

Львівський національний університет імені Івана Франка  
Національний педагогічний університет імені М.П. Драгоманова  
Брестський державний технічний університет  
Проект Linux Vacation / Eastern Europe  
Львівська група користувачів Linux



# Матеріали другої міжнародної науково- практичної конференції

## FOSS Lviv 2012

26-28 квітня 2012 р.

<http://conference.linux.lviv.ua>





Друга міжнародна науково-практична конференція FOSS Lviv 2012:  
Збірник наукових праць/ Львів, 26-28 квітня 2012 р.

Під редакцією:

**Злобін Г.Г., Апунович С.Є., Апунович С.В., Ванькевич Д.Є.**

**ISBN 978-966-8460-99-9**

## Зміст

Использование свободного программного обеспечения в университетах и исследовательских учреждениях Российской Федерации Алексеев Е.Р., Брагилевский В.Н.....	5
Про досвід використання компілятора GCC та технології OPENMP при розв'язанні задач обчислювальної математики Алексеев С.Р., Чеснокова О.В., Савченко М.А., Хохлова А.А.....	11
Обробка результатів виконання лабораторних робіт як спосіб навчання елементарного програмування Апуневич С.В., Апуневич С.Є.....	14
Потужний, надійний та доступний корпоративний брандмауер/маршрутизатор на основі ОС Debian Gnu/Linux та системи Shorewall Апуневич С.Є.....	15
Соціальні сервіси в дистанційному навчанні Артеменко В.Б.....	16
MOZILLA — вчора, сьогодні, завтра Бабич Т.Ю.....	18
Побудова тернарних графіків засобами MAXIMA Баранецький В.І.....	20
Використання ВПЗ для тестування апаратного забезпечення ПЕОМ в навчальному процесі факультету електроніки ЛНУ імені Івана Франка Батюк А.Я., Злобін Г.Г.....	23
Використання технології віртуалізації в навчальному процесі факультету електроніки ЛНУ імені Івана Франка Бойко Я., Ванькевич Д., Злобін Г.....	24
Автоматизація фізичного експерименту засобами фірми Cypress Semiconductor Бублик П.І., Ткач О.І., Злобін Г.Г.....	26
Emacs Org-Mode у навчанні та науковій роботі Булка Ю.А.....	28
Вільний програмний каркас для роботи із блок-діаграмами Чмихало О.С.....	29
Обзор возможностей существующего открытого ПО для работы с музыкальными данными Чубарьян А.С.....	30
ВПЗ — можливість цікавого навчання від малечі до науковця Чоповський С.С.....	32
ВПЗ для астрономічних спостережень Білінський А.І., Мартинюк-Лотоцький К.П.....	36

OpenQA – система автоматизованого тестування дистрибутивів ОС Linux Дідюк І.....	38
JOOMDLE: інтеграція CMS JOOMLA! та LCMS MOODLE Франчук В.М.....	43
Україномовна локалізація MLE-MOODLE Франчук Н.П.....	45
Використання бібліотеки Qt для написання гітарного тюнера Голубник М.Г.....	47
Системи дистанційного навчання Іванов В.Г., Гвозденко М.В., Кошева Н.А.....	49
Спотворення спектрів сигналів внаслідок джиттеру частоти дискретизації Яремчук О.М., Дзюба С.П.....	50
Перспективы развития свободного обеспечения ў беларускіх “Аблоках” Карагода А.І., Стальбоўская Н.І.....	53
Правові аспекти використання ІКТ у навчальних закладах Хачіров Т.С.....	55
Використання електронного навчального курсу математики як засобу підготовки до ЗНО Харченко В.М., Ваврикovich Л.В., Карпенко І.І.....	58
Побудова ВЕБ-інтерфейсу для XML-даних в мобільних пристроях Павлишенко Б.М., Квач З.А.....	63
FREE/LIBRE software usage in the Belarusian system of higher educational institutions Derechennk S.S., Kostiuk D.A., Pynkin D.A.....	64
Використання відкритого програмного забезпечення для організації дистанційного навчання в технічному університеті Кривенко О.В., Чичкар'юв С.А.....	68
Вільні ліцензії у законодавчій неволі Бічук А.А., Наушко М.В.....	71
Інформаційні технології в підготовці магістрів фізико-математичних спеціальностей педагогічного університету Лазурчак І.І., Кобильник Т.П., Грозовський В.В.....	74
Scientific inquiry based education and FOSS Levitshi A., Duca M.....	76
Використання відкритого програмного забезпечення у науково-дослідних роботах і навчальному процесі кафедри комп'ютерних систем та мереж ТНТУ ім. І.Пулюя Луцків А.М.....	78
Віртуальний лабораторний практикум на базі GEANT4 в середовищі ОС Linux для вивчення процесів взаємодії випромінювання з речовиною Малихіна Т.В.....	81

Використання вільного ПЗ для управління телескопом Мартинюк-Лотоцький К.П.....	84
ALT LINUX как пример свободной платформы Новодворский А.Е.....	85
Застосування вільних математичних пакетів до розв'язання задач теоретичної механіки Періг О.В., Кисіль А.В., Матвеев І.А., Міхєнко Д.Ю.....	86
Використання ВПЗ для підвищення рівня захищеності комп'ютерних мереж та систем Піскозуб А.З. ....	90
Програмная реализация декодера одного класса помехоустойчивых кодов на алгебраических кривых. Проектирование на основе шаблонов обобщённого и мета-программирования Пеленицын А.М.....	94
Досвід та перспективи використання вільного ПЗ у видавничо-поліграфічній галузі Піскозуб Й.З., Миклушка І.З., Піскозуб Л.Г.....	96
Використання вільного програмного забезпечення при створенні науково-освітніх електронних бібліотек Прилуцка Н.С. ....	99
Державна політика у сфері ВПЗ ДЦНТП з розповсюдження ПЗ з відкритим кодом: велика перемога чи велика профанація? Радченко Ю.А. ....	102
Використання крос-платформного інструментарію розробки програмного забезпечення Qt для створення навчальних програм. Рудий М.Ф.....	107
Дистрибутив ZENTYAL для побудови серверів для малого та середнього бізнесу Сало М.Я., Ванькевич Д.....	110
Поштовий сервер IREDMAIL в бізнесі Сейдаметов Г. С. ....	112
Підтримка шкільної математики вільно поширюваним ПЗ Семеніхіна О.В., Друшляк М.Г.....	113
Застосування ВПЗ для підготовки спеціалістів зі створення судових інформаційних системи і комп'ютерних мереж Шапо В.Ф., Шевченко Т.І.....	116
Виконання завдань розподіленої обробки зображення під управлінням системи CONDOR Шийка Ю.Я., Шувар Р.Я.....	119
Використання обчислювального кластера з ОС Linux у вищому навчальному закладі Сідун Н.М., Чичкарьов Є.А.....	121



Новий підхід до створення INITRAMFS Сігало В.С., Ясько М.М.....	124
Технологія AJAX у інформаційних системах автоматизованого збору та опрацювання даних Словінський О.В., Словінська О.Д.....	127
Огляд дистрибутивів LINUX, створених на базі DEBIAN/UBUNTU, що використовуються у сфері освіти Смалько О.А.....	128
Організація ІНТЕРНЕТ-доступу до віддалених баз даних Столярчук О.В., Шувар Р.Я., Продивус А.М.....	132
Переваги використання вільного ПЗ в організації наукової діяльності Університету Ткаченко Н.О.....	133
Використання вільно-поширюваного ПЗ для створення систем управління навчальним процесом у ВНЗ Триус Ю.В., Герасименко І.В.....	136
Перехід університетів України на вільне програмне забезпечення Тверезовська Н.Т., Липська В.В.....	139
Автоматизовані системи створення та супроводу пакетів у дистрибутиві ALT LINUX. Крок у майбутнє. Власенко І.Ю.....	142
«Хмарні» обчислення як основа формування персональних навчальних середовищ Воронкін О. С.....	143
Library for advanced functions in algorithms, data structures and AI implemented in C/C++ - Olib Wszeborowska A. and Swierczewski L.....	147
Досвід впровадження відкритого програмного забезпечення в стартапах компанії Медіа-Ключ Зелінський О.Л.....	149
Использование свободного ПО для реализации функций контроля при обучении студентов специальности «Математика. Информатика» в педагогическом ВУЗе Зенько С.И., Ненартович М.В.....	151
Using Puppet to reduce Time of Deployment V. Zhzhuta.....	154
Мюнхен за рік заощадив 4 млн євро завдяки переходу на Linux: хроніка подій Г. Злобін .....	154
Аналіз використання ВПЗ в закладах освіти України Г. Злобін .....	157
Статический анализ ПО с помощью его промежуточных представлений и технологий с открытым кодом Зубов М.В., Пустыгин А.Н., Старцев Е.В.....	165

**Использование свободного программного обеспечения в университетах и исследовательских учреждениях Российской Федерации**

**Алексеев Е.Р., Брагилевский В.Н.**

*Донецкий национальный технический университет, Южный федеральный университет, EАlekseev@gmail.com, bravit@sfnedu.ru*

The experience is represented of utilization and deployment of free/open software in educational institutions as well as research institutions of Russian Federation.

За последние годы в Российской Федерации накоплен большой опыт использования свободного программного обеспечения как в университетской, так и академической среде, а также в средней школе (как в отдельных школах, так и в целых регионах).

Ряд университетов Российской Федерации, таких как Московский государственный университет (МГУ), Удмуртский государственный университет (УдГУ), Санкт-Петербургский торгово-экономический институт (СПбТЭИ), Томский государственный педагогический университет (ТГПУ), Университет города Переславля, Южный федеральный университет (ЮФУ), Елецкий государственный университет (ЕГУ) и др., широко используют свободное программное обеспечение как при обучении специалистов в области информационных технологий, так и при информационной подготовке специалистов различного направления.

Значительный интерес представляет опыт Санкт-петербургского торгово-экономического института (СПбТЭИ) [1] и Томского государственного педагогического университета (ТГПУ) [2].

В СПбТЭИ на всех компьютерах к 2007 году в учебных лабораториях была установлена ОС семейства Linux. Внедрение началось в 2003 году [1] с одного учебного класса (12 рабочих мест), в котором проводились занятия со студентами 3-4 курсов, а также работали аспиранты. Положительные результаты эксперимента позволили распространить опыт, в конце 2006 года уже в пяти (из семи) компьютерных классах (114 рабочих мест) в основном использовался дистрибутив Linux, а в качестве второй ОС иногда использовалась Windows. В этих классах к началу 2007 года уже не осталось коммерческих прикладных программ. Выбор операционной системы для проведения занятий определялся административным принципом «в Интернет из Windows не ходят!». Далее по заказу СПбТЭИ в ALT Linux был разработан специализированный дистрибутив для учебного процесса, который практически вытеснил ОС Windows из учебных лабораторий.

При переходе на свободное ПО одной из проблем является обучение персонала. Интересный опыт переподготовки преподавателей и сотрудников для работы со свободным программным обеспечением накоплен

в Томском государственном педагогическом университете (ТГПУ) [2]. В университете было принято решение о переходе на свободное ПО и необходимости переобучения всех сотрудников и преподавателей, а также учителей школ Томской области, лишь небольшая часть из которых освоила свободное ПО самостоятельно. Для остальных были организованы курсы и было разработано пособие «Практическая работа на компьютере в среде Linux». Однако при организации курсов возникла следующая проблема. Обучить свободному программному обеспечению надо было всех быстро и качественно, для чего возможностей инициативной группы внедрения СПО было недостаточно. В результате была принята «сетевая модель обучения с тьюторской поддержкой». Каждый из членов инициативной группы обучил группу «тьюторов», каждый из которых в свою очередь обучил одну или несколько групп сотрудников и т. д. Таким образом, довольно быстро были обучены все сотрудники и преподаватели.

В Нижегородском радиотехническом колледже (НРТК) был осуществлён полный переход на свободное программное обеспечение как в учебном процессе, так и в работе подразделений и администрации [3]. Инициативной группой осуществляющей переход всего колледжа на свободное ПО была проведена большая работа:

- по подбору и адаптации программного обеспечения для учебного процесса и административных нужд;
- по подключению и настройке различного периферийного оборудования, некоторые типы сканеров и принтеров оказались не совместимы со свободным ПО, этот опыт был использован при замене старого оборудования; при покупке нового оборудования обязательно производится предварительная проверка на совместимость со свободными платформами и программным обеспечением.

Опыт НРТК говорит о том, что в 2008-2009 гг. самой большой проблемой при полном переходе на свободное ПО стало взаимодействие с административными органами управления учебными заведениями. Подразделения министерства образования ведут переписку исключительно в форматах MS Office. Многие on-line системы сбора данных об учебных заведениях ориентированы на работу лишь с web-браузером Internet Explorer и некорректно работают в других. Программное обеспечение для проведения ЕГЭ ориентировано исключительно на проприетарную платформу. Не смотря на то, что ранние версии этих программ работали в среде Wine, новые версии перестали корректно выполнять свои функции в этой среде. К тому же для правильного формирования отчётов требуется MS Office определённой версии. Все эти сложности усугубляются обязательностью представления данных в сжатые сроки, санкциями в отношении учебного заведения при нарушениях сроков и отсутствия какой либо поддержки по данным продуктам при вопросах об их использовании в средах с набором свободного программного обеспечения.

Следует отметить, что для практически полного переходе ряда российских университетов на свободное программное обеспечение (ТГПУ, СПбТЭИ, НРТК и др.) принималось административное решение, и только после этого начался реальный и повсеместный переход.

В ряде ведущих российских университетов (МГУ, СПбГУ, ЮФУ, УдГУ, Петрозаводский Университет) преподавание ведётся как с использованием свободного, так и проприетарного программного обеспечения. Программное обеспечение выбирается под конкретный курс, а иногда и под конкретную тему. Некоторые специалисты (проф. Н.Н. Непейвода, УдГУ, ИПС РАН) вообще не ограничивают студентов в выборе программного обеспечения.

В Южном федеральном университете (г. Ростов-на-Дону) свободное ПО имеет две важные сферы применения: в организации сетевой инфраструктуры университета с одной стороны, и в учебном процессе на некоторых факультетах с другой [4]. Выбор ПО под конкретный курс осуществляется исключительно профессорско-преподавательским составом, поэтому применение свободного ПО большей частью определяется интересами преподавателей. К примеру, при обучении программированию свободное ПО играет достаточно большую роль; курсы по внутреннему устройству и принципам функционирования операционных систем также ориентированы в первую очередь на свободные ОС. Основная часть преподавателей при этом ориентируется на проприетарные системы.

Можно заметить, что при наличии достаточных финансовых средств на приобретение проприетарного ПО, у вузов нет стимулов к тотальному переходу на свободное ПО. При этом свободное ПО, тем не менее, в них используется, что является прямым следствием академических свобод преподавателей. Во всяком случае, ни в одном из государственных образовательных стандартов конкретные программные средства (будь то свободные или проприетарные) не упоминаются.

В 2009 году в Елецком государственном университете им. И.А. Бунина прошёл семинар «Информационные технологии на базе свободного программного обеспечения», материалы которого представлены в виде сборника трудов ([http://fosscenter.elsu.ru/pdf/Sbornic\\_2009\\_SPO.pdf](http://fosscenter.elsu.ru/pdf/Sbornic_2009_SPO.pdf)).

Более полную картину об использовании СПО в университетах России можно получить, ознакомившись с материалами, представленными на сайте <http://lists.raspo.ru/Plone/publichnye-drafty-dokumentov/dokumenty-komiteta-po-obrazovaniyu-i-vysshei-shkole/spo-v-rossiiskih-vuzah>.

Опыт отдельных вузов и энтузиастов получил развитие в России на государственном уровне. 17 декабря 2010 года в Российской Федерации было принято распоряжение №2299-р правительства РФ, в котором был утверждён план перехода федеральных органов исполнительной власти и федеральных бюджетных учреждений на использование свободного программного обеспечения на 2011 – 2015 годы. Пункты 2 и 4 этого плана касаются непосредственно вузов:



- разработка и утверждение перечня образовательных программ и учебно-методических материалов, рекомендованных для применения в образовательных учреждениях;
- подготовка и утверждение методических рекомендаций для образовательных учреждений высшего профессионального образования о замене используемого в учебном процессе проприетарного программного обеспечения аналогичным свободным программным обеспечением.

Таким образом, на сегодняшний день в России существует государственная программа перехода бюджетных организаций на свободное ПО.

В исследовательских организациях РФ широко используются как unix-подобные операционные системы, так прикладное свободно-распространяемое программное обеспечение. В университетах России функционирует целая сеть суперкомпьютеров, объединённых в программу «Университетский кластер» (<http://www.uniclust.ru>).

Так на суперкомпьютерах МГУ «ЛОМОНОСОВ» (<http://parallel.ru/cluster/lomonosov.html>) и «ЧЕБЫШЕВ», ([http://parallel.ru/cluster/skif\\_msu.html](http://parallel.ru/cluster/skif_msu.html)) наряду с коммерческими компиляторами компании Intel используются свободно-распространяемые компиляторы версии 4.4 – g++, gcc, gfortran.

В конце 2011 года в Северо-Восточном федеральном университете им. Н.К. Аммосова (Якутск) был запущен вычислительный кластер «Ариан Кузьмин». Первоочередными пользователями якутского суперкомпьютера стали математики. В ближайшее время к ним присоединятся и прикладники – геологи и геофизики. На кластере совместно с Институтом системного программирования РАН (который является национальным центром СПО) был развернут комплекс свободного программного обеспечения (СПО) для математического моделирования SALOME, OpenFOAM, ParaView, а также для геофизического моделирования компании Schlumberger.

Осенью 2012 года в России начнёт функционировать государственная автоматизированная система «Гособоронзаказ» (ГАС «ГОЗ»), призванная упростить процессы заключения и выполнения оборонных контрактов. Разработкой ГАС «ГОЗ» занимается ОАО «Системы управления», генеральный директор которой Владимир Минаев сообщил, что ГАС «ГОЗ» будет работать на российских компьютерах «Эльбрус» под управлением операционной системы Astra Linux Special Edition.

Отдельные учёные и исследовательские лаборатории РФ широко используют СПО, как при проведении численных и аналитических расчётов, так и при оформлении результатов исследований. Интересен опыт использования свободного программного обеспечения в Казанском научном центре при исследованиях турбулентности [5]. Были использованы такие инструменты на языке Python, как NumPy – поддержка многомерных массивов, библиотека математических функций, PyTables – ввод/вывод информации в файлы формата hdf5, Matplotlib – библиотека

научной графики с поддержкой формул LaTeX и русского языка, ViTables – просмотр файлов формата hdf5.

Кроме большого опыта использования свободного программного обеспечения в России накоплен опыт адаптации и разработки свободного программного обеспечения. Основанное в 2010 году ЗАО «РОСА» (<http://www.rosalab.ru>) является российским центром разработки решений на основе СПО. Продукты компании – адаптированные под российского пользователя дистрибутивы на основе Mandriva Linux. Интересным дистрибутивом был разрабатываемый компанией ASPLinux одноимённый дистрибутив. К сожалению, в настоящее время проект закрыт. Последняя версия вышла в 2008 году. Группа энтузиастов из Томского педагогического университета занимается русификацией и адаптацией под отечественного пользователя детского дистрибутива DoudouLinux. Сайт проекта – <http://doudoulinux.org/web/russkij/index.html>.

AltLinux ([www.altlinux.ru](http://www.altlinux.ru)) занимает особое положение на российском IT-рынке. Компания занимается не адаптацией, а разработкой собственной линейки дистрибутивов, среди которых выделяются дистрибутивы, ориентированные на использование в образовательных учреждениях (линейка дистрибутивов «Школьный»). Также специалисты компании много времени уделяют внедрению свободного программного обеспечения в школах и университетах. Несколько лет компания занималась проектом «Свободное программное обеспечение в школах России» (<http://freeschool.altlinux.ru/>). Ежегодно с 2006 года в последние выходные января компания проводит конференцию «Свободное программное обеспечение в высшей школе» с участием работников сферы образования России, Украины и Белоруссии. Сообщество пользователей AltLinux разрабатывает несколько очень интересных дистрибутивов, некоторые из них переходят в разряд официальных дистрибутивов компании: Simply Linux (<http://slinux.ru>), ALT Linux LXDEsktop (<http://lxdesktop.altlinux.org/>).

В Российской Федерации издаётся достаточно большое количество литературы, посвящённой свободному программному обеспечению. Этим занимаются такие коммерческие издатели IT-литературы, как «Питер», «ДМК-Пресс», «ВНУ» и др. Кроме того, компания ALT Linux занимается изданием учебной литературы по свободному программному обеспечению. AltLinux оплачивает труд авторов, издаёт книги на основе лицензии GNU FDL, вследствие чего все изданные книги в электронном виде доступны для свободного скачивания и распространения на сайте проекта [http://www.altlinux.org/Books:Main\\_page](http://www.altlinux.org/Books:Main_page).

В России издаётся ежемесячный журнал по свободному программному обеспечению – Linux Format ([www.linuxformat.ru](http://www.linuxformat.ru)) (с DVD диском). Практически в каждом номере журнала печатаются материалы, посвящённые использованию Linux и СПО в образовании.

Конечно, существуют и ряд проблем с использованием Linux и СПО в России. Внедрение свободного программного обеспечения в универси-

татах и исследовательских центрах зачастую остаётся делом энтузиастов (не смотря на наличие решения кабинета министров), качество ряда книг издаваемых коммерческими издательствами оставляет желать лучшего, процесс внедрения СПО встречает серьёзное сопротивление как среди разработчиков проприетарного ПО, так и среди пользователей. Но в целом за последние 10 лет ситуация в России изменилась кардинально – свободные программы пришли в университеты, исследовательские центры, на личные компьютеры рядовых пользователей. Это совместная заслуга разработчиков, пользователей и энтузиастов, распространяющих и популяризирующих Linux и СПО. Причём роль последних невозможно преуменьшить. Ведь «двигает» в массы свободное программное обеспечение именно сообщество пользователей.

### **Литература**

1. И.А. Хахаев . Свободное ПО и лицензирование ВУЗа // Тезисы докладов четвёртой конференции «Свободное программное обеспечение в высшей школе». Россия, Переславль, 30 января–1 февраля 2009 г. С. 19–21.
2. Ю.П. Немчинова, Е. Г. Пьяных . Проблема развития ИКТ-компетентности педагогических кадров в условиях перехода на свободное программное обеспечение и организационно-педагогические условия ее решения // Тезисы докладов пятой конференции «Свободное программное обеспечение в высшей школе». Россия, Переславль, 30-31 января 2010 г. С. 79–83.
3. В.Г. Слугин . Внедрение, использование и сопровождение СПО в учебном заведении на примере ФГОУ СПО «НРТК» // Тезисы докладов пятой конференции «Свободное программное обеспечение в высшей школе». Россия, Переславль, 30-31 января 2010 г. С. 27–30.
4. Брагилевский В.Н. Гуда С.А., Худолей Г.В. СПО на мехмате Южного федерального университета // Тезисы докладов седьмой конференции «Свободное программное обеспечение в высшей школе». Россия, Переславль, 28–29 января 2012 г. С. 26–28.
5. Ф.С. Занько . Свободные инструменты на языке Python в исследованиях турбулентности // Тезисы докладов седьмой конференции «Свободное программное обеспечение в высшей школе». Россия, Переславль, 28–29 января 2012 г. С. 95–98.

***Про досвід використання компілятора GCC та технології OPENMP при розв'язанні задач обчислювальної математики***

*Алексєєв Є.Р., Чеснокова О.В., Савченко М.А., Хохлова А.А.*

*Донецький національний технічний університет, EAlekseev@gmail.com,  
chesn\_o@list.ru*

Solving large-scale mathematical and engineering problems demands the long computations, even if powerful computing facilities are engaged. In this contribution authors share own experience of using free collection of compilers GCC along with OpenMP technology for solving the problems of linear algebra. The results are represented of benchmarking the codes for solving the systems of linear algebraic equations with large number of unknowns (500-5000) as well as multiplication of high-dimension matrices (500-5000).

GCC (GNU Compiler Collection) – це вільно розповсюджуваний набір компіляторів для різних мов програмування. На даний момент GCC є кросплатформним і може працювати практично під будь-якої сучасної операційної системою і компілювати програми під різні типи процесорів.

Особливість GCC в тому, що він здатний аналізувати імена файлів і визначати, які дії необхідно виконати. Так файли з розширенням .c/cc (.cpp) розглядаються, як файли на мові C/C++, файли з розширенням .f90 або .F95 – це коди мови Fortran, а файли з розширенням .o вважаються всередині машинними кодами.

Відмінною рисою GCC є управління компіляцією за допомогою ключів командного рядка, які детально описані в [1]. При компіляції програм розв'язання задач обчислювальної математики слід звернути увагу на ключі «-O1», «-O2», «-O3», які встановлюють різні рівні оптимізації програми за швидкістю - мінімальна, середня і висока відповідно.

Як інструмент для налагодження програм за допомогою GCC можна використовувати вільне середовище розробки програмного забезпечення Geany. Цей багатоплатформний програмний продукт має досить гнучкий інтерфейс, автоматично визначає мову програмування і виділяє програмний код відповідно з синтаксисом мови.

Автори вже кілька років використовують GCC в навчальному процесі та науковій діяльності. Одним з напрямків досліджень є паралельне програмування з використанням технології OpenMP і компілятора GCC.

OpenMP – це відкритий стандарт для розпаралелювання програм. Він реалізує паралельні обчислення за допомогою багатопоточності. На даний момент автори досліджували розпаралелювання алгоритмів множення матриць і рішення систем лінійних алгебраїчних рівнянь (СЛАР). Використовувався наступний ПК: системна плата MSI H67MA-E45; ОЗУ – 4Гб; процесор Intel Core I5 (4 ядра) з тактовою частотою 3.3 Ггц; 32-х розрядна ОС Linux Mint Debian с рає ядром.



У таблиці 1 представлені результати тестування програми вирішення СЛАУ ітераційним методом за допомогою компілятора g++ з опцією оптимізації «-O3».

Таблиця 1. Час розв'язання СЛАУ ітераційним методом в g++

Процес:	Розмірність СЛАУ				
	500	1000	2000	3000	5000
послідовний	0.0145874	0.031860	0.05	0.27771	0.77832
паралельний	0.0035095	0.014160	0.020507	0.108582	0.265625

Таблиця 2 містить результати тестування програми розв'язання СЛАУ методом Гауса (компілятор g++ з опцією оптимізації «-O3»).

Таблиця 2. Час розв'язання СЛАУ методом Гауса в g++

Процес:	Розмірність СЛАУ				
	500	1000	2000	3000	5000
Послідовний	0.157532	1.24301	1.92383	34.2341	159.008
паралельний	0.0566101	0.421143	1.30005	12.1223	56.0793

Результати тестування програми множення матриць показані в таблиці 3 (компілятор g++ з опцією оптимізації «-O3»).

Таблиця 3. Час множення квадратних матриць у g++

Процес:	Розмірність матриць				
	500	1000	2000	3000	5000
послідовний	0.689331	8.81665	107.583	443.823	2253.58
паралельний	0.231598	2.64703	31.3691	120.983	1033.91

Для порівняння наведемо результати тестування задач у середовищі Microsoft Visual C++ 2008 (MS VC++ 2008) (див. табл. 4). Тут тестувалися СЛАУ з 2000 рівнянь і матриці розмірністю 2000-2000.

Таблиця 4. Час розв'язання задач у MS VC++ 2008

Процес:	Задача		Множення матриць
	Метод розв'язання СЛАУ		
	ітераційний	Гауса	
послідовний	0.126587	9.20386	115.092
паралельний	0.0561523	2.58533	33.223

Проведені дослідження говорять про те, що вільний компілятор GCC g++ – це потужний інструмент програмування, який не поступається пропрієтарному MS VC++ 2008. А технологія OpenMP за грамотного використання дає змогускоротити час виконання програм у декілька разів.

В ході досліджень були розроблені програми вирішення СЛАР на мові Fortran. В таблицях 5 та 6 приведені результати їхнього тестування.

*Таблиця 5. Час розв'язання СЛАР ітераційним методом в gfortran*

Процес:	Розмірність СЛАР				
	500	1000	2000	3000	5000
послідовний	0.00049	0.00220	0.12376	0.31449	0.9096
паралельний	0.00025	0.00133	0.00664	0.14449	0.42084

*Таблиця 6. Час розв'язання СЛАР методом Гауса в gfortran*

Процес:	Розмірність СЛАР				
	500	1000	2000	3000	5000
послідовний	9.23514	3.70463	42.18344	157.8577	835.6785
паралельний	4.76315	1.41757	13.4700	140.22	585.639

У програмних кодах використовується велика кількість матричних операцій і вбудованих функцій мови Fortran. Це в деяких випадках унеможливорює процес розпаралелювання і отже, часто порівнювати швидкість роботи компіляторів g++ и gfortran недоцільно.

За результатами тестування програмних кодів у gfortran можна зробити висновок, що у разі збільшення кількості рівнянь у СЛАР послідовні процеси значно сповільнюються і погіршується результат розпаралелювання.

Існує багато способів порівняння компіляторів. Можна, наприклад, порівняти швидкість компіляції або розмір генерованого коду. У своїх дослідженнях автори в основному приділяли уваги швидкодії виконання програм. У підсумку можна сказати, що набір компіляторів в GCC – це потужний відкритий засіб програмування, що дозволяє створювати програми високого ступеня складності і обробляти велику кількість вихідних даних

## **Література**

1. Гриффитс Артур. *СС. Настольная книга пользователей, программистов и системных администраторов.* – К.: ООО «ТИД ДС», 2004. – 624 с.

**Обробка результатів виконання лабораторних робіт як спосіб навчання елементарного програмування**  
Апуневич С.В., Апуневич С.Є.

Львівський національний університет ветеринарної медицини та біотехнологій імені С.З. Гжицького, [sofiya.apunevych@gmail.com](mailto:sofiya.apunevych@gmail.com)  
Львівська група користувачів Лінукс, [apusbird@gmail.com](mailto:apusbird@gmail.com)

In this contribution we propose to use IDE Kuzya along with well-known textbook "Numerical Recipes in C" by Press et al., 1992 for processing the results of experiments at laboratory works. This would promote the understanding and the sense of doing computations among students, and bring very useful skills to them.

Під час курсу загальної фізики студенти виконують лабораторні роботи, одним із елементів котрих є обробка результатів. Зазвичай ці нескладні обчислення виконуються на калькуляторі, за допомогою табульованих функцій, також їх можливо виконати за допомогою табличних процесорів. Проблема полягає у тому, що студенти перших курсів рідко розуміють сенс цих обчислень. Хоча якраз розуміння базової статистики у застосуванні до даних експерименту є важливим для будь-якого спеціаліста, який виконуватиме тести і експерименти (наприклад, клінічні). Також навичка провадити обчислення за допомогою комп'ютера (нині вже забута функція ЕОМ) -- цінний здобуток вищої освіти, корисний у практичній діяльності.

На нашу думку, найкраще таким завданням у педагогічному сенсі може відповідати безпосереднє програмування студентом таких обчислень і виконання програми. Постає питання, котру мову програмування використати та котре середовище. Ми пропонуємо варіант програмування мовою C в інтегрованому середовищі IDE Kuzya, що розробляється Львівською спільнотою --- групою програмістів Programming Lviv Linux User Group (<http://www.pllug.totalh.com/>).

C -- одна із найстаріших і найбільш поширених мов програмування. Її синтаксис ліг в основу багатьох спеціалізованих мов програмування. Проблема хіба становить надмірна гнучкість цієї мови програмування, та відсутність вбудованих у мову стандартних засобів опису матриць та операцій із ними. Але це ускладнення можна обійти, якщо приховати від новачків цю складність за допомогою стилю програмування, що викладений у відомому підручнику "Numerical Recipes. Art of Scientific Computing" ("Числові рецепти. Мистецтво наукових обчислень.", Пресс та ін. 1992 р.). Цей підручник також слугує джерелом готових бібліотек (функцій) для роботи із матрицями і векторами, щоб не обтяжувати студента інформацією. Як платформу для таких обчислень ми пропонуємо Лінукс (будь-яку популярну збірку, наприклад Ubuntu), де завжди присутній компілятор мови Cі.

Інтегроване середовище IDE Kuzya пасує до розв'язання цієї задачі, бо воно надзвичайне просте, адаптації до нього легко приймаються авторами програми, і воно може виконуватися не лише на ОС Linux, але й під управлінням ОС Windows чи MacOS. Зрештою, програмісти-початківці і є цільовою аудиторією цієї програми.

Отже, подаючи студентам середовище Kuzya із заготовкою програми, в котру студентові достатньо вписати або виправити кілька лексем, можна сформуванати розуміння суті програмування і обчислень, базової статистики.

### ***Потужний, надійний та доступний корпоративний брандмауер/маршрутизатор на основі ОС Debian Gnu/Linux та систему Shorewall***

*Апуневич С.Є.*

*Компанія "Елекс", [stepan.apunevych@eleks.com](mailto:stepan.apunevych@eleks.com)*

Author shares his own experience in using the mid-scale application of firewalling and routing. The proposed solution is an ordinary PC, supplied with several network interfaces, with Debian Gnu/Linux installed, supplemented with Shorewall system for configuring NetFilter capabilities of Linux kernel. The solution has proved to be low-cost, reliable, extremely powerful and flexible. The configuration is easy and clearly understood, and need no special training to implement. I would say, no proprietary appliances can withstand the comparison.

На певному етапі зростання кожна організація стикається із проблемами росту. Кількість працівників зростає, інфраструктура розростається, і вимоги до стабільності та швидкості з'єднання до Інтернет різко зростають. Водночас на кардинальні зміни немає можливості, бо організація не спинить роботи навіть у вихідні. Особливо гостро така проблема стоїть для розробницьких компаній, які переважно, окрім обслуговування внутрішніх клієнтів, надають ІТ-послуги стороннім організаціям.

Одним із варіантів є придбання спеціалізованого обладнання від відомих постачальників, наприклад Cisco, Checkpoint, чи Juniper. Ці прилади відомі своєю високою надійністю і високими характеристиками, але без відповідної вузько-спеціалізованої підготовки фахівця реалізувати складну конфігурацію нереально. Водночас, навіть вже добре підготовленому фахівцю знадобиться чимало часу на те, щоб розібратися із поточною конфігурацією. Далі постає питання про постійну підтримку системи у робочому стані та внесення модифікацій. І так до наступного циклу, який має середню по галузі тривалість 5 років. Отже, сукупна вартість такого рішення стає значною, і ці витрати не цілком виправдані для підприємства середнього масштабу.



Інший відомий спосіб -- це самостійно виготовити аналогічну систему на базі ОС родини Unix, користуючись наявними у ній інструментами. Однак це потребує часу на розробку і налагодження, і у достатньо складній конфігурації підтримка таких рішень є також коштовна. На щастя, є вже готова розробка для керування конфігурацією -- Shorewall [1]. По суті, це система сценаріїв мовою Perl, які на основі конфігураційних файлів виготовляють "атомарне" представлення правил для Netfilter, а також таблиці маршрутизації, контролю трафіка, тощо. Таким чином досягається висока потужність, наприклад, звичайний ПЕОМ може маршрутизувати до 300 Мб/с. Дуже корисною рисою такого підходу є те, що під час перевантаження конфігурації не обриваються вже наявні сесії.

### **Література**

1. <http://www.shorewall.net/>

## **Соціальні сервіси в дистанційному навчанні** **Артеменко В.Б.**

*Львівська комерційна академія, artem@iac.lviv.ua*

Key social services Web 2.0, which are based on personal learning environments and social networks, are considered. We are discussing these services, tools and information materials, aimed to provide a remote user to a comfortable environment for learning via Internet, on the basis of which formed a similar approach under the name of e-Learning 2.0. Here the ways are proposed to use social services, aimed at developing personal learning environment in such a learning management system (LMS), as Moodle.

До соціальних сервісів Web 2.0, за допомогою яких користувачі можуть не тільки спілкуватися між собою, але найголовніше – самі створювати контент веб-сторінок, відносяться наступні платформи та онлайн-інструменти:

- блоги та мікроблоги (Twitter, Blog.com, LiveJournal – ЖЖ);
- персоналізована стартова сторінка або персоналізований інтернет-портал (iGoogle, Netvibes), реалізовані на основі побудови користувацьких інтерфейсів веб-застосувань, за яких веб-сторінка, не переавантажуючись, у фоновому режимі відправляє запити на сервер і сама звідти довантажує потрібні користувачу дані;
- соціальні мережі і системи соціальних презентацій (Facebook, Ning, MySpace, Vkontakte, Odnoklassniki.ru);
- вікі-проекти (відкрита багатомовна енциклопедія Wikipedia, освітній проект Letopisi.ru);
- соціальні закладки (diigo, delicious, bobrdobr);

- мультимедійні системи розповсюдження інформації (Flicker, Picasa, YouTube, SlideShare);
- системи спільних редакторських офісів (Google.docs, Spread-sheets);
- системи машапа та бриколажа (mashaps&vricolage), що дають змогу форматувати і змішувати різні формати представлення даних, обробляти веб-сторінки без знання мови HTML (PingMe services, Del.icio.us, SkypeMe, Yahoo Pipes);
- вебінар – це термін для позначення різних онлайнзаходів: семінарів, дискусій, презентацій, тренінгів та мережових трансляцій тих чи інших подій (під час вебінару зв'язок між учасниками підтримується через Інтернет на базі спеціальної веб-платформи – "віртуальний клас" фірми Web-soft, dimdim, wiziq);
- карти знань – зручна техніка альтернативного запису (bubbl.us) або спосіб зображення процесу загального системного мислення за допомогою схем (mind map);
- сучасні технології синдикації та нотифікації інформації RSS (Really Simple Syndication), що дає змогу публікувати і транслювати практично будь-який матеріал з певного сайту (починаючи з новин і закінчуючи особистими мережевими щоденниками).

Соціальні сервіси Web 2.0 створили можливість для комунікації і роботи в Інтернеті, на основі яких потім сформувався аналогічний за назвою підхід e-Learning 2.0 (термін запропонований Стефаном Доунс).

