацию, которая выводится по клику на этот маркер. Информация будет содержать данные о расстоянии между А и В. Производится это функцией `computeDistanceBetween()`. Добавляем обработчик событий по клику на этот маркер и получаем нужный результат. Так же эта функция вызывает внутри другую (а именно `getPath()`) для нахождения краткого пути от А до В при условии, что вы едете на машине.

Функция `getPath()` создает сервис для нахождения этого пути, а также объект для рисования результата. Для выполнения этих задач используется функция `route`, которая принадлежит сервису. Первый параметр – точки, которые нужно соединить и способ соединения (например, пешком, на велосипеде, на машине), а второй параметр – `callback`

функция, в которой в нашем случае производится отрисовка результатов. Также я добавил создание маркеров в том месте, где нажали мышкой. Для этого был использован обработчик событий по клику, который вызывает функцию `placeMarker()`. Обработчик передает в `placeMarker()` карту, на которую кликнули, и позицию клика.

Функция `placeMarker()` осуществляет добавления маркера и поиск близлежащих мест в радиусе 200 метров, которые записываются в виде информации к маркеру. Для поиска мест используется `PlacesService` и его метод `search()`. Поиск происходит по критериям, заданным в переменной `searchRequest`. Обработка найденных мест производится `callback` функцией `searchCallback()`. Результат поиска заносится в глобальную переменную `searchStringResult`, которая затем передается как содержимое подписи маркера.

При работе с любой библиотекой не бывает все столь гладко:

1. Нельзя инициализировать карту во внешнем файле, что неудобно для работы.
2. Существует много разных API библиотек, поэтому необходимо смотреть, какие библиотеки нужны классам, которые используются в приложении/web-сайте.
3. Callback-функции не возвращают значений, поэтому нужно использовать глобальные переменные для возврата значений.
4. Расстояния измеряются напрямую, а значит преграды, которые увеличивают расстояние, не учитываются.

**Список использованных источников:**

<a href="https://developers.google.com/maps/documentation/?hl=ru">https://developers.google.com/maps/documentation/?hl=ru</a>	1.	Режим	доступа	:
<a href="http://www.w3schools.com/googleapi/default.asp">http://www.w3schools.com/googleapi/default.asp</a>	2.	Режим	доступа	:
<a href="http://econym.org.uk/gmap/">http://econym.org.uk/gmap/</a>	3.	Режим	доступа	:

УДК 004.418

## ПРОЕКТ СИСТЕМИ ЗАХИСТУ ВЕБ-РЕСУРСІВ ВІД DDOS-АТАК

О. Федоров, А. Хижняк

*Чернігівський національний технологічний інститут*

Тема «захист інтернет-ресурсів» є актуальною проблемою сучасності. Мережа Інтернет стала частиною майже усіх сфер діяльності людини і є її невід'ємною складовою. І кожна частина піддається впливу. У випадку з мережею Інтернет, найпростіший спосіб вивести цю ланку із ладу – зробити DoS/DDoS-атаку.

Так як Інтернет не має внутрішніх механізмів, які б дозволили звести нанівець такий вид діяльності, то захист перекладається на плечі програмістів та адміністраторів. Так склалось, що програмного забезпечення, яке може відтворити такий тип атак, більше, ніж програмного забезпечення, яке б могло надати захист. Багато компаній з усього світу працюють над системами захисту для веб-ресурсів. Готові програмні комплекси, які б могли надати ефективний рівень захисту, дуже дорогі і недоступними для усіх. У цьому випадку випадає сегмент малого бізнесу, капітал якого спрямований на розвиток, а не на захист.

Головна ідея цього проекту – створити та надати інструмент захисту веб-ресурсів, який був би простий у використанні й у вільному доступі для усіх бажаючих.

У системі захисту можна виділити три базових компоненти системи (рис. 1).



Рис. 1. Головні компоненти системи

Це платформа – компонент системи, на якому вона базується; база даних – компонент, який відповідає за збереження інформації про роботу системи та дані, які проходять через неї; алгоритми – компонент системи, який відповідає за реалізацію аналізу вхідних даних, управлінням системою тощо.

Кожен компонент системи впливає один на одного, і фінальна продуктивність системи залежить від того, як ці компонент взаємодіють між собою.