В Інтернеті представлені сьогодні різні точки зору щодо впровадження дистанційного навчання: на базі систем управління дистанційним навчанням (LMS), або ж соціальних сервісів Web 2.0, на засадах яких утворюються персональні навчальні середовища (PLE). Використовуються також підходи щодо підтримки дистанційного навчання на засадах інтеграція соціальних сервісів у LMS.

Ми маємо на меті висвітлити підходи до застосування в одному з вищих навчальних закладів деяких соціальних сервісів, які спрямовані насамперед на формування персонального навчального середовища у такій системі управління дистанційним навчанням, як MOODLE (Modular Object Oriented Distance Learning).

У Веб-центрі Львівської комерційної академії використовуються LMS Moodle з інтегрованими в неї такими базовими соціальними сервісами: вікі, блоги, вебіари (<http://virt.lac.lviv.ua>).

Вікі-проекти зазвичай використовуються, коли декілька учасників певних дистанційних курсів (ДК) одночасно працюють над створенням спільного документу. Вікі починається з однієї сторінки, а потім кожен автор може додавати посилання на будь-які сторінки.

У Веб-центрі ЛКА існує три типи вікі: викладач, групи, студент.

Використання сервісу «Wiki» дозволить створити базу знань для будь-якого дистанційного курсу без великих витрат часу.

Уваги заслуговує також і такий різновид соціального сервісу як блоги.

Блоги (weblog) – персональний журнал. У Веб-центрі ЛКА блог – це власний щоденник учасників ДК, куди вони заносять свої думки. Записи даного щоденника можуть бути особистими або відкритими іншим користувачам.

Режим доступності блогів встановлюється адміністратором для всього сайту:

1. *Всі користувачі сайту можуть бачити все блоги.*
2. *Доступність блогу можна обмежити, щоб користувачі могли бачити блоги тільки учасників свого ДК або учасників своєї групи.*
3. *Систему блогів на сайті може заборонити адміністратор.*

У Веб-центрі ЛКА використовуються вебінари, що проводяться тьютором на платформі dimdim. Зареєструвавшись учасники вебінару в призначений час повинні клікнути на надіслане тьютором ДК посилання та підключитися до відповідного сайту. Отож, у Веб-центрі ЛКА вбудовано мінімальний набір інструментів, що надає можливість у дистанційному навчанні вирішувати як викладачам, так і студентам поставлені перед ними завдання.

## **MOZILLA — вчора, сьогодні, завтра** Бабич Т.Ю.

Mozilla Україна, [tim.babych@gmail.com](mailto:tim.babych@gmail.com)

Mozilla is a global community striving to promote openness, innovation and opportunity on the web, creator of Mozilla Firefox browser. This talk is about its origins, structure, current tasks, challenges and plans for the future.

Mozilla є світовою спільнотою людей, об'єднаних спільною метою: забезпечити розвиток інтернету як середовища, відкритого до інновацій та доступного для всіх.

Mozilla була створена у 1998-му році для продовження розвитку браузера Netscape, знаменитою стала у 2004-му, випустивши першу версію браузера Firefox. Іншими відомими проектами є поштовий клієнт Thunderbird, інтегрований пакет Seamonkey та календар Lightning. Всі ці проекти з минулого року перейшли на режим швидких випусків, нові версії тепер з'являються строго за графіком, кожні шість тижнів. Одночасно існують чотири стадії кожного продукту: нічні збірки, альфа-збірки(Аврора), бета та реліз.

Організаційно Мозілла складається з кількох сотень співробітників, котрі працюють на повний день віддалено та в офісах (Маунтін Вью, Париж, Токіо та кілька менших); кількох тисяч добровільних помічників та сотні тисяч тестерів.

Для комунікації використовуються групи новин, irc, bugzilla. Код зберігається у репозиторії mercurial, а команди веб-розробки та B2G використовують github.

Крім браузера та інших програм Mozilla розробляє крос-браузерні проекти, націлені на покращення мережі, такі як зручна система авторизації BrowserID, документація MDN, служба субтитрування довільних відео UniversalSubtitles, онлайн-редактор Bepin (Ace), менторська ініціатива WebFWD.

Велику увагу Mozilla приділяє мобільним технологіям, оскільки вже невдовзі кількість мобільних інтернет-пристроїв перевищить к-ть настільних, і далі тільки зростатиме.

За місяць вийде радикально оновлений Firefox Mobile під Android, котрий використовує нативний інтерфейс, має синхронізацію з десктопною версією та підтримує додатки. Триває робота над проектом Boot to Gecko, повноцінної мобільної ОС на базі Linux, драйверів від Android, та всього інтерфейсу побудованого на Gecko та веб-технологіях.

Багато нового планується додати і в десктопну версію Firefox. Невдовзі він буде працювати ще швидше та стабільніше, і оновлюватись тихо, не турбуючи користувачів перепитуваннями. Будуть покращені інструменти розробника, додатки можна буде синхронізувати так само як і історію з закладками, оновлено зовнішній вигляд, додано підтримку нових технологій (SPDY тощо)

Проекту потрібна ваша допомога, є багато задач і кожен зможе вибрати собі якусь по плечу. Потрібні розробники, веб-розробники, перекладачі, тестери, просвітяни. Долучайтесь!

## **Література**

1. <http://mozilla.org>
2. <http://mozilla.org.ua>
3. <irc://irc.mozilla.org#mozilla-ua>
4. <https://wiki.mozilla.org/Roadmaps>
5. <https://wiki.mozilla.org/Firefox/Roadmap>

## Побудова тернарних графіків засобами МАХІМА

Баранецький В.І.

*Дрогобицький державний педагогічний університет імені Івана Франка,  
bvolodi@gmail.com*

The diagram content-property is considered herein for 3-component compounds. The program implementation of such diagram is represented by means of computer algebra system Maxima.

Активне використання комп'ютерів для проведення символічних і графічних обчислень звільняє дослідника від проведення рутинних, але трудомістких і багатих помилками перетворень, істотно скорочує час реалізації наукових і технічних проєктів.

Багато спеціалізованих комп'ютерних програмних продуктів для моделювання та аналізу даних успішно використовуються в галузі економіки, біології, хімії, фізиці, інженерії та інших областях науки.

Створення складних моделей пов'язане насамперед з використанням сучасної обчислювальної техніки, що дозволяє вивчати великі системи, які залежать від багатьох факторів. Однією з таких задач є побудова діаграм склад-властивість, де функція описує залежність певної характеристики сполуки від співвідношення її компонентів.

Тернарна система координат часто використовується у фізиці, хімії, біології, матеріалознавстві та ін. Перевага її використання в тому, що зображення композиції трьох змінних може бути легко побудоване засобами двовимірної графіки. Кожна точка на потрійній ділянці являє собою композицію різного складу з трьох компонентів.

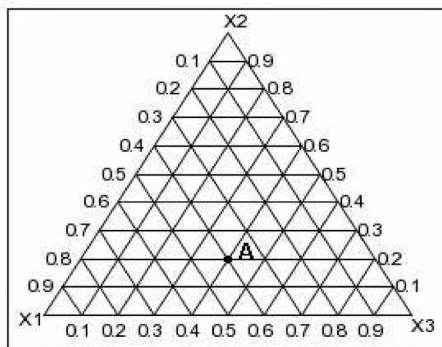


Рис 1. Порожня тернарна діаграма.

Один із способів визначення складу композиції базується на сітці тернарної діаграми. Концентрація кожного фактора становить 100% (чиста речовина) в своєму кутку трикутника і 0% в лінії навпроти неї. Частка окремих факторів убуває зі збільшенням відстані від цього кута. Проводячи паралельні лінії через однакові проміжки між нульовою лінією і точкою в кутку, частка фактору може бути встановлена відповідно до зручності оцінки змісту того чи іншого фактору. Так, наприклад, точці А на рисунку 1 відповідає наступне співвідношення компонентів:  $x_1 - 40%$ ,  $x_2 - 20%$  і  $x_3 - 40%$ .

Розглянемо побудову такої “трикутної сітки”. На рисунку 2 наведено програмний код, що будувє у системі Махіма мітки та паралельні прямі для фактору  $x_1$ .

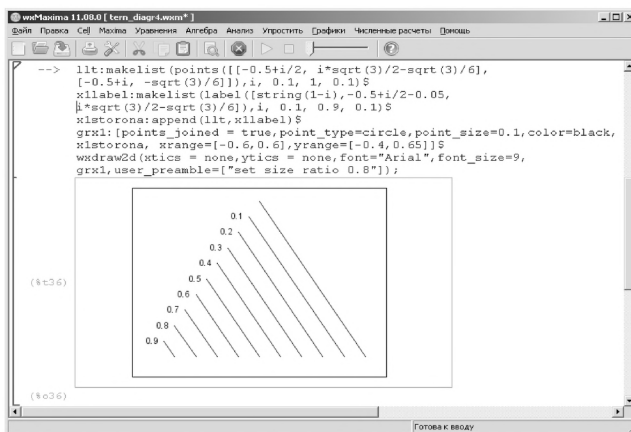


Рис 2. Програмна побудова “сторони”  $X_1$  тернарної системи координат.

Аналогічно будувємо дві інші сторони тернарної системи  $X_1 X_2 X_3$ .

Оскільки функція, результатом роботи якої повинен бути тернарний графік, залежить від трьох змінних  $x_1, x_2$  і  $x_3$  та має зображатися на площині, то для побудови графіка необхідно привести функцію  $f(x_1, x_2, x_3)$  до вигляду  $f(x, y)$ . Для цього використаємо формули переходу (Дрейпера-Лоуренса) від «трикутних» координат до декартових:

$$x_1 = 1/3(-3z_1 - z_2\sqrt{3} + m),$$

$$x_2 = 1/3(3z_1 - z_2\sqrt{3} + m),$$

$$x_3 = 1/3(2z_2\sqrt{3} + m),$$

де  $m = 1$  (100%), причому  $z_1 = x, z_2 = y$ .

Продемонструємо це на прикладі полінома четвертого порядку (див. рис.3).

Рис 3. Перевід функції від трьох змінних у функцію від двох змінних.

Для побудови ліній однакових значень (ліній рівня) скористаємось командою `implicit_plot`.

На рисунку 4 наведена тернарна діаграма склад-властивість для згаданого вище полінома.

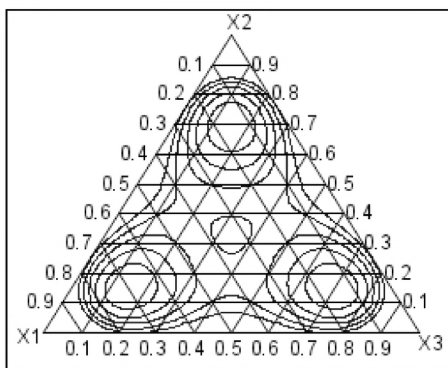


Рис 4. Графік функції для значень  $f(x, y) = (0.5, 0.6, \dots, 0.9)$ .

Наведений приклад свідчить про досить розвинуті обчислювальні та графічні можливості системи комп'ютерної математики Maxima, що дає змогу використовувати її для розв'язування різноманітних задач, а безкоштовність та відкритість коду робить систему Maxima доступним продуктом для широкого кола користувачів.

**Використання ВПЗ для тестування апаратного забезпечення ПЕОМ в навчальному процесі факультету електроніки ЛНУ імені Івана Франка**  
*Батюк А.Я., Злобін Г.Г.*

*Львівський національний університет імені Івана Франка,  
batiuk@electronics.wups.lviv.ua*

The report is considering open source software (OSS), which is used for testing, installation, upgrade and repair of PCs.

Операції з налагодження апаратного забезпечення ПЕОМ не є повсякденною задачею, проте вони є необхідними для стабільної роботи ПЕОМ. Окрім того сучасний фахівець в галузі інформаційних технологій повинен мати базові знання про роботу складових частин персонального комп'ютера. Ці знання надалі можуть пригодитись при постановці перед ним завдань щодо модернізації та програмного чи апаратного ремонту персонального комп'ютера. За умовами поширення програмні засоби для тестування складових ПЕОМ можна розділити на такі групи:

- відкриті програми (free/libreware);
- вільно-поширювані програми (freeware);
- умовно платні програми (shareware);
- комерційні програми з закритим кодом.

Незважаючи на добре ознайомлення викладачів та студентів із програмами тестування, які належать до умовно платних (shareware) або комерційних програм з закритим кодом, використання їх в навчальному процесі небажане через практичну неможливість підтримки їх ліцензійності. Навіть наявність пробних версій з обмеженим терміном користування (зазвичай 30 днів) не розв'язує проблему, бо семестр триває 4 місяці. За таких умов надійне забезпечення навчального процесу можливе лише у разі використання вільного або вільно-поширюваного програмного забезпечення.

Базовими елементами ПЕОМ є: процесор(и), оперативна пам'ять, системна плата та носії інформації (жорсткий диск, оптичні диски тощо). Зупинимось детальніше на вільному програмному забезпеченні для кожної з цих компонент. кілька прикладів використання ВПЗ для роботи з апаратним забезпеченням:

- тестування процесора: stress, CPU Burn, Im-sensors (температура процесора);
- тестування оперативної пам'яті: memtester, memtest86+, ramspeed;
- отримання загальної інформації про стан апаратного забезпечення: lspci, lsusb, lshw, pciconf;
- відновлення даних та аналіз цілісності поверхні жорсткого/оптичного диску: testdisk, photorec, badblocks, dd\_rescue, parted/gparted.



Якщо мова йде про ремонт апаратного забезпечення, то безумовно корисними в цьому випадку будуть програми послідовної передачі даних - telnet, minicom.

Зазвичай вищезгадані утиліти розрізнено присутні у дистрибутивах GNU/Linux. Нашу увагу привернув дистрибутив, що спеціально призначений для тестування ПК — stresslinux [1], який базується на openSUSE 11.4. Його можна завантажити як iso- або usb-образ. Зважаючи на більшу універсальність usb-образ, використовується саме він. Для систем, які напряду не підтримують завантаження з usb (або підтримують його неповністю), ми використовуємо Plop Boot Manager [2] разом з grub або grub4dos [3] (додатково використовується дискета або компакт-диск).

### **Література**

1. <http://www.stresslinux.org/sl/>
2. <http://www.plop.at/en/bootmanagers.html>
3. <https://gna.org/projects/grub4dos/>

## **Використання технології віртуалізації в навчальному процесі факультету електроніки ЛНУ імені Івана Франка**

*Бойко Я., Ванькевич Д., Злобін Г.*

*Львівський національний університет імені Івана Франка,  
[boyko@electronics.wups.lviv.ua](mailto:boyko@electronics.wups.lviv.ua)*

The virtualization technologies are considered which are used at learning process in Ivan Franko LNU.

Широке поширення технології віртуалізації створило необхідні передумови для їх використання в навчальному процесі. На факультеті електроніки ЛНУ імені Івана Франка технології віртуалізації використовуються у двох лекційних курсах:

1. *Операційні системи*
2. *Системне адміністрування ОС Linux*

Потреба у використанні віртуалізації в спецкурсі “Системне адміністрування ОС Linux” виникла несподівано — наприкінці серпня 2011 р. лектора, який мав читати цей спецкурс, повідомили, що приміщення, позичене підготовчим відділенням ЛНУ імені Івана Франка до першого липня 2001 р., звільнене не буде. А це означало, що приміщення для проведення лабораторних робіт на живих ПЕОМ ( процесором Intel Celeron 400 МГц, 128 МБ оперативної пам’яті) немає. Через це виникла потреба у швидкому розгортання програмних і апаратних засобів для

проведення лабораторних робіт курсу «Системне адміністрування ОС Linux» в лабораторії загального користування.

Для проведення лабораторних робіт було використане таке обладнання:

1. сервер з процесором AMD Phenom (tm) II X4 945, 8ГБ оперативної пам'яті і операційною системою GNU Debian і системою віртуалізації OpenVZ;
2. 5 робочих місць студента з процесором Intel Pentium 4, 512 МБ оперативної пам'яті і операційною системою Microsoft Windows XP.

На сервері було створено 8 віртуальних машин - 5 для студентів + 3 службові (кешуючий проксі пакетів apt-cacher-ng, wiki з методичними матеріалами, локальний DNS-сервер). Кожному студенту була виділена власна операційна система Debian GNU / Linux в контейнері OpenVZ і доступ до управління контейнером за допомогою ovz-web-panel [4]. На клієнтських ПЕОМ для управління віртуальними машинами використовувалися ssh-клієнт (putty) і Веб-переглядач.

В межах спецкурсу «Системне адміністрування ОС Linux» студенти повинні були встановити та налаштувати систему управління веб-хостінгом ISPConfig [2] в ході послідовного виконання таких робіт:

- лабораторна робота № 1 "Ознайомлення з програмами OpenVZ, PuTTY, vi";
- лабораторна робота № 2 "Установка компонент поштового сервера";
- лабораторна робота № 3 "Установка компонент Веб-сервера";
- лабораторна робота № 4 "Робота з сервером електронної пошти";
- лабораторна робота № 5 "Перескерування стандартних потоків вводу/виводу в ОС Linux".

Для забезпечення від виходу з ладу віртуальних машин під час виконання лабораторних робіт було передбачено створення контрольних точок для відновлення працездатної конфігурації. Незважаючи на експромтну реалізацію полігону в цілому результати проведення лабораторних робіт курсу «Системне адміністрування ОС Linux» можна вважати задовільними.

Технології віртуалізації використовуються також і для організації лабораторного курсу з дисципліни «Операційні системи». Як систему віртуалізації застосовано Oracle VM VirtualBox — кросплатформний програмний продукт, який випускається під ліцензією GNU GPL [5]. Пропріетарні розширення від Oracle при цьому не використовуються. У вказаному середовищі встановлено і налаштовано для навчальних цілей гостьові операційні системи Linux (Ubuntu 10.04 LTS) і FreeBSD 9.0. Віддалений доступ через віртуальні інтерфейси типу «проміжний адаптер» типу організовано за протоколами SSH (для обох ОС) і FreeNX [5] (для ОС Linux). Особливістю технології FreeNX є ефективність і налаштуваність компресії трафіку та його шифрування, що дає змогу надавати клієнтам віддалені графічні робочі середовища через мережі з різною пропускну здатністю. Завдяки цьому студенти мають змогу працювати з

CLI та GUI-інтерфейсами операційних систем і використовувати для виконання лабораторних робіт інтегровані системи розробки (QT Creator, Code::Blocks тощо). Описане програмне середовище, яке ґрунтується на технологіях віртуалізації, як показує тривалий досвід його використання, має значні переваги. Зокрема воно надає можливість швидкого переналаштування, доповнення (наприклад, додавання за потребою нових операційних систем чи систем з різною конфігурацією). При цьому клієнтські робочі місця, оснащені обмеженим набором програмного забезпечення, не потребують жодної модифікації.

### *Література*

1. <http://www.ispconfig.org/>
2. [http://wiki.openvz.org/Main\\_Page](http://wiki.openvz.org/Main_Page)
3. <http://code.google.com/p/ovz-web-panel/>
4. <http://www.virtualbox.org/>
5. <http://freenx.berlios.de/>

## **Автоматизація фізичного експерименту засобами фірми Cypress Semiconductor**

*Бублик П.І., Ткач О.І., Злобін Г.Г.*

*Львівський національний університет імені Івана Франка,  
факультет електроніки,  
[bublykpetro@gmail.com](mailto:bublykpetro@gmail.com)*

The development is described of hardware and software parts of the automatization system at the physical experiment implemented on the basis of programmable system on chip (PSoC) by Cypress Semiconductor. The connection to a computer is implemented by means of universal serial bus, USB. The project documentation is published according to international license standard on free hardware and software.

Завдяки високу рівню розвитку мікропроцесорної техніки сьогодні є можливість спростити процес експериментальних досліджень. Дуже потужним рішенням в цьому плані є використання мікроконтролерів PSoC Cypress які містять на одному кристалі процесорне ядро, цифрову та аналогову периферію. В мікроконтролерах PSoC Cypress реалізовані такі користувацькі модулі, як:

- VDAC8 - ЦАП (цифро-аналоговий перетворювач);
- ADC\_DeSig - АЦП (аналого-цифровий перетворювач);
- PWM - ШІМ (широтно-імпульсний модулятор);
- Counter - лічильник;
- Character LCD - контролер рідкокристалічного дисплею;
- Clock - тактовий генератор;

USBFS - контролер USB;  
інші модулі.

Першим реалізованим нами блоком є генератор періодичних сигналів з можливістю вибору дискретних значень частот. Принцип роботи генератора полягає у видачі на вихід VDAC8 масиву значень (20 точок), які і будуть генерувати один період сигналу. Повторення цієї операції нескінченну кількість разів дає змогу отримати потрібний сигнал. Залежно від потреби в масив будуть занесені значення для генерування сигналів різної форми (синус, меандр, пилка). Зміна частоти сигналу відбувається за рахунок зміни частоти дискретизації, яку ми подаємо на відповідний вхід VDAC8. Вибір частоти і форми сигналу здійснюється в прикладній програмі на ПЕОМ, зв'язок з якою відбувається через універсальну послідовну шину USB.

Другим реалізованим нами блоком є цифровий частотомір. З допомогою цього модуля можна визначити частоту вхідного періодичного сигналу. За допомогою Delta Sigma ADC ми записуємо в масив вхідний сигнал, тоді проводимо над отриманими даними дискретне перетворення Фур'є. Частотою сигналу буде частота найбільшої гармоніки сигналу. Визначену частоту сигналу можна спостерігати на рідкокристалічному дисплеї, а також можна провести додаткові дослідження на ПЕОМ.

Третім реалізованим нами блоком є цифровий вольтметр з можливістю накопичення експериментальних даних. Отриманні результати можна передати на ПЕОМ для подальшої обробки та візуалізації експериментальних даних.

Четвертим реалізованим нами блоком є блок керування кроковими двигунами (КД). Реалізована можливість задання кута повороту, швидкості а також напряму обертання крокового двигуна. Керування КД базується на використанні компоненти PWM (широтно-імпульсний модулятор). Керування відбувається через USB в прикладній програмі проекту в режимі реального часу.

Всі модулі мають зв'язок з ПЕОМ з використанням універсальної послідовної шини USB. В PSOC Cypress зв'язок з USB реалізується за допомогою компоненти USBFS, яка дає змогу здійснювати передачу даних у режимі FAST. Також в ній відбувається оголошення дескрипторів, які потрібні для розпізнавання пристрою операційною системою.

На ПЕОМ реалізована програма керування модулями та опрацювання отриманих результатів для подальшого їхнього аналізу та візуалізації. Програма використовує готові компоненти фірми Cypress для роботи з USB.

Проектна документація є у вільному доступі відповідно до міжнародної ліцензії про вільне апаратне забезпечення [1].

Таким чином створена система дає змогу значно спростити і оптимізувати процес дослідження. А відкритість системи дає змогу залу-

чити ентузіастів з різних куточків світу для створення нових та вдосконалення вже наявних модулів.

1. <http://www.ohwr.org/projects/cernohl/wiki>

## ***Emacs Org-Mode у навчанні та науковій роботі***

*Булка Ю.А.*

*Львівська національна музична академія ім. Миколи Лисенка  
yurkobb@gmail.com*

Emacs Org-mode, being developed by a scientist with PhD in physics Carsten Dominic, seems to have a good potential as a tool for students and authors of scientific papers. Org-mode provides wide range of features, but the core concept is to assist in organizing information, storing it efficiently in plain text, as well as allowing rich text and structural markup in exported result.

Emacs Org-mode, створений німецьким науковцем-фізиком, судячи з усього, має гарний потенціал як знаряддя для студентів та авторів наукових робіт. Org-mode містить широкий ряд функцій, але в основі лежить ідея організації інформації та ефективного її зберігання у форматі простого тексту, а також можливість включення форматування та структурного розмежування в похідні формати.

Org-mode — додаток до розбудованого текстового редактора GNU Emacs. Emacs вже давно перетворився на своєрідне робоче середовище завдяки численним функціям, які виходять за межі власне редагування текстових даних. До них належить і Org-mode. Оскільки Emacs починали розробляти ще тоді, коли не було графічного дисплею, він має розбудовану систему клавіатурних скорочень, що роблять можливою роботу без миші; саме такий спосіб роботи є найефективнішим у Emacs і його сповідує також Org-mode.

Сам Org-mode втілює функції кількох категорій. В основі всіх їх лежить можливість створення структурованого документа, що розділяється ієрархією заголовків. Зміст розділів можна приховати, частково приховати (бачити лише заголовки), і повністю показувати; між заголовками можна швидко перестрибувати комбінацією клавіш, заголовки можна легко переставляти місцями і навіть переносити між файлами чи архівувати.

На основі заголовків Org створює систему особистого організатора — кожен заголовок можна позначити як завдання до виконання (**TODO**), призначити дату завдання та кінцеву дату завершення; дати можуть включати інтервал повторення. Існує система пріоритетів, категорій та ступеню завершення завдань. Також є ряд типів звітів, які автоматично генеруються, зокрема розпорядок на день, тиждень чи іншу кількість



днів. Інше вельми елегантна функція обліку часу, який витрачається на виконання завдань.

Будь-який файл Org-mode може містити форматування: грубий та нахилений шрифти, списки, графічні файли, гіперлінки, таблиці, примітки, фрагменти коду, формули (у LaTeX). Мова маркування Org значно зручніша для вдрукування, ніж LaTeX чи HTML, зокрема щодо таблиць, що робить його зручним препроцесором для створення файлів такого типу, адже підтримує перетворення у LaTeX (і, відповідно, PDF), HTML, DocBook. Недавно введено також перетворення в ODT (Open Document Text), з якого, як відомо, можна створити ще й файли doc та rtf. Це дає змогу писати наукову роботу, будучи відносно незалежними від цільового формату файлу, і користуватись усіма зручностями, які надає Emacs сам по собі та Org-mode.

Таблиці Org-mode можуть містити формули, що є доволі подібною функцією до програм т. зв. електронних таблиць (зокрема LibreOffice Calc). Формули тут можна писати двома мовами: в синтаксисі модуля Emacs Calc, або й на самому Emacs Lisp. Попри доволі широкий набір функцій електронних таблиць, Org-mode не копіює архітектури інших програм цього типу, і натомість пропонує деякі власні ідеї, як то формула-стовбець чи відносні координати комірок.

Це далеко не повний огляд функцій Org-mode але вже вони розкривають потенціал цієї програми як помічного інструмента для студента чи науковця, чи просто для людей, що мають схильність конспектувати інформацію та організувати свій час з допомогою цифрових технологій. GNU Emacs, на якому працює Org-mode, запускається на всіх основних платформах: Windows, GNU/Linux, Mac OS X, FreeBSD та інших. Org-mode, як і GNU Emacs розповсюджується під вільною ліцензією.

## ***Вільний програмний каркас для роботи із блок-діаграмами***

*Чмихало О.С.*

*Львівський національний університет імені Івана Франка,  
[alex.chmykhalo@gmail.com](mailto:alex.chmykhalo@gmail.com)*

There is a variety of software visualizes data structures or models of systems in a form of block diagrams (e. g. software for the diagram creation (UML diagrams, state diagrams), creation of mind maps, visual programming, graphical system design etc.). The open graphical diagram creation and analyzing framework will greatly facilitate the development of tools for creation, edition and analysis of diagrams.

Існує велика кількість програмного забезпечення, яке використовує задання даних або моделей систем у вигляді блок-діаграм. Серед таких програм:

- програми призначені для створення та редагування блок-схем;
- програми для створення та редагування діаграм (UML-діаграми, діаграми станів тощо);
- ПЗ для створення ментальних карт (так звані карти думок або mind-maps);
- засоби візуального програмування;
- засоби візуального конструювання систем (graphical system design) тощо.

Програмне забезпечення володіє наступними особливостями:

- створення, редагування та збереження блок-діаграм;
- засоби для візуального відображення та редагування структури блок-діаграм;
- різноманітний аналіз структури та взаємозв'язків складових елементів у діаграмі.

Серед існуючих програмних засобів можна виділити GEF Project (створення, редагування, збереження діаграм, які мають вигляд графів), JUNG (Java Universal Network/Graph Framework), Graph-tool (маніпуляція графами, аналіз графів), Graphviz (візуалізація діаграм, програмні засоби для візуалізації діаграм), OpenJGraph (Java Graph and Graph Drawing Project), Graphical Modeling Framework for Eclipse та інші, які надають інструменти для створення програмного забезпечення такого виду.

Створення програмного каркасу має значно полегшити розробку програм для створення, редагування та аналізу блок-діаграм. Відкритий програмний код, можливість розширення, кросплатформність зроблять його зручним та доступним для використання.

### ***Обзор возможностей существующего открытого ПО для работы с музыкальными данными*** Чубарьян А.С.

*Федеральное государственное автономное образовательное учреждение высшего профессионального образования «ЮЖНЫЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ», [armaxis@mail.ru](mailto:armaxis@mail.ru)*

This paper presents a classification of the software to work with the music data in accordance with its functionality. Examples are given and opportunities and prospects are discussed for open source software from every category. Special attention is paid to the software libraries for music generation; specific examples of their use for the generation of classical canons are given.

В наши дни существует целый ряд открытого ПО для работы с музыкальными данными. Большую его часть условно можно разделить на 3 группы:



Нотные редакторы (программы, позволяющие записать и распечатать ноты в привычном классическом виде);

Секвенсоры, виртуальные синтезаторы, семплеры (программы, используемые для микширования и редактирования аудиозаписей);

Библиотеки для программирования музыки.

Все нотные редакторы позволяют вводить, редактировать и печатать нотный текст разной сложности от простых мелодий до симфонических партитур. Разработано огромное количество ПО, которое предоставляет возможность форматировать, настраивая форму и расположение элементов при помощи графического интерфейса или изменяя параметры в текстовом виде, и проигрывать нотный текст через MIDI-выход. Большинство нотных редакторов являются платными, однако существует целый ряд бесплатных программных решений, таких как Rosegarden [1] или MuseScore [2].

К отдельной категории нотных редакторов стоит отнести проекты Lilypond [3] и MusiXTeX [4] – редакторы, построенные на основе TeX'a. Гибкость и лаконичность данной компьютерной верстки позволяют им успешно конкурировать со своими GUI-аналогами.

Дальнейшее развитие нотных редакторов происходит в следующих направлениях: распознавание сканированного нотного текста, диапазон операций, выполняемых автоматически, гибкость и конфигурабельность программы, интеграция с другими видами музыкального программного обеспечения; работа с электронными нотными библиотеками.

Секвенсоры позволяют производить запись с последовательным наложением нескольких партий, а также редактировать записанный ранее материал. Благодаря тесной интеграции с виртуальными синтезаторами (VST, а также его бесплатные аналоги LADSPA и LV2) секвенсоры становятся мощным средством, позволяющим записать полноценное музыкальное произведение при полном отсутствии реальных музыкальных инструментов. Многие программные секвенсоры уже включают в себя возможности нотного редактора (и наоборот, современные нотные редакторы содержат возможности секвенсора), однако их основной задачей является работа непосредственно с музыкальным материалом, а не отображение нот. Данная группа является наиболее популярной среди музыкантов и разработчиков музыкального ПО. Среди большого количества проприетарного ПО встречается и свободное, например: LMMS [5], Qtractor.

Число существующих библиотек, предназначенных для облегчения программирования музыки, невелико. Среди них стоит выделить два наиболее удачных проекта: CFugue [6] и jMusic [7]. Первый из них является библиотекой для C/C++, а второй – библиотекой для Java. Основная цель этих библиотек – построить мост между классическим понятием о нотах и некоей абстракцией, которой удобно оперировать программисту. Становится возможным выполнение более сложных задач от гармонизации мелодий, поиска закономерностей в наборе музыкальных

произведений, формализации существующих методик композиции до генерации музыки с применением возможностей искусственного интеллекта.

С помощью библиотеки jMusic были апробированы возможности генерации музыки, строящейся по принципам классического канона. За основу был взят известный Канон Ре-Мажор Иоганна Пахельбеля, который был тщательно проанализирован с помощью классической гармонии и формализован в виде музыкальных паттернов, наиболее точно описывающих его структуру, и при этом позволяющих настраивать параметры канона. Средствами jMusic был написан код-генератор, способный создавать каноны любой тональности и любой продолжительности. Сгенерированные компьютерные каноны были выложены в Интернете [8], и их звучание оценили как близкое к оригиналу.

Таким образом, библиотеки позволяют провести компьютерный анализ творчества известных композиторов, определяя авторство неизвестных произведений, и создать новые композиции, стилистически близкие к первоисточнику. Дальнейшее развитие музыкальных библиотек позволит генерировать музыки разных стилей и направлений от барокко до авангарда XXI века.

### *Литература*

1. Rosegarden - <http://www.rosegardenmusic.com/>
2. Musescore - <http://musescore.org>
3. Lilypond - <http://lilypond.org>
4. MusiXTeX and Related Software - <http://icking-music-archive.org/software/indexmt6.html>
5. Linux MultiMedia Studio – <http://lmms.sourceforge.net/>
6. CFugue - <http://gopalakrishna.palem.in/CFugue.html>
7. jMusic - <http://explodingart.com/jmusic/>
8. Computer-generated music  
[http://vk.com/audio?album\\_id=20845269&id=2316355](http://vk.com/audio?album_id=20845269&id=2316355)

### **ВПЗ — можливість цікавого навчання від малечі до науковця** Чоповський С.С.

Львівський професійний ліцей залізничного транспорту, [auslemborg@meta.ua](mailto:auslemborg@meta.ua)

Використання сучасних педагогічних, інформаційних та комп'ютерних технологій у навчальному процесі вимагають нових підходів до вирішення завдання щодо підготовки висококваліфікованих фахівців у різних галузях господарства.

Отже розглянемо проєкт GNU/Linux як платформу для організації навчального процесу:

- для дітей дошкільного віку;
- для учнів шкіл та ПТНЗ;
- для студентів ВНЗ;
- для науковців, інженерів та інших фахівців.

Використання ВПЗ в освіті підтримується багатьма країнами Євросоюзу, Бразилією та Китаєм. Але в Україні спостерігається майже 99% домінування власницького програмного забезпечення.

Крім того закупівля ліцензійного програмного забезпечення від Microsoft, навіть так званих пільгових "академічних" ліцензій, є дуже дорогою. Проблемним є і наявне методичне забезпечення, підручники з інформатики, які в першу чергу спрямовані на вивчення розташування ярликів та піктограм інтерфейсу операційної системи Windows та пакету Microsoft Office, що робить з учнів "кнопкодавів із розвинутою м'язовою пам'яттю" замість людей з розвинутою логікою.

Але й не все так погано. Діючі навчальні програми дозволяють використовувати ВПЗ у вивченні курсу інформатики та інформаційних технологій у повному обсязі.

Річард Столлман одного разу в своєму виступі зазначив: "Школа повинна вказати дітям шлях життя, що буде дуже корисним суспільству в цілому ... Вільне програмне забезпечення заохочує кожного вчитися ... Школи, які використовують вільне програмне забезпечення, заохочують обдарованих до руху вперед."

Олександр Зленко, (прес-секретар проектів СЕНСАР) ще в жовтні 2004 р. в статті "Український комп'ютер: сучасний стан і перспективи" (<http://sd.org.ua/news.php?id=5033>) зазначив: "Не можна казати, що Україна повністю залишається на узбіччі цих світових процесів..." "Поширення нової операційної системи в навчальних закладах всіх рівнів сприятиме не тільки активнішій їх комп'ютеризації за рахунок збережених коштів, але й дозволить підвищити рівень комп'ютерної грамотності української молоді..."

Але на щастя є в Україні деякі небайдужі ентузіасти, які не лякаються експериментувати та широко застосовують досвід своїх зарубіжних колег.

Наприклад: в РФ з 2007 року впроваджується державна програма по використанню вільного програмного забезпечення в освіті.

5 грудня 2011р. стало відомо що Кабінет Міністрів України ухвалив цільову програму використання ПЗ в органах державної влади на 2012-2015 роки. Згідно з нею, до 2015 року органи державної влади повинні перейти на використання ПЗ з відкритим кодом.

Отже, як бачимо, використання в освітньому процесі вільного програмного забезпечення може принести для нашої держави значну економію бюджетних коштів та зробити великий крок у справі підвищення загального рівня комп'ютерної грамотності населення.

Але цього, на мою думку, не достатньо для докорінної зміни ситуації з ВПЗ. Потрібна не тільки всебічна підтримка ВПЗ державними органами,

але й залучення до процесу широкі громадські маси (ВНЗ, ПТНЗ, школи) для популяризації використання ВПЗ в навчанні та вдома.

Особливо це стосується студентів педвузів, як майбутніх вчителів, що будуть спроможними використовувати всі можливості GNU/Linux, бо давно відомо, на чому навчають — те й потім використовують.

А зараз пропоную вашій увазі невеликий огляд засобів організації цікавого навчання від малечі до науковця на базі проектів GNU/Linux.

*ВПЗ для дітей дошкільного віку:*

doudoulinux — (<http://www.doudoulinux.org/>) — це система що спроектована для дітей. DoudouLinux містить десятки програм, які призначені для дітей у віці від двох років, і уявляє собою робоче середовище, що подібне до ігрової консолі.

Дитячий Linux дистрибутив Qimo 4 Kids — (<http://www.qimo4kids.com/>) на основі Xubuntu 10.04 призначений для дітей дошкільного і шкільного віку, та оснащений простим, зрозумілим й барвистим інтерфейсом.

*ВПЗ для учнів шкіл та ПТНЗ:*

Skolelinux — (<http://www.slx.no/>) — операційна система, призначена для освітніх цілей та побудована на дистрибутиві Debian (тому вона також називається Debian-Edu).

Edubuntu — (<http://www.edubuntu.org/>) — Розробка Edubuntu ведеться разом із вчителями. Специфічне програмне забезпечення, що входить до складу дистрибутиву розраховане на вікову аудиторію 6-18 років.

EduMandriva — (<http://edumandriva.ru/>) — Освітній проект, направлений на покриття запитів освітніх систем всіх рівнів від дошкільного до університетського.

Дистрибутиви ALT Linux — ([www.altlinux.ru](http://www.altlinux.ru)) — це сімейство дистрибутивів Linux, що є окремою гілкою розвитку русифікованого Linux компанії «Альт Лінукс» та її партнерів, заснований на розробках команди програмістів ALT Linux Team.

OpenSUSE — ([www.opensuse.org](http://www.opensuse.org)) — один з дистрибутивів GNU/Linux, базується на дистрибутиві Slackware,

*ВПЗ для студентів ВНЗ, науковців, інженерів та інших фахівців:*

Scientific Linux (SL) — (<http://scientific-linux.ru/>) — дистрибутив операційний системи Linux, орієнтований на використання в наукових установках та ВНЗ, який створений сумісними зусиллями Fermilab і CERN, при підтримки лабораторій і університетів із всього світу.

OpenSUSE Medical — ([http://en.opensuse.org/openSUSE:Medical\\_DVD](http://en.opensuse.org/openSUSE:Medical_DVD)) — в склад дистрибутива внесена одна із самих повних колекцій програм для лікарів, студентів медичних ВНЗ та адміністративного персоналу лікарень і поліклінік.

Tango Studio — (<http://tangostudio.tuxfamily.org/en>) — повноцінний дистрибутив для музикантів і професіоналів, що працюють в студіях звукозапису, містить набір кращих програм з відкритим кодом для створення аудіо та підтримки професіональних і аматорських студій звукозапису.



Mathlinux —(<http://mathlinux.com/>) — це портативний дистрибутив Linux, оснований на Arch Linux . Він має велику кількість математичних, статистичних пакетів, пакетів числового аналізу та є дистрибутивом для обчислень і розробок в області математики.

Vinux—(<http://vinuxproject.org/>)— це ремастеринг версія дистрибутива Ubuntu Linux, який оптимізований для користувачів з послабленим зором. Він забезпечує читання з екрана, повноекранне збільшення та підтримку дисплеїв Брайля із коробки!

Navigatrix —(<http://navigatrix.net>) —(Linux для навігаційних цілей) повноцінна операційна система для комп'ютера, яка суміщає електронні інструменти для навігації, комунікації, зв'язку, інформації та безпеки. Рекомендовано кожному штурману для особистого арсеналу.

Bio-Linux —(<http://nebc.nerc.ac.uk/>) — (Дистрибутив для біологів, біоінформатиків, медиків ...), має в собі більше 500 різноманітних біоінформатичних програм.

Шановні колеги!

Своїм виступом я намагався зацікавити вас до використання вільного програмного забезпечення для організації сучасного навчального процесу.

В зв'язку з тим, що вивчення інформатики та інформаційних технологій носить практичний, прикладний характер, то вкрай необхідно виробляти у учнів відношення до комп'ютера, як до інструменту, здатного суттєво полегшити виконання повсякденних професійних завдань. Щоразу покращувати власні інформаційно-комунікаційні компетенції. Незважаючи на велике різноманіття програмного забезпечення учні повинні вміти вибрати оптимальні засоби розв'язання поставлених завдань.

У сучасній психології наголошується вплив вивчення інформатики і використання комп'ютерів в навчанні на формування нового типу мислення, так званого операційного мислення, скерованого на вибір оптимальних рішень.

Саме таких компетенцій вимагає життя від людини, і тому одним з основних цілей сучасного навчання є розвиток операційного мислення тих, що навчаються. Сподіваюсь, що вам сподобалась наша маленька екскурсія по найбільш яскравих сузір'ях галактики на ім'я GNU/Linux.

Дякую за увагу!

Автор висловлює щирю вдячність всім авторам публікацій матеріалів, які суттєво допомагають при створенні навчальних презентацій. І зокрема:

Григорій Злобін — ЛНУ ім.І.Франка; Алексей Новодворский — ALT Linux; Григорий Громко — Нечаевская общеобразовательная школа; Алексей Алешин — ОНУ им. И.И. Мечникова; Дмитрий Сподарец —RootUA Media; ALEKS\_USB — Розробник цілої низки чудових збірок LiveDVD (USB) на базі GNU/Debian. А також спільноті Internet, що всебічно розвивають та роз'яснюють усі тонкощі використання ВПЗ.

## **ВПЗ для астрономічних спостережень** **Білінський А.І., Мартинюк-Лотоцький К.П.**

*Астрономічна обсерваторія Львівського національного університету імені Івана Франка, bilian1831@gmail.com, langure@mail.ru*

With new century, the series of equipment in AO LNU for optical observations of artificial satellites of Earth were modernized. As a result of this, the problem of appropriate software was appeared. The report brought the results of implementation of OSS for astronomical observations in AO LNU in the research and students education.

В інформаційну епоху розвитку суспільства ВПЗ займає все більший сегмент у різних галузях, й найбільше, напевно, у наукових дослідженнях. Зокрема це відноситься до астрономічних спостережень – від підготовки вихідних даних до обробки й аналізу отриманих результатів, які проводять на базі науково-дослідного комплексу апаратури для вивчення штучних небесних тіл ближнього космосу АО ЛНУ ім.І.Франка.

Використовуване ВПЗ можна умовно розділити на такі категорії:

- Готове навчальне ПЗ та програми для любителів астрономії.
- Засоби розробки ПЗ та бібліотеки.
- Професійні системи.

### **Готове навчальне ПЗ та програми для любителів астрономії.**

Яскравим прикладом такого типу ВПЗ є вільний віртуальний планетарій **Stellarium** – програма із вільним відкритим кодом, що є доступно відповідно ліцензії GNU General Public License version 2.0 (GPLv2). Використовується під час навчально-ознайомчої практики студентів-астрофізики, а також при підготовці до спостережень у режимі “реального часу” для візуальної оцінки умов видимості небесних об’єктів.

**Засоби розробки ПЗ та бібліотеки.** Для розробки програмного забезпечення, а також оновлення наявного, з метою задоволення потреб розвитку ПЗ, а також вимог відповідних організацій (напр. Міжнародної служби лазерної локації супутників IIRS) використовується низка спеціальних вільних бібліотек та відкриті коди ряду проєктів. ВПЗ у АО для проведення спостережень та обробки їх результатів розробляється на мовах програмування **C/C++**, **Perl**, **PHP**, а також із використанням скриптів (**csH**, **bash**).

Для розробки ПЗ, пов’язаного із задачами небесної механіки та розрахунками видимих положень штучних супутників Землі, використано такі засоби:

**1.libnova** – універсальна бібліотека для розробників у **C/C++** (астрономів та програмістів), кого цікавлять питання розрахунку видимих положень світил та проблеми небесної механіки. Поширюється за ліцензією GNU LGPL.

2.**SGP4** – непропрістарна, узагальнена із багатьох сторонніх джерел (технічної документації, частин коду), версія бібліотеки для розрахунку у геоцентричній системі руху штучних супутників Землі, елементи орбіт яких є опубліковані у форматі TLE (введеного Міністерством оборони США більше 25 років назад), що є найпоширенішим для таких даних.

3.**CPF** – введений міжнародною службою ILRS формат даних та одно-ім'яний базовий пакет текстів програм для використання у прецизійних розрахунках орбіт штучних супутників Землі. Розроблений в Інституті астрономії Бернського університету та Центрі космічних досліджень Техаського університету, згодом викладений для вільного використання.

Для роботи із телевізійними камерами, встановленими у фокальній площині телескопів, використано вільну утиліту **VirtualDub**, яка поширюється з ліцензією GNU GPL. Також планується розробити ВПЗ для автоматичного супроводу небесних тіл (автогід) на основі текстів програм вільного проекту **open-phd-guiding**, що поширюється із ліцензією New BSD License.

Перетворення форматів графічних файлів здійснюється із використанням **ImageMagick** – вільного крос-платформенного пакету програм (ліцензія Apache 2.0 license) та **Netpbm** – пакету графічних програм і бібліотек із відкритим кодом (ліцензії поширення: Artistic License, GPLv2, MIT License).

При обробці астрономічних даних у форматі FITS, є незамінною бібліотека **cfitsio**. Також використовується модуль **CFITSIO.pm** (Perl-інтерфейс до самої бібліотеки).

**Професійні системи.** В продовження теми обробки результатів астрономічних спостережень, а саме зображень у форматі FITS, використовується:

• **IRAF** – система для обробки та аналізу астрономічних зображень, розроблена Національною оптичною астрономічною обсерваторією (США). Розповсюджується вільно, проте із дещо конфліктною ліцензією, яка найближче слідує MIT License.

• **SAOImage DS9** – самодостатня прикладна програма (Tk/Tcl) візуалізації та аналізу астрономічних даних у форматі FITS, складена із майже 20 ВПЗ, що поширюються за ліцензіями як стандартними GNU GPL, LGPL чи BSD, так і власними ліцензіями на поширення відкритого коду.

Для кінцевого відображення результатів наукових досліджень у графічному вигляді використовуються такі програми:

- **QtiPlot** – крос-платформенне графічне ВПЗ для інтерактивного аналізу та відображення наукових даних (подібний до Origin), із ліцензією GNU GPL.
- **gnuplot** – консольне ВПЗ для створення графіків, що може працювати як інтерактивно, так і з використанням скриптів. Поширюється із власною ліцензією – лише офіційний текстовий код вільно.



## **OpenQA – система автоматизованого тестування дистрибутивів ОС Linux**

Дідюк І.

Львівський національний університет імені Івана Франка, [Ivan.Didyuk@gmail.com](mailto:Ivan.Didyuk@gmail.com)

The report reviews the characteristics, advantages and disadvantages of OpenQA, which aim to simplify the testing operating systems of family GNU/Linux.

OpenQA – це інструмент для всебічного тестування ОС, який дає можливість в повністю автоматичному режимі проводити всебічне тестування працездатності різних компонентів дистрибутивів, від таких стадій як робота завантажувача, ініціалізація та завантаження ядра, до перевірки окремих графічних програм, таких як Firefox і LibreOffice. Пакет не обмежений підтримкою openSUSE і може використовуватися для тестування Fedora, Ubuntu, Debian, FreeBSD і навіть OpenIndiana. Код openQA повністю відкритий і поширюється в рамках ліцензії GPLv2.

В основі openQA лежать дві незалежні підсистеми: тестовий пакет OS-autoinst і реалізація керуючого Веб-інтерфейсу. OS-autoinst є багатоплатформним додатком, що дає можливість протестувати будь-яку систему, яка може бути запущена у віртуальній машині (використовується KVM і Qemu). Тестування може проводитися як для системи встановлення та оновлення дистрибутива, так і для перевірки типових операцій з додатками в графічній оболонці.

Особливий увагу привертає метод проведення тестування - на різних стадіях роботи дистрибутива і при запуску різних додатків створюються знімки екрану, які порівнюються з еталонними знімками екрану для визначення пройдено черговий тест чи ні. При цьому OS-autoinst виконує певні сценарії роботи користувача з системою, симулюючи раніше записані дії з клавіатурою та мишею. Система не просто маніпулює порівнянням скріншотів, а й намагається відреагувати на поточний зміст екрану, з урахуванням різноманітних відхилень, що дає можливість продовжити тестування навіть при виникненні проблем.

Керуючий Веб-інтерфейс поки підтримує тільки роботу в openSUSE та Debian, оскільки інтегрований з деякими сервісами проєктів, але в майбутньому буде адаптований і для інших систем. Через Веб-інтерфейс інженер з контролю якості може наочно оцінити суть проблеми - формується зведений звіт, а для кожної позаштатної ситуації створюються знімки екрану. Весь процес тестування фіксується у формі відеоролика. Зараз OpenQA вже використовується для тестування збірок на базі репозиторію 'Factory', в рамках якого формується випуск openSUSE 12.2.

Загалом з допомогою OpenQA розробники виконують динамічне тестування ОС (аналіз програми виконується під час її роботи). При динамічному тестуванні мають задовольнятися наступні умови:

- Для завантаження програми потрібні деякі вхідні дані;
- Динамічний аналіз виявляє дефекти тільки на шляху, який визначається конкретними вхідними даними; дефекти, що знаходяться в інших частинах програми, не будуть виявлені;
- У більшості реалізацій поява помилкових спрацьовувань виключена, тому що виявлення помилки відбувається в момент її виникнення в програмі, таким чином, виявлена помилка є не передбаченням, зробленим на основі аналізу моделі програми, а констатацією факту її виникнення.







Звідси можна зробити висновок, що основна мета цієї системи – перевірка дистрибутиву на стабільність і працездатність.











Основна перевага OpenQA полягає в тому, що ця система дає можливість автоматизувати тестування всієї операційної системи в цілому, а не окремих додатків, і позбавити розробників від нудної, рутинної роботи.

### Література




1. <http://news.opensuse.org/2011/10/11/opensuse-announces-first-public-release-of-openqa/>
2. <http://openqa.opensuse.org/results/?sort=distrib.-mtime&hours=96>
3. <http://www.protesting.ru/testing/testcase.html>
4. <http://os-autoinst.org/>




















### Додаток 1. Веб-інтерфейс OpenQA










link	distri ↑	type ↑	arch ↑ ↓	build ↑ ↓	extra ↑	testtime ↑ ↓	OK ↑ ↓	unk ↑ ↓	fail ↑ ↓
 	debian	netinst	amd64	sid_20120112		2012-01-12 00:43	11	1	
 	debian	netinst	amd64	testing_20120112		2012-01-12 00:42	11	1	
 	debian	bc	i386	testing_20120112		2012-01-12 00:04	9	3	1





 	debian	netinst	i386	testing_20120112		2012-01-11 23:21	9	3	1
 	debian	netinst	amd64	sid_20120111		2012-01-11 01:57	11	1	
 	debian	netinst	amd64	testing_20120111		2012-01-11 01:56	11	1	1
 	debian	bc	i386	testing_20120111		2012-01-11 01:18	9	3	1
 	debian	netinst	i386	testing_20120111		2012-01-11 00:34	9	3	1

Таблиця 1. Огляд результатів тестування (уривок)

Test	Result	References
timeout	not- autochecked	
<a href="#">bootloader</a>	not- autochecked	
<input type="checkbox"/> <a href="#">locale</a>	OK	

<a href="#">hostname</a>	not-autochecked			
<a href="#">userconf</a>	not-autochecked	 		
<input type="checkbox"/> <a href="#">timezone</a>	OK	 		
<a href="#">partitioning</a>	not-autochecked	 		
<a href="#">kernel_select</a>	not-autochecked	 		
<a href="#">popcon</a>	OK	 		
<a href="#">packageselection</a>	not-autochecked	 		
<a href="#">samba_workgroup</a>	not-autochecked	 		
<a href="#">grubinstall</a>	OK	 		
<a href="#">confirmreboot</a>	not-autochecked	 		

<a href="#">booted</a>	not-autochecked			
<a href="#">dist_upgrade_sid</a>	not-autochecked			
<a href="#">firstlogin</a>	not-autochecked			
<input type="checkbox"/> <a href="#">xterm</a>	OK			
<input type="checkbox"/> <a href="#">firefox</a>	OK			
<a href="#">nautilus</a>	not-autochecked			
<input type="checkbox"/> <a href="#">evolution</a>	OK			
<a href="#">reboot_gnome</a>	not-autochecked			
<input type="checkbox"/> <a href="#">reboot_wait_for_grub</a>	unknown			

<a href="#">reboot_wait_booted</a>	not-autochecked			
<a href="#">gdm_login_after_reboot</a>	not-autochecked			
<input type="checkbox"/> <a href="#">gimp</a>	OK			
<input type="checkbox"/> <a href="#">inkscape</a>	OK			
<a href="#">shutdown</a>	not-autochecked			
<a href="#">isozize</a>	OK			
<a href="#">overall</a>	OK			

## ***JOOMDLE: інтеграція CMS JOOMLA! та LCMS MOODLE*** ***Франчук В.М.***

*Національний педагогічний університет імені М.П. Драгоманова, Інститут інформатики, vfranчук@ukr.net*

Some schools and other organizations are actively using web-oriented educational computer systems to provide educational services to the global Internet using freely distributed content management system Joomla! and learning content management system MOODLE. Access to the resources of these two systems can be restricted to users who visit the site without registration. For a single user base and integration of these systems can use the component Joomdle.

Деякі навчальні заклади і інші організації активно використовують web-орієнтовані навчальні комп'ютерні системи для надання освітніх послуг в глобальній мережі Інтернет з використанням вільнопоширюваних системи управління контентом (CMS – Content Management System) Joomla! та системи управління навчальним контентом (LCMS – Learning Content Management Systems) MOODLE.

Joomla! – це вільнопоширювана система управління вмістом сайту (офіційний сайт <http://joomla.org>). Стандартний пакет системи Joomla! може бути легко і швидко встановлений користувачами без спеціальної підготовки. Після встановлення та запуску системою Joomla! можна додавати та редагувати вміст сторінок, зокрема завантажувати картинки і коригувати дані. За допомогою простого інтерфейсу можна додавати нові статті і новини, редагувати сторінки і створювати необхідну кількість контенту на сайті. Користувач, який вміє набирати текст за допомогою стандартних комп'ютерних засобів, може швидко оволодіти правилами роботи з системою Joomla! [2].

MOODLE (Modular Object Oriented Distance Learning Environment) – це система управління навчальним контентом (офіційний сайт [www.moodle.org](http://www.moodle.org)). За допомогою даної системи можна створювати навчальні курси і проводити як аудиторне (очне) навчання, так і навчання на відстані (заочне/дистанційне). Використання цієї системи забезпечує студентам доступ до навчальних ресурсів, можна надсилати нові повідомлення студентам, розподіляти, збирати та перевіряти завдання, вести електронні журнали обліку оцінок та відвідування, налаштовувати різноманітні ресурси курсу і т.д. [1].

Система MOODLE, так само як і система Joomla!, розповсюджується безкоштовно за принципами Open Source. Кожний користувач має доступ до джерела-коду системи і може його змінювати залежно від своїх цілей та бажань.

Доступ до ресурсів цих двох систем може бути обмеженим для користувачів, які відвідують сайт без реєстрації. Для створення єдиної бази користувачів та інтеграції цих систем можна використати компонент Joomla!

Joomdle – вільнопоширюваний компонент, за допомогою якого забезпечується єдиний вхід (SSO – Single Sign On) до CMS Joomla! та LCMS MOODLE (офіційний сайт <http://www.joomdle.com>). За допомогою цього компоненту користувачі можуть авторизуватися і отримати доступ до ресурсів цих двох систем через сторінку входу CMS Joomla! або сторінку входу в LCMS MOODLE [3].

В доповіді планується більш детально розкрити досвід впровадження та організації надання освітніх послуг у ВНЗ із використанням CMS Joomla! та LCMS MOODLE з використанням компонента Joomdle.

Список використаних джерел

### **Література**

1. Moodle.org: open-source community-based tools for learning. [Електронний ресурс] – режим доступу: <http://www.moodle.org>.
2. Joomla! [Електронний ресурс]. – Режим доступу: <http://www.joomla.org>.
3. Joomdle [Електронний ресурс]. – Режим доступу: [http://www.joomdle.com/wiki/Main\\_Page](http://www.joomdle.com/wiki/Main_Page).



## Україномовна локалізація MLE-MOODLE

Франчук Н.П.

Національний педагогічний університет імені М.П. Драгоманова, Інститут інформатики, [netyfr@ukr.net](mailto:netyfr@ukr.net)

MLE-MOODLE is a M-Learning extension for MOODLE (a popular open source learning content management systems). The official website does not contain Ukrainian translation. Therefore, we did a translation of this software. To work with a software tool MLE-MOODLE on a mobile device it's necessary to install a block "Mobile Access" on the home page of MOODLE.

MLE-MOODLE – вільнопоширювальний програмний засіб для підтримки мобільного доступу до системи управління навчальним контентом (LCMS – Learning Content Management Systems) MOODLE. MLE-MOODLE оснащений зручним інтерфейсом та системою допомоги, засобами для підтримки всіх етапів процесу навчання, що виділяє його з переліку інших програмних засобів цього ж класу [1].

Для роботи з програмним засобом MLE-MOODLE на мобільному пристрої доцільно на головній сторінці системи управління навчальним контентом MOODLE налаштувати блок мобільного доступу Mobile Access (Мобільний доступ).



Є два способи доступу до системи управління навчальним контентом MOODLE з мобільного пристрою:

- за допомогою спеціального додатка для мобільних пристроїв (потрібно завантажити спеціальну програму та встановити її на конкретній моделі мобільного пристрою, посилання Завантажити).
- за допомогою браузера мобільного пристрою (потрібно ввести, наприклад, адресу

<http://www.moodle.ii.npu.edu.ua/blocks/mle/browser.php> у рядок адреси браузера мобільного пристрою).

Використовуючи другий спосіб, отримуємо доступ до системи дистанційного навчання під управлінням системи MOODLE з мобільного пристрою.

Наявність доступу до навчальних курсів за допомогою мобільних пристроїв надає можливість:

- викладачеві: здійснювати оперативний контроль за станом навчальних матеріалів курсу, стежити за навчальною діяльністю студентів, зокрема за звітуванням про виконання індивідуальних

завдань, завдань до лабораторних робіт, проходженням тематичного тестування тощо;

- студенту: переглядати навчальні матеріали, зокрема відео-лекції, читати новини курсу, отримувати повідомлення від викладача, спілкуватися з одногрупниками тощо.

На жаль, у встановленій версії MLE-MOODLE не підтримуються деякі важливі функції, що є в системі MOODLE, зокрема режими меню курсу «Управління», як з боку викладача, так і з боку студента, є проблеми з режимом перегляду і управління завданнями, які були надіслані студентами у вигляді файлів, та деякі інші, недоступні в мобільній версії й пункти меню курсу «Діяльності» [2].

Зазначені обмеження в управлінні навчальним курсом дещо звужують можливості застосування системи управління мобільним навчанням MLE-MOODLE, але є сподівання, що вони будуть усунені найближчим часом, оскільки система досить швидко розвивається.

На офіційному сайті програмного засобу MLE-MOODLE немає перекладу його інтерфейсу українською мовою, тому переклад було зроблено в НПУ імені М.П. Драгоманова на кафедрі теоретичних основ інформатики.

Переклад програмного засобу українською мовою можна зробити за допомогою стандартних засобів системи MOODLE, для цього потрібно скористатися послугою головного меню: Мова інтерфейсу ->Редагування, та перекласти вміст таких файлів:

- mle.php; block\_mle.php;
- block\_mobile\_repository.php;
- block\_online\_users\_mobile.php;
- flashcardtrainer.php; mlo.php; mobiletags.php.

Також можна завантажити перекладені файли цього програмного засобу українською мовою з сайтів: НПУ імені М.П. Драгоманова <http://www.moodle.ii.npu.edu.ua/mod/resource/view.php?id=2999> або «Система дистанційного навчання ЧДТУ» <http://ias.cdtu.edu.ua/moodle19-test/mod/resource/view.php?id=31886>.

У доповіді планується більш детально розкрити досвід розробки українського інтерфейсу програмного засобу MLE-MOODLE та його використання в навчальному процесі у вищих навчальних закладах.

## **Література**

1. MLE – Mobil Learning Engine <http://mle.sourceforge.net/>
2. Триус Ю.В., Франчук В.М., Франчук Н.П. Організаційні й технічні аспекти використання систем мобільного навчання // Науковий часопис нпу імені М.П. Драгоманова. Серія № 2. Комп'ютерно-орієнтовані системи навчання: Зб. наук. праць /Редрада. – К.: НПУ імені М.П.Драгоманова, 2012. – № 12 (19). – С. 53-62.

## Використання бібліотеки Qt для написання гітарного тюнера Голубник М.Г.

Львівський національний університет імені Івана Франка, [gmaryg@email.ua](mailto:gmaryg@email.ua)

Digital technologies are filling our environment more and more every day, and this process is accelerating. The activities for which a computer has caused real revolution are audio playback and music. At sound processing the main role is played by transformations to and from electric signals and recording to the media. Also, various tuners are used for tuning musical instruments. This particular tuner is developed for single score playing and allow to tune a guitar without even hearing.

Гітарний тюнер реалізований за допомогою кросплатформної бібліотеки Qt мови C++. Наша програма отримує сигнал із мікрофону. Для того щоб працювати із аудіовходом (мікрофоном) ми використовуємо клас бібліотеки Qt QAudioInput (модуль QtMultimedia).

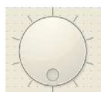
Модуль QtMultimedia- набір класів, які надають низькорівневу мультимедійну функціональність. Цей модуль застосовується для запису і відтворення мультимедійних даних.

- QAudioDeviceInfo – це клас , який надає інтерфейс для запиту аудіо-пристроїв та їх функціональних можливостей. QAudioDeviceInfo дає змогу записувати дані для аудіопристроїв - таких, як звукові карти і USB-гарнітури - що в даний час доступні в системі. Аудіопристрої залежать від платформи або аудіоплагінів.
- QAudioFormat - цей клас складається з параметрів аудіоінформації. Аудіоформат визначає, як влаштовані дані в звуковому потоці, тобто, як потік має бути інтерпретований. QAudioFormat містить інші параметри, які вказують, як аудіодані організовані. Це частота, кількість каналів, обсяг вибірки, тип зразка, і послідовність байт. Можна отримати аудіоформи, сумісні з аудіоприроями, що використовується через функції в QAudioDeviceInfo. Цей клас також дозволяє робити запит доступних значень параметра для пристрою, так що ви можна встановлювати параметри самостійно.
- QAudioInput — клас, який надає інтерфейс для отримання аудіоданих з пристрою введення звуку. Можна створити QAudioInput з конкретними QAudioDeviceInfo. Дає змогу записувати звук з аудіопристроїв введення. Конструктор за замовчуванням цього класу буде використовувати звуковий пристрій за замовчуванням, але ви можете також вказати QAudioDeviceInfo для конкретного пристрою. Ви також повинні зайти в QAudioFormat, в якій ви хочете записати. Запуск QAudioInput це просто питання виклику Start () з QIODevice відкритий для запису. QAudioInput надає кілька способів виміру часу, що пройшов з моменту початку () запису.

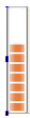
- QAudioOutput - клас, який надає інтерфейс для відправки звукових даних на пристрій виводу звуку. Ви можете побудувати аудіовихід за замовчуванням системи пристрій виводу звуку. Крім того, можна створити QAudioOutput з конкретними QAudioDeviceInfo. При створенні аудіовиходу, також повинні направити в QAudioFormat дані, які будуть використовуватися для відтворення.

Алгоритм роботи програми полягає в наступному:

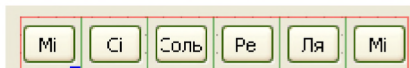
1. Програма обробляє звук, отриманий з мікрофонного входу комп'ютера.
2. Обчислюється дискретне перетворення Фур'є.
3. Отримана частота порівнюється з еталоном.
4. Відправляється команда на пристрій налаштування струни.  
Форма гітарного тюнера



- QDial – для виставлення рівня чутливості мікрофону;



-QProgressBar – для представлення рівня чутливості мікрофону;



- QPushButton – для представлення ноти відповідної частоті прийнятій із мікрофона;

## **Системи дистанційного навчання** **Іванов В.Г., Гвозденко М.В., Кошева Н.А.**

Національний університет “Юридична академія України імені Ярослава Мудрого”, [Gvozdenko@rambler.ru](mailto:Gvozdenko@rambler.ru)

Modern technologies of teaching are widely used system of distance education (e-learning) on the basis of OpenSourceLMS\LCMS. The report considered the advantages of distance learning systems on the basis of Open SourceLMS\LCMS, examples of modern e-learning and their capabilities.

Сучасні технології навчання широко використовують системи дистанційної освіти (СДО) на базі OpenSourceLMS\LCMS. Не поступаючись комерційним системам дистанційної освіти, системи на основі OpenSourceLMS\LCMS мають безперечні переваги:

- система може бути адаптована під потреби конкретної моделі освітнього процесу без додаткових витрат;
- у розвитку системи бере участь велика кількість фахівців, що дає змогу гнучко і оперативно реагувати на змінні методичні і технічні запити освітнього процесу;
- відсутність жорсткої прив'язки до постачальника дає змогу отримувати оновлення і нові версії системи в режимі вільного доступу;
- саме налаштування і адаптація системи, розробка нових модулів і підсистем може бути використана як навчальний матеріал у вузах певного профілю.

Засоби OpenSourceLMS\LCMS постійно розвиваються, збільшуючи функціональність, адаптаційні можливості і стаючи все більш універсальними. Лідером ринку СДО з із відкритим кодом на сьогоднішній день є Moodle. Нова версія відкритої системи дистанційного навчання (СДО) Moodle 2.2 написана на PHP з використанням SQL-СУБД (MYSQL, POSTGRESQL, Microsoft SQL Server і ін. — використовується ADOdb XML). Moodle може працювати із об'єктами SCO і відповідає стандарту SCORM. Система поширюється за ліцензією GNU GPL і дає змогу:

- створювати сайти для онлайн-навчання|;
- використовувати рубрики оцінювання|, що реалізують багатокритеріальне оцінювання знань - після оцінки викладача, яка виставляється з урахуванням багатьох критеріїв, оцінки автоматично нормуються за рівнем інших учнів.
- використання розширеної підтримки стандарту IMS, що дає змогу під'єднувати до курсів завдання, розташовані на сторонніх сайтах або написані іншими мовами програмування. Завдяки стандарту IMS зовнішні завдання можуть отримувати інформацію про учня, що складає іспит (або залік), і повертати отриману оцінку.
- тема оформлення MyMobile оптимізована для роботи із смартфонів.



Новим продуктом на ринку СДО з відкритим кодом є додаток Open Class, спільна розробка компанії Pearson і Google. Система базується на хмарному хостингу Pearson. Доступ до Open Class здійснюється через Google Apps for Education, вхід в систему - через аккаунт Google. Система має функції соціальної мережі і дає змогу управляти дистанційними курсами. У системі використовуються Gmail і GoogleDocs. OpenClass має стрічку новин, що містить коментарі і активність користувачів та сторінки відгуків з питаннями і пропозиціями учнів.

Для забезпечення проведення всіх форм дистанційного навчання в режимі відеоконференцій розроблено багатокористувацьке застосування BigBlueButton. Це застосування працює через Веб-переглядач і не вимагає встановлення додаткового програмного забезпечення на комп'ютери користувачів. Застосування дає змогу використовувати свою камеру, велику кількість налаштувань для презентацій і обмін файлами з іншими користувачами.

### **Спотворення спектрів сигналів внаслідок джиттеру частоти дискретизації**

*Яремчук О.М., Дзюба С.П.*

*м. Миколаїв, Чорноморський державний університет ім. Петра Могили, [olga-yaremchuk.77@mail.ru](mailto:olga-yaremchuk.77@mail.ru)*

This article describes how to use the software LTspise IV to simulate the distortion of the signal due to jitter sampling.

Джиттер сигналу має різні характеристики в залежності від його причин та джерел. Джиттери поділяють на дві основні категорії: випадковий (random jitter - RJ) і регулярний (deterministic jitter - DJ).

Випадковий джиттер обумовлений шумовими процесами, що відбуваються у всіх напівпровідниках і компонентах

Регулярний джиттер викликається діючими на сигнал процесами, що відбуваються в системному обладнанні, а також, може з'явитися за певних способах подання переданих даних. Регулярний джиттер залежить від характеристик цифрової системи. Приклади джерел системного джиттера:

- Перехресні перешкоди від випромінюваних або переданих сигналів;
- Вплив дисперсії при поширенні сигналу;
- Неузгодженість опорів.

Дуже часто відбувається фазове тремтіння цифрового сигналу даних - небажані фазові і/або частотні випадкові відхилення сигналу, що передається. Виникають вони внаслідок нестабільності передавального генератора, змін параметрів лінії передачі в часі і різної швидкості поширення частотних складових одного і того ж сигналу. У цифрових системах це проявляється у вигляді випадкових швидких (з частотою 10 Гц і більше

) змін місця розташування фронтів цифрового сигналу в часі, що призводить до розсинхронізації і, як наслідок, спотворення переданої інформації.

Дискретизація сигналу за часом відповідає тому, що вихоплюється з безперервного сигналу його значення в окремі, рівновіддалені на період опитування, моменти часу. Ця операція еквівалентна множенню сигналу на періодичний імпульсний сигнал рівний 1 в точках опитування і рівний 0 у всіх інших точках. При цьому спектр вихідного сигналу піддається певним спотворенням.

Моделювання та дослідження спектрів сигналів, спотворених джиттером, відбувалося у середовищі LTspice IV. При розгляді результатів моделювання, стає помітною різниця часових характеристик між ідеальним сигналом (рис.1) та спотвореним сигналом.

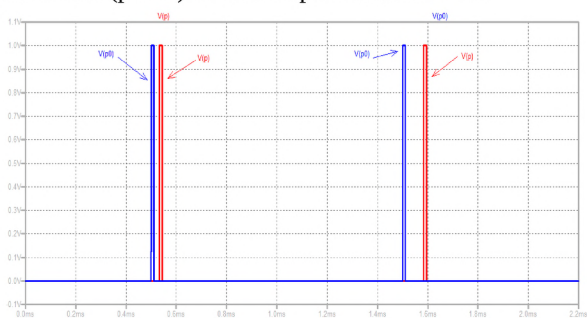


Рис.1. Графічне представлення джиттера сигналу.  $V(p0)$  – ідеальний сигнал (показаний синім кольором),  $V(p)$  – сигнал, спотворений джиттером (червоний колір)

При проведенні швидкого перетворення Фур'є стає очевидною різниця між спектрами ідеального та спотвореного сигналу. Збільшивши масштаб на графіку, детально видно різницю між ідеальним імпульсним сигналом, та сигналом з джиттером (рис. 2).

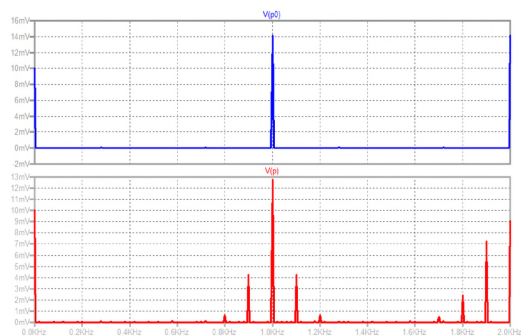


Рис. 2. Спектр вихідних сигналів при відношенні частоти сигналу до частоти джиттера, рівному 10.



У спотвореного сигналу з'являються додаткові спектральні піки – частоти джиттера, які повторюють форму спектра сигналу. Розташування даних піків на частотній осі відповідає 0.9 кГц та 1.1 кГц, адже опорна частота дорівнює  $f_0 = 1$  кГц, а частота джиттера  $f_j = 100$  Гц. Відтак при згортці сигналів  $f_0 + f_j = 1.1$  кГц, а  $f_0 - f_j = 0.9$  кГц.

При згортці синусоїдального сигналу та імпульсного сигналу без/з джиттером отримано наступні графіки вихідних сигналів, різниця яких ззовні ледве помітна. Однак у спектральному представленні вона стає очевидною (рис. 3).

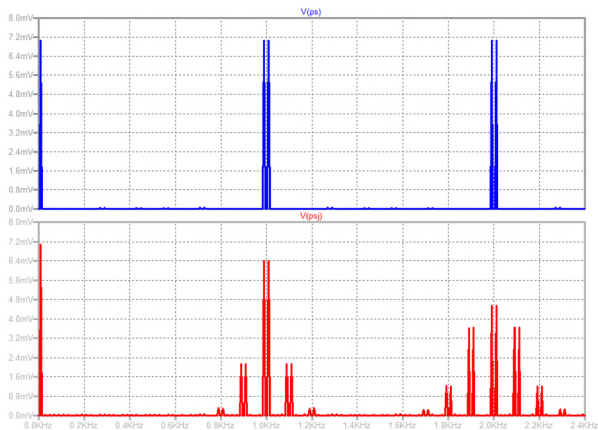


Рис. 3. Згортка спектрів синусоїди та імпульсного сигналу без/з джиттером,  $V(ps)$  та  $V(psj)$  відповідно

Як бачимо, згортка спектрів дала дві опорні частоти 0.9 кГц та 1.1 кГц, з кожною наступною гармонією амплітуда яких стає все меншою за рахунок утворення додаткових спектральних піків внаслідок джиттеру

Для практичного використання при дискретизації сигналу треба обов'язково звертати увагу на джиттер, який може суттєво спотворити спектр вихідного сигналу та призвести до помилкової інтерпретації результатів вимірювання. При джиттері відбувається зменшення амплітуди спектрального піка опорної частоти за рахунок утворення бокових піків на кожній гармоніці.

### Література

1. *Indepth: Jitter*. – [Цум. 2011, 11 лютого]. – Доступний з: <http://www.voiptroubleshooter.com/indepth/jittersources.html#top>.
2. *Джиттер. Теорія. Частина 1*. – [Цум. 2011, 21 лютого]. – Доступний з: <http://www.ixbt.com/proaudio/jitter-theory-part1.shtml>.
3. *Jitter explained - Part 1.3 [English].mht*. – [Цум. 2011, 21 лютого]. – Доступний з: [http://www.tnt-audio.com/clinica/jitter1\\_e.html](http://www.tnt-audio.com/clinica/jitter1_e.html).

## **Перспективы развіцця свабоднага забеспячэння ў беларускіх “Аблоках”**

*Карагода А.І., Стальбоўская Н.І.*

*Беларускі нацыянальны тэхнічны універсітэт, [zelenu\\_ochi@mail.ru](mailto:zelenu_ochi@mail.ru)*

Інфармацыйныя тэхналогіі змяняюцца вельмі хутка і сёння яны ўтвараюць своеасабівы пласт, які ўсе больш значна ўплывае на ўсе аспекты нашага жыцця.

Апошнія некалькі год усё больш ідзе гаворка пра “аблаковыя вылічэнні”. Гэта новая мадэль ІТ-паслуг пад назвай cloud computing (“аблаковыя вылічэнні”) за апошні час дазволіла значна паменшыць ІТ-выдаткі і складанасць інфраструктур, а таксама аптымізаваць працоўныя напетаванні і функцыянаванне ІТ-сэрвісаў.