Кожен компонент системи повинен розроблятися з використанням п'ятих характеристик: масштабування; швидкості дії; керованість; надійність; вартість.

В готовому варіанті дана система представляє собою проксі-мережу, головне завдання якої – очищення вхідного трафіку, аналізування вхідних, передбачання майбутніх атак. Принцип роботи системи можна ілюструвати (рис. 2).



Рис. 2. Принцип роботи системи

Розвиток таких систем захисту дозволить піднятися на одну ступінь вище в захисті внутрішніх ресурсів мережі Інтернет. Для більш швидкого розвитку даний проект повинен бути вільним, доступним та простим у використанні.

**Список використаних джерел:** 1. S. Agarwal, T. Dawson, C. Tryfonas DDoS Mitigation via Regional Cleaning Centers. – 2011. 2. Мельников Д. А. Информационная безопасность открытых систем / Д. А. Мельников. – М. : ФЛИНТА, 2012. – С. 448. 3. Устоять любой ценой. Методы борьбы с DoS/DDoS-атаками [Электронный ресурс]. – Режим доступа : <http://hacker.ru/articles/49752/>. 4. Касперски К. Компьютерные вирусы изнутри и снаружи / Крис Касперски. – СПб. : Питер, 2006. – С. 526. 5. David Dittrich, Jelena Mirkovic, Peter Reiher, Sven Dietrich Internet Denial of Service: Attack and Defense Mechanisms. – 1. – М. : Pearson Education, 2004. – С. 400. 6. Фергюсон Н. Практическая криптография / Н. Фергюсон, Б. Шнайер. – М.: Вильямс, 2005. – С. 416. 7. J. Reynolds, R. Braden Request for Comments (RFC). – 2004.

УДК 004.4

## ПРОЕКТИРОВАНИЕ УСТРОЙСТВ НА БАЗЕ ПЛАТФОРМЫ ARDUINO С ПОМОЩЬЮ ОТКРЫТОГО ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ

В. Шаповал, А. Роговенко

*Черниговский национальный технологический университет*

Arduino – это электронный конструктор и удобная платформа быстрой разработки электронных устройств для новичков и профессионалов. Устройство программируется через USB без использования программаторов. Arduino позволяет компьютеру выйти за рамки виртуального мира в физический и взаимодействовать с ним. Устройства на базе Arduino могут получать информацию об окружающей среде посредством различных датчиков, а также могут управлять различными исполнительными устройствами.

Платформа Arduino пользуется огромной популярностью во всем мире благодаря удобству и простоте языка программирования, а также открытой архитектуре и программному коду. И все бы хорошо.. А теперь представьте, что вы решили создать что-то свое и готовых шилдов для этого не существует в природе. Пусть это даже не будет

устройство для перехвата смс-сообщений из соседней галактики, а простой шилд, на котором Вы хотите разместить экран со всей обвязкой и пару датчиков.

Оказывается, все придумали уже за нас. Для создания макетов проектов есть множество программного обеспечения. Вот некоторые из них:

1) VirtualBreadboard – хорошая программа для проектирования устройств на базе платформы Arduino, которая еще позволяет эмулировать работу платформы в режиме реального времени. И все бы хорошо, но это коммерческий проект, который стоит денег.

2) Proteus - пакет программ для автоматизированного проектирования (САПР) электронных схем. Разработка компании Labcenter Electronics (Великобритания). Очень мощный инструмент, но данный пакет является коммерческим. Бесплатная ознакомительная версия характеризуется полной функциональностью, но не имеет возможности сохранения файлов.

3) Fritzing – Open Source проект, который позволяет легко создавать прототипы устройств и имеет множество плат Arduino и различной периферии.

Fritzing – это простая в работе программа, заточенная под Arduino-проекты и предназначена для разработки электронных устройств от прототипа в виде макетной платы до конечного продукта в виде печатной платы. Программный пакет Fritzing был разработан в 2009 году в Потсдамском университете прикладных наук за счет субсидий, выделяемых государством на исследования научной программы под названием «From prototype to product» (от прототипа к продукту). В настоящее время создатели САПР предлагают талантливым программистам сотрудничать с ними по вопросам дальнейшего развития программы.