“Аблаковыя вылічэнні” – тэхналогія, як і раней застаецца вельмі маладой, і стварэнне дадаткаў, якія працаюць на серверах cloud-правайдэра патрэбна ўвесь час дапрацовываць. Больш за тое, паслугі патрабуюць ад кліентаў вялікі шэраг навыкаў у воблаці праграмавання. Windows Azure выкарыстоўваюць платформу для разпрацоўшчыкаў Microsoft .NET; Google App Engine выкарыстоўваюць папулярную мову праграмавання Python, і сэрвіс Salesforce.com выкарыстоўвае ўласную мову разпрацоўкі дадаткаў Apex. Сёння найбольш буйнымі прадпрыемствамі індустрыі аблаковых вылічэнняў з’яўляюцца Google, Amazon, Microsoft, IBM, Sun, Ubuntu і інш.

Сёння на тэрыторыі Рэспублікі Беларусь працуе больш за 320 ІТ-кампаній, большая частка якіх з’яўляюцца часткай еўрапейскіх. Канешне іх нельга параўноўваць з такімі найбуйнейшымі сусветнымі асілкамі як Microsoft, Google і IBM, асабліва ў тых пытаннях, якія датычацца “аблаковых вылічэнняў”, аднак, сёння іх развіцце павінна быць больш буйным.

Для большай часці беларускіх карыстальнікаў, як і для беларускіх ІТ-кампаній, стаіць пытанне вялікага кошта здзяйснення “аблаковых” паслуг. У Беларусі cloud computing атрымлівае свае развіццё толькі дзякуючы існаванню безкаштоўнага праграмнага забеспячэння, платформ і сэрвісаў. Трэба адзначыць, што большая частка якіх з’яўляецца распрацоўкай замежных кампаній. Разгледзім далей самыя папулярныя сярод іх.

Cloudstack — гэта кансоль кіравання вылічальнымі рэсурсамі датацэнтра. На гэтай платформе пабудавалі свае аблогі такія буйныя кампаніі як Zynka, Nokia Research Center, Cloudcentral і інш. Развіццё праекта ажыццяўляецца кампаніяй Citrix. Трэба адзначыць, што сёння Cloudstack безкаштоўны і распаўсюджваецца пад ліцэнзіяй GNU Public License Version 3. Калі карыстальніка зацікавіць грашовая падтрымка, то за інфармацыяй трэба звярнуцца да разпрацоўшчыкаў праз email.

Eucalyptus — яшчэ адна платформа для будавання аблокаў. Свае Private Cloud, пабудавалі на гэтай платформе, такія буйныя кампаніі як: Sony,

Puma, NASA, Trend micro і інш. Існуе 2 рэдакцыі Eucalyptus: грашовая і безкаштоўная. У гэтых версіях вельмі значна адрозніваецца функцыянал. Асноўная вартасць, якую Eucalyptus спрабуе данесці да карыстальнікаў, гэта тое, што яго API цалкам сумяшчальна з Amazon API. Eucalyptus падтрымлівае 3 гіпервізара: XEN, KVM, ESXi. Апошні (ESXi) падтрымліваецца толькі ў Enterprise версіі. Трэба адзначыць, што безкаштовая версія, нажаль, не вельмі актыўна развіваецца.

Openstack 2011.3 – платформа з адкрытым зыходным кодам для будавання аб'ектаў. У праект Openstack уваходзяць 3 прадукта: Nova (аналаг Amazon EC2), Swift (аналаг Amazon S3), Glance (сэрвіс для выдаткоўвання вобразу). Сёння Nova падтрымлівае на высокім узроўні 2 гіпервізара: KVM і XEN. Платформа хутка развіваецца і абяцае вялікі функцыянал. Мае па-за сабою вельмі вялікую супольнасць і падтрымку з боку такіх карпарацый як: Cisco, Dell, NASA, Intel, AMD, Citrix, Rackspace, Rightscale. Прадукт з'яўляецца безкаштоўным.

У межах канферэнцыі WWDC 2011 кампанія Apple выдаткавала iCloud - набор аб'ектавых паслуг, працуючых з дадаткамі для iPhone, iPad, iPod touch, Mac або уласных кампутараў, якія дазваляюць аўтаматычна захоўваць змест у iCloud а таксама аўтаматычна перадаваць інфармацыю ў выглядзе push-паведамленняў на іншыя прылады па бяспроўнаму злучэнню. 5 Гбайт дыскавай прасторы, дадзенага карыстальнікам, дазваляюць захоўваць музыку, фотаздымкі, дадаткі, пошту, календары і гэтак далей.

Windows Live SkyDrive – безумоўна першы нумар па памеру дыскавай прасторы, безкаштоўна дадзенай ў карыстанне зарэгістраваным карыстальнікам — 25 Гбайт! Можна захоўваць файлы розных фарматаў.

Кампанія Canonical дала на карыстанне Windows-версію свайго папулярнага карыстальніка аб'ектавай паслугі захоўвання дадзеных Ubuntu One, якая раней была дасяжна толькі карыстальнікам Ubuntu, Android і iPhone. У выніку уладальнікі Windows-УК таксама атрымалі магчымасць безкаштоўна захоўваць да 5 Гбайт анлайн-дадзеных.

І гэта далёка не поўны спіс усіх безкаштоўных сэрвісаў, якія выдаткоўваюць сёння кампаніі сваім карыстальнікам. Большая частка гэтых кампаній выдаткоўваюць толькі частку сваіх рэсурсаў дзеля безкаштоўнага выкарыстання, большай часткай для знаёмства з асноўнымі магчымасцямі і функцыяналам. І канешне ж, гэта вельмі паўплывае на студэнтаў беларускіх ВНУ. Таму што, напрыклад, для таго каб напісаць курсавы праект, у якім поўнасцю або часткова выкарыстоўваецца “аб'екты”, патрэбна шукаць або безкаштоўны сэрвіс, або шукаць уласныя сродкі для аранды сродкаў. Гэта з'яўляецца адной з асноўных нагод, чаму з кожным годам усё больш студэнтаў, маладых навукоўцаў і працоўнікаў ІТ- кампаній, якія актыўна працуюць развіццю і папулярнасцю безкаштоўнага праграмага забеспянення.

## **Правові аспекти використання ІКТ у навчальних закладах** *Хачіров Т.С.*

*Харківський національний педагогічний університет ім. Г.С. Сковороди*  
*ktim@ukr.net*

The aim of a thesis is to cover issues of using intellectual property and copyrights in the educational institutions of Ukraine. It highlights provisions in international agreements and laws concerning the settlement of these issues. The necessity of legal training teachers of computer science is substantiated.

Інформаційно-комунікаційні технології все глибше проникають у сферу освіти. За офіційними даними Міністерства освіти, молоді та спорту України, на сьогодні вже 96% українських шкіл оснащені комп'ютерами, 86% підключені до мережі Інтернет [1]. Державною цільовою програмою впровадження у навчально-виховний процес загальноосвітніх навчальних закладів інформаційно-комунікаційних технологій «Сто відсотків» на період до 2015 року передбачено повну інформатизацію всіх загальноосвітніх навчальних закладів та їх безкоштовне підключення до Інтернету [2].

Міністр МОНМС Д. Табачник в інтерв'ю російському інтернет-виданню «Сучасна школа Росії» (02.11.2011) зазначив «Разом з Європейським Союзом ми впроваджуємо нову програму, яка називається «Life-long Learning», тобто, «Навчання впродовж життя». В цьому році ми поставили завдання 100%-но підготувати всіх вчителів математики, геометрії, алгебри та інформатики з точки зору комп'ютерної грамотності.» [3].

У зв'язку з викладеним вище особливо гостро постає питання про дотримання авторських прав у процесі використанні інтелектуальної власності. Технічна доступність електронних творів (текстів, програм, відео, тощо), можливість скопіювати практично все, що представлено в мережі Інтернет, призвели до масових порушень авторського права [4]. Відповідно до чинного законодавства, вчитель/викладач інформатики несе відповідальність за використання програмного забезпечення на комп'ютерах навчальній аудиторії [5]. Сьогодні не надається належної уваги питанням підготовки учнів загальноосвітніх навчальних закладів (шкіл, гімназій, ліцеїв, ін.) з основ інтелектуального права. Найчастішими причинами такого стану є недостатня обізнаність учителів з відповідними законодавчими актами і брак методичної літератури. З іншого боку, попри безкоштовність і надійність таких продуктів FOSS, як операційна система LINUX, OPEN OFFICE та інші, на цей час у навчальних закладах вони використовуються вкрай мало. На Першій міжнародній конференції «FOSS Sea 2008» (вересень, 2008) Г. Злобіним, одним з ініціаторів запровадження вільного програмного забезпечення у школи Львівщини, було зазначено, що незацікавленість у переході на вільно поширюване

програмне забезпечення спричинена відносною безкарністю використання піратських копій Windows.

Слід зазначити, що за останні 20 років були прийняті 15 міжнародних угод [6], які регулюють авторське право. Серед них:

- 2008. Prioritizing Resources and Organization for Intellectual Property (PRO-IP)
- 2005. Family Entertainment and Copyright Act:
- 2004. Intellectual Property Protection and Courts Amendments Act

Більше 10 законів були прийняті в період з 1990 р. по 1999 р. у зв'язку з швидкими темпами розвитку цифрових технологій. Останнім часом були здійснені спроби прийняти такі законодавчі акти як:

- SOPA (Stop Online Piracy ACT) – закон дозволяє спершу закрити або блокувати сайт за однією підозрою в розповсюдженні інформації без дотримання авторського права. Він дозволяє закрити весь сайт на основі одного випадку порушення авторських прав, а так само дозволяє вважати спрямованим на порушення закону будь-який сайт, який може містити генерований користувачем контент. Винесено на розгляд конгресу США 26 жовтня 2011 року [7].
- PIPA (PROTECT Intellectual Property Act) – дозволяє блокувати, використовуючи платіжні системи, фінансування сайтів, тільки підозрюваних в нелегальному використанні/розповсюдженні об'єктів авторського права (після отримання санкції суду), відключати DNS і змушувати провайдерів від'єднувати сервери, на яких розміщений сайт. Винесено на розгляд конгресу США 12 травня 2011 р. [8].

SOPA і PIPA є проектами законів, які розроблені в США, однак і в Європі існують подібні закони:

- Директива ЄС про авторське право (англ. EUCD, European Union Copyright Directive) [9].
- Міжнародна угода по боротьбі з контрафактною продукцією АСТА (The Anti-Counterfeiting Trade Agreement) [10].

Таким чином, сьогодні вкрай важливо розробити і включити в програму навчання як вчителів та викладачів інформатики, так і в шкільну програму набори тематичних лекцій з авторського права та інтелектуальної власності. Разом з тим слід вносити в програму навчання молодих фахівців більше тем, присвячених вільному ПЗ та ПЗ з відкритими вихідними кодами.

Правові знання необхідні вчителю для:

1. *Коректного використання ресурсів (інформаційних, програмних) в роботі;*
2. *Для проведення роботи з учнями щодо формування норм поведінки в мережі (Інтернет стає доступним на уроці, значить, відповідальність за поведінку учня несе вчитель);*
3. *Для профілактики масово поширеної практики плагіату. (подолання хибного уявлення про Інтернет-ресурси як «загальну» - «нічийну» власність).*



На жаль, жодна з існуючих чинних програм шкільного курсу інформатики, що затверджена МОНМС, не відводить жодної години навчального часу для ознайомлення з існуючими нормативними документами та правовими нормами. Учні не вивчають існуючі публічні ліцензійні угоди, не набувають навичок та вмінь роботи з правовою документацією, і це призводить до порушень авторського права в сфері інформаційних технологій.

### **Література:**

1. Табачник пообещал до конца 2013 года оснастить все школы компьютерами. <http://ua.rian.ru/politics/20111216/78944660.html>
2. Новый стандарт начатковой школы будет запроваджено з 1 вересня 2012 року. [http://www.kmu.gov.ua/control/publish/article?art\\_id=244795395](http://www.kmu.gov.ua/control/publish/article?art_id=244795395)  
<http://www.mon.gov.ua/index.php/ua/7206-dmitro-tabachnik-poinformuvav-zmi-pro-zdobutki-v-seredni-osviti-za-2011-rik>
3. Дмитрий ТАБАЧНИК: «Увеличение престижа профессии учителя – одна из моих основных стратегических задач». <http://www.remote-teach.oworld.ru/default.aspx?s=0&p=212&0a0=t&0n0=374>
4. Торговая палата США накажет Украину за пиратство. 1 <http://www.bezpeka.com/ru/news/2012/02/10/us-to-fine-ukraine.html>.
5. Рекомендації щодо забезпечення правомірності використання комп'ютерних програм у діяльності суб'єктів господарювання. [http://sips.gov.ua/ua/soft\\_recommendations.html](http://sips.gov.ua/ua/soft_recommendations.html)
6. How Much Is Enough? We've Passed 15 'Anti-Piracy' Laws In The Last 30 Years. <http://www.techdirt.com/articles/20120215/04241517766/how-much-is-enough-weve-passed-15-anti-piracy-laws-last-30-years.shtml>
7. SOPA. <http://thomas.loc.gov/cgi-bin/query/z?c112:H.R.3261>
8. PIPA. <http://thomas.loc.gov/cgi-bin/query/D?c112:2:./temp/~c1128t5wza>
9. Directive 2001/29/EC (EUCD, European Union Copyright Directive) <http://www.ipo.gov.uk/200121ec.pdf>
10. ACTA. [http://trade.ec.europa.eu/doclib/docs/2011/may/tradoc\\_147937.pdf](http://trade.ec.europa.eu/doclib/docs/2011/may/tradoc_147937.pdf)
11. Jan Hargreaves. A review of Intellectual Property and Growth. <http://www.ipo.gov.uk/ipreview.htm>



## **Використання електронного навчального курсу математики як засобу підготовки до ЗНО**

*Харченко В.М., Ваврикович Л.В., Карпенко І.І.*

*Ніжин НДУ ім. Миколи Гоголя, Ніжинський обласний педагогічний ліцей Чернігівської обласної ради, volmkhar@gmail.com*

Розвиток освіти неможливий без аналізу показників її якості. На даний час до основних таких показників відносять зовнішнє незалежне оцінювання (ЗНО).

Згідно з офіційним звітом Українського центру оцінювання знань за 2011 р. кількість випускників, які набрали від 100 до 135,5 балів, становить 18,75%, а від 183,5 до 200 – 14,9% від загальної кількості тих, хто брав участь у тестуванні з математики [12, 343]. Це означає, що якість знань з математики тих випускників, які хочуть здобувати не гуманітарну освіту, бажає бути кращою. Як стверджують дослідники проблем ЗНО, для випускників є дещо незвичними завдання в тестовій формі, оскільки в підручниках з математики недостатня кількість завдань такого типу. Для покращення ситуації в [16, 212] запропоновано поступову підготовку школярів до «специфічної діяльності з виконання завдань ЗНО через органічне включення тестових форм поточного контролю, допомагаючи учням оволодівати технікою роботи з тестами».

Метою публікації є ознайомлення з можливостями використання відкритих систем дистанційного навчання для підготовки учнів до ЗНО, представлення окремих аспектів використання електронного навчального курсу під час підготовки учнів до ЗНО.

Проблеми створення і впровадження в навчальний процес комп'ютерно-орієнтованих методичних систем навчання природничо-математичних дисциплін і інформатики в школах і вищих навчальних закладах досліджували М.І. Жалдак, В.І. Клочко, Н.В. Морзе, С.А. Раков, І.В. Роберт та інші.

Загальні аспекти дистанційного навчання розкрито у роботах Т. Андерсона, А.А. Андреева, В.Ю. Бикова, В.М. Кухаренка, Є.С. Полат, психолого-педагогічні аспекти і технології створення дистанційного курсу – В.Ю. Бикова, В.М. Кухаренка, Н.Г. Сиротенка, Н.В. Морзе, Ю.С. Рамського, В.П. Сергієнка, Є.М. Смирнової-Трибульської; аспекти використання освітніх вимірювань – В.С. Аванесова, В.Є. Бахрушина, Л.М. Гриневич, Л.П. Дворецької, М. Зелмана, С.А. Ракова; методичні та дидактичні проблеми і перспективи використання інформаційних технологій у навчанні – Г.А. Атанова, М.І. Жалдака, А.П. Єршова, В.І. Монахова, Н.В. Морзе, А.В. Хуторського, О.І. Скафи, Ю.В. Триуса. Проте досліджень психолого-педагогічних аспектів використання дистанційних курсів при вивченні математики в загальноосвітніх школах не достатня кількість.

У ЗНО з математики передбачається перевірка навчальних досягнень випускників з програмового матеріалу за 5 - 11 класи. Така перевірка використовує педагогічний тест. Згідно з [5, 7], «педагогічний тест ... – це набір завдань з певної галузі знань (навчального предмета), побудованих за певними правилами викладу умови і надання відповіді, за допомогою яких оцінюється рівень навчальних досягнень учнів (знання, вміння, предметна компетентність тощо)». Хоча педагогічне тестування має багато недоліків, але воно дає змогу:

- перевірити результати навчальних досягнень водночас із багатьох тем і розділів програми;
- об'єктивно оцінити рівень засвоєння навчального матеріалу;
- створити для всіх учасників тестування рівні умови складання тестів;
- стандартизувати та автоматизувати процедуру перевірки результатів;
- охопити тестуванням велику кількість учнів [5, 7-8].

Зрозуміло, що добре підготуватися до ЗНО тільки за рахунок відведених програмами з алгебри та геометрії годин на повторення матеріалу не вдасться. Важливу роль відіграє при цьому самостійна робота випускників під керівництвом вчителя. Для організації такої роботи доречно буде створити курс математики за зразком електронного навчального курсу (ЕНК) [8; 13]. Згідно з [8, 63; 13, 1] «ЕНК передбачений для оволодіння студентами навчальним матеріалом під керівництвом викладача». Використовуючи його розділи учитель зможе організувати ефективну діяльність учнів з підготовки до незалежного оцінювання.

Для організації курсу з математики потрібно вибрати систему дистанційного навчання на основі якої можна організувати такий курс. Оскільки сучасне матеріальне становище загальноосвітніх навчальних закладів, у переважній більшості, не дозволяє використовувати пропрієтарне програмне забезпечення, то доречно зупинитися на відкритих дистанційних системах. У дослідженні [2] подано коротку характеристику шести таких систем. Враховуючи той факт, що систему Moodle можна встановити й для локальної мережі шкільних комп'ютерів, вона достатньо добре описана в літературі, - доречно вибрати саме її. У нашому випадку курс створено в LMS Moodle, який розміщений на сервері НДУ ім. Миколи Гоголя.

Під час створення курсу математики було враховано, що програма підготовки до ЗНО 2012 містить 6 розділів [14]. Врахувавши цей факт, а також необхідність допоміжних розділів, був створений курс із 9 розділів. У розділі новин розміщені програма ЗНО, специфікація тесту ЗНО, короткий опис курсу, відомості про авторів, інструкція до тестування та форум для обговорення проблем підготовки до ЗНО. Зауважимо, що інструкція до тестування потрібна лише на початковому етапі роботи учнів. Оскільки у разі самостійного проходження перших тестів в школярів виникають певні проблеми із переглядом отриманих результатів.

Кожен із наступних шести розділів містить мотиваційний вступ, короткі теоретичні дані, методичні рекомендації щодо розв'язання завдань даного розділу та тести. У кожному розділі створено тести, які перевіряють досягнення у вивченні конкретної теми, а також підсумковий тест. Тематичні тести розраховані на 15-20 хвилинну роботу учнів, а підсумкові – на 40 хвилинну. Окремим розділом виділено 5 пробних тестів, виконання яких потребує 150 хвилинної роботи. Список рекомендованої літератури також виділено окремим розділом. Він містить шкільні підручники з математики за 5-11 класи, наприклад, [1; 6-7; 9-10] та посібники типу [4; 11].

Для того, щоб під час проходження тестування в учнів були різні завдання з одного й того ж розділу, створено банк запитань, який містить більше 1000 задач. Задачі взято із варіантів ЗНО за минулі роки та посібників аналогічних до [4]. Як і на ЗНО кожна категорія задач розбита на підкатегорії: задачі на відповідності, на вибір однієї відповіді та уведення короткої відповіді. Найменша кількість тестових завдань на відповідності – близько 100. Створений таким чином банк задач дає змогу по кілька разів тестувати учнів з однієї й тієї теми. При цьому повтори завдань – мінімальні. Залежно від призначення тесту учні його можуть проходити лише один раз або ж необмежену кількість разів. Єдине обмеження – інтервал часу між сусідніми тестуваннями – 1 доба. Це зроблено для того, щоб учні могли виявити прогалини в своїх знаннях та усунути їх. А після цього ще раз перевірити якість своїх знань. Частина тестів призначена для контролю знань на уроках алгебри чи геометрії. Тому учні можуть проходити їх лише один раз. Повторне виконання таких тестів на уроці не передбачено.

Цей курс проходить апробацію в Ніжинському обласному педагогічному ліцеї Чернігівської обласної ради. Специфіка цього закладу полягає в тому, що до нього в десяті класи набираються учнів сільських шкіл Чернігівської області. Незважаючи на певний відбір поступаючих до класу з профільним вивченням математики, їх знання на початку навчання в ліцеї дуже відрізняються. Для вирівнювання знань активно почали використовувати створений ЕНК.

Оскільки ЕНК з математики не розрахований на його перегляд гостями, то попередньо учителю необхідно зареєструвати свій клас, створити відповідну групу. Перше тестування проходить під час повторення навчального матеріалу за 7-9 класи. Тоді ж пояснюється не тільки сам процес тестування, а й перегляду своїх результатів та правильних відповідей. Проте на перших порах учням складно запам'ятати всі необхідні дії. Тому у розділ новин, після проведених кількох тестувань, було додано інструкцію щодо процесу проходження і перевірки тестування.

Передбачається, що не всі школярі пам'ятають ті теоретичні відомості, які потрібні для активної навчальної діяльності на уроках алгебри й геометрії в ліцеї. Тим, хто забув певний теоретичний матеріал, пропонується скористатися коротким довідником, що знаходиться у відповідному роз-

ділі. Багатьом учням настійливо рекомендується розібратися із зразками розв'язань певних типів задач.

У ході навчання ліцеїстам рекомендується проходити тематичні й розділові тести доти, поки рівень їхньої підготовки не стане достатньо високим. Оскільки переважна частина учнів проживає в гуртожитку і має доступ до комп'ютерного класу в позаурочний час, то серед домашніх завдань, які задають вчителі математики, є й самостійне проходження тестів з певної теми чи розділу. Учням наголошується, що кількість проходження таких тестів не обмежена. Приклад такого проходження тестування подано на рис.1.


<input type="checkbox"/>	 <b>Клименко Марина</b>	16 січень 2012, 07:18	16 січень 2012, 08:49	1 година 30 хв	<b>31</b>	-/1.63	1.63/1.63	1.22/1.63	0.81/1.63
<input type="checkbox"/>		17 січень 2012, 07:52	17 січень 2012, 09:07	1 година 14 хв	<b>36</b>	-/1.63	0/1.63	1.63/1.63	1.63/1.63

Рис.1

Відомо, що найкраще усвідомлюються ті теми, які учень не тільки самостійно опрацював, а й допоміг іншому їх засвоїти. Тому в курсі передбачено форум для обговорення проблемних завдань. Він адмініструється учителем математики. Якщо серед учнів не знайдеться того, хто допоміг би правильно розв'язати поставлену задачу, то учитель у форумі залишає підказку або й повне розв'язання задачі. Деякі гілки обговорення у форумі доводиться редагувати, бо учні починають перетворювати його на «стіну» соціальної мережі.

Автори свідомі того, що дані, які отримані в ході тестування, не дають змогу зробити остаточні висновки про причини виникнення проблем у навчальних досягненнях. Тому ЕНК з математики використовується не на кожному уроці, а лише у ході засвоєння раніше здобутих знань, умінь та навичок. Хоча усвідомлення багатьма учням невідворотності тестування з кожної теми, що вивчається, стимулює їх до кращої навчальної діяльності.

Особливістю ЗНО є те, що його учасникам потрібно виконати значну кількість завдань за досить обмежений проміжок часу. Часто трапляються випадки, коли учні без навичок розв'язання задач у тестовій формі, не можуть показати реальні свої знання. Саме тому перед вихідними рекомендуємо пройти пробні тестування, які містять 32 задачі й мають структуру тесту ЗНО.

Як показав досвід використання ЕНК, учні починають об'єктивніше відноситись до рівня своїх знань, умінь та навичок. Хоча на початковому етапі багато з них стверджує, що помилки, допущені під час тестування, були зумовлені лише неухважністю. Вони більш уважно ставляться до прочитання умов завдань, виконання тестів, звикають до форми подання

задач. Загалом це повинно сприяти кращому виконанню тесту ЗНО з математики.

### **Література**

1. Бевз Г.П. Геометрія: 11 кл.: підруч. для загальноосвіт. навч. закл.: академ. рівень, проф. рівень / Г.П. Бевз, В.Г. Бевз, Н.Г. Владімірова, В.М. Владіміров. – К.: Генеза, 2011. – 336 с.
2. Думанський Н.О. Відкриті системи дистанційного навчання / Н.О.Думанський // Вісник Національного університету „Львівська політехніка”. Інформаційні системи та мережі. – 2011. - №699. – С. 94-103.
3. Жалдак М. І. Педагогічний потенціал комп’ютерно-орієнтованих систем навчання математики / М.І. Жалдак // Комп’ютерно-орієнтовані системи навчання: зб. наук. праць. – К.: НПУ імені М. П. Драгоманова. – 2003. – Випуск 7. – С. 3-16.
4. Захарійченко Ю.О. Повний курс математики в тестах / Ю.О.Захарійченко, О.В.Школьнік, Л.І.Захарійченко, О.В.Школьнік – Х.: Ранок, 2011. – 496 с.
5. Зовнішнє незалежне оцінювання навчальних досягнень випускників загальноосвітніх навчальних закладів. 2008 р.: Інформаційні матеріали / Український центр оцінювання якості освіти: Уклад.: І.Л. Лікарчук (наук. ред.) та ін. – К., 2007. – 288 с.
6. Мерзляк А.Г. Алгебра і початки аналізу: підруч. для 10 кл. загальноосвіт. навч. закладів: проф. рівень / А.Г. Мерзляк, Д.А. Номіровський, В.Б. Полонський, М.С. Якір. – Х. : Гімназія, 2010. – 416 с.
7. Мерзляк А.Г. Алгебра. 11 клас: підруч. для загальноосвіт. навчальн. закладів: академ. рівень, проф. рівень / А.Г. Мерзляк, Д.А. Номіровський, В.Б. Полонський, М.С. Якір. – Х. : Гімназія, 2011. – 431 с.
8. Морзе Н.В. Критерії якості електронних навчальних курсів, розроблених на базі платформ дистанційного навчання / Н.В.Морзе, О.Г.Глазунова // Інформаційні технології в освіті. – Херсон: Херсонський державний університет. – 2009. – Випуск 4. – С. 63 – 75.
9. Нелін Є.П.Алгебра і початки аналізу: підр. для 10 кл. загально-освіт. навч. закладів: профільн. рівень / Є.П.Нелін – Х.: Гімназія, 2010. – 416 с.
10. Нелін Є.П. Алгебра. 11 клас: підр. для загальноосвіт. навч. закладів: академ. рівень, проф. рівень / Є.П.Нелін, О.Є.Долгова – Х.: Гімназія, 2011. – 448 с.
11. Нелін Є.П. Математика. Експрес-підготовка. ЗНО-2012 / Є.П.Нелін – К.: Літера ЛТД, 2011. – 240 с.
12. Зайцева І.П. Офіційний звіт про проведення зовнішнього незалежного оцінювання навчальних досягнень випускників загальноосвітніх навчальних закладів у 2011 році. / І.П.Зайцева – К.: Український центр оцінювання якості знань, 2011. – 359 с.



13. Положення про атестацію електронного навчального курсу на рівні ВНЗ та МОН України.  
[www.mon.gov.ua/images/gr/obg/2010/08\\_06\\_10.pdf](http://www.mon.gov.ua/images/gr/obg/2010/08_06_10.pdf)
14. Програма зовнішнього незалежного оцінювання з математики  
<http://testportal.gov.ua/index.php/text/math/>
15. Хара О.М. Мотивація навчальної діяльності в дистанційному курсі з математики. / О.М.Хара // *Didactics of mathematics: Problems and Investigations*. – Issue # 32. – 2009. – С. 77-81.
16. Чашечникова О.С. Математична грамотність як одна зі складових інтелектуальної компетентності учнів./ О.С. Чашечникова, І.М. Москаленко, Л.О. Калюсенко // *Педагогічні науки: теорія, історія, інноваційні технології*. – Суми: Сумський державний університет, 2009. – №2. – С. 209 – 216.

### ***Побудова ВЕБ-інтерфейсу для XML-даних в мобільних пристроях***

*Павлишенко Б.М., [Квач З.А.](mailto:zaxari4@gmail.com)*

*ЛНУ імені Івана Франка, [zaxari4@gmail.com](mailto:zaxari4@gmail.com)*

Сучасне інформаційне суспільство, за твердженням виконавчого директора компанії Apple Т. Кука, вступає в еру пост-персональних комп'ютерів. Суттєву роль в обміні інформацією починають відігравати мобільні пристрої, зокрема смартфони та планшетні комп'ютери. Це зумовлює доцільність створення програмних систем клієнт-серверної взаємодії для мобільних пристроїв.

З існуючих технологій обміну даними можна виділити веб-технології, а одним із універсальних форматів даних можна розглядати XML. Об'єднання веб-технологій та XML формату даних є перспективним напрямом побудови комунікаційних систем для мобільних пристроїв. Найбільш поширеними операційними системами для них є Android та IOS . В цій роботі розглядається побудова веб-інтерфейсу для XML-даних у пристроях під управлінням ОС Android, яка в свою чергу побудована на основі ОС Linux. Для досягнення поставленої мети було зроблено наступне:

- встановлено необхідні засоби розробки (Eclipse IDE, Android SDK, ADT Plugin для Eclipse);
- розроблено структуру класів;
- протестовано додаток шляхом розбору RSS-стрічок, що містяться на веб-сайтах.

На основі проведеної апробації розробленої системи показано ефективність запропонованого підходу розробки веб-інтерфейсів для мобільних пристроїв на основі ОС Android, з використанням формату даних XML.



## ***FREE/LIBRE software usage in the Belarusian system of higher educational institutions***

*Derechennk S.S., Kostiuk D.A., Pynkin D.A.*

*Brest State Technical Institute, [cm@bstu.by](mailto:cm@bstu.by)  
Brest State Technical Institute, [dmitriykostiuk@gmail.com](mailto:dmitriykostiuk@gmail.com)  
Belarusian State University of Informatics & Radioelectronics,  
[denis.pynkin@gmail.com](mailto:denis.pynkin@gmail.com)*

The particular features of free/libre software usage in the academic activity of Belarusian higher institutions are reviewed. The analysis of supporting software for the educational process is undertaken as far as of one studied in practical courses. Changes in free/libre software share in the academic activity are discussed in light of the graduates' up-to-date employment situation.

The scope of free/libre/open source software (FLOSS) in higher educational institutions includes three categories:

- 1. Software supporting the academic process (mainly system software on servers and workstations),*
- 2. Additional application software, particularly one used by students at preparing and formatting their term papers,*
- 3. Software studied in academic and laboratory courses.*

Labour market demands studying a lot of proprietary software products ranging from Windows platform to specialized CAD/CAM systems. This tendency is accompanied with weak motivation of FLOSS usage, caused by the fact that license violation risk is still in the area of assumptions, with no real actions of the inspection bodies in Belarus. Legally purchased software share was increasing in last years due to special discounts from vendors and also because of high economic indices before the year 2011. Nevertheless, economic factor is mainly not the determining one for making choice between a free/libre software and a proprietary one. As a result, FLOSS usage in the universities is usually reasoned by its technical advantages over proprietary analogues, or by the labor market demands. Server software choice may be the sole exception, as it is substantially influenced by the administrators' personal tastes.

FLOSS of the second and third categories can obviously be seen more often in the engineering higher institutions, especially ones training the IT-specialists. Server software is less influenced by the university specialization [1].

### *1. Software supporting the academic process*

In most cases free/libre system software on workstations is presented by a GNU/Linux in the multiboot mode, which plays the role of an alternative OS in computer classrooms of departments, teaching programming to students with engineering specialization. In pedagogical universities Linux desktops are rarely seen due to their low demand in Belarussian schools [2], but successful usage of Linux-based thin clients with a terminal Windows server was also

noticed in several universities, e.g. in Yanka Kupala State University of Grodno (GRSU).

In a period of 2007–2011 Ubuntu distribution had leadership in the segment of Linux desktops in the universities, because of its broad availability and good quality. But starting from the year 2011 and the withdrawal of the desktop metaphor in Ubuntu, system administrators are searching alternative distribution for the regular systematic upgrade of computer classrooms. Nevertheless there is no general approach among administrators, and choice is a matter of personal taste to some extent. Beside Ubuntu, ALT Linux, Fedora and Mandriva are also predictably popular on workstations, while latest Ubuntu versions are seen mostly on the students' notebooks, as far as Linux Mint was until the most recent times.

Broader variety of distributions is typical to the server segment. Leadership obviously belongs to Debian, and from the aforementioned desktop list only the choice of Mandriva is not typical on servers. But as server distribution choice has not much influence on the end users, so administrators have more freedom to follow their personal likes. Therefore Gentoo or some binary distribution with manually recompiled packages can be often seen on university servers. Modified versions of standard software packages are also not a rare case, with more functions and/or non-typical configurations added.

GNU/Linux platform is traditionally used on servers to provide firewalls, mail and web-hosting, personalized Internet access, networked users authentication and file-servers. BSD systems are used rarely. The Internet access for students and teachers is often organized with Squid, but billing systems based on RADIUS-servers (particularly FreeRADIUS) also can be found. Most typical free authentication server is a Samba-based domain controller, while more complex solutions involve OpenLDAP with Kerberos or without it, and/or the VPN-segment. ClamAV is used for antivirus protection of Linux-based mail and file servers with different levels of its integration. FLOSS server databases are mainly represented by MySQL and PostgreSQL (used in web-sites, logs storage, and in other system components more often than in education process). Distributed solutions based on DRBD are also used.

The distribution choice and the complexity of a sever configuration is not strongly correlated to the university profile. For example, Maxim Tank Belarusian State Pedagogical University (BSPU) has complex sever Linux segment, which is totally built on the source-based approach.

Belarussian higher educational institutions usually have heterogeneous server segment. Windows systems are needed to run server versions of studied software (Microsoft SQL Server, 1C:Enterprise system of programs, etc.) and to use hardware keys of licensed applications. Old Novell NetWare servers still work in some places also. Beside that, computer department often has its own independent segment in the local network, with different set of software and services [1].

A tendency can be seen in some universities to place Windows and legacy Novell servers into the virtualized environments, working atop of GNU/Linux — as to provide the hardware abstraction, so to improve the system safety and to simplify its failure recovery [1, 3]. VirtualBox and KVM are the most used FLOSS virtualization systems. Desktop virtualization in computer classrooms is also used in some computer departments: successful VirtualBox usage in Computers dept. of Belarusian State University of Informatics and Radioelectronics (BSUIR) and highly-integrated approach to VirtualBox-based isolation of Windows desktops in Computers & Systems dept. of Brest State Technical University (BrSTU) should be mentioned here. These approaches improve speed and stability of virtualized Windows in comparison with its functioning on bare hardware; students are allowed to work with administrator privileges if they need so, and computer classroom administrators have an instrument of quick rollback of the guest system to its previous state snapshots (or stateless configuration can be used with automatic rollback after each reboot) [3, 4].

## *2. Additional application software*

Complimentary software proposed to students includes the standard set of the OpenOffice.org office suite & the Firefox web browser. Firefox is widely spread, while OpenOffice/LibreOffice is more often installed on teachers' workstations than on students ones in non-IT departments — supposedly due to its more conservative interface approach and the ease of PDF export. IT departments often have both proprietary and free office suites installed in classrooms. Some technical universities also use LaTeX document preparation system (mainly those having developed scientific school). Moodle and iTest projects are used for examination.

In non-IT engineering specializations other FLOSS is mostly used for mathematical calculations. Perhaps SciLab is most popular because of its ease of installation, small size, good set of features and Matlab compatibility [5]. Among non-engineering universities computer departments of medical institutions have most noticeable interest to mathematical FLOSS [6] due to essential possibility of the analysis of used algorithms and their correctness, which improves the reliability of medical researches.

Free CAD/CAM systems (mainly QCAD/LibreCAD, and Blender, which also can be related to this area) are used occasionally and mainly for non-core specialization students, while core engineering subjects are more strictly bounded to professional proprietary products. As an exception, some universities of the radioelectronics direction use GNU Electric as an integrated circuit CAD system.

There is a special case of building and studying platform for the GRID systems and clusters, where GNU/Linux has no alternatives. FLOSS is used as key components in the national GRID system of Belarus, which incorporated resources of the leading universities (Belarussian State University, GRSU, BSUIR, Belarusian National Technical University among them), scientific institutions and enterprises of the country under the joint Russian & Belarusian

SKIF-GRID program [7, 8]: OpenMPI, Torque, Unicore, GLite, etc. are expanded on the classical cluster stack based on GNU/Linux systems.

### 3. Studied software

Theory and practice of the GNU/Linux command line basics are widely studied in IT-oriented specializations. Until the recent time, system programming for this platform was taught in several departments only — e.g. set of subjects related to low-level programming and computer architecture in BrSTU, or network programming in BSUIR; in both cases Linux was used as the basis due to such its advantages as no unnatural limitations from the owner, simplicity of development, simple documentation, compliance with standards and wide spread of the platform.

In last years there is a growth of corporate employers' interest to the Linux development, mainly for embedded and server systems. In a related move the computer departments are considering courses of system programming and Linux architecture, and large players of the market are actively announcing their own trainings and seminars dedicated to the same subject.

Among hardware-independent FLOSS platforms Java and PHP have leading positions in the academic load. The last one can be met not only in engineering universities, but also is included into web programming courses in pedagogical ones. From the other side, many perspective programming languages, such as Python and Ruby, are not studied for limited number of class hours and base curriculum bounded to classic C/C++ language. As additional measures to amend this issue, individual study of such FLOSS platforms is stimulated for the term and diploma projects, and in the student research contests.

Authors are grateful to Denis Koganovich from BSPU and Andrew Shadoura from BSUIR for their useful comments.

### References

1. Пойта П.С. *et al.* Обеспечение информационной безопасности при организации учебного процесса в рамках локальной сети университета // *Комплексная защита информации: матер. XVI научно-практич. конф., Гродно, 17-20 мая 2011 г. Гродно, 2011.* – pp. 237-240
2. Soroko V. Особенности внедрения свободного программного обеспечения в Республике Беларусь // «Linux Vacation / Eastern Europe 2009» 5th international conference. <http://lvee.org/ru/articles/157>
3. Костюк Д.А., Приступчик А.М. Особенности прозрачной виртуализации небезопасных и уязвимых систем для упрощённого администрирования учебных классов // *Шестая конференция «Свободное программное обеспечение в высшей школе»: Тезисы докладов / Переславль, 29–30 января 2011 года. М.: Альт Линукс, 2011.* – pp. 66-68.
4. Пынькин Д.А., Глецевич И.И. Проектирование IT-инфраструктуры учебных заведений на базе ОС Linux // 5-я конференция «Свободное



- программное обеспечение в высшей школе»: Тезисы докладов / Переславль, 30-31 января 2010 г. М.: Альт Линукс, 2010. – pp. 41-44.
5. Журавлев В., Котяшев Е. Использование математической программы scilab в учебном процессе в БПУ и МГУ им. А.А. Кулешова // «Linux Vacation / Eastern Europe 2007» 3rd international conference. <http://lvee.org/ru/articles/77>
6. Chubukov J. Опыт применения приложений с открытым исходным кодом в медицинских научных исследованиях. // «Linux Vacation / Eastern Europe 2009» 5th international conference. <http://lvee.org/ru/articles/140>
7. Пынькин Д.А. Создание специализированного дистрибутива Linux для подключения к национальной GRID-сети // «Linux Vacation / Eastern Europe 2011» 7th international conference. <http://lvee.org/en/articles/242>
8. Пынькин Д.А. Изоляция GRID-задач с помощью LXC // «Linux Vacation / Eastern Europe 2010» 6th international conference. <http://lvee.org/en/articles/207>
9. Костюк Д.А., Жук А.М. Построение практикумов по программированию периферийных устройств и архитектуре ЭВМ на базе GNU/Linux // Тези міжнародної НПК FOSS LVIV–2011, Львів, 1–6 лютого 2011р. – Львів: Вид. ЛНУ ім. І. Франка, 2011. – pp. 73-75.

**Використання відкритого програмного забезпечення для організації дистанційного навчання в технічному університеті**  
Кривенко О.В., Чичкар'єв Є.А.

Державний вищий навчальний заклад  
«Приазовський державний технічний університет»  
[KrivenkoOV@gmail.com](mailto:KrivenkoOV@gmail.com)

We consider the use of open source software Moodle for distant learning at the Technical University. The experience of teachers of public higher education institution "Priazovsky State Technical University" online-training courses in the system to Moodle.

На даний час все більшою популярністю користується навчання за допомогою Internet технологій. Такі технології існують в OpenSource системах LMS \ LCMS, таких як ATutor, Claroline, Dokeos, LAMS, Moodle, OLAT, OpenACS, Sakai. До одних з основних достоїнств OpenSource систем відносять можливість доопрацювання та адаптації конкретної системи до своїх потреб і поточної освітньої ситуації; відсутність прив'язки до конкретної операційної системи, конкретному Web-браузеру; реалізація набору функціональних можливостей з істотно меншими економічними витратами в порівнянні з комерційними.

Найбільш популярною серед OpenSource систем є Moodle.

Moodle – це система управління вмістом сайту (Content Management System - CMS), спеціально розроблена для створення якісних online-курсів викладачами.

Відкрите програмне забезпечення (ПЗ) Moodle в ДВНЗ «ПДТУ» використовується вже більше п'яти років.

Дистанційну освіту (ДО) Moodle в ДВНЗ «ПДТУ» застосовують для навчання студентів дистанційної, заочної, прискореної та денної форм навчання, курсів підвищення кваліфікації, системи перепідготовки кадрів та студентів філій та представництв.

Отримуючи доступ до дистанційного курсу, студенти мають можливість поглиблено вивчити теоретичну частину матеріалу, виконати додаткові завдання і вправи, а так само пройти пробне тестування.

Електронні навчальні матеріали для очної форми використовуються переважно як допоміжний навчальний матеріал, тестування, віртуальних лабораторних робіт та електронного довідника. Найбільш часто застосовують електронні навчальні матеріали при самостійній роботі студентів над навчальними посібниками, електронними підручниками, при курсовому і дипломному проектуванні, тестуванні при самоконтролі.

В ДВНЗ «ПДТУ» створено 586 дистанційних курсів (ДК), що містять ресурси (веб-сторінки) лекційного та методичного матеріалів, тести.

У відділі дистанційного навчання (ДН) ДВНЗ «ПДТУ» навчаються студенти за заочно- і прискорено-дистанційною формами навчання за чотирма напрямками: УКП № 1 Волноваха, УКП № 2 Комсомольське, УКП № 3 Бердянськ і УКП № 4 Новоазовськ.

Система ДН ГВНЗ «ПДТУ» налічує 1849 користувачів, серед яких понад 160 викладачів, які працюють над створенням дистанційних курсів.

Сайт ДН використовується для тестування студентів всіх форм навчання. Зареєстровано більше 50 груп студентів.

ДН дозволяє студентам різних рівнів підготовки з курсів, що викладаються, самостійно виконувати деякі завдання і вправи з метою ліквідації прогалин освіти. В рамках курсу є можливість відпрацювань пропущеного студентом лекційного матеріалу, лабораторних занять та контрольного тестування за певним заліковим модулем.

Проведено навчання науково-педагогічного складу ДВНЗ «ПДТУ» на курсах по темі «Розробка ДК в MOODLE». Всього пройшли навчання понад 100 викладачів, у тому числі з видачею посвідчень про підвищення кваліфікації. Навчання проводиться постійно по мірі набору груп.

В ДВНЗ «ПДТУ» розроблений необхідний комплект нормативних документів, який забезпечує реалізацію навчального процесу з використанням технологій ДН. Всі ці документи затверджені методичною радою ДВНЗ «ПДТУ», наказами ректора та розміщені на сайтах системи ДН ДВНЗ «ПДТУ».

Одним з варіантів вирішення проблеми навчання студентів за індивідуальним робочим планам з різних регіонів України, без створення



Локальних центрів ДН, є розміщення в кожному дистанційному курсі настановної відеолекції, застосування елементів флеш-анімації, презентацій, віртуальних лабораторних практикумів.

Сформовано по всіх спеціальностях ДН база даних Кейс-пакетів, які видають своїм студентам.

Таким чином, в ДВНЗ «ПДТУ»:

- створено структурний підрозділ, який організаційно і технологічно забезпечує розвиток системи ДН в ДВНЗ «ПДТУ»;
- затверджені Вченою радою університету понад 20 спеціальностей (технічного та економічного напрямків) для проведення педагогічного експерименту з ДН;
- введено в дію виділений сервер системи ДН ДВНЗ «ПДТУ», на якому розміщені 3 веб-сайту ДН (сайти адмініструють і обслуговують фахівці відділу Дистанційного навчання та екстернату);
- проведено навчання науково-педагогічного складу ПДТУ на курсах по темі «Розробка ДК в MOODLE»;
- створена лабораторія ДН, в якій оснащено 14 робочих місць, з підключенням їх до локальної мережі ДВНЗ «ПДТУ» та Інтернет. Всі комп'ютери ДВНЗ «ПДТУ» (близько 1000 шт.), задіяні в навчальному процесі, підключені до локальної мережі ДВНЗ «ПДТУ» і мають вихід в Інтернет;
- наказами ректора за кожною кафедрою затверджено персональний склад тьюторів та осіб, відповідальних за ведення сторінок сайту «Каталог навчально-методичних матеріалів»;
- створені 3 Локальних центру ДН в регіонах Донецької і Запорізької областей, з якими підтримується цілодобовий комунікаційний зв'язок;
- використання вільного ПЗ Moodle при навчанні студентів дозволило не тільки підвищити якість їх навчання, але і створити міцну основу їх подальшої постійної самоосвіти.

**Вільні ліцензії у законодавчій неволі**  
Бічук А.А., Наумко М.В.

Creative Commons Україна, [maksym.naumko@intelactive.com.ua](mailto:maksym.naumko@intelactive.com.ua)

The report is on the feasibility of legal use of objects if intellectual property in Ukraine, available under so called free public licences.

На жаль, сьогодні юридична доктрина і законодавство не встигають дослідити та врегулювати суспільні відносини з приводу використання об'єктів права інтелектуальної власності, що виникають в умовах безупинного руху технологій. Одним з найяскравіших прикладів є законодавча нерегульованість та тотальне незрозуміння проблематики використання об'єктів «відкритої інтелектуальної власності».

У Рекомендаціях щодо забезпечення правомірності використання комп'ютерних програм вільного користування Державна служба визначає **вільну публічну ліцензію** як загальнодоступний договір приєднання, що надає особі, яка приєдналася до такого договору, безоплатний дозвіл на таке: використання комп'ютерної програми з будь-якою метою; доступ до вихідного коду; будь-які дослідження механізмів функціонування програми; використання механізмів (принципів) функціонування будь-яких довільних частин коду програми для створення інших програм та (або) адаптації до потреб користувача; відтворення комп'ютерної програми й розповсюдження її примірників будь-яким способом та в будь-якій формі; внесення змін і вільне розповсюдження як оригінальної комп'ютерної програми, так і зміненої, на тих самих умовах, під які підпадає оригінальна комп'ютерної програма, якщо інше не передбачено ліцензією;

На жаль в Україні на даний момент використання об'єктів права інтелектуальної власності, що розповсюджуються на умовах вільних публічних ліцензій вкрай ускладнене. Головним чином така ситуація обумовлена неготовністю українського законодавства сприймати електронні тексти ліцензій та посилання на такі тексти на веб-сторінках, де автори розміщують свої твори, чи навіть електронні тексти ліцензій, включені до дистрибутивів програмного забезпечення, як підтвердження факту надання правовласником користувачам певного обсягу прав.

Частина 1 статті 32 Закону України „Про авторське право і суміжні права” (далі – Закон) встановлює, що автору та іншій особі, яка має авторське право, належить виключне право надавати іншим особам дозвіл на використання твору будь-яким одним або всіма відомими способами на підставі авторського договору. Використання твору будь-якою особою допускається виключно на основі авторського договору, за винятком випадків, передбачених статтями 21 – 25 Закону.

Згідно з частиною першою статті 639 Цивільного кодексу України (надалі - ЦК), договір може бути укладений у будь-якій формі, якщо

вимоги щодо форми договору не встановлені законом. Стаття 33 Закону встановлює вимоги до договорів на право використання творів. Зокрема, ця стаття містить вимогу про те, що договори про передачу прав на використання творів укладаються в письмовій формі. Частина 2 статті 1107 Кодексу передбачає, що договір щодо розпорядження майновими правами інтелектуальної власності укладається в письмовій формі. Хоча ч.1 ст. 207 ЦК і визнає укладеним у письмовій формі правочин, якщо воля сторін виражена за допомогою електронного зв'язку, ст. 6 Закону України «Про електронні документи та електронний документообіг» визначає обов'язковим реквізитом електронного документу електронний підпис. Електронні підписи є лише у дуже невеликій кількості суб'єктів права інтелектуальної власності, не кажучи вже про звичайних користувачів.

В контексті чинного законодавства постає ще й питання про правомірність розповсюдження дистрибутивів вільного програмного забезпечення на лазерних дисках.

Відповідно до статті 3 Закону України „Про розповсюдження примірників аудіовізуальних творів, фонограм, відеограм, комп'ютерних програм, баз даних” (далі – Закон про розповсюдження) розповсюдження на території України примірників, зокрема, комп'ютерних програм, а також їх прокат дозволяються лише за умови їх маркування контрольними марками.

Контрольні марки видаються імпортерам, експортерам та відтворювачам (далі – Заявники) примірників комп'ютерних програм на підставі документів, передбачених частиною першою статті 5 Закону про розповсюдження. Зокрема, відповідно до пункту „е” частини першої згаданої статті заявники надають копію договору про передачу (відчуження) майнових прав авторів або договори про передачу прав на використання комп'ютерних програм шляхом їх відтворення та розповсюдження примірників, починаючи з договору, за яким майнові права авторів або права на використання передаються на територію України, або починаючи з договору, за яким майнові права авторів або права на використання передаються від первинних суб'єктів авторського права (у разі виникнення цих прав на території України).

Звичайно, можливість надати копії письмових договорів, м'яко кажучи, є далеко не завжди, що стає ще однією перешкодою на шляху використання вільного ПЗ.

Як результат описаних вище законодавчих, на наш погляд, прогалин, абсолютно незрозумілим є на якій правовій підставі український користувач може використовувати, наприклад, операційну систему Linux або матеріали Вікіпедії.

В Україні неодноразово робилися спроби створити законодавчу базу для забезпечення можливості використання відкритих публічних ліцензій. Так, 15.08.2002 р. до Верховної Ради було внесено Проект Закону «Про використання Відкритого (Вільного) програмного забезпечення в

державних установах та державному секторі господарства». 01.12.2005 на заміну йому було подано Проект Закону «Про використання Відкритих і Вільних форм інтелектуальної власності, Відкритих форматів даних та Відкритого (Вільного) програмного забезпечення в державних установах і державному секторі економіки». Ці законопроекти мали на меті ввести в законодавче поле України ряд понять та термінів, без яких сучасне законодавство у галузі авторського права в цілому, та у сфері виробництва й використання програмного забезпечення зокрема, не може бути ані цілісним, ані дієвим (що яскраво ілюструє наявна в Україні ситуація з «боротьбою» проти нелегального використання програмного забезпечення та специфічних для нього носіїв).

Друга спроба мала місце влітку 2009 року, коли до Верховної Ради було подано Проект Закону «Про використання відкритого програмного забезпечення в органах державної влади та органах місцевого самоврядування». Законопроект мав ввести відповідну світовій практиці термінологію різних об'єктів та суб'єктів інформаційної сфери та визначити пріоритетні напрямки інформаційної політики України щодо використання відкритих та пропрієтарних комп'ютерних програм в органах державної влади, місцевого самоврядування та державних закладах освіти, а також врегулювати використання в органах державного управління програмного забезпечення і забезпечити запобігання утворенню монополій при постачанні програмних продуктів та комплексних інформаційних рішень. В лютому 2010 р. законопроект було відхилено та знято з розгляду.

Позитивним зрушенням є прийняття Постанови КМУ «Про затвердження Державної цільової науково-технічної програми використання в органах державної влади програмного забезпечення з відкритим кодом на 2012-2015 роки» від «30» листопада 2011 р.

Також, наразі на розгляді у Верховній Раді знаходиться законопроект № 6523 Про внесення змін до деяких законодавчих актів України щодо врегулювання питань авторського права і суміжних прав, який минулого року пройшов перше читання. Серед ряду пропозицій, що викликали ряд неоднозначних реакцій у суспільстві, даний проект містить пропозицію у статтю 33 Закону внести доповнення такого змісту:

*„Договір на використання комп'ютерної програми може бути укладений як шляхом підписання його обома сторонами, так і шляхом приєднання другої сторони до запропонованого договору в цілому. Умови таких договорів можуть бути викладені в письмовій, у тому числі електронній формі, у примірниках комп'ютерних програм чи додаватись до них як окремий документ. Приєднання може здійснюватись шляхом виконання особою, яка приєднується до договору, певних дій, що виражають її волю до такого приєднання.“*

Залишається сподіватися, що цьому законопроекту пощастить більше, ніж попереднім.

**Інформаційні технології в підготовці магістрів фізико-математичних спеціальностей педагогічного університету**  
Лазурчак І.І., Кобильник Т.П., Грозовський В.В.

*Дрогобицький державний педагогічний університет імені Івана Франка*  
[taras2408@mail.ru](mailto:taras2408@mail.ru)

The benefits are indicated, that are gained by magistrants of physics-mathematics specialties with the study of Computer Algebra Systems (CAS) and typesetting system LaTeX. Such composition let us to prepare high-quality electronic documents with examples of mathematical calculations, data visualisation, everything is done within one system.

Актуальність вивчення курсу «Інформаційні технології в освіті та науці» магістрантами педагогічного університету фізико-математичних спеціальностей незалежно від їхньої майбутньої професійної діяльності (педагогічної, наукової, прикладної чи практичної) обумовлена:

- потребами і необхідністю підвищення рівня інформаційної культури магістрантів педагогічних університетів;
- впровадженням та використанням комп'ютерної техніки з відповідним програмним забезпеченням практично у всі сфери діяльності людини (зокрема у навчальний процес загальноосвітніх шкіл і ВНЗ, у наукову, технічну, економічну діяльність, в заклади охорони здоров'я тощо).
- Основним методом навчання дисципліни «Інформаційні технології в освіті та науці» для магістрантів обирається метод проектів. Це обумовлюється кількома факторами, зокрема можливостями:
- поєднати метод проектів з традиційною системою навчання без суттєвих (значних) організаційних перетворень;
- організувати планування процесу засвоєння знань магістрантами як впродовж тривалого відрізка часу, так і оперативно, при поточному контролі;
- своєчасно визначати прогалини у знаннях магістрантів.

Крім того, метод проектів є дослідницьким методом, використання якого дає можливість сформувати у студентів досвід творчої діяльності. Робота над проектом виробляє стійкі інтереси, постійну потребу до творчих пошуків, оскільки без діяльності інтереси та потреби не виникають.

Для фізико-математичних спеціальностей актуальним є освоєння спеціалізованого програмного забезпечення для проведення розрахунків (вибрана система комп'ютерної математики) та оформлення результатів дослідження (видавнична система LaTeX).

Пропонується за основу навчання вибрати СКМ Махіта. Вибір системи Махіта за основу навчання обумовлений тим, що:

- система Махіта є вільно поширюваною;

- система надзвичайно легка для опанування (Mathiа оснащена системою меню, є україномовний інтерфейс);

СКМ Mathiа є однією з кращих для виконання символічних перетворень. Це по суті єдина система, яка може конкурувати з комерційними математичними пакетами Mathematica і Maple в галузі символічних обчислень.

Цікавим є такий напрям інтеграції як поєднання видавничої системи LaTeX з математичними пакетами, що дозволяє готувати електронні документи високої якості з прикладами математичних розрахунків, візуалізації даних, виконаних у певній СКМ.

LaTeX – це комп'ютерна видавнича система. Основне її призначення – підготовка наукових документів (тези доповідей, статті, посібники, автореферати, дисертації тощо). Необхідність використання системи LaTeX обумовлена тим, що більшість наукових видань (як правило, з фізико-математичних і технічних наук) приймають тексти в tex-форматі.

Назвемо деякі переваги системи LaTeX:

- зручні засоби відтворення алфавітного покажчика, списку використаних джерел, графічних об'єктів і таблиць, автоматична нумерація математичних формул, посилань та інших подібних об'єктів поряд з ефективним механізмом перехресного цитування;
- безкоштовне розповсюдження;
- можливість набору вхідного файлу у будь-якому текстовому редакторі (на відміну від файлу, набраного в редакторі MS Word, tex-файл займає значно менший обсяг);
- автоматичне форматування тексту (щільність символів, розміщення таблиць і графіків, граматично правильна вставка знаків переносу тощо) та математичних формул (розмір вкладених дужок, знаків інтегралу та інших спеціальних значків);
- автоматичне збереження документа після компіляції → перегляду останнього відкоректованого варіанту;
- розвиток абстрактного мислення студентів, набуття навичок програмування.

До недоліків LaTeX можна віднести те, що вона не є системою типу WYSIWYG: створення tex-документа і перегляд підготовленого до друку вихідного документа, є різними операціями.

Система LaTeX в усьому світі широко використовується науковцями, а особливо фахівцями з фундаментальних наук. Усі професійні математичні журнали у світі приймають статті до публікації виключно у tex-форматі. LaTeX зручно використовувати під час підготовки, наприклад, книжки або звіту на кілька сотень сторінок з великою кількістю математичних формул, рисунків, таблиць, з безліччю перехресних посилань і великим списком використаних джерел.

Протягом навчання курсу «Інформаційні технології в освіті та науці» магістранти виконують проект, кінцевим результатом якого є звіт про виконання індивідуального завдання, тематика якого пов'язана з



використанням СКМ Maxima до розв'язування задач з певного розділу математики чи фізики. Тема проекту вибирається магістрантом за погодженням з викладачем. У кінці навчання курсу студент презентує свою роботу. Звіт і презентація оформляється засобами системи LaTeX.

Таке поєднання СКМ Maxima та видавничої системи LaTeX обумовлене кількома факторами. По-перше, використання СКМ значно розширює межі застосування математичних методів і моделей для дослідження процесів у різних сферах людської діяльності. Широкий набір засобів для комп'ютерної підтримки аналітичних, обчислювальних і графічних операцій роблять сучасні СКМ одними з основних засобів у професійній діяльності вчителя-новатора, фізика-теоретика і дослідника, математика-аналітика, програміста, інженера, економіста-кібернетика та ін. Тому їх освоєння і використання у навчальному процесі педагогічного університету магістрантами фізико-математичних спеціальностей надасть можливість підвищити рівень їх професійної підготовки. По-друге, як правило, кінцевим результатом дослідження є певна публікація (презентації, тези доповідей, статті, науково-методична література тощо). У цьому випадку у пригоді стає система LaTeX, що у поєднанні з СКМ надає можливість створювати високоякісні продукти (як електронні, так і друковані-).

### ***Scientific inquiry based education and FOSS***

*Levitichi A., Duca M.*

*University of the Academy of Sciences of Moldova, Republic of Moldova*  
[cbm\\_bioinformatica@unasm.asm.md](mailto:cbm_bioinformatica@unasm.asm.md)

FOSS represents an important component in promotion and implementation of scientific inquiry based education concept in frame of a modern University. There are presented some examples of application of FOSSes in organization of educational, research and management activity of the University of the Academy of Sciences of Moldova.

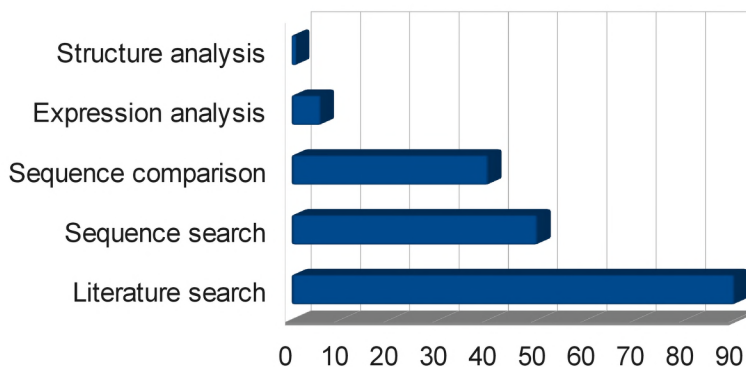
The University of the Academy of Sciences of Moldova (UnASM) is promoting scientific inquiry based education concept that means all of the students are involved in research activities and apply directly knowledge in science. In order to sustain this, Scientific Research Cluster UniverSCIENCE was created, comprising 18 research institutes, technological transfer organizations, University and High School.

Besides, there is University Center of Research, including three laboratories, one of which is Laboratory of Bioinformatics (<http://edu.asm.md/ro/node/78>). The Laboratory promotes Free and Open Source software exploitation in education and research process, by providing

access to the computer room with Linux Ubuntu installed on the machines. Unfortunately, there is a very low level of Unix systems usage, because most of the high dependency to Windows and Microsoft software. One of the basic features offered by Linux and appreciated by users is the wide range of open and free tools and software. It, also, offers possibility to save money for licensed OS and software acquisition in frame of the Center.

The Laboratory was designed for education, training and research. There are two basic courses assisted by the lab, Biostatistics for undergraduates and Bioinformatics for master students. The structure of the courses includes individual training component, that offers the possibility to teach students to work with LibreOffice [1] features, especially Text editor, Spreadsheet, Drawing and Presentation. Later, they apply these skills in theses elaboration, data analysis and research presentations during the defence.

During Bioinformatics courses students receive a wider image of the possibilities offered by open source software and free resources (especially, from NCBI and EBI portals). Thus, this course becomes one that learns the application of FOSS, with the following implementation of them in research. According to our survey, at least half of the students who passed Bioinformatics course occasionally apply free and open resources in their research, while about one forth in their day-by-day activity. Main resources and tools they use are literature databases and text mining tools.



*Figure 1. Level of application of Bioinformatic tools depending on the research purpose*

Moreover, we are implementing R language [2] application in bioinformatic analysis, with the link to Bioconductor [3].

Besides of training support, the Laboratory, in collaboration with the Information Society Development Institute, is maintaining e-learning UnASM MOODLE platform [4]. For the moment, it is filled with over 80 courses taught at the University, both basic and optional curricula. It contains all necessary materials and documentation, but also uses the advantages of linking to various open education resources. For the moment, it is in development

university registry system, where all data about students' successes and promotions are stored, with the aim to emphasize Moodle utility for statistical analysis of the academic situation of the students.

University is promoting its activity through a web page created on the open source content management system DRUPAL, which is continuously adjusted.

During years of study, students are involved in the activity of UnASM, contributing to the development of the e-learning platform and website, research or data analysis in frame of investigation projects, learning and exploiting FOSS. Later they apply their skills in frame of the UnivER SCIENCE Cluster. It represents one of the ways to sustain scientific inquiry based education concept.

### **References**

1. <http://www.libreoffice.org>
2. *Ihaka, R., Gentleman, R. (1996). R: A Language for Data Analysis and Graphics. Journal of Computational and Graphical Statistics, 5(3): 299–314.* /<http://www.r-project.org/>
3. *Gentleman, R. C.; Carey, V. J., Bates, D. M., Bolstad, B., Dettling, M., Dudoit, S., Ellis, B., Gautier, L. et al (2004). Bioconductor: Open software development for computational biology and bioinformatics. Genome Biology, 5(10): R80.* /[www.bioconductor.org](http://www.bioconductor.org)
4. [www.moodle.org](http://www.moodle.org)

### **Використання відкритого програмного забезпечення у науково-дослідних роботах і навчальному процесі кафедри комп'ютерних систем та мереж ТНТУ ім. І.Пулюя Луцків А.М.**

*Кафедра комп'ютерних систем та мереж, Тернопільський національний технічний університет імені Івана Пулюя, e-mail: l.andriy@gmail.com*

The peculiarities of open source, free and closed commercial software in research work and teaching process of the Computer Systems and Networks department of Ternopil Ivan Pul'uj National Technical University are analyzed. The basic constraints of open source software, and examples of successful implementation of open source software are submitted. The main advantages of using open source software for teaching specialists according the specialty computer systems and networks and in the scientific research projects are analyzed.

Автор доповіді працює на кафедрі комп'ютерних систем та мереж Тернопільського національного технічного університету імені Івана Пулюя [1], яка готує бакалаврів за напрямом 6.050102 — Комп'ютерна

інженерія, а також спеціалістів та магістрів спеціальності 8.05010201-Комп'ютерні системи та мережі, викладає курси «Паралельні та розподілені обчислення», «Системне програмне забезпечення», «Інформаційні системи паралельної та розподіленої обробки даних», «Програмування мовою JAVA», «Комплексне забезпечення інформаційної комп'ютерної безпеки». У ході викладання відповідних курсів, в основному, використовується відкрите програмне забезпечення — дистрибутив ALT Linux [2] та необхідні пакети програм, що входять у його репозиторій програмного забезпечення.

Варто зазначити, що Тернопільський національний технічний університет імені Івана Пулюя має академічні ліцензії на використання закритого комерційного програмного забезпечення, зокрема компаній Microsoft (операційні системи, утиліти, офісні пакети та засоби розробки-), Autodesk Autocad, АСКОН Компас-3D та цілої низки іншого. Таким чином студенти мають можливість використовувати програмне забезпечення різних виробників та з різними ліцензіями.

Використання на кафедрі дистрибутиву ALT Linux обрано на основі власного успішного досвіду автора (з 2002-го року). До переваг даного дистрибутиву варто віднести наступне:

- висока стабільність системи;
- хороша продуманість архітектурних особливостей системи та використання передових технологій у галузі відкритого програмного забезпечення;
- великий репозиторій пакетів;
- технічна підтримка та документованість (список розсилки та форуми);
- висока гнучкість дистрибутиву, яка забезпечила велику кількість похідних дистрибутивів на спільній пакетній базі (ALTSP, Skif, ALT LXDesktop та інші);
- можливість використання на застарілому апаратному забезпеченні;
- зручність у використанні як початківцями так і досвідченими користувачами UNIX-систем;
- урахування потреб україномовних користувачів.

Загалом, до переваг використання відкритого програмного забезпечення, зокрема ОС Linux, у навчальному процесі за напрямом 6.050102 — Комп'ютерна інженерія, варто віднести:

- наочність демонстрації роботи ключових компонент системного програмного забезпечення в аспекті “бачимо вихідний код — модифікуємо — компілюємо — бачимо результати роботи”;
- швидше опанування навичками мережевого та системного адміністрування комп'ютерних мереж та інфраструктур;
- краще розуміння принципів роботи апаратного забезпечення, зокрема спеціалізованого мережевого та вбудованого (embedded);
- знайомство з принципами розробки великих відкритих проектів.

Також можна спостерігати цікаву тенденцію — студентам, які активно використовують відкрите програмне забезпечення, а, як правило, це один із дистрибутивів ОС Linux, набагато простіше дається опанування профільних курсів, нових пакетів програмного забезпечення, технологій системного, прикладного та веб-програмування (незалежно від операційної системи, на яку вони орієнтовані). Досвід показує, що у таких студентів краще розвиваються здібності до самостійного навчання, а тому їх можна залучати до технічно складніших та наукомісткіших курсових та дипломних робіт й проектів. Ці студенти беруть участь у виконанні науково-дослідних робіт кафедри і в результаті їх технічний фаховий рівень забезпечує вищі конкурентні переваги на ринку праці.

В аспекті світової фінансової кризи, дефіциту бюджету спостерігається недофінансування бюджетних закладів освіти та брак коштів у приватних структур, тому можна прогнозувати підвищення інтересу до відкритого програмного забезпечення, а відповідно, й потреби на ринку праці у кваліфікованих фахівцях у даній сфері. Не можна не відзначити позитивні зрушення у цьому напрямку, зокрема постанову Кабінету міністрів України № 1269 від 30 листопада 2011-го року «Про затвердження Державної цільової науково-технічної програми використання в органах державної влади програмного забезпечення з відкритим кодом на 2012 - 2015 роки».

На кафедрі комп'ютерних систем та мереж Тернопільського національного технічного університету імені Івана Пулюя виконувалась і виконується на даний час ціла низка науково-дослідних робіт, а саме: ВК 21-06 «Математичне моделювання, методи обробки та імітації біометричних циклічних сигналів в інформаційних системах», ВК 34-11 «Моделювання та розробка алгоритмів криптоаналізу з використанням паралельних та розподілених комп'ютерних систем» та ДІ 188-12 «Розробка математичного та програмного забезпечення інформаційних систем діагностики та аутентифікації людини за циклічними біометричними сигналами».

У рамках наведених тем проводиться розробка програмного забезпечення, яке планується розповсюджувати під відкритими ліцензіями [3]. На основі власного досвіду автора варто зазначити, що використання відкритого програмного забезпечення при створенні науково-технічної продукції у рамках науково-дослідних робіт суттєво знижує «порог входження» у відповідну предметну область, оскільки, дає змогу науковцю створювати власні програмні продукти, як похідні від інших, з урахуванням усіх ліцензійних вимог. Іншою перевагою для дослідника є наявність великої кількості інструментальних засобів для проведення експериментальних досліджень..

На думку автора, основними стримуючими факторами використання відкритого програмного забезпечення в Україні є:

1. у цілому ряді випадків, *невисокий рівень технічних навичок науково-педагогічних кадрів та технічного персоналу, щоб забезпечити проведення лабораторних практикумів та науково-дослідних робіт;*



2. *неадекватне розуміння можливих юридичних наслідків використання нелегального програмного забезпечення.*

Причини цих факторів криються у низькій матеріальній та моральній мотивації відповідних кадрів.

Важливою проблемою є створення нового та розширення функціональних можливостей існуючого наукомісткого програмного забезпечення, яке є необхідним при виконанні науково-дослідних робіт у фундаментальних та прикладних галузях науки й техніки. Доцільно, щоб таке програмне забезпечення розроблялося у рамках державних програм за державним фінансуванням та публікувалось під відкритими ліцензіями.

Ще одна важлива задача — необхідність запровадження на державному рівні відкритих форматів документообігу при оформленні статей, звітів, дисертацій, посібників у сфері науки і техніки (ДСТУ 3008-95).

### **Література**

1. *Кафедра комп'ютерних систем та мереж [Електронний ресурс]. - Режим доступу: URL: <http://kaf-ks.tntu.edu.ua/> — Назва з екрану.*
2. *ALT Linux [Електронний ресурс]. - Режим доступу: URL: <http://www.altlinux.ru/> — Назва з екрану.*
3. *Бойко І.Ф. Відкрите програмне забезпечення для розробки інформаційних систем: порівняльний аналіз і перспективи розвитку в Україні / І.Ф. Бойко, С.А. Лупенко, А.М. Луцків // Міжнародний науково-технічний журнал "Комп'ютинг". -Тернопіль: "Економічна думка", 2005, - Т.4,№1 - С.99-104.*

## **Віртуальний лабораторний практикум на базі GEANT4 в середовищі ОС Linux для вивчення процесів взаємодії випромінювання з речовиною**

*Малихіна Т.В.*

*Харківський національний університет імені В.Н.Каразіна  
tmalykhina@univer.kharkov.ua*

The Virtual laboratory practicum environment is presented. It was developed at the V.N.Karazin Kharkov National University at the Faculty of Computer Science. This practicum is developed together with leading experts of the Faculty of Physics and Technology as well as the Faculty of Energy Physics. Virtual laboratory practicum is a complex of computer programs developed in the environment of the Linux SLC4.8 operating system using Geant4 toolkit that is open-source software. Virtual laboratory classes will be used to train students of the Department Faculty and Technology as well as the Faculty of Energy Physics.



В теперішній час набули подальшого розвитку програмно-апаратні комплекси, “Віртуальні лабораторії”, спрямовані на удосконалення методик дистанційної освіти. В той же час, деякі лабораторні роботи неможливо виконати, не маючи спеціального устаткування. До таких лабораторних робіт можливо віднести практикуми з ядерної фізики і радіаційних технологій, з фізики елементарних частинок тощо. Навіть за наявності унікального устаткування в учбових закладах існує проблема обмеженого доступу до такого обладнання, зумовлена або коштовністю проведення експерименту, або шкідливими чи небезпечними умовами цього експерименту.

В багатьох галузях людської діяльності використання комп’ютерних технологій дозволяє істотно оптимізувати виробничі процеси й покращити якість результату цієї діяльності. Одним з напрямків розвитку інформаційних технологій є розробка програмних засобів для моделювання різноманітних процесів, що дозволяє економити ресурси, знижує енерговитрати, людські ресурси, а також дозволяє оптимізувати керівництво виробничими процесами. Зокрема, при використанні інформаційних технологій в науці стає можливим проведення модельного експерименту без використання коштовного обладнання, витрат матеріалів, без значних витрат електроенергії тощо, та без притягнення великої кількості персоналу лабораторій.

Важливою перевагою віртуальної лабораторії є можливість наочної імітації реального фізичного експерименту, а також можливість використання раніш отриманих експериментальних даних.

В Харківському національному університеті імені В.Н.Каразіна на факультеті комп’ютерних наук сумісно з провідними спеціалістами Інституту високих технологій ХНУ імені В.Н.Каразіна розроблюється комплекс віртуальних лабораторних робіт з вивчення процесів взаємодії випромінювання з речовиною. Віртуальний лабораторний практикум являє собою комплекс комп’ютерних програм, що розроблюється в середовищі операційної системи Linux SLC4.8 і використовує бібліотеку класів Geant4 – вільне програмне забезпечення.

Geant4 забезпечує повну систему інструментальних засобів для моделювання ядерно-фізичних процесів взаємодії випромінювання з речовиною: створення геометрії детекторів і установки в цілому, опис частинок і фізичних процесів; транспорт частинок і слідкування за ними; моделювання відгуку детектора; моделювання подій, керування треком тощо. Бібліотека класів Geant4 має великий набір службових функцій, генераторів випадкових величин і т.ін., а також є можливість використовувати різні модулі візуалізації [1–3].

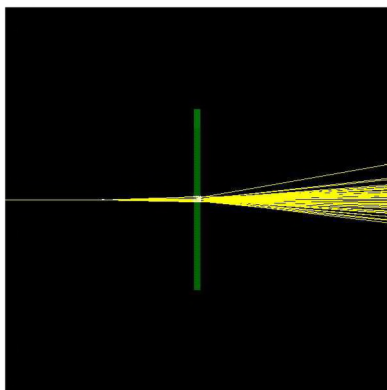
Колаборація Geant4 надає підтримку користувачів та повний комплект необхідної документації. Документація має в своєму складі довідку з інсталяції, документацію кінцевого користувача та довідники [2,3], а також ряд підручників, що охоплюють широке коло користувачів – від початківців до досвідчених користувачів Geant4. Підтримка користувачів

охоплює допомогу у вирішенні проблем, пов'язаних з кодом, консультації з використання інструментарію та реагування на запити про удосконалення коду. Користувач має також розраховувати на допомогу у дослідженні аномальних результатів [1]. На сайті проекту Geant4 [4] можна ознайомитись з веб-системою звітності та списком питань, що частіше задаються (FAQ). Колаборація Geant4 також керує веб-форумом користувачів, з підфорумами згідно з галузями наукових інтересів. У вільному доступі в Інтернеті існують актуальні робочі версії відкритого коду і документація, а також учбові приклади та комплекти учбових матеріалів.