Устройства создаются из готовых элементов и узлов, которые включают в себя: все модели плат Arduino, различные микросхемы, транзисторы, диоды, резисторы, конденсаторы, макетные платы, батарейки.

Чтобы поместить нужный элемент на схему – нужно выбрать его из списка и перетащить на рабочее пространство левой кнопкой мышки. Проектировать устройство можно как в режиме макетной платы, так и в режиме принципиальной схемы. Второй вариант будет строиться автоматически.

Помимо всего этого вы получаете рисунок печатной платы для своей модели Arduino, который можно сохранить в PDF для последующей распечатки и изготовления по технологии ЛУТ. Кстати, модель Arduino можно легко менять в процессе разработки. Программа сама сделает новую разводку по нужным контактам. Таким образом вы получаете универсальный бесплатный инструмент с поддержкой русского интерфейса для создания своих проектов. Лазерно-утюжная технология (ЛУТ) – название технологии производства печатных плат в домашних условиях, использующей лазерный принтер для создания рисунка печатных проводников, и утюг для его переноса на плату. Метод основан на том обстоятельстве, что слой расплавленного тонера устойчив к действию травящего раствора.

Fritzing также поддерживает добавление новых элементов. Данное действие не представляет особой сложности: выбираем наиболее подходящий для нас элемент (так проще) и открываем редактор. Для создания нового элемента нам потребуются 4 картинки в векторном формате: картинка для макета, для принципиальной схемы, для печатной платы и иконка. При этом достаточно открыть нужное изображение и связать графику с выводами.

Выводы: Fritzing является бесплатным открытым пакетом, который является конкурентным аналогом дорогих коммерческих проектов. Он позволяет легко создавать макеты проектов и не требует для этого особых усилий. Программу легко освоить, для этого не потребуется много времени и вряд ли вам понадобится чтение документации по Fritzing. Она поможет в разработке не только новичкам, но опытным разработчикам.

**Список использованных источников:** 1. *Fritzing*. Electronic made easy [Электронный ресурс]. – Режим доступа : <http://fritzing.org/home/>. 2. Программный пакет Fritzing [Электронный ресурс]. – Режим доступа : <http://cxem.net/software/fritzing.php>. 3. Программное обеспечение для Arduino [Электронный ресурс]. – Режим доступа : <http://a-bolshakov.ru/index/0-145>.

## РАЗРАБОТКА ПРИЛОЖЕНИЙ ДЛЯ ANDROID В ECLIPSE IDE

Ю. Шоломий, В. Соломаха

Операционная система Android является одной из наиболее популярных платформ для мобильных устройств. Одним из ее основных преимуществ является открытость. Данная операционная система построена на основе открытого исходного кода и распространяется на свободной основе. Доступ к исходным кодам позволяет разработчикам максимально гибко использовать возможности платформы при разработке прикладного программного обеспечения.

При разработке приложений Android в большинстве случаев используется язык программирования Java – кроссплатформенный язык, позволяющий разрабатывать приложения, не зависящие от особенностей аппаратного обеспечения различных устройств. Для поддержки данного языка программирования в рабочей среде должен быть установлен фреймворк Java SE Software Development Kit, также необходима установка фреймворка Android Software Development Kit. Данный инструментальный набор включает средства для разработки, тестирования и отладки приложений Android. Android SDK распространяется на свободной основе и доступен по адресу [3].

Процесс разработки приложений может быть значительно ускорен при помощи интегрированных сред разработки. Одной из наиболее широко используемых сред является Eclipse. К ее преимуществам можно отнести открытость, высокую гибкость настройки среды за счет модульности и поддержку наиболее популярных языков программирования и фреймворков. В состав среды входит редактор кода, отладчик, также предоставляется поддержка систем контроля версий, автоматизированного рефакторинга. Интегрированная среда разработки Eclipse доступна по адресу [4].

Плагин ADT (Android Development Tools, Инструменты разработки Android-приложений) для Eclipse – это расширение интегрированной среды Eclipse, которое предоставляет возможность разработки, выполнения, отладки приложений для Android, доступа к Android SDK, а также подготовки и сборки приложения для публикации [1]. Плагин ADT также включает в себя визуальный инструмент, применяемый для создания графического интерфейса пользователя. Чтобы спроектировать новый интерфейс с помощью этого интерфейса, достаточно воспользоваться уже готовыми компонентами, однако существует возможность добавлять и пользовательские компоненты.