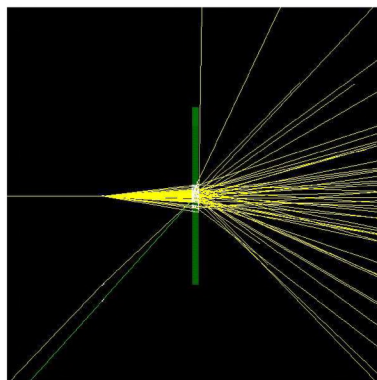
Великою перевагою Geant4, окрім його функціональності, гнучкості й надійності, є доступність його коду, можливість модифікації й адаптації практично до будь-яких задач комп'ютерного моделювання, в яких взаємодія випромінювання з речовиною відіграє роль.

На рисунку 1 наведено фрагмент лабораторної роботи з вивчення процесів багатократного кулонівського розсіювання електронів. Використовується бібліотека класів Geant4, візуалізація проходження 50 електронів з початковою енергією 1.5 MeV через шар кремнію 300 мікрон. Перед шаром кремнію на невеликій відстані розташована дуже тонка алюмінієва фольга, яка на рисунку не представлена.

На рис. 1а наведено візуалізацію експерименту, коли не враховуються процеси багатократного кулонівського розсіювання, на рис. 1б – експеримент має ті ж самі умови, але процеси багатократного кулонівського розсіювання враховуються. Первинні електрони спрямовані за нормаллю до поверхні кремнієвої площини, напрямок руху електронів зліва направо. Траєкторії електронів представлені жовтим кольором.



а



б

Рис. 1. Фрагмент лабораторної роботи з вивчення процесів багатократного кулонівського розсіювання електронів

В процесі виконання цієї лабораторної роботи студенти мають можливість задавати будь-які енергії частинок, вказувати товщини шарів, обирати різні типи первинних частинок і матеріали (з числа визначених в програмі), керувати рівнями деталізації результатів, що записуються у текстовий файл для подальшої обробки.

Віртуальний лабораторний практикум, що розроблюється, планується використовувати в процесі навчання студентів старших курсів фізико-технічного та фізико-енергетичного факультетів.

### **Література**

1. J. Allison et.al. *Geant4 – a simulation toolkit*. // *Nuclear Instruments and Methods in Physics Research*. – 2003. А 506. – P. 250 – 303.
2. *Geant4 User's Guide for Application Developers*. [Електронний ресурс]. – Режим доступу: <http://geant4.web.cern.ch/geant4/UserDocumentation/UsersGuides/ForApplicationDeveloper/fo/BookForAppliDev.pdf>. Електронна версія друкованої публікації, 2011. – PDF формат, 350 с.
3. *Geant4 Physics Reference Manual*. [Електронний ресурс]. – Режим доступу: <http://geant4.web.cern.ch/geant4/UserDocumentation/UsersGuides/PhysicsReferenceManual/fo/PhysicsReferenceManual.pdf>. Електронна версія друкованої публікації, 2011. – PDF формат, 554 с.
4. Веб-ресурс проекту Geant4. – Режим доступу: <http://geant4.cern.ch>

### **Використання вільного ПЗ для управління телескопом Мартинюк-Лотоцький К.П.**

Львівський національний університет імені Івана Франка, астрономічна  
обсерваторія, langure@mail.ru

During astronomical observations it is necessary to provide a precise telescope guiding for a celestial object. For this goal two programs have been written: the first of them is interface for telescope operation and the second is a micro-controller program. To move telescope around polar or declination axes stepper motors are used. Control motion of telescope is performed via the interface program, which transfers control commands to the micro-controller program. The micro-controller forms signals for the stepper motors.

В астрономічних спостереженнях за космічними об'єктами необхідно управляти рухом телескопа. Різні спостережувані об'єкти рухаються з різними швидкостями по небесній сфері. Найпростіше спостерігати зорі, рух відбувається за годинним кутом навколо полярної осі з швидкістю 15 кутових секунд за секунду. Інші об'єкти рухаються з іншими швидкостями-

ми і необхідно за допомогою двигунів відслідковувати їхній рух так, щоб отримувати нерухоме зображення у фокальній площині телескопа.

Для цього було розроблено контролер з мікропроцесором, який забезпечує формування сигналів на крокові двигуни, які обертають трубу телескопа. Під час розробки були використані і модифіковані бібліотеки для завантаження програм в мікропроцесор – bootloader. Контролер двигунів під'єднаний через перехідник USB-COM до комп'ютера. Для управління можна застосовувати стандартні термінали, які працюють з послідовними портами такі як: minicom, putty чи інші.

Під час розробки і написання програм були використані частини відкритих кодів для мікропроцесорів серії Atmega, зокрема код для оновлення вбудованого програмного забезпечення через канал зв'язку з комп'ютером. Написання програм, відлагодження і програмування мікропроцесора було проведено на системі Linux Debian і може бути перенесено на інші linux-системи.

Також розробляються програми для управління телескопом, які можуть взаємодіяти з іншими програмами для автоматизації спостережень. Усі програми взаємодіють через мережеві протоколи і можуть розташовуватись на різних комп'ютерах для зручності роботи.

## ***ALT LINUX как пример свободной платформы*** *Новодворский А.Е.*

ООО «Альт Линукс», [aen@altlinux.ru](mailto:aen@altlinux.ru)

This report is concerned with the principles of free software platforms construction with ALT Linux taken as an example, and the future development possibilities are examined. Development and deployment of ALT's Fifth and Sixth platforms are discussed, and the new Seventh platform's features are laid out.

Программные платформы ALT Linux представляют из себя комплекс средств коллективной работы, сборки, тестирования, создания дистрибутивных интегрированных решений и поддержки для аппаратных платформ i586,x86\_64, ARM на основе единой пакетной базы с поддерживаемой целостностью. Отличительной особенностью платформ ALT Linux являются строгий контроль за зависимостями пакетов при сборке и установке, оптимизация ресурсов, возможность создания сложных интегрированных решений, удобство разработки собственных независимых решений и их интеграции в платформу.

Примеры решений на основе платформы ALT Linux:

- Решения для платформы ARM от заказных встроенных и терминалов до серверов.

- Пакет решений для школ, включающий интегрированные сервер, терминальный сервер дистрибутивы для нетбуков, рабочих станций, планшетов.
  - Решения для медицинских учреждений: только в Москве 22 тысячи рабочих мест врачей.
  - Сертифицированное ФСТЭК решение для доступа к персональным данным (сервер/клиент).
  - Первая очередь системного ПО портала gosuslugi.ru, оригинальная технология быстрого обновления ПО.
  - Суперкомпьютеры в МГУ и ЮУрГУ .
- Седьмая платформа ALT Linux:
- Новые защищенные решения, интегрированные с партнерскими.
  - Решения для планшетов и компьютеров с сенсорным интерфейсом
  - Новые сервисы для пользователей и разработчиков.
  - Deepsolver – новый проект ALT Linux, потенциальная замена apt и других систем управления пакетами (эксперимент).

### ***Застосування вільних математичних пакетів до розв'язання задач теоретичної механіки***

*Періг О.В., Кисіль А.В., Матвеев І.А., Міхєєнко Д.Ю.*

*Донбаська державна машинобудівна академія, м. Краматорськ, Україна  
E-mail: [alexander.perig@gmail.com](mailto:alexander.perig@gmail.com), [alexander.perig@dgma.donetsk.ua](mailto:alexander.perig@dgma.donetsk.ua)*

This article is focused on the educational application of open source math software like Maxima, Xcas, Smath Studio and Scilab-Xcos to the solution of material particle dynamics problems in classical mechanics.

Наразі в системі вищої технічної освіти України має місце постійне скорочення годин на вивчення курсів вищої математики та математичного аналізу, що робить актуальним застосування у навчальному процесі сучасних систем комп'ютерної алгебри. Водночас висока вартість сучасних комерційних систем із відповідними модулями Maplesoft Maple–MapleSim, Wolfram Mathematica-MathModelica та MathWorks Matlab-Simulink не сприяють їх широкому розповсюдженню у вітчизняному освітньому процесі. Частково можливості середовища Wolfram Mathematica є доступними при застосуванні комерційної on-line системи Wolfram Alpha [1], яка може додатково показувати кроки розв'язку поставленої математичної задачі, але у безкоштовному режимі не дозволяє експорту графічних результатів, що зумовлює актуальність проблеми застосування вільних математичних пакетів.



Виконаємо порівняльний аналіз можливостей вільних систем на прикладі інтегрування задачі Коші  $x''-10x'+50x=0$ ,  $x'(0)=1$ ,  $x(0)=1$ , яка описує зростальні коливання матеріальної точки під дією змінної збурювальної сили.

У GPL CAS Maxima [2] синтаксис розв'язку має наступний вигляд:

```
(%i1) atvalue(x(t),t=0,1); {Shift+Enter, задання початкової умови x(0)=1}
(%o1) 1 {результат виконання команди в системі Maxima}
(%i2) atvalue('diff(x(t),t),t=0,1); {Shift+Enter, задання початкової умови
x'(0)=1}
(%o2) 1 {результат виконання команди в системі Maxima}
(%i3) 'diff(x(t),t,2)-10*'diff(x(t),t)+50*x(t)=0; {Shift+Enter, запис рівнян-
ня}
(%o3) 'diff(x(t),t,2)-10*('diff(x(t),t,1))+50*x(t)=0 {Maxima-запис рівнян-
ня}
(%i4) desolve(%x(t)); {розв'язання заданого диференціального рівнян-
ня}
(%o4) x(t)=%e^(5*t)*(cos(5*t)-(4*sin(5*t))/5) {аналітичний розв'язок
задачі}
(%i5) plot2d(%e^(5*t)*(cos(5*t)-(4*sin(5*t))/5),[t,0,2]); {побудова графі-
ка}.
```

Відзначимо, що складність збереження графічного результату із вікна `gnuplot graph` створює певні незручності при використанні GPL CAS Maxima.

Зазначимо, що аналітичний розв'язок зазначеної задачі Коші легко одержати і в системі GNU GPL Giac/Xcas [3], розробленою проф. V. Parisse:

```
> desolve([y''-10*y'+50*y=0,y(0)=1,y'(0)=1],y) {Enter, запис рівн. і поч.
умов}
[cos(5*x)*exp(5*x)+(-4*sin(5*x)*exp(5*x))/5] {результат інтегрування у
Xcas}
> plotfunc([cos(5*x)*exp(5*x)+(-4*sin(5*x)*exp(5*x))/5], x=0..2,
xstep=0.0005, display=magenta+remply) {побудова графіка розв'язку рівнян-
ня}. До вад реалізації середовища також відноситься проблема з
експортом графіків.
```

Чисельний розв'язок даної задачі у відкритій системі Smath Studio [4] здійснюється заміною диф. рівняння II порядку  $x''-10x'+50x=0$  системою рівнянь I порядку  $z=x'$ ;  $z'=10z-50x$  і введенням позначень  $x_2=z$ ;  $x_1=x$ , що дозволяє представити систему I порядку у вигляді  $x'_1=x_2$ ;  $z'=10*x_2-50*x_1$ . Тоді алгоритм розв'язку матиме наступний вигляд: `Cond ← mat(1,1,2,1)` {початкові умови}; `Begin ← 0` {поч. знач. незалеж. змінної  $t=0$ }; `End ← 2` {кінц. знач. незалеж. змінної  $t=2$ }; `Steps ← 200` {кількість кроків розбиття відрізка  $0 \leq t \leq 2$ }; `System(t,x) ← mat(el(x,2),10*el(x,2)-50*el(x,1),2,1)` {система диф. рівнянь}; `Solve ← rkfixed(Cond,Begin,End,Steps,System(t,x))` {побудова чисельного розв'язку із застосуванням ODESolvers.dll Plugin}.

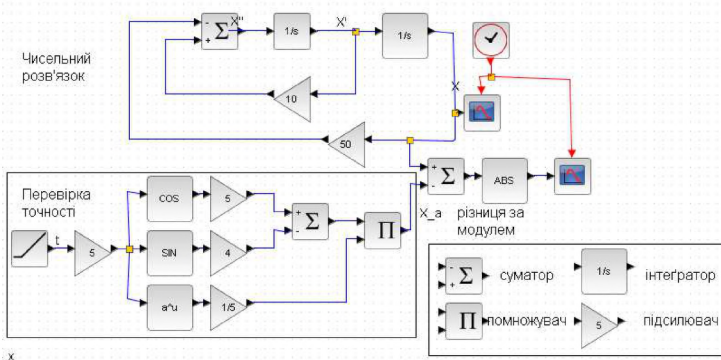


Друк матриці результатів здійснюється командами  $T \leftarrow \text{col}(\text{Solve},1)$ ;  $X1 \leftarrow \text{col}(\text{Solve},2)$ ;  $X2 \leftarrow \text{col}(\text{Solve},3)$ .

До наявних незручностей середовища Smath Studio можна віднести певну некоректність відображення графіків та відсутність бібліотеки ODESolvers.dll у дистрибутиві Smath StudioDesktop і, отже, необхідність її додаткового завантаження на форумі проекту за розташуванням “SMath Studio Forum » SMath Studio » Samples » rkfixed()” [5].

Цікавий підхід до чисельного інтегрування зазначеної задачі Коші можна реалізувати у відкритій системі GNU Scilab-Xcos [6-7]. Для цього диференціальне рівняння представимо у вигляді інтеграторів та суматорів на верхній частині рис. 1, причому кількість інтеграторів визначається порядком рівняння. Наразі початкові умови задамо як початкові значення суматорів. Для перевірки розв'язку у нижній частині рис. 1 зберемо схему, яка моделює розв'язання і на вхід якої необхідно подати потік змінної час. Оскільки модуль Xcos не дозволяє подати потік часу від таймеру, то застосуємо джерело сигналів Ramp. Зі зміною часу джерело Ramp генерує лінійно-зростальний сигнал, який можна вважати часом (рис. 1). Для оцінки точності побудованого чисельного розв'язку виконаємо віднімання за модулем вихідних сигналів двох систем у верхній та нижній частинах на рис. 1.

Рис. 1. Аналогова модель задачі Коші у середовищі GNU Scilab-Xcos



Наразі різниця сигналів для чисельного та аналітичного розв'язків не перевищує  $6 \cdot 10^{-5}$ , що переконливо свідчить про коректність чисельного інтегрування задач Коші у середовищі GNU Scilab-Xcos.

До певних складнощів застосування середовища GNU Scilab-Xcos при вивченні студентами Вишів курсу теоретичної механіки, можна віднести необхідність знання таких додаткових дисциплін як мехатроніка, схемотехніка та теорія автоматизованого керування, що дозволяє широко використовувати інструмент Scilab-Xcos лише на заняттях зі студентами старших курсів та магістрами.

**Висновки**

1. Застосування комерційних середовищ комп'ютерної алгебри при викладанні класичної механіки у вітчизняному освітньому просторі зовсім не підтримується державою, а висока вартість *Mathematica Home Edition* 295 US\$ та *Maple Personal Edition* 239 US\$ унеможливають їх широке використання на домашніх комп'ютерах викладачів та студентів в Україні. Найбільший подив викликає єдина цінова політика *Wolfram Research* та *Maplesoft* щодо розповсюдження їх продуктів як у розвинених, так і у пострадянських країнах. Зазначені обставини зумовлюють нагальну необхідність застосування відкритих систем комп'ютерної алгебри.
2. Переважна більшість диференціальних рівнянь класичної механіки впливають із основного закону динаміки і являють собою рівняння другого порядку, причому побудова аналітичних розв'язків задач Коші для диф. рівнянь другого порядку можлива лише із застосуванням двох відкритих систем комп'ютерної алгебри *GPL CAS Maxima* [2] і *GNU GPL Giac/Xcas* [3].
3. Чисельне інтегрування задач Коші для диф. рівнянь другого порядку у таких середовищах як *Smath Studio* [4-5], *GNU Scilab* [6], *GNU Octave* та *GNU R* (разом із модулем *GPL deSolve*) вимагає від користувача як попереднього зведення досліджуваного диф. рівняння II порядку до системи рівнянь I порядку, так і коректного запису матриці, що містить систему диф. рівнянь I порядку та початкові умови для їх чисельного розв'язання.
4. Можливості для аналогового чисельне інтегрування задач Коші для диф. рівнянь другого порядку реалізовані лише у відкритому середовищі *GNU Scilab-Xcos* [7]. Застосування *GNU Scilab-Xcos* вимагає представлення диференціального рівняння у вигляді інтеграторів та суматорів, а початкові умови для задачі Коші задаються як певні початкові значення суматорів.

### Література

1. <http://www.wolframalpha.com/>
2. <http://maxima.sourceforge.net/>
3. <http://www-fourier.ujf-grenoble.fr/~parisse/giac.html>
4. <http://en.smath.info/forum/>
5. <http://en.smath.info/forum/default.aspx?g=posts&t=726>
6. <http://www.scilab.org/>
7. <http://www.scilab.org/products/xcos>

**Використання ВПЗ для підвищення рівня захищеності  
комп'ютерних мереж та систем**  
Піскозуб А.З.

*Національний університет «Львівська політехніка»,  
azpiskozub@gmail.com*

The problems of different kinds of software security instruments, such as IDS, vulnerability assessment and penetration testing tools, log auditing and analysis tools are been discussed in this paper. The trend toward establishing minimum required level of security has affected many security safeguards and thus determined the integrated approach of mentioned above tools. Modern security tools simplify network discovery and vulnerability verification for specific exploits, help organizations improve stability/availability, meet security event management objectives and adhere to demands of regulatory compliance requirements.

Питання підвищення рівня захищеності ІТ-середовища є на сьогодні актуальним не лише для великих корпорацій, але і для невеликих організацій, як для потреб бізнесу, так і для навчальних установ.

Як правило, такі питання вимагають інвестування певної суми коштів на організацію програмно-апаратних засобів підвищення рівня захищеності ІТ-інфраструктури, причому лєвова частка тих коштів виділяється на закупівлю ліцензій відповідних програмних продуктів. Сьогодні, без сумніву, заслуговує особливої уваги питання використання вільного та відкритого ПЗ (ВВПЗ) для потреб підвищення рівня захищеності комп'ютерних мереж і систем.

Як правило, більшість організацій для підвищення рівня захищеності ІТ-середовища використовують антивірусний захист та захист на рівні використання міжмережєвих екранів (firewalls). Подекуди цей список доповнюється використанням рядом організацій засобів виявлення/запобігання втручань (intrusion detection/prevention systems – IDS/IPS), які допомагають в залежності від налаштувань цих засобів оперативнo реагувати на події в системах.

Насправді, список інструментів підвищення рівня захищеності комп'ютерних мереж та систем доповнюють сканери вразливостей, системи тестування на проникнення та системи журналізації подій, описи та рекомендації використання яких можна знайти в багатьох відомих документах, зокрема стандарті як ISO/IEC 27001:2005 [1], що описує методи захисту та системи менеджменту захисту інформації в інформаційних технологіях.

Сканери вразливостей дозволяють сканувати мережі, комп'ютери та програми на предмет виявлення можливих проблем в системі безпеки, оцінювати і рекомендувати усунення вразливостей. Побутує думка, що сканери при виявленні окремих потенційних вразливостей виконують злам системи, використовуючи отриману інформацію та відповідні про-

грами-експлоїти. Насправді на цьому етапі застосовуються інші системи – системи тестування на проникнення, які на базі знайденої сканерами інформації моделюють атаки зловмисників, використовуючи при цьому активний аналіз системи на наявність потенційних вразливостей, які, своєю чергою, можуть спровокувати некоректну роботу цільової системи, або повну відмову в обслуговуванні. Аналіз ведеться з позиції потенційного атакуючого і може включати в себе активне використання вразливостей системи. Результатом роботи є звіт, який містить в собі всі знайдені вразливості системи безпеки, а також може містити рекомендації щодо їх усунення. Мета випробувань на проникнення - оцінити його можливість здійснення і спрогнозувати економічні втрати в результаті успішного здійснення атаки. Випробування на проникнення є частиною аудиту безпеки, який повинна пройти кожна компанія, яка має на меті отримати сертифікати відповідності міжнародним стандартам, таким як ISO/IEC 27001:2005.

Доповнюють список інструментальних засобів системи журналізації подій. Роль журналізації подій в комп'ютерних системах не можна недооцінити, оскільки вони є невід'ємною частиною будь-якої системи захисту інформації і дозволяють не лише бути доказом подій, що сталися в системі, але і допомагають підвищити рівень захищеності системи.

Журнали подій містять важливу інформацію, що дозволяє визначити вразливості в системі, зупинити несанкціоноване втручання і визначити місце чи сервіс в системі, які вимагають негайної уваги. В основному, журналізація подій використовується для детектування та аналізу інцидентів, пов'язаних з проблемами безпеки чи продуктивності, для відповідності вимогам політик безпеки, нормативних документів, аудиту чи стандартів, правових питань, а також для мінімізації простоїв.

Одною з основних проблем на сьогодні є опрацювання даних журналів подій, а особливо в реальному масштабі часу. Аналіз журналів подій по суті зводиться до того, що необхідно якісно виокремити важливе і відкинути непотрібне, а це є насправді важко.

В більшості випадків, IT-персонал звертається до журналів подій тоді, коли щось трапилось. Ніхто, практично, не здійснює моніторинг систем цілодобово, 7 днів в тиждень, цілий рік. Ця ситуація може бути неприпустимою в системах з підвищеними вимогами до захисту інформації, до систем з високою відмовостійкістю тощо. Хіба що будуть застосовані найсучасніші системи управління інформацією та повідомленнями безпеки (Security Information and Event Management (SIEM)).

У цьому випадку стає можливим централізований онлайн-моніторинг подій з різноманітних джерел – операційних систем, прикладних сервісів, мережевих пристроїв тощо з відображенням в режимі реального часу, кореляція результатів, надання звітів, ранжування за критичністю сервісів, нотифікація/попередження користувачів при настанні певних подій, що дозволяє забезпечити відповідальний персонал та користувачів системи повною інформацією про її стан і, тим самим, забезпечити мож-

ливість ефективно управляти ризиками системи. SIEM-системи дозволяють забезпечити інцидент менеджмент подій, пов'язаних з безпекою інформації, з побудовою послідовностей кроків по усуненню цих інцидентів відповідно до попередньо заданих правил.

Опишемо приклад конкретної реалізації системи захисту IT-інфраструктури із використанням вищезгаданих типів інструментальних засобів підвищення рівня захищеності комп'ютерних мереж та систем на базі ВВПЗ.

Кінцева реалізація системи захисту складається з 3 операційних систем (ОС), розгорнутих в віртуальному режимі під управлінням VMWare ESXi 5, розташованих на одному фізичному хості (Intel Core i5, 8GB, 500 Gb HDD):

- системи централізованого збору та обробки інформації журналів подій (ОС Linux CentOS 6.2);
- системи оцінки вразливостей та тестування на втручання (Back Track 5 на базі ОС Linux Ubuntu 10.04);
- системи управління інформацією та повідомленнями безпеки (OSSIM на базі ОС Linux Debian 6).

Критерії вибору операційних систем виходять за рамки даної статті, проте зазначимо, що вибір ОС CentOS та Debian в даному випадку обґрунтований тим, що вони вважаються одними з найстабільніших серверних Linux-дистрибутивів. Система оцінки вразливостей та тестування на втручання використовується як клієнтська станція без доступу ззовні, тому для цих задач був вибраний спеціалізований Linux-дистрибутив на базі Ubuntu з величезною кількістю програм для оцінки вразливостей та тестування на проникнення.

На системі централізованого збору та обробки інформації журналів подій під управлінням ОС Linux CentOS 6.2 використовуються наступне ВВПЗ: фаєрвол iptables, системи журналізації syslog-подій у складі rsyslogd, loganalyzer, які забезпечують централізований збір інформації з усіх відповідальних вузлів мережі – активного мережевого обладнання та серверів під управлінням ОС Linux та ОС Windows (за допомогою використання програми Snare), системи Sagan – монітору журналів подій реального часу, який інтегрується з IDS-системою виявлення втручань на базі Snort, встановленою на даному хості. Snort забезпечує моніторинг мережі через дзеркалюючий SPAN-порт кореневого комутатора мережі. Окрім цього даний хост відіграє роль центрального сервера часу IT-інфраструктури.

Для оцінки вразливостей систем та тестування на їх проникнення на однойменній системі використовуються наступне ВВПЗ: фаєрвол iptables, сніфери tcpdump, wireshark; сканер портів nmap; сканери вразливостей openvas, Nexpose Community edition (окремо відзначимо, що продукт Nexpose компанії Rapid7, freeware-версію якого ми використовуємо, вважається одним з найкращим в своєму класі [2]), веб-сканери w3af, wapiti,

нікто, arachni; системи тестування на проникнення Metasloit Framework, Metasploit Community edition.

OSSIM (open source SIEM) [3] вибрана з поміж 85 продуктів класу SIEM-систем [4]. Серед списку з 85 позицій лише 8 відносяться до безкоштовних продуктів і лише 2 з них належать до продуктів з відкритим кодом. Вона являє собою повноцінну безкоштовну відкриту SIEM-систему. Разом з AlienVault Unified SIEM, AlienVault OSSIM використовується в більшій кількості організацій, ніж усі інші SIEM-продукти разом взяті. AlienVault OSSIM забезпечує усю функціональність, яка вимагається для детектування та профілювання атак, і забезпечує всебічну, інтелектуальну платформу управління безпекою з набором відповідних інструментальних засобів.

Резюмуючи наведений матеріал, можна відзначити необхідність комплексного підходу – використання широкого спектру програмних засобів для підвищення рівня захищеності комп'ютерних мереж та систем. При цьому є можливо забезпечити ці рішення на базі ВВПЗ.

### ***Література:***

1. *ISO/IEC 27001:2005, Information technology — Security techniques — Information security management systems — Requirements*
2. *Gartner: MarketScope for Vulnerability Assessment 2011.*  
<http://www.gartner.com/technology/media-products/reprints/qualys/article1/article1.html>
3. *AlienVault Open Source SIEM (OSSIM).* <http://www.alienvault.com/>.
4. *Mosaic Security Research. Log Management & Security Information and Event Management (SIEM).* <https://mosaicsecurity.com/categories/85-log-management-security-information-and-event-management>.



**Программная реализация декодера одного класса  
помехоустойчивых кодов на алгебраических кривых.  
Проектирование на основе шаблонов обобщённого и мета-  
программирования**  
Пеленицын А.М.

*Южный федеральный университет, [apel@sfedu.ru](mailto:apel@sfedu.ru)*

We present the decisions made in our software implementation of decoder for a class of algebraic geometry codes. The decisions are concerned with generic programming approach and show some metaprogramming techniques which, we believe, are of considerable importance for similar problems from the field of error-correcting codes or, more generally, computational algebraic geometry. We also describe some features of C++11 which are aimed to improve generic programming capabilities of the C++ programming language.

Построение помехоустойчивых кодов на основе алгебраических кривых (АГ-кодов) и методы их декодирования [1] используют ряд абстрактных математических объектов, программная реализация которых представляет собой более сложную задачу, чем в случае классических кодов. Поскольку доступно лишь ограниченное число открытых реализаций декодеров для АГ-кодов [2,3], опыт их реализации представляет интерес как с теоретико-кодовой точки зрения, так и в контексте целого класса наукоёмкого программного обеспечения. Тот факт, что конструкции помехоустойчивых кодов имеются в некоторых системах компьютерной алгебры, например GAP (пакет GUAVA [4]), позволяет рассматривать вопрос о возможности интеграции проделанной работы в такие системы.

Язык программирования C++ является достаточно универсальным инструментом для создания наукоёмкого открытого ПО: один из ведущих компиляторов C++, входящий в состав GCC, хорошо переносим и имеет свободную лицензию. Также важно то, что данный компилятор открывает доступ к последним изменениям в языке C++, отражённым в новом стандарте, который принято обозначать C++11 (ранее: C++0x) [5]. Важным является тот факт, что значительная часть нововведений языка посвящена расширению поддержки обобщённого программирования, а также (или — в частности), функционального программирования на C++.

Одной из интересных особенностей данного языка являются шаблоны C++. В ряде работ [6,7,8] было продемонстрировано, насколько мощными являются выразительные средства, представляемыми шаблонами. Здесь можно выделить самостоятельный подход к программированию, называемый «обобщённое программирование», а также набор приёмов, объединяемых понятием «метапрограммирование».

В докладе рассматривается ряд техник, использованных в полученной автором реализации декодера: рекурсивное инстанцирование шаблонов,

классы характеристик типов и одна общая проблема, связанная с ними, а также её решение, классы стратегий. Данные техники основаны в той или иной степени на механизме шаблонов C++. Особое внимание уделено применению средств C++11.

### **Литература**

1. Nøholdt T., van Lint J.H., Pellikaan R. *Algebraic geometry codes* // *Handbook of Coding Theory*, V.S.Pless, W.C. Huffman, and R.A. Brualdi, Eds. Amsterdam, The Netherlands: Elsevier, pp. 871–961, 1998.
2. Madelung Y. “Implementation of a decoding algorithm for AG-codes from the Hermitian curve,” Rept. IT 93-137, Inst. Circuit Theory Telecom., Tech. Univ. of Denmark, Lyngby, Denmark, Apr. 1993.
3. Маевский А.Э., Пеленицын А.М. О программной реализации алгебро-геометрического кодера с применением алгоритма Сакаты // В сб. «Материалы X Международной научно-практической конференции „Информационная безопасность“», ч.2. Таганрог. ЮФУ. 2008. С. 55–57.
4. GUAVA A GAP4 Package for computing with error-correcting codes <http://sage.math.washington.edu/home/wdj/guava/htm/chap0.html>
5. C++0x/C++11 Support in GCC <http://gcc.gnu.org/projects/cxx0x.html>
6. Abrahams D., Gurtovoy A.. C++ Template Metaprogramming: Concepts, Tools, and Techniques from Boost and Beyond. Addison-Wesley, 2004.
7. Вандевурд Д., Джосаттис Н. Шаблоны C++: справочник разработчика. М.: Вильямс, 2003. 544 с.
8. Александреску А. Современное проектирование на C++. М.: Вильямс, 2002. 336 с.

## **Досвід та перспективи використання вільного ПЗ у видавничо-поліграфічній галузі**

*Піскозуб Й.З., Миклушка І.З., Піскозуб Л.Г.*

*Українська академія друкарства, факультет видавничо-поліграфічних інформаційних технологій, кафедра прикладної математики і комп'ютерних інформаційних систем,*

*[www.pmkis.uad.lviv.ua](http://www.pmkis.uad.lviv.ua), [pmkis@uad.lviv.ua](mailto:pmkis@uad.lviv.ua)*

The comparative review of free software for specific professional needs in the field of publishing and printing is done. Experience of implementing and using this software in the field of training for publishing and printing industry is discussed.

Цифрові технології у поліграфії та видавничій справі сьогодні швидко витісняють традиційні, хоч і не замінюють їх повністю. Сучасні комп'ютери удосконалили методи підготовки документів до друку і запропонували нові засоби та інструменти, щоб спростити процес їх створення. Технології цифрової додрукарської підготовки, які налічують трохи більше десятка років, радикально змінили всю видавничу сферу. Знаковим явищем у комп'ютерному світі стала поява мультимедійних систем, які є результатом розвитку нових технологій і яким прогнозують великі перспективи. Очевидно, що для успішної та продуктивної реалізації сучасного типового проекту у видавничій галузі потрібні групи спеціалістів, які досконало володіють новими технологіями і засобами. На передній план, крім фахового володіння професійними навичками, виходять проблеми правильного вибору комп'ютерного обладнання, відповідного програмного забезпечення і оптимізації робочого середовища для виконання конкретного класу завдань.

У сучасних реаліях як для підприємств, так і для персональних користувачів комп'ютерної техніки на передній план виступають питання жорсткої економії фінансів і використання ліцензійно "чистих" програмних продуктів. Багато видавців вважають за краще використовувати для вирішення своїх виробничих завдань комерційні програми, стверджуючи що якщо програма платна, то і її функціональність буде адекватна її вартості. Останнім часом ситуація починає змінюватися: росте зацікавлення вільним програмним забезпеченням, особливо в умовах розробки Державної цільової програми з впровадження вільного ПЗ в навчальних закладах та державних установах України, провадяться наукові дослідження з цього питання. Використання вільного програмного забезпечення в освіті підтримується багатьма країнами Євросоюзу, Бразилії, Китаю. У Росії аналогічна Державна програма прийнята ще у 2004 році.

У разі застосуванні пропрієтарного ПЗ відсутня можливість вибору комплексних інформаційних рішень, що породжує технологічну залежність, в межах якої розробник односторонньо диктує конфігурації програм та їх ціни. Відкрите ПЗ позбавлене цього недоліку і дає змогу

користувачеві змінювати і виправляти програму, пристосовуючи її до специфічних потреб.

Висновки авторів європейських досліджень говорять про існування чотирьох основних мотивів для використання вільного програмного забезпечення державними організаціями:

- усунення залежності від одного постачальника програмного забезпечення;
- зменшення витрат на інформатизацію;
- безпека;
- прозорість (доступ до вихідного програмного коду).

Потреби галузі спонукають до введення змін у зміст підготовки майбутніх фахівців та необхідності їх ознайомлення з існуючим вільним ПЗ для фахових задач. Особливим є становище УАД, як головного в Україні вузу з напрямом “Видавничо-поліграфічна справа та поліграфія”, обов’язком якого є розробка та впровадження державних стандартів для спеціалістів цього напрямку.

Наш досвід впровадження в навчальний процес вільного ПЗ налічує кілька років:

- З 2002-3 р.р. почали вивчати сучасні бази даних на прикладі СУБД MySQL (з використанням веб-сервера Apache для організації локального веб-середовища і мови скриптів PHP для вводу інформації у базу зі скрипта та публікації вмістимого баз даних на веб-сторінках).
- 2005 р. – започаткували 3-ох-семестровий курс веб-програмування для спеціалістів у галузі електронних видань з ґрунтовним вивченням сценарних мов – клієнтської JavaScript та серверної PHP. Курс завершується курсовою роботою – створенням сайту або інформаційної системи з веб-інтерфейсом. Є цікаві приклади створення повноцінних навчальних мережових видань, які вже використовуються.
- З 2007 р. для тестування знань студентів використовуємо адаптовану до вимог болонського процесу та наших потреб систему OpenTEST, базовану на вільних програмних продуктах. Сервер системи працює на ОС Linux Ubuntu.
- З 2009 р. вивчаємо вільні операційні системи – на прикладі Linux Ubuntu.
- 2010-11 р.р. – вивчаємо пакети вільного ПЗ для наукових розрахунків, професійної верстки та опрацювання графічної інформації.

Крім цього, студенти досліджують проблеми OpenSource ПЗ у дипломних роботах, розробляємо новий проект з веб-інтерфейсом для тестування знань школярів та студентів, який базується на сучасних версіях вільних програмних продуктів. Вийшли друком наші навчальні посібники з вільного ПЗ з грифом МОНмолодьспорту.

Окремої уваги заслуговує проблема неповної тотожності функціональності вільних пакетів професійної верстки та опрацювання графічної інформації зі своїми пропрієтарними відповідниками. Мова йде

про ПЗ: GIMP, LibreOfficeDraw, Inkscape, Scribus – аналоги комерційних програмних рішень фірм Adobe, Corel чи Quark.

На наш погляд найвужчим місцем видавничої технології GPL-програм була операція верстання. Сьогодні версію 1.4 програми Scribus (реліз від 1 січня 2012 року) можна сміливо рекомендувати в якості видавничої платформи. Ця програма підтримує безліч професійних функцій, включаючи використання технологій з форматами PDF, EPS, SVG, кольорову схему CMYK, систему керування кольором ICC, може виконувати кольороподіл та генерувати інтерактивні PDF-документи з вбудованими скриптами і ефектами презентації, підтримує різноманітні шрифти в кодуванні Unicode.

В новій версії програми Scribus застосований покращений алгоритм перенесення рядків і форматування абзацу, реалізований новий двигун оконтурювання, який дозволяє використовувати лігатури і розширені можливості OpenType для тексту, а також забезпечує підтримку національних писемностей, наприклад арабських мов та івриту. Пропонується нова система керування стилями (навіть символні стилі) і вирівнювання меж блоків. З'явилася можливість гнучкого управління відстанню між буквами та словами, використання типографічних прогалів, які не передбачені в шрифті.

Окремо варто відзначити непогану інтеграцію з програмами растрової та векторної графіки, відповідно GIMP та Inkscape. Наприклад, показовими є робота з плашковими кольорами типу Pantone, реалізація підтримки формату TIFF та PSD, які можуть містити шари і відсічні контури в зображеннях.

Вже без обмежень застосовуються рамки, видимі і невидимі з підтримкою округлених меж і кутів для тексту та інших об'єктів, які можна обертати і масштабувати; шаблони сторінок для легкого управління документами, багатошаровість сторінок документів. Нарешті, Scribus вміє встановлювати в документ мітки обрізу, суміщення тощо, здатний виводити інформацію на високоякісному професійному обладнанні за допомогою мови PostScript (причому Level3, а це вже прямий крок до JD-F-специфікації). Вивідні файли вже можуть без обмежень опрацьовуватись в PrePress-програмах і на різноманітних RIP-ах. В новій версії зняті обмеження з імпортом в OpenXML.

В Scribus є можливість генерування та редагування ISBN- і штрихкодів, що підтверджує повноцінний видавничий процес. А прибічникам автоматизації низки рутинних операцій цілком доречно надана можливість використання макросів на основі Python-скриптів.