Рассмотрим компоненты Android SDK, к которым можно получить доступ с помощью ADT и их назначение. В состав Android SDK входят [2]:

1. Android Virtual Device Manager (AVD Manager).
2. Dalvik Debug Monitor Server (DDMS).
3. Traceview.
4. Hierarchy Viewer.
5. Android Lint.

Android Virtual Device Manager (AVD Manager) – менеджер виртуальных Android-устройств. Данный компонент представляет собой полнофункциональный эмулятор виртуальных машин Android. Эмулятор запускает полный стек системы Android, включая ядро Linux. Виртуальное устройство Android полностью моделирует все аппаратные и программные функции реального устройства. Виртуально могут совершаться звонки, отправляться сообщения, эмулироваться данные с различных датчиков (GPS, гироскоп, датчик освещенности).

Dalvik Debug Monitor Server (DDMS) – сервер отладки и мониторинга выполнения процессов. Данный компонент позволяет разработчикам отслеживать и взаимодействовать с виртуальными машинами и реальными устройствами Android. Доступно перенаправление

портов, захват экранного изображения, доступ к состоянию процессов и потоков, доступ к информации о использовании heap-памяти, файловый браузер, логгер и много других функций.

Traceview – компонент, позволяющий анализировать время процессора, потраченное на выполнение различных процессов.

Hierarchy Viewer – в состав Android SDK входит компонент, позволяющий отлаживать и оптимизировать графический интерфейс пользователя. Визуальные компоненты предоставляются в виде иерархии, для каждого из них доступно профилирование, показывающее время прорисовки компонента.

Android Lint – данный компонент предназначен для сканирования Android-приложения на наличие ошибок и уязвимостей. Отслеживаются также несоответствия расположения визуальных компонентов, использования ресурсов.

Таким образом, рабочая среда, состоящая из Java SE SDK, Android SDK, Eclipse IDE и ADT Eclipse plugin позволяет эффективно разрабатывать приложения Android, выполнять их отладку, тестирование, публикацию и поддержку. Все рассмотренные компоненты распространяются на свободной основе и поддерживаются различными сообществами и компаниями, в том числе и Google.

**Список использованных источников:** 1. *Android* для программистов: создаем приложения / П. Дейтел, Х. Дейтел, Э. Дейтел, М. Моргано. – СПб. : Питер, 2013. – 560 с. 2. *Android Apps with Eclipse* [Электронный ресурс]. – Режим доступа : [http://books.google.com.ua/books/about/Android Apps with Eclipse.html?id=AdZntwAACAAJ&redir\\_esc=y](http://books.google.com.ua/books/about/Android Apps with Eclipse.html?id=AdZntwAACAAJ&redir_esc=y). 3. *Get the Android SDK* [Электронный ресурс]. – Режим доступа : [developer.android.com/sdk/index.html](http://developer.android.com/sdk/index.html). 4. *Eclipse Luna (4.4)* [Электронный ресурс]. – Режим доступа : [www.eclipse.org/downloads/](http://www.eclipse.org/downloads/).

НАУКОВЕ ВИДАННЯ

# **ВІЛЬНЕ ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ В ОСВІТІ, НАУЦІ ТА БІЗНЕСІ**

**Тези доповідей п'ятої  
Міжнародної науково-практичної конференції**

Літературний редактор  
Коректор  
Комп'ютерна верстка і макетування

Л.М. Сила  
О.С. Смєлова  
В.М. Олефіренко  
Т.В. Коваленко

Підписано до друку 10.07.14. Формат 60x84/16. Умов. друк. арк. – 4,82.  
Тираж 100 пр. Замовлення № 225/14.

---

Редакційно-видавничий відділ Чернігівського державного технологічного університету  
14027, Україна, м. Чернігів, вул. Шевченка, 95.  
Свідоцтво про внесення суб'єкта видавничої справи до Державного реєстру видавців,  
виготівників і розповсюджувачів видавничої продукції  
серія ДК № 840 від 04.03.2002 р.