На завершення зазначимо, що аналіз використання вільного ПЗ у видавничо-поліграфічній галузі цілком може екстраполюватися на інші галузі та сфери застосування і, безперечно, свідчить про його зростаючі перспективи.



**Використання вільного програмного забезпечення при  
створенні науково-освітніх електронних бібліотек**  
Прилуцка Н.С.

*Житомирський державний університет імені Івана Франка, prilutska@gmail.com*

The main objects of free software system Eprints are considered which was used for developing Electronic library of Ivan Franko Zhytomyr State University.

У XXI столітті, в процесі перехідного періоду від інформаційного суспільства до суспільства знань, у який ввійшла Україна, важливим завданням постає створення мережевого розподіленого середовища для забезпечення доступу до ресурсів бібліотек, архівів, музеїв та інших інформаційних систем у сфері науки та освіти. [Спірін О. М. Проектування системи електронних бібліотек наукових і навчальних закладів АПН України [Електронний ресурс] / О. М. Спірін, В. М. Саух, В. А. Резніченко, О. В. Новицький // Інформаційні технології і засоби навчання. – 2009. – №6 (14). – Режим доступу до журн. : [http://www.journal.iitta.gov.ua.](http://www.journal.iitta.gov.ua)]

В Україні в 2009 році прийнята Концепція Державної цільової національно-культурної програми створення єдиної інформаційної бібліотечної системи "Бібліотека - XXI", що спрямована на вирішення проблеми ефективності використання і забезпечення доступності документів, які зберігаються у бібліотечних, архівних та музейних фондах.

Бібліотеки вузів та наукових установ акумулюють в своїх фондах твори в різних формах подання, створені викладачами та науковими співробітниками. Одночасно з цим розвиваються електронні бібліотеки (ЕБ) для доступу до наукових та гуманітарних знань, що представляють досить потужний мережевий ресурс. Особливу роль у розширенні доступу до інформації відіграють електронні бібліотеки, які забезпечують подання інформаційних ресурсів в електронному вигляді. Вони на даний момент є частиною освітнього інформаційного простору, а також національного бібліотечно-інформаційного фонду країни.

ЕБ бібліотека будується на окремій платформі, з використанням спеціалізованого програмного забезпечення (ПЗ) такого, як Dspace [<http://www.dspace.org/>], Eprints [<http://www.eprints.org/>], Greenstone [<http://www.greenstone.org/>] та ін. Модель інформаційного середовища у цих програмних продуктах відображає всі особливості електронної бібліотеки у порівнянні з електронними каталогами.

Розглянемо основні об'єкти системи програмного забезпечення Eprints, що адаптується до використання при створенні електронних бібліотек (на прикладі Електронної бібліотеки Житомирського державного університету імені Івана Франка) в Україні.



Eprints – вільно розповсюджене програмне забезпечення під ліцензією GNU, що використовується для формування та керування відкритих архівів. Нині у світі створено більше 200 архівів, що використовують ПЗ Eprint.

Eprints функціонує як централізований сервіс.

Система має такі функціональні аспекти:

- для базової організації даних зафіксована певна модель;
- система зберігає інформацію про користувача;
- можна контролювати ступінь доступу користувачів;
- можливе редагування, доповнення архівного матеріалу;
- система може імпортувати велику кількість елементів для пакетного завантаження контенту;
- матеріали в архіві є доступними через посилання, що наведені в описі конкретного елемента, за яким можна робити бібліографічні посилання на даний елемент;
- система має дружній інтерфейс і зручну навігацію;
- для інтегрованого пошуку документів підтримується протокол збору метаданих OAI-PMH;
- система забезпечує автоматичну розсилку з повідомленнями про нові матеріали електронною поштою через службу підписки;
- система надає можливість обробляти дані різних форматів, від простих текстових документів до наборів даних і цифрових відео.

ПЗ Eprints надає можливості [Gutteridge C. EPrints 2.3 Documentation. October12, 2005. <http://www.EPrints.org/documentation/tech/EPrints-docs.pdf>]

:

- створення електронних архівів;
- підтримки файлів різного формату;
- індексації файлів PDF, ASCII, Microsoft Word, HTML;
- перегляду формул у документах, написаних мовою LaTeX;
- виконання повнотекстового й розширеного пошуку (по метаданим);
- гнучкого адміністрування прав доступу;
- гнучкої інтеграції з основним сайтом (з використанням основного стилю оформлення Web-Сайту організації).

У системі Eprints розроблена стандартна конфігурація, що підходить для більшості архівів. Усі електронні документи розміщені в Eprints у вигляді архіву. Під архівом (репозитарій) в Eprints розуміють колекцію електронних документів. Архів містить документи або об'єкти та записи метаданих, що описують ці документи. Архів може містити різні типи документів, різних форматів і розмірів. Всі документи архіву можна згрупувати за певним и метаданими.

Створювати відкритий науковий архів можна: 1. коли цим займається безпосередньо редакція, що видає електронні видання з відкритим доступом за аналогією із друкованими виданнями; 2. коли автори самостійно вносять свої статті в архіви, цей процес називається .

Eprints обробляє різні типи записів, що описують документи. Кожний тип документів має свій набір полів метаданих максимально корисний при описі. Для повноти архіву деякі поля метаданих є обов'язковими для заповнення. Крім того, кожне поле має детальний опис залежно від типу документа.

У системі Eprints виділені наступні типи документів:

- періодичні видання.
- окремі статті і препринти.
- книги та монографії.
- науково-дослідні звіти.
- автореферати і дисертації.
- матеріали конференцій.
- інші інформаційні ресурси

Сервіс пошуку Eprints індексує всі наявні файли у всіх архівах, що дозволяє реалізувати базовий (simple search) і розширений (advanced search) пошук.

Сторінки перегляду створюються автоматично. Ці сторінки статичні і генеруються скриптом, що запускається через рівні проміжки часу Крім того, ПЗ Eprints надає можливість перегляду нових надходжень за останній тиждень (рис. 1).

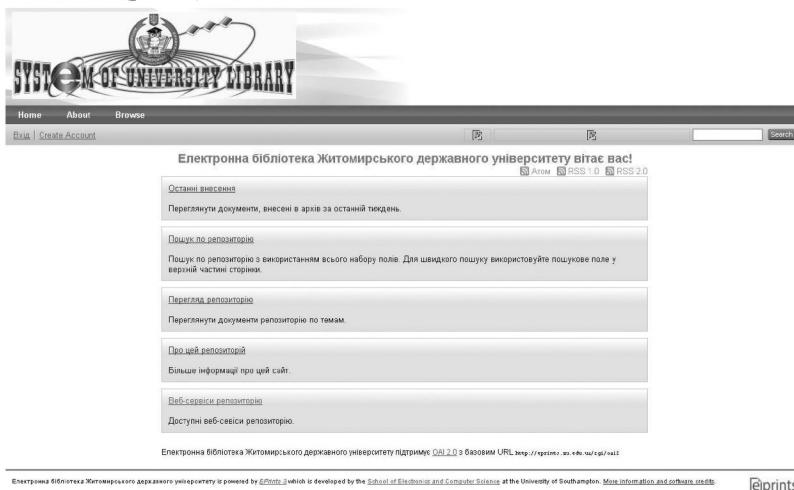


Рис. 1. Стартова сторінка електронної бібліотеки Житомирського державного університету імені Івана Франка

Процес наповнення бібліотеки проходить у кілька етапів. Всі операції здійснюються через Web-Інтерфейс. Вносити електронні документи можуть тільки зареєстровані користувачі.

EPrints це зручне і просте в налагодженні програмне забезпечення, що задовольняє всі вимоги сучасних програмних продуктів в області створення електронних бібліотек. У майбутньому розробники Eprints обіцяють

поліпшену підтримку технології, що забезпечить безпосереднє цитування документа (дозволить додавати в основний документ цитати з першоджерела, не завантажуючи весь документ).

### **Література**

1. Gutteridge C. *EPrints 2.3 Documentation*. October 12, 2005.  
<http://www.EPrints.org/documentation/tech/EPrints-docs.pdf>
2. Концепція Державної цільової національно-культурної програми створення єдиної інформаційної бібліотечної системи "Бібліотека - XXI" [Електронний ресурс] / [схвалено розпорядж. Кабінету Міністрів України від 23 груд. 2009 р. № 1579-р] // Веб-сайт Верховної Ради України. – 2009. – Режим доступу : <http://zakon1.rada.gov.ua/cgi-bin/laws/main.cgi?nreg=1579-2009-%F0>.
3. Спірін О. М. Проектування системи електронних бібліотек наукових і навчальних закладів АПН України [Електронний ресурс] / О. М. Спірін, В. М. Саух, В. А. Резніченко, О. В. Новицький // Інформаційні технології і засоби навчання. – 2009. – №6 (14). – Режим доступу до журн. : <http://www.journal.iitta.gov.ua>.

## **Державна політика у сфері ВПЗ ДЦНТП з розповсюдження ПЗ з відкритим кодом: велика перемога чи велика профанація?**

**Радченко Ю.А.**

*Інститут економіки та прогнозування НАНУ. Координатор УАВВПЗ, член громадської ради при ДАНИ, Київ. [yuriradchenko@gmail.com](mailto:yuriradchenko@gmail.com)*

The "State Special-Purpose Scientific-Technical Programme of Implementing Free Software in Governmental Institutions" is finally approved by the government. However, the detailed list of activities seems to be totally unexpected and somewhat fantastic, so that the apprehensions arise on the subject is it just a trivial "screen" for washing budget funding. Whether it is so or not, the scrupulous analysis of main finance statement can tell.

"Державну цільову науково-технічну програму використання в органах державної влади програмного забезпечення з відкритим кодом" було вперше затверджено ще на початку 2010 року. Характерно, що спершу у тодішньому ДКІ (Державний комітет інформатизації України) було визначено обсяг фінансування програми (45 мільйонів гривень), і лише через рік здійснено спробу розгортання цієї суми до відповідної "Таблиці завдань та заходів". Причому цю спробу не можна було вважати вдалою: адже жодна з методик розрахунків (якщо такі й існували) ніколи не була представлена громадськості.

З 2011 року процес очолив заново сформоване "Державне агентство з питань науки, інновацій та інформатизації України" (ДАНІ), а 5 грудня урядовий портал повідомив, що на засіданні уряду програму остаточно затверджено. Незабаром он-лайн з'явився і основний плановий та економічний документ програми – "Таблиця завдань та заходів" [1].

На подив представників ІТ-громадськості і фахівців, аналіз відповідного документу на загальну суму майже у 42 мільйони гривень (37.668.000 грн. з державного бюджету та 4.155.000 "з інших джерел") наводить на думку, що за більшістю пунктів програми державні чиновники намагалися продати самі собі стандарти, формати файлів, технічні протоколи і програмне забезпечення, які а) вже існують і використовуються в світовому масштабі; б) емітуються або виробляються широковідомими технічними спілками та міжнародними колективами розробників; в) не є у компетенції виконавчої структури будь-якого національного уряду. Як і раніше, жодна з цифр програми не є аргументованою економічно, і за жодним з пунктів програми не наведено методики розрахунків.

Громадянам, які не є фахівцями в ІТ-галузі, простий перелік заходів програми нічого не скаже: необхідно проаналізувати їх з фахової точки зору, по суті. Тому наведемо нижче найбільш красномовні приклади, узяті безпосередньо із вищезгаданої "таблиці".

Пункт 2, завдання: "Виконання науково-дослідних робіт з розроблення та використання програмного забезпечення з відкритим кодом".

"Науково-дослідні роботи з розроблення" вже розробленого, тим більше "науково-дослідні роботи з використання" чогось вже розробленого – це, як казав Гоголь, "Андрони ідуть, нонсенс, нісенітниця, та чоботи, зварені некруто". Дивимось конкретніше, за окремими "заходами" програми.

Заходи 1-3: "Розроблення вимог до комплексу програм і форматів,.. розроблення методики відбору ПЗ,.. вивчення питання щодо визначення шляхів створення та розвитку інфраструктури для формування і розповсюдження ВВПЗ". Вартість: 1 млн. грн.

Тут наявна серйозна неадекватність у сенсі "форматів" та "шляхів створення і розповсюдження".

1) Україна не може (за фактом) вводити нові формати у сфері ІТ: все, що потрібно, вже існує. Це не означає, звісно, що український програміст в принципі не може придумати якийсь новий формат даних (іншими словами – формат файлів). Але питання: навіщо? Що є такого в масі вільно доступного програмного забезпечення, що не вдовольнило б з функціональної точки зору, прямо скажемо, скромних вимог з боку працівників державного сектору? Введення ж якихось штучних форматів даних на локальному рівні, не прийнятих міжнародною спільнотою розробників, автоматично означає, що цей формат не буде ані "відкритим", ані "вільним".

2) Ще гірше – зі "шляхами створення і розповсюдження". Вільне/відкрите програмне забезпечення тому і зветься "вільним" та "відкритим", що той, хто його зробив, розміщує його в Інтернет а) для вільного доступу і б) з усіма початковими кодами. Отже, відповідні "шляхи створення і розповсюдження" вже існують і використовуються в світовому масштабі у повній незалежності від розпорядників коштів (Держінформнауки, Мінмолодьспорту, ДССЗ31). Виникає питання, на що беруться гроші. Чи не на розповсюдження посилання на сайт [orepo-office.org](http://orepo-office.org), наприклад?

Той же пункт "завдань", "захід" 4: "розроблення стандартів створення та використання програмного забезпечення з відкритим кодом" (всього – на 12 стандартів). Ціна питання – 480 тис.грн.

Повна неадекватність.

1) Стандарти створення і використання ПЗ емітуються міжнародними інженерними спілками і дослідницькими консорціумами (ISO, W3C, IEEE CS тощо), а пропонуються – чільними розробниками ПЗ. За фактом Україна може лише використовувати вже розроблені та імплементовані (!) до конкретних продуктів стандарти.

2) Технічні стандарти створення ПЗ як для вільної, так і для пропрієтарної (приватновласницької) моделі є загальними, і нічим не відрізняються. Простіше кажучи, мови програмування та вимоги до інтерфейсу користувача і тут, і там абсолютно тотожні.

3) У сфері організації виробництва та використання кінцевого продукту жоден орган державної влади в принципі не може нав'язати якісь власні "стандарти" для розробників, бо міжнародні колективи розробників не підвітні національним урядам, а користувачі по всьому світові приймають Вільну ліцензію, що розроблена FSF [2] шляхом простого приєднання до вимог ліцензії. Намагання обмежити Вільну ліцензію у будь-який спосіб, або привласнити вільний/відкритий продукт, переслідуються юридичною службою FSF так само, як переслідуються порушення прав на пропрієтарну інтелектуальну власність асоціацією BSA [3] (Business Software Association).

Той же пункт "завдання", "захід" 5: "розроблення стандартів створення та використання програмного забезпечення, що створюється на замовлення органів державної влади": 400 тис. грн.

Повна неадекватність. І не тільки тому, що і "відкриті", і "закриті" стандарти вже за фактом існують, і ви лише можете або використовувати речі, створені з їх застосуванням, або ні. А й тому, що важко навіть уявити, що в глибинах ДКНІІ або Мінмолодьспорту, де люди здебільшого працюють на піратській Windows XP, знайдуться фахівці, здатні запропонувати новий стандарт у галузі ІКТ. Такий, що його було б прийнято не те що IEEE CS або ISO, а й просто розробниками того або іншого дистрибутиву (цілісного, функціонально-закінченого набору програмного забезпечення).

Той же пункт "завдання", "захід" 6: "Підготовка глосарія з питань використання програмного забезпечення з відкритим кодом". 150 тис. грн бюджетних коштів; 50 - "з інших джерел".

Відповідний глосарій існує в готовому вигляді у межах статті 1 т. зв. "Законопроект Олійника-Сироти", зареєстрованого на сервері Верховної Ради України за номером 2065 від 01.12.2005 року...

Що далі – то щільніше. Завдання 3, "Створення інфраструктури формування та розповсюдження ПЗ з відкритим кодом" (неначе б то її за фактом ще немає...)

Захід 1: "створення програмно-технічного комплексу реєстру ПЗ, що використовується в органах державної влади" та забезпечення його ведення: 750 тис. грн. Розрахунок відсутній, ціль формування реєстру – невідома...

Захід 4: "створення дослідного комплексу (!) для тестування ПЗ з відкритим кодом: 560 тис. грн". Що таке "дослідний комплекс" – ніхто не знає (адже йдеться явно не про ядерну фізику).

Захід 5: "створення необхідних програмних засобів для забезпечення доступу до устаткування, визначення сумісності та переведення даних... 1270 тис. грн." Про яке "устаткування" йдеться – невідомо. А якщо немає характеристик "устаткування" – то як ви визначили суму?!.. Що ж до форматів – то практично усі відомі "вільні" формати на рівні офісних документів вже є вільно конвертованими до пропрієтарних, і навпаки, а переважна більшість прикладних програм "вільної/відкритої" юридичної форми просто працює із загальногалузевими форматами.

Захід 6: "створення... інфраструктури формування і розповсюдження ВВПЗ. 300 тис. грн." Вже створено: будь-який дистрибутив ВВПЗ вільно розповсюджується в Інтернет.

Захід 7: "розроблення та впровадження єдиних прикладних програмних інтерфейсів, плагінів та динамічних бібліотек, необхідних для забезпечення сумісності форматів даних". 800 тис. грн. Існує в готовому вигляді, або створюється і модифікується постійно у рамках вже наявних дистрибутивів...

Особливо цікавими, з огляду на текст таблиці, видаються намагання узяти бюджетні гроші на якісь суто-технічні речі: окремі програмки, якими користується весь світ, абощо. Ось "захід" 8 із "завдання" 3: "створення автоматизованої системи проведення моніторингу наявності в Інтернеті оновлених версій програмного забезпечення з відкритим кодом": 250 тис. грн. Халепа, звичайно, але кожен системний адміністратор знає, що це програми "ушп", "арт" та подібні... Або такі заходи пункту 5 "Створення локалізованого базового комплекту програмного забезпечення" (на загальну суму, до речі, 14.202 тис. грн бюджетних коштів!), як "розробка засобів інсталяції", що не лише присутні в будь-якому дистрибутиві ПЗ, а й частково є тими самими "стандартами", що їх Держінформнауки та ДССЗЗІ вже заплановано "розробити" у ряді попередніх пунктів...



Окремі пункти програми взагалі майже неможливо зрозуміти. Так, за завданням 7 "Створення загальнодержавного фонду алгоритмів ПЗ з відкритим кодом" автори держпрограми беруться "створити сховище (!) еталонного зберігання проектної документації щодо ПЗ з відкритим кодом... на 800 об'єктів". Коштує ця титанічна праця 1.626 тис. грн. Але про що тут йдеться? "Об'єкти" – це що таке людською мовою? "Програмні пакети"? Тож мова про створення стандартного програмного репозиторію? Що ж, багато хто з розробників буде аж ніяк не проти узяти таку суму за специфічне налагодження кількох серверів... Але тоді чому "на 800 об'єктів"? Адже стандартна кількість програмних пакетів, доступних для будь-якої "вільної" операційної системи – не менше 3 – 5 тисяч... До того ж, дійсно логічне завдання створення такого репозиторію вже вирішено вище за текстом "таблиці", і за окрему суму.

До переліку, звісно, необхідно додати вивчення чиновниками "світового досвіду" (на це візьмуть близько 3 млн. грн.: на місці ж, за загальнодоступними у Мережі статистикою, книжками та спеціальними дослідженнями Євросоюзу, вивчити той досвід, вочевидь, неможливо)...

\* \*\*

Загальну мету програми сформульовано цілком благородно: "по-ступово забезпечити перехід органів державної влади на програмне забезпечення з відкритим кодом, що дозволить скоротити бюджетні витрати на легалізацію та придбання пропрітарного програмного забезпечення. Бюджетні кошти будуть переспрямовані на підтримку вітчизняних розробників програмного забезпечення..."

Проте основним результатом будь-якої державної цільової програми є не гасла, а дії, а у фінансових і планових документах – повна прозорість.

Звичайно, документи програми вже дістали належного висновку Громадської Ради при ДАНІ [4], яка, практично, порекомендувала згорнути всі напрямки витрат окрім навчання чиновництва та курсів ІТ-сертифікації. І це логічно: адже з наведених прикладів ясно, що програму ДЦНТП необхідно докорінно перероблювати. Проте адекватного висновку на зауваження Громадської Ради з боку ДАНІ не надійшло: в отриманій відповіді йшлося здебільшого про те, що виконання пропозицій Громадської Ради планом програми "не передбачається" [5]...

Хотілося б помилятися у негативних висновках – проте, з огляду на доступні документи програми, доводиться із жалем констатувати, що поки що бажаної прозорості у політиці держави на ринку ІКТ, і конкретно – в рамках широко анонсованої "ДЦНТП використання в органах державної влади програмного забезпечення з відкритим кодом" немає.

І не намічається навіть позитивної тенденції...

### **Література:**

1. *Постанова Кабінету міністрів України "Про затвердження Державної цільової науково-технічної програми використання в орга-*

- нах державної влади програмного забезпечення з відкритим кодом на 2012 - 2015 роки" [Електронний ресурс]. – Режим доступу: [http://search.ligazakon.ua/l\\_doc2.nsf/link1/KP111269.html](http://search.ligazakon.ua/l_doc2.nsf/link1/KP111269.html)
2. Free Software Foundation [Електронний ресурс]. – Режим доступу: <http://www.fsf.org/>
3. Business Software Alliance. [Електронний ресурс]. – Режим доступу: [http://www.bsa.org/country.aspx?sc\\_lang=uk](http://www.bsa.org/country.aspx?sc_lang=uk)
4. Протокол №5 засідання Громадської ради при Держінформнауки від 18 січня 2012 року [Електронний ресурс]. – Режим доступу: <http://dki.org.ua/node/120>
5. ДАНИІ. Лист № 1/20-2-36 від 27.02.12 [щодо розгляду пропозицій громадської ради, надісланої листом від 26.01.2012 р.]

### **Використання крос-платформного інструментарію розробки програмного забезпечення Qt для створення навчальних програм.**

Рудий М.Ф.

*Львівський національний університет імені Івана Франка, [rudym@meta.ua](mailto:rudym@meta.ua)*

In a lecture give the example of the use of cross-platform tool of development of Qt for creation of educational shell of numeral integration of the system of differential equalizations of first-order.

Вже більше ніж 30 років тому програмування та комп'ютерні технології почали використовувати для наукових і навчальних цілей. Але широке застосування програмування почало набувати тільки в останні 10 років. Сьогодні майже у всіх навчальних закладах використовують різне програмне забезпечення для навчання учнів чи студентів. Але велика кількість програм, які використовують для навчання, мають поганий графічний інтерфейс, незрозумілий користувачеві. Тому з розвитком технології почали розробляти якісні графічні інтерфейси. Сьогодні великою популярністю користується оболонка Qt, яка дає змогу створювати потужні програмні засоби з інтуїтивним графічним інтерфейсом.

Qt (варіант вимови від розробників *cute* — к'ют) — крос-платформний інструментарій розробки програмного забезпечення мовою програмування C++. Дає змогу запускати написане за його допомогою програмне забезпечення на більшості сучасних операційних системах шляхом простої компіляції тексту програми для кожної ОС без зміни початкового коду.

Qt дає змогу робити платформонезалежне програмування реальністю, не обмежуючи свій вибір і творчі можливості. Qt надає користувачам те, чого вони хочуть: швидкі програми, які мають хороший інтерфейс і пра-

цюють належним чином, а розробникам програм те, чого вони бажають: середовище, яке надає можливість писати менше коду, створюючи при цьому більше. Завдяки цьому програмування стає цікавішим.

Основні характеристики оболонки Qt:

- використання Meta Object Compiler — попередньої системи обробки початкового коду (загалом, Qt, це бібліотека не для чистого C++, а для його особливого діалекту, з якого і «перекладає» МОС для подальшої компіляції будь-яким стандартним C++ компілятором). МОС дає змогу у багато разів збільшити потужність бібліотек, вводячи такі поняття як слоти (slots) і сигнали (signals);
- Qt комплектується графічним середовищем розробки графічного інтерфейсу QTDesigner, що дає змогу створювати діалоги і форми «мишею»;
- ідеологія створення форм в Qt базується на використанні менеджерів розташування, які створюють «гумовий» дизайн, при якому розмір і розташування елементів форм визначаються автоматично, що значно прискорює розробку графічного інтерфейсу;
- в поставці Qt є «Qt Linguist» — могутня графічна утиліта, що дає змогу спростити локалізацію і переклад вашої програми багатьма мовами;
- «Qt Assistant» — довідкова система Qt, що спрощує роботу з документацією, а також дає змогу створювати крос-платформну довідку для розробленого на основі Qt програмного забезпечення;
- Qt також може бути використаним в багатьох інших мовах програмування.

Прикладом навчальної програми, створеної за допомогою оболонки Qt є програма “Ds”, яку я б хотів розглянути детальніше. “Ds”- це програма для чисельного інтегрування систем диференціальних рівнянь першого порядку. Програма поширюється з ліцензією GPL.

Основні можливості програми:

- користувач може створювати опис нових систем дифрівнянь, ввівши назву та розмір системи рівнянь(рис. 1.), або відкривати із файлу збережені системи;

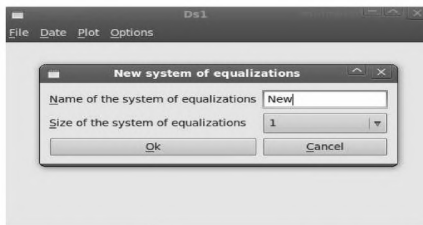


Рис. 1. Створення опису нової системи дифрівнянь

- створені описи систем дифрівнянь можна зберігати у файл;
- після розрахунку програма автоматично виводить на графік залежність знайдених розв’язків від часу(рис. 2.);

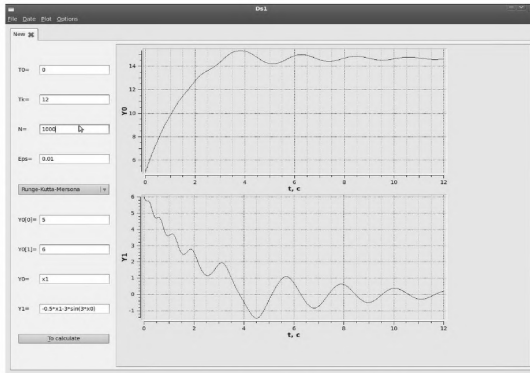


Рис. 2. Головне вікно програми

- користувач має змогу вивести на графік будь-яку іншу залежність, для цього йому потрібно скористатись пунктом меню, де він має ввести відповідну залежність(рис. 3.);

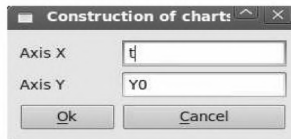


Рис. 3. Вікно меню “Графік”

- користувач має змогу зберегти дані знайдених розв’язків у файл;
- у програмі є можливість змінювати мову діалогу (українська, англійська, російська, німецька) .

Таким чином, оболонка Qt дає змогу створювати якісні програмні засоби для навчальних цілей. Перевагою створених програм за допомогою Qt є їхня простота у використанні, простий та інтуїтивний графічний інтерфейс, а також можливість зміни мови діалогу, що дає змогу працювати на даних програмах користувачам, які володіють тільки певною мовою.

## ***Дистрибутив ZENTYAL для побудови серверів для малого та середнього бізнесу*** *Сало М.Я., Ванькевич Д.*

*Львівський національний університет імені Івана Франка, факультет електроніки, e-mail: [mishas89@gmail.com](mailto:mishas89@gmail.com)*

Zentyal is a free, open source, released under the GNU General Public License (GPL) Linux-based Small Business Server, that offers easy and efficient computer network administration for small and medium-size businesses. Considered as alternative to Windows Small Business Server.

Малий і середній бізнес (small and medium businesses [SMBs]) сьогодні у всьому світі все частіше звертають свою увагу на інформаційні технології, які стають основою ефективної роботи підприємств. Однак активне використання ІТ потребує фінансових ресурсів для їх впровадження. І найчастіше власники невеликого бізнесу приймають рішення на користь недорогого ІТ-продукту.

Найдоступніший варіант для невеликих або тільки заснованих компаній - це, безумовно ОС Windows Small Business Server - серверне рішення, призначене для такого сегменту ринку. Але думаючи так, підприємці не беруть до уваги те, що ліцензія на використання такої ОС вимагає значних витрат. А чому б не використати безкоштовні альтернативи?

У зв'язку з цим, доцільно розглянути Linux як більш цікавий, альтернативний сервер SMB. Одним із таких Linux дистрибутивів є Zentyal на основі пакетної бази Ubuntu 10.04 LTS. Мета якого "зближення" Linux до малого та середнього бізнесу, щоб дозволити йому максимально використовувати свій потенціал, як корпоративного серверу.

Zentyal - це пакет серверного програмного забезпечення з відкритим вихідним кодом, орієнтований на малі й середні корпоративні мережі. Zentyal може виступати в ролі мережевого шлюзу, єдиного центру безпеки мережі, Office Server, сервера уніфікованих комунікацій або комбінувати будь-які з перерахованих функцій (рис.1).

Під час розробки Zentyal акцент був поставлений на зручність використання, тому управління всіма налаштуваннями роботи дистрибутива проводяться через веб-інтерфейс, в рамках якого об'єднано 35 різних відкритих платформ для управління мережею.

Підтримується швидка організація роботи шлюзу, брандмауера, поштового сервера, VoIP (Asterisk), VPN-сервера, проксі (Squid), файлового сервера, систем обміну миттєвими повідомленнями та моніторингу, сервера для резервного копіювання, системи забезпечення мережної безпеки (Unified Threat Manager) та інших.

Особливої уваги заслуговує інтегрована система групової роботи Zafafa, що позиціонується як заміна MS Exchange, випущена під відкри-

тою Affero GPL3 та комерційної ліцензії, яка надає стандартні можливості зв'язків Exchange/Outlook: пошту, адресні книги, завдання, нотатки, календарі, персональні/публічні каталоги, планування заходів.

Дистрибутив Linux - Zentyal, як і будь-який із множини UNIX\LINUX працює стабільно і не вимагає значних апаратних потужностей, тому знижуються фінансові витрати на розгортання сервера.

Zentyal забезпечує низький поріг входження для новачків в адміністрування ОС Linux, завдяки правильно продуманому підходу до проектування інструментів управління.

Принцип побудови дистрибутива Zentyal - модульний, завдяки чому на його основі можна налаштувати сервер практично будь-якого призначення, починаючи від найпростішого файл-сховища або контролера домену і закінчуючи SIP-шлюзом. За ідеологією роботи до числа близьких аналогів Zentyal можна віднести Clear OS від Red Hat.



## **Поштовий сервер IREDMAIL в бізнесі** *Сейдаметов Г. С.*

*Кримський інженерно-педагогічний університет, [girey2004@mail.ru](mailto:girey2004@mail.ru)*

The main advantage of large and small businesses is the use of Open Source solutions for deploying and managing e-mail.

На сьогоднішній день не можна уявити організацію з великою кількістю співробітників без електронної пошти. В таких організаціях кількість поштових облікових записів досягає сотні, для їхнього існування необхідно розгорнути власний поштовий сервер. Єдине, що відрізняє в такій ситуації велику компанію від малої - це вимоги до поштової системи і фінансові можливості. Зазвичай великі підприємства можуть дозволити собі придбати комерційну поштову систему і оплатити послуги третіх осіб за її інтеграцію, а в перспективі - знайти достатньо коштів для підтримки системи власними фахівцями. Для невеликих організацій подібні витрати нерідко стають недозволеною розкішшю: навіть закупівля комерційного пакета ПЗ є надто дорогою справою, а вже оплата фахівців - і зовсім неможливою.

Тому найчастіше вдаються до безкоштовного ПЗ, що працює в складі ОС поширюваних на основі GNU / GPL (Linux, FreeBSD).

Прикладом такого сервера є iRedMail - повнофункціональний поштовий сервер. Всі компоненти є безкоштовними та відкритими. До його основних переваг належать [1]:

Контроль над власними даними. Всі дані розташовані на сервері компанії;

Легкий, швидкий в розгортанні (менше однієї хвилини). Повнофункціональна Веб-панель адміністратора - iRedAdmin Pro за допомогою якої можна завжди налаштувати поштовий сервер вручну; Postfix: SMTP-сервіс; Apache: Веб-сервер; MySQL: зберігання даних програм та/або поштових скриньок; OpenLDAP: зберігання облікових записів електронної пошти.

В додаток, можна зазначити, що даний поштовий сервер, найпростіший і надійний з усіма функціями необхідними для адміністраторів.

### ***Література:***

*1. iRedMail – Free, Open Source Email Server Solution – [www.iredmail.org](http://www.iredmail.org)*

## **Підтримка шкільної математики вільно поширюваним ПЗ** Семеніхіна О.В., Друшляк М.Г.

Сумський державний педагогічний університет імені А. С. Макаренка, [marydru@mail.ru](mailto:marydru@mail.ru)

In the article the possibility of the use of the wide-spread package of dynamic geometry Geogebra is considered, its use in Ukrainian schools is grounded.

Інформатизація суспільства зумовила тотальне впровадження інформаційних технологій в освіту. Навчальний процес сьогодні обов'язковою складовою містить комп'ютерні технології, серед яких підтримка навчального матеріалу презентаціями, електронними підручниками, тестовими технологіями тощо є невід'ємною складовою шкільного навчання. Разом з цим в українських школах, як правило, використовується операційна система Windows, яка крім того, що вимагає ліцензування, ще є нестійкою по відношенню до вірусів. Також варто зазначити, що рівень комп'ютерної техніки, яка використовується у пересічній українській школі навіть не середній, а занижений порівняно з сучасними інформаційними технологіями (нове покоління комп'ютерної техніки, відповідне до нього програмне забезпечення тощо). Всі комерційні програми досить дорогі і більшість шкіл не спроможні придбати достатню кількість ліцензій. Тому часто можна почути про використання вільно поширювальних операційних систем, які не вимагають великих апаратних ресурсів, стійкі до вірусів і постійно оновлюються.

Через це актуальним наразі є питання про встановлення вільного ПЗ, визначення тих програмних засобів, які варто використовувати на уроках з навчальною метою і поряд з цим до інтерфейсу яких вже звикли і напруцювання на яких можна «безболісно перенести» в іншу операційну систему з мінімальними витратами часу.

Нами були проаналізовані програми підтримки шкільного навчання математики. Зокрема, аналіз стосувався програм динамічної геометрії на предмет можливості їх безпосереднього використання в ОС Linux. Результати дослідження представлено в таблиці 1.

Таблиця 1.

Програма	Операційна система	Розповсюдження
Cabri	Windows, Mac OS X	Ліцензія
Geometers' SketchPad	Windows, Linux, Mac OS X (Java)	Ліцензія
GeoGebra	Windows, Linux, Mac OS X	Вільно
Cinderella	Windows, Linux, Mac OS X (Java)	Ліцензія
GeoNext	Windows, Linux, Mac OS X	Ліцензія
Живая Геометрия	Windows, Mac OS X	Ліцензія
Матем. конструктор	Windows, Linux, Mac OS X	Ліцензія
DG	Windows	Ліцензія
GRAN2d	Windows	Ліцензія
Dr Geo	Linux	Вільно



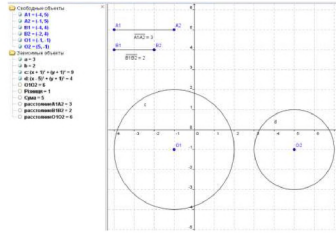


Рис.1

Висновок. Кола не перетинаються, тобто не мають спільних точок.

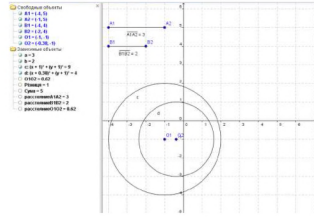


Рис.2

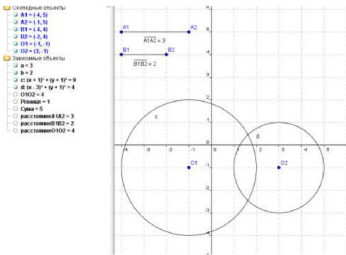


Рис.3

Висновок. Кола перетинаються.

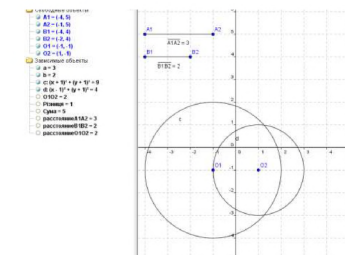


Рис.4

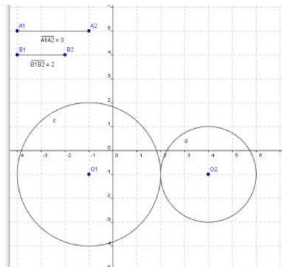


Рис.5

Висновок. Кола торкаються, зовнішній дотик і внутрішній дотик.

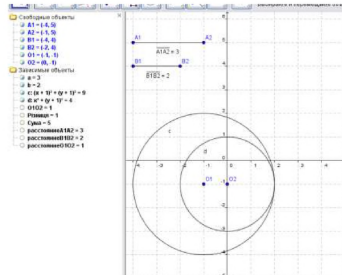


Рис.6

**Література**

1. Інститут GeoGebra Харків, Україна – <http://kafinfo.org.ua/geogebra>.
2. Інститут GeoGebra Чернігів, Україна – <https://sites.google.com/site/geogebrachernigiv/>

## **Застосування ВПЗ для підготовки спеціалістів зі створення суднових інформаційних системи і комп'ютерних мереж**

*Шапо В.Ф., Шевченко Т.І.,*

*Одеська національна морська академія, Одеська державна академія будівництва та архітектури, stani@te.net.ua, avk@te.net.ua*

Possibilities and teaching experience of free and open source OpenSCADA system creating and exploitation basics for maritime and river transport specialists during creation and exploitation of ship automatized systems, information systems, computer networks and field buses are analyzed.

За останні роки в світовій економіці суттєво зросла роль морських перевезень вантажів, не зважаючи на фінансово-економічні кризи, що стають частішими та потужнішими. Нові морські та річкові судна усіх типів та класів, що будуються замість морально та фізично застарілих, обладнуються сучасними системами та засобами автоматизації, мікропроцесорними системами, судовими комп'ютерними мережами, що дозволяють забезпечити не тільки роботу с судовими документами та обмін даними між представниками екіпажу, а й повне контролювання усіх технологічних параметрів та автоматичне управління ними. У зв'язку із збільшенням кількості та ускладненням судового електронного обладнання, а також неможливістю вирішення основних задач без його використання 24 червня 2010 року у Манілі були прийняті додатки до міжнародної конвенції STCW/ПДНВ (Standards of Training, Certification and Watchkeeping / Підготовка і Дипломування моряків і Несення Вахти), яка була впроваджена у 1978 році та модифікована у 1995 році, що стосуються саме відповідних знань та вмінь щодо роботи з судовим електронним обладнанням, комп'ютерною технікою та судовими комп'ютерними мережами. Для вчасного підготування кваліфікованих кадрів в Одеській національній морській академії на старших курсах введено відповідні навчальні дисципліни, у тому числі «Суднові комп'ютерні мережі» на 4-му курсі спеціальності «Електричні системи і комплекси транспортних засобів».

Важливим засобом комплексної автоматизації технологічних процесів в усіх галузях промисловості та виробництва є SCADA-системи (Supervisory Control And Data Acquisition, системи моніторингу та збирання даних-). Ці системи забезпечують зв'язок між вельми широким спектром промислового обладнання різних класів і сфер використання й дуже широким спектром комп'ютерних систем, використовуючи ряд стандартних методів, інтерфейсів і протоколів. Визначимо найбільш важливі з них: використання Веб-серверів на базі технологій AJAX Web 2.0 й аналогічних; програмування сценаріїв користувача на багатьох мовах програмування (TCL\Tk, Visual Basic, C++ і т. ін.); підтримка найбільш поширених технологій промислових мереж (Modbus, Siemens, Interbus, M-Bus, EN62056, LON, CAN, ASI, Profibus і т. д.); організація запитів до баз

даних (БД) з використанням мови SQL; інтерфейси DDE, ODBC, OPC клієнт-сервер, VBA/Automation, OLE та Active X, що дозволяють обмінюватися даними з програмними додатками різних розробників; наявність широкого вибору драйверів, що забезпечують зв'язок з логічними контролерами, що програмуються (ПЛК) ведучих виробників.

SCADA-системи мають інтуїтивно зрозумілий процес графічного програмування; широкі можливості збирання, обробки й аналізу даних, управління віртуальними й реальними приладами; різноманітні форми генерації звітів й візуалізації графіків; передавання даних з використанням технологій USB, Bluetooth, Ethernet, Wi-Fi; можливості підключення десятків модулів АЦП й ЦАП, аналізаторів спектру, вольтметрів й генераторів; множину готових шаблонів додатків; працюють на сучасних операційних системах (ОС) сімейств Windows, Unix, Linux, а також реалізаціях ОС реального часу.

Надзвичайно широкі й можливості їх застосування: автомобільна промисловість; телекомунікації; аерокосмічна промисловість; напівпровідникова промисловість; нафтовидобувача промисловість; розробка й виробництво електроніки; управління технологічними процесами; біомедицина і т. д. Найбільш популярними є системи Ant Studio (Польща), Каскад-САУ (Росія), АСМО (Україна), Open SCADA (Україна), Citect (Росія), Free SCADA (міжнародний проєкт), IGSS (Данія), PCVue (Франція), Clear SCADA (Канада), Simp Lite (Росія), S3 (Росія), Master SCADA (Росія), Trace Mode (Росія), ZETView (Росія), Wonderware (США/Росія), Factory Link (США), InTouch (США/Росія), RealFlex (Ірландія), RSView (США), КРУГ 2000 (Росія) й ряд інших, що створені у багатьох інших країнах.

Система Open SCADA відноситься до категорії FOSS, що дозволяє залучити велику кількість розробників та зацікавлених до її розробки, тестування, розвитку, розповсюдження й використання. Основні характеристики системи: відкритість; надійність; гнучкість; можливості масштабування; багатоплатформеність; безпека; фінансова доступність; наявність сучасного інтерфейсу управління. OpenSCADA може бути використана для виконання типових функцій SCADA та в близьких галузях інформаційних технологій: на промислових об'єктах як повнофункціональна SCADA-система; у вбудованих системах як середовище виконання; для побудови моделей технологічних, хімічних, фізичних, електричних процесів, що особливо важливо для підготовки спеціалістів; на комп'ютерах, серверах і кластерах для підготовки спеціалістів; представлення і архівації інформації.

В якості ОС для розробки й використання розробниками обрано ОС Linux, яка є стандартною POSIX-сумісною та є оптимальним рішенням в питаннях надійності, гнучкості, доступності й т. д. Система є модульною. В залежності від того, які модулі підключено, система може виконувати функції різноманітних серверів або клієнтів клієнт-серверної архітектури будь-якої складності. Система OpenSCADA має наступні підсистеми.



1. Підсистема безпеки забезпечує перевірку прав користувачів та їх груп на можливості доступу до елементів системи.
2. Модульна підсистема баз даних забезпечує доступ до баз даних.
3. Модульна підсистема транспортів забезпечує зв'язок із зовнішнім середовищем шляхом використання різноманітних комунікаційних інтерфейсів.
4. Модульна підсистема комунікаційних протоколів обміну тісно пов'язана з підсистемою транспортів й забезпечує підтримку різноманітних протоколів обміну із зовнішніми системами.
5. Модульна підсистема збирання даних забезпечує збирання даних від зовнішніх джерел (контролерів, датчиків й т. ін.) та може надавати середовище для написання генераторів даних (моделі, регулятори й т.д.).
6. Модульна підсистема архівів містить архіви повідомлень й архіви значень. Спосіб архівування визначається алгоритмом, що заложений в модулі архівування.
7. Модульна підсистема інтерфейсів користувача містить функції інтерфейсів користувача.
8. Підсистема управління модулями забезпечує контроль над модулями.
9. Модульна підсистема спеціальних функцій містить функції, які не увійшли в інші підсистеми (у теперішній час це функції тестування).

Проаналізовані вище модульні підсистеми можуть отримати розширену функціональність шляхом підключення модулів відповідного типу.

Модульне ядро системи OpenSCADA реалізовано як статичні й використовує сумісно бібліотеки. Це дозволяє вбудовувати функції системи в існуючі програми, а також створювати нові програми на основі модульного ядра системи OpenSCADA. Однак модульне ядро є самодостатнім й може бути використано нескладною запускаючою програмою.

Модулі системи OpenSCADA зберігаються в динамічних бібліотеках. Кожна динамічна бібліотека може містити багато модулів різного типу. Наповнення динамічних бібліотек модулями визначається функціональним зв'язком між самими модулями. Динамічні бібліотеки допускають гарячу заміну, що дозволяє виконувати оновлення модулів безпосередньо у процесі роботи. Метод зберігання кода модулів в динамічних бібліотеках є основним для системи OpenSCADA, оскільки підтримується практично всіма сучасними ОС.

Перший досвід викладання основ роботи з системою OpenSCADA показав достатню простоту та швидкість засвоєння матеріалу, непідробне зацікавлення й гарні перспективи її використання для початкового навчання та подальшого підвищення кваліфікації спеціалістів морського та річкового транспорту, чиїми посадовими обов'язками є створення та експлуатація судових систем автоматизації, інформаційних систем і комп'ютерних мереж.

**Виконання завдань розподіленої обробки зображення під управлінням системи CONDOR**  
*Шийка Ю.Я., Шувар Р.Я.*

*Львівський національний університет імені Івана Франка,  
факультет електроніки, e-mail: shiyka@mail.lviv.ua*

The peculiarities are considered of implementation of distributed computing system under the management of Condor. The project is described of distributed segmentation of images taken by remote sensing system.

Технології паралельних та розподілених обчислень дозволяють виконувати ресурсозатратні завдання та обробляти великі обсяги даних, використовуючи обчислювальні можливості комп'ютерів об'єднаних в обчислювальну мережу (грід, кластер). Такі обчислювальні мережі мають різну топологію і інфраструктуру, залежно від завдань і вимог, які до них ставляться. Грід або грід-інфраструктура – це розподілена програмно-апаратна комп'ютерна мережа з особливою організацією обчислень і управління потоками завдань і даних.

В лабораторії комп'ютерних інформаційних технологій для розподілених обчислень впроваджено систему [Condor](#) під управлінням кластерної системи [Rocks](#). Виконання незалежних підзавдань відбувається в кластерній комп'ютерній мережі, що складається з 14 двохядерних обчислювальних вузлів (нод) та вузла управління. Система Condor дає можливість створювати черги завдань, які виконуються відповідно до заданих правил у строго визначеному чи невизначеному порядку на тих чи інших вузлах кластера.

Розподілені обчислення полягають в обробці незалежних між собою підзавдань для отримання одного кінцевого результату. Розроблено методику розподіленої сегментації растрових зображень ДЗЗ. Згідно цієї методики обробка зображень здійснюється в чотири етапи.

Перший етап – формування незалежних підзавдань. Створено програму генератор незалежних підзавдань, яка велике вхідне зображення нарізає за принципом мозаїки на менші зображення підзавдань; крім того до кожного підзавдання створюється файл з інформацією про розміри і позицію зображення підзавдання у вхідному зображенні а також параметри для виконавчої програми.

Другий етап – обробка незалежних завдань виконавчою програмою в розподіленому режимі. Створена адаптована до розподілених обчислень виконавча програма виконується на обчислювальних нодах. Ця програма за двома вхідними файлами створює два вихідних: перший файл – сегментоване зображення, другий файл – інформація про розміри і позицію зображення у великому вхідному зображенні. Для того, щоб додати набір підзавдань в чергу завдань Condor створено файл опису завдань Condor, в якому вказуються вибрані параметрами та середовище виконання зав-

дань, шлях та вигляд вхідних та вихідних файлів, параметри доступу і пересилання файлів між обчислювальними вузлами і т.п.

Третій етап – формування кінцевого результату після виконання всіх підзавдань. Для цього створено програму, яка утворює результуюче зображення шляхом вставлення зображень результатів підзавдань на відповідні місця у результуючому великому зображенні.

Четвертий етап – постпроцесінг. Після склеювання незалежно сегментованих зображень на границях зображень утворюються зайві границі розділу сегментів. Для того, щоб отримати кінцеве зображення сегменти розділені на краях потрібно об'єднати. Створено програму постпроцесінгу, яка здійснює кінцеву обробку сегментів у результуючому зображенні за інформацією про властивості та характеристики сегментів утворених в результаті сегментації незалежних підзавдань.

Таким чином, в результаті роботи створено систему розподіленої сегментації растрових зображень. Ця система дозволяє здійснювати обробку великих об'ємів даних, що є неможливо на окремих комп'ютерах. Шляхом організації розподілених обчислень обчислювальні потужності окремих вузлів об'єднуються для виконання одної задачі. При виконанні завдань під управлінням системи Condor використовуються тільки вільні ресурси обчислювальних вузлів, що не перешкоджає їх звичайній роботі в складі кластера паралельних обчислень. Впроваджена система є масштабованою і гнучкою. Система дозволяє виконувати різні завдання розподілених обчислень, змінювати параметри і порядок виконуваних завдань. Зміна кількості обчислювальних нод, топології та інфраструктури не впливає на працездатність проектів розподілених обчислень. Існуючі проекти легко налаштовуються відповідно до змін системи.

Отриману кластерну систему та розроблений проект розподілених обчислень використано для обробки даних дистанційного зондування території Шацького національного природного парку.

**Використання обчислювального кластера з ОС Linux у вищому навчальному закладі**  
Сідун Н.М., Чичкарьов Є.А.

*Маріуполь, Донецька область, Україна, Приазовський державний технічний університет, email: influence@meta.ua*

The contribution is devoted to the usage of open source software in teaching process at technical university during the courses of parallel programming. The peculiarities are considered of implementation of cluster system on the base of teaching laboratory, equipped with personal computers.

В теперішній час виникає нагальна потреба у вирішенні актуальних завдань фундаментальної і прикладної науки, для аналізу і дослідження яких необхідні високопродуктивні системи, що дозволяють в найкоротші терміни проводити розрахунки і моделювання процесів. Перспективним напрямком є використання обчислювальних систем на основі кластерів, що представляють собою набір обчислювальних вузлів, пов'язаних високошвидкісною мережею і об'єднаних в логічне ціле спеціалізованим програмним забезпеченням.

В даній роботі проаналізовано можливості розгортання обчислювального кластера під управлінням ОС Linux, представлені оцінки швидкості обчислень при використанні різних компіляторів стосовно вирішення рівнянь з приватними похідними і деяких тестових завдань.

Обчислювальний кластер - це набір обчислювальних вузлів, призначених для вирішення розподілених завдань в паралельному режимі, з продуктивністю в сотні мільярдів операцій в секунду. Кожен обчислювальний вузол має свою оперативну пам'ять і працює під управлінням своєї операційної системи [1,2]. Програми, які виконуються на кластері, повинні бути адаптовані для використання в кластерному оточенні, для того щоб отримати переваги в продуктивності.

Багато з існуючих у світі обчислювальних кластерів працюють під управлінням операційних систем сімейства UNIX (у тому числі систем з відкритим кодом - Linux і FreeBSD). Серед них є як вільно поширювані системи, так і комерційні системи [1,2]. Використання UNIX-платформ може дати деякі переваги у вигляді оптимізації початкових витрат на придбання програмного забезпечення і реалізувати максимальну продуктивність. Крім цього, частина спеціалізованого програмного забезпечення для обчислювальних кластерів працює тільки на UNIX-платформах. Розробка програм для обчислювального кластеру здійснюється за допомогою вільно розповсюджуваних бібліотек і компіляторів, існують утиліти, що дозволяють забезпечити віддалений моніторинг завантаженості обчислювальних вузлів і доступ для копіювання даних і запуску виконуваних модулів. Основою кластера є не стільки операційна система, скільки комунікаційна середовище, що забезпечує можливість части-

нам паралельної програми, що виконується на різних комп'ютерах, ефективно взаємодіяти між собою. Відомі засоби для побудови кластера мають реалізації як для ОС сімейства UNIX (Linux, FreeBSD та ін.), так і для ОС Майкрософт [1].

Побудова кластера на базі Windows HPC Server 2008 або Windows Compute Cluster Server 2003 спрощує завдання його адміністрування (фактично від адміністратора потрібно тільки налаштувати керуючий вузол, забезпечивши його роботу в режимі контролера домена і запустивши необхідні служби). Дані рішення дозволяють забезпечити інтеграцію кластерного ПЗ з іншою інфраструктурою організації. Однак плюсом Linux в якості кластерної ОС є «прозорість» для користувача і системного адміністратора, що дозволяє швидше і простіше вирішувати всі виникаючі проблеми [1,2]. Оскільки бібліотеки обміну повідомленнями при реалізації паралельних обчислень є кросплатформеною, то вибір операційної системи (Windows vs Linux) не визначає працездатність кластера. Однак слід врахувати той факт, що Linux є помітно менш ресурсомісткою системою. Наприклад, при використанні PelicanHPC GNU Linux система займає в оперативній пам'яті не більш 40Мб (за результатами тесту). Вся інша пам'ять доступна паралельній програмі. Це дуже важливий фактор в тому випадку, коли кластер використовується з метою моделювання процесів на як можна більш докладній сітці. Очевидно, що при використанні багатопроцесорних систем каменем спотикання стає проблема так званого «розпаралелювання алгоритмів» для програмування на багато-процесорної структурі.

Для оцінки ефективності розпаралелювання, дослідження впливу числа використовуваних ядер на час вирішення, а також аналізу швидкодії програм, скомпільованих різними засобами, в якості тестової використовувалася крайова задача Діріхле для одновимірного рівняння теплопровідності, для вирішення якої використовувалися паралельний варіант явної різницевої схеми [1,3] і метод прямих з дискретизацією по просторовій координаті [4], що дозволяє звести задачу інтегрування диференціального рівняння з приватними похідними до задачі інтегрування системи звичайних диференціальних рівнянь (ОДУ). Для інтегрування системи ОДУ використовувалася паралельна реалізація методу Рунге-Кутта [2,5]. На думку [6], дана схема забезпечує високу обчислювальну ефективність. Програмна реалізація всіх необхідних алгоритмів виконана на Фортрані, С ++ і Octave (аналог Matlab з відкритим кодом). Компіляція та збирання програм виконувалася з використанням gfortran 4.5 та g++ або Intel ® Fortran Compiler 11 та icc. Кількість використовуваних в обчисленнях ядер змінювалося від 1 до 16 (при підключенні всіх вузлів). Обчислювальний експеримент виконувався на сітках змінної розмірності (кількість вузлів варіювалося на кілька порядків).

У результаті проведення обчислювального експерименту встановлено, що за інших рівних умов (конкретні значення граничних умов, кількість вузлів сітки, алгоритм вирішення задачі (1)) час обчислення сильно зале-

жить від способу оптимізації коду з урахуванням використовуваної архітектури процесора. При виконанні розрахунків час обчислення програми, скомпільованою Intel® Fortran Compiler, виявилось менше, ніж з використанням gfortran 4.5. Зниження часу рахунки за рахунок використання компілятора Intel склало від 5-10% до 100-110%, в залежності від способу оптимізації, обраному при компіляції за допомогою gfortran. Досліджуваний кластер був зібраний з 7 системних блоків ідентичною конфігурації (DualCore Intel Pentium, 1800 MHz cache 1 Mb; RAM: 1Gb DDR2-667; 1 HDD 250Gb SATA 2) і одного керуючого блоку (2 CPU Intel Core 2 Duo P8400, 2266 MHz cache 3 Mb; RAM: 4Gb DDR2-667; 1 HDD 320Gb SATA 2). Всі вузли кластера були об'єднані в локальну мережу на основі технології Gigabit Ethernet (фізичний тип з'єднання - «зірка»), яка забезпечує пропускну здатність 16 Gbit/s. Теоретична обчислювальна потужність кластера склала 851 GFlops, а кращий результат при використанні тестів Linpack склав 634 GFlops. Для окремих експериментів використовувалася та ж конфігурація кластера, але з використанням мережі Fast Ethernet. В якості операційної системи на всіх вузлах була встановлена ОС Linux Ubuntu 11.04 i386. Випробувана також робота кластера із завантаженням PelicanHPC GNU Linux. До складу системного програмного забезпечення був використаний пакет засобів для паралельного програмування Intel Cluster Toolkit Compiler Edition (компілятор Intel C++ та Fortran; бібліотека Intel MPI Library; інструменти Intel Trace Analyzer і Intel® Trace Collector; бібліотека Intel Math Kernel; еталонні тести Intel MPI), а також паралельний відлагоджувач Total View Debugger. Як технології паралельного програмування на даному кластері були доступні технології OpenMP і MPI (mpich-1.1.0, mpich2-1.2p1, openmpi-1.3). З інших засобів розробки встановлені GNU C++, GNU Fortran.

При варіюванні числом використовуваних обчислювальних вузлів встановлено, що найбільший приріст швидкості обчислень досягається при переході з обчислень на одному вузлі до обчислень на 2-3 вузлах. Подальше збільшення кількості обчислювальних вузлів і ядер дає істотно менший відносно зменшення часу рахунку. Оптимальне по приросту ефективності обчислень число використовуваних для розрахунку ядер процесорів залежить від швидкості міжпроцесорних комунікацій. З ростом пропускну здатності мережних з'єднань (перехід з Fast Ethernet на Gigabit Ethernet) оптимальна кількість обчислювальних ядер закономірно збільшилася. Тестування методів вирішення тієї ж задачі (рівняння теплопровідності з граничними умовами Діріхле) з використанням відкритої реалізації m-мови Matlab - пакета Octave - показала, що за рахунок векторизації і распараллеівання швидкість виконання інтерпретуються програм на Octave цілком порівнянна зі швидкістю виконання скомпільованих програм на C++ (навіть при використанні компілятора Intel). Проте швидкість обчислень для програм на m-мові виявилася досить чутливою до стилю програмування.



## **Література**

1. Сбитнев Ю.И. Кластеры / Ю.И. Сбитнев. – Екатеринбург, 2009. – 119 с.
2. Высокопроизводительные вычисления на кластерах: Учебн. пособие/ Под ред. А.В. Старченко. – Томск: Изд-во Том. ун-та, 2008. – 198 с.
3. Воеводин В. В. Параллельные вычисления / В. В. Воеводин - СПб.: БХВ - Петербург, 2002.- 608 с.
4. Самарский А.А. Введение в численные методы /А.А.Самарский. – Спб.:Лань, 2009. – 288 с.
5. Ващенко Г.В. Параллельная реализация явных методов Рунге-Кутты / Г.В. Ващенко, Е.А. Новиков. - Вестник Красноярского государственного аграрного университета. – 2010.- № 2. – С.14-18.
6. Швачич Г.Г. О концепции неограниченного параллелизма в задачах теплопроводности / Г.Г. Швачич, А.А. Шмукин // Восточно-Европейский журнал передовых технологий. - 4 (10). - 2004. – с. 132-135.

## **Новий підхід до створення INITRAMFS**

*Сітало В.С., Ясько М.М.*

*Дніпропетровський національний університет імені Олеся Гончара, факультет прикладної математики, [root@stvad.org](mailto:root@stvad.org)*

Considered the principles of operation, uses and benefits of live systems. Proposed a new approach to the construction of the initial file system and described the benefits of this approach. Described the steps necessary to implement the approach. Approach demonstrated on the example of Puppy Linux.

Одним із засобів ширшого розповсюдження вільного програмного забезпечення, а конкретніше для ширшого ознайомлення користувачів з операційними системами GNU/Linux, є використання LiveCD. LiveCD – це операційна система, яка завантажується зі змінного носія (CD, DVD, USB-накопичувача і т.п.) та не вимагає для свого функціонування встановлення на жорсткий диск. Цим же поняттям позначають і носії з такими ОС (в залежності від носія розрізняють LiveCD, LiveDVD і LiveUSB).

LiveCD дозволяють швидко почати роботу з комп'ютером, уникаючи тривалого процесу установки ОС в сталу пам'ять (вінчестер). Запуск LiveCD звичайно триває декілька хвилин, в той час як перший запуск (встановлення) традиційних ОС часто вимагає від однієї до декількох годин.

Першим LiveCD, який базувався на Linux, був Yggdrasil Linux, вперше виданий у 1992р. На більш пізньому етапі розвитку з'явилась можливість

виконувати завантаження ОС також і з інших змінних носіїв. В останні десять років були створені сотні таких дистрибутивів, здебільшого на базі операційних систем Linux та BSD.

Найчастіше LiveCD використовуються для ознайомлення з можливостями операційних систем. Інше застосування LiveCD - відновлення даних та / або працездатності основної операційної системи після збою. LiveCD можна застосовувати на комп'ютерах, які не мають жорсткого диска і на комп'ютерах, що мають жорсткий диск дуже невеликого обсягу.

При створенні таких систем зазвичай використовується початкова файлова система в оперативній пам'яті (initrd або initramfs). Initrd пов'язана з ядром і завантажується як його частина в процесі завантаження системи. Ядро монтує образ initrd, в якому знаходяться необхідні модулі для монтування кореневої ФС і подальшого переходу в цей корінь як основний. В initrd міститься мінімальний набір директорій і виконуваних файлів для завантаження модулів. У випадку настільної системи або сервера, initrd — це тимчасова файлова система. Час її життя досить малий і слугує вона переважно лише проміжною ланкою до кореневої ФС.

В initramfs зазвичай розміщують модулі ядра, критичні для завантаження системи (драйвери дискового контролера, драйвери кореневої файлової системи, а при необхідності й модулі підтримки NFS, LVM и RAID) і спеціальний скрипт (init або linuxrc), який і запускається на ранніх стадіях завантаження. Одна з його основних функцій полягає в пошуку розділу або файлу, де заходиться основна файлова система. Саме він збирає софтові RAID-масиви, активує LVM і т.п. Потім, коли основна файлова система знайдена, він переключає корінь системи на неї (а на ранніх стадіях роль тимчасової кореневої системи грає саме initramfs). Далі запускається системний init, який і веде подальше завантаження системи. Швидкість завантаження таких систем, як правило, не дуже висока. Для створення initramfs створені певні інструменти, такі як mkinitrd або більш сучасний Dracut.

В даній роботі пропонується новий підхід до створення початкової файлової системи, який полягає в наступному:

*1. Розмістити основну файлову систему (у стисненому вигляді як squashfs) безпосередньо у initramfs. (Уперше ця ідея була запропонована професором Барі Каулером, основним розробником дистрибутива Purru Linux. Але авторам не відомі якісь її існуючі реалізації)*

*2. Початкову команду init, яка зазвичай являє собою скрипт, написати мовою C.*

Початкова команда init повинна виконувати такі дії:

*1. Завантаження необхідних модулів ядра. В більшості випадків таких модулів буде не більше трьох (fuse, squashfs, unionfs або aufs). Завантаження модуля може здійснюватись за допомогою наступної процедури:*

```
void LoadModule(const char *ModPath){
    int fd;
    void *image;
    struct stat st;
    fd=open(ModPath,O_RDONLY);
    fstat(fd, &st);
    image = mmap(NULL, st.st_size, PROT_READ, MAP_PRIVATE, fd, 0);
    close(fd);
    init_module(image, st.st_size, "");
}

```

2. Монтування основної файлової системи у форматі *squashfs* може здійснюватись наступним чином:

```
fdev=open(devname,O_RDWR);
lodev=open("/dev/loop0",O_RDWR);
ioctl(lodev, LOOP_SET_FD, fdev);
close(lodev);
mount("/dev/loop0", "/d_root", "squashfs", 0, NULL);

```

3. Створення тимчасової файлової системи у пам'яті та сумісне монтування двох файлових систем за допомогою *unionfs* можна реалізувати наступним чином:

```
mount("none", "/d_rw", "tmpfs", 0, size);
mount("none", "/d_new", "unionfs", 0, "dirs=/d_rw=rw:/d_root=ro");

```

4. Остання дія полягає в заміні кореневого каталогу на каталог */d\_rw* та виконання команди */sbin/init* як початкового процесу:

```
chdir("/d_new");
stat("/", &st);
rootdev = st.st_dev;
stat(".", &st);
statfs("/", &stfs);
if(mount(".", "/", NULL, 8192, NULL)) return 13;
chroot(".");
chdir("/");
execl("/sbin/init", "init", (char*)NULL);

```

Якщо скопіювати таку програму з використанням *dietlibc* (стандартна бібліотека мови програмування С, що розроблена Феліксом вон Лейтнером з метою компіляції та лінкування програм у виконуваний файл якомога меншого розміру), то розмір результуючої програми буде 2-3 Кб. Її слід розмістити в кореновому каталозі разом з основною файловою системою, яка в даному випадку знаходиться у файлі *Y.sfs*. Також потрібно створити три порожні каталоги */d\_root*, */d\_rw*, */d\_new*, а в каталозі */dev* слід створити файл блочного пристрою *loop0*. Цю файлову систему слід помістити в архів *сріо*.

Цей підхід має наступні переваги:

1. Під час завантаження не потрібно шукати основну файлову систему. Це спрощує і прискорює процес завантаження;
2. Спрощення завантаження по мережі (для завантаження потрібно лише 2 файли);
3. Після завантаження носій може бути видалений, тому що вся основна файлова система буде знаходитися в оперативній пам'яті у стисненому вигляді.

Для демонстрації цього підходу були перероблені дистрибутиви Purru Linux та Fat Dog. Змінені таким чином дистрибутиви продемонстрували вищу швидкість завантаження та високу надійність в роботі.

**Технологія AJAX у інформаційних системах автоматизованого збору та опрацювання даних**  
Словінський О.В., Словінська О.Д.

*Житомирський державний університет імені Івана Франка, alexslv@zu.edu.ua*

The report highlights the main benefits and the description of methods and techniques using AJAX technology to develop information systems of automated data collection and data processing. We consider the introduction of technology in information system "Master" at Zhytomyr State University named after Ivan Franko.

Сучасний світ характеризується широким використанням систем опрацювання інформації, які ґрунтуються на застосуванні розвинутих інформаційних технологій, зокрема локальної комп'ютерної мережі. Новітні інформаційні системи об'єднують в собі інформаційні процеси, які задовольняють потреби в доступі до інформації на всіх рівнях управління та прийняття рішень. Для ефективного використання інформації важливими завданнями стають способи її накопичення, реєстрації, обробки і передачі. Сьогодні архітектура клієнт-сервер є домінуючою концепцією у створенні розподілених мережних інформаційних систем та застосувань і передбачає взаємодію та обмін даними між ними. Одним з найважливіших завдань, що вирішується при проектуванні інформаційних систем, є створення зручного і відповідного цілям інформаційної системи призначеного для користувача інтерфейсу. Звичайно задача ергономічності не формалізується, але в той же час вона є дуже істотною. Користувачі часто судять про якість системи в цілому, виходячи з якості інтерфейсу користувача. Більш того, від його якості залежить ефективність інформаційної системи в цілому.

Останнім часом широкого розповсюдження набула технологія AJAX, яка представляє собою сучасний підхід до побудови інтерфейсу користувача. Дана технологія передбачає відправку запитів на сервер у фоновому режимі і довантаження потрібних користувачу даних, при цьому не перезавантажується вся веб-сторінка. AJAX – це скоріш за все концепція використання декількох суміжних технологій, не самостійна технологія в чистому вигляді. Підхід до розробки призначених для користувача зручних інтерфейсів за допомогою AJAX поєднає декілька основних методів і прийомів:

- використання Dynamic HTML для динамічної зміни вмісту сторінки та надання інтерактивності;

- використання [XMLHttpRequest](#) для здійснення HTTP-запитів до сервера повністю не перезавантажуючи всю сторінку;
- [динамічне довантаження JavaScript коду в тег <SCRIPT>](#) з використанням моделі [DOM](#), для зміни вмісту, структури та оформлення HTML-документу;
- [динамічне створення дочірніх фреймів](#).

Використання цих підходів дозволяє створювати набагато зручніші веб-інтерфейси користувача адже для інформаційних систем характерна активна взаємодія з користувачем. AJAX – асинхронний, тому користувач може далі переглядати інформацію, поки сервер все ще обробляє запит. Браузер не перезавантажує веб-сторінку і дані посилаються на сервер «у фоні». Використання AJAX стало найбільш популярне після того, як компанія Google почала активно використовувати його при створенні своїх ресурсів, таких як Gmail, Google Maps і Google Suggest. Це підтвердило ефективність використання даного підходу та зручність роботи з системами, які реалізовані на основі технології AJAX.

Інформаційна система «Магістр», розроблена на базі Житомирського державного університету імені Івана Франка для автоматизованого оперативного контролю за надходженням заяв абітурієнтів для вступу на навчання за освітньо-кваліфікаційним рівнем магістр та спеціаліст, пройшла успішне випробування під час вступної кампанії 2010 року. Протягом року були проведені конструктивні зміни в системі, враховуючи побажання технічних секретарів. Практично весь інтерфейс змінено та вдосконалено використовуючи можливості технології AJAX. Сподіваємось, що робота з оновленою системою дозволить більш швидко і зручно шукати, вносити, змінювати, редагувати дані про абітурієнтів та формувати звіти.

### ***Огляд дистрибутивів LINUX, створених на базі DEBIAN/UBUNTU, що використовуються у сфері освіт***

*Смалько О.А.*

*Кам'янець-Подільський національний університет імені Івана Огієнка, sm-lena@mail.ru*

Leaning on electronic publications, the short review of Linux distributions which are based on Debian/Ubuntu and used in educational establishments of different countries of the world is done in theses.

У даний час з усіх дистрибутивів, що активно використовується в освіті, найбільш широко представлено сімейство проектів, що беруть свій початок від системи Debian GNU/Linux та її популярного нащадку Ubuntu.

На основі Debian в Бразилії у Федеральному університеті Парани за технічної підтримки Центру освітніх технологій розробляється дистрибутив Linux Educacional, що містить безкоштовне програмне забезпечення, у тому числі освітні програми, орієнтовані на використання в середній школах [13]. Відповідно до розпорядження Міністерства освіти Бразилії, Linux Educacional масово встановлюється у школах країни.

Створений у Республіці Індія дистрибутив BOSS GNU/Linux, доступний майже на всіх мовах, на яких розмовляють в Індії, активно використовують в державних і освітніх колах, зокрема у південному штаті Таміл Наду. Освітній варіант проекту EduBOSS постачається також з програмами і додатковими матеріалами, що можуть знадобитись вчителям під час навчання [6].

В межах проекту Debian-Edu у Норвегії на основі Debian GNU/Linux створено дистрибутив Skolelinux, який пристосований для комп'ютерного класу на базі бездискових машин для завантаження з мережі [15]. Цей дистрибутив гнучко налагоджується з урахуванням наявних апаратних засобів як для використання на нових апаратних платформах, так і для роботи на застарілому обладнанні. Використовується Skolelinux у навчальних закладах Норвегії і в школах федеральної землі Рейнланд-Пфальц Німеччини.

Технологічною основою французького проекту DoudouLinux — іграшкового Linux, призначеного для дітей від 2 до 12 років, також є дистрибутив Debian GNU/Linux. DoudouLinux містить набір простих у використанні освітніх ігор, навчальних і пізнавальних програм. Його можна запускати з Live-CD або з USB-носія [5]. Тепер розробкою дистрибутива DoudouLinux займається співтовариство добровольців з різних країн під керівництвом засновника проекту Жан-Мішеля Філіппа.

Організацією Educa у школи Швейцарії в рамках фінансованого державою проекту постачається на USB-флешках Debian-based дистрибутив Lernstick, в якому зібрано понад 140 програм навчального спрямування [7]. Існує також багато збірок, побудованих на основі дистрибутиву Ubuntu, в яких міститься велика кількість освітніх програм. Наприклад, в рамках проекту Edubuntu [2] створюються відкриті програмні засоби та цифрові матеріали, призначені для навчальних цілей і розраховані на вікову аудиторію 6-18 років.

В американському дистрибутиві Qimo for kids зібрано багато розвиваючих ігор та програм, розрахованих на дітей, починаючи з двохлітнього віку [18]. Якщо на комп'ютері вже функціонує Ubuntu, то для встановлення цього дитячого дистрибутиву потрібно лише довантажити оточення Qimo.

Ще один американський дистрибутив UberStudent, що базується на Ubuntu, розробляється командою на чолі зі Стівеном Юеном. Розрахований він на студентів і викладачів вищих навчальних закладів [3].

В Іспанії активно розвивається кілька вільних операційних систем, побудованих на основі Ubuntu і націлених на задоволення потреб освіт-



ньої системи. Такими є, зокрема, дистрибутиви LliureX (локалізований до кастильської мови і валенсійського діалекту) [10], MAX: Madrid\_Linux (пристосований для використання у навчальних закладах автономного співтовариства Мадрид) [11], Molinux (призначений для освітніх установ автономної спільноти Кастилія-Ла-Манча) [12].

У Бразилії на основі Ubuntu створено дистрибутив Poseidon Linux, орієнтований на наукове і навчальне використання. Розвивається і обслуговується Poseidon Linux групою вчених Федерального університету Піу-Гранді-ду-Сул (Бразилія) і Центру екології моря MARUM (Німеччина) [14].

У Вірменії фахівці Російсько-Вірменського (Слов'янського) Університету в результаті реконструювання Ubuntu створили дистрибутив NUR Linux (від Nur's Ubuntu Reconstructed). Версія дистрибутиву NUR Linux Education, що містить багато програм, призначених для навчання та загального розвитку дітей та дорослих, призначається для використання в освітніх закладах, а також тими, хто займається самоосвітою [1].

Спеціально для навчання дітей та опрацювання даних в освітніх установах розроблено французький дистрибутив AbulÉdu. За допомогою Live-DVD-диску з цією системою можна забезпечити безпечне відвідування сайтів Інтернет, з'єднання з шкільним сервером для виконання домашньої роботи або читання електронних листів, з'єднання клієнта AbulÉdu з сервером LTSP (Linux Terminal Server Project) [8].

В рамках освітнього проекту Guardabarranco GNU/Linux в Республіці Нікарагуа з 2008 року розробляється однойменний дистрибутив, орієнтований на використання у публічних школах на застарілій техніці з LTSP [17].

Похідна від Ubuntu литовсько-латиська Live-CD/USB-система Baltix GNU/Linux, призначена для користувачів з країн балтійського регіону, починаючи з 2006 року за рекомендацією Міністерства освіти Литви, встановлюється у школи. До Baltix GNU/Linux включено освітні застосунки, ігри та інші програми, яких немає в Ubuntu [4]. Ця система може успішно функціонувати і на застарілих комп'ютерах, яких досить багато у школах.

Останні версії таїландського дистрибутиву Suriyan, орієнтованого на використання у шкільній освіті, також базуються на Ubuntu. Його можна експлуатувати в якості Live-CD або інстальювати на комп'ютер [16].

Створений на базі Debian/Ubuntu у португальському науково-дослідницькому центрі ADETTI (підрозділі Лісабонського університету) дистрибутив Caixa Mágica, що тепер розвивається однойменною компанією, містить набір комп'ютерних навчальних програм, орієнтованих на використання дітьми 6-12 років. Відповідно до державної програми електронної освіти він постачається у португальські школи, зокрема на нетбуках [9].

Реліз "Educational" ірландської операційної системи Zorin OS, що містить кілька десятків навчальних програм, визначається розробниками,

як дистрибутив для школярів і студентів. Особливо він підходить для тих користувачів, хто прагне спробувати перейти з Windows на Linux [19].

В Україні також створюються дистрибутиви на основі Ubuntu. Зокрема, у збірці Ubuntu DesktopPack міститься освітнє програмне забезпечення, орієнтоване на використання у школах. Спеціально для організації та спрощення процесу навчання створено також пакет доповнень Ubuntu EducationPack. До його складу входить пакет освітніх програм для тих, хто навчається у молодших, середніх і вищих навчальних закладах, а також програми для організації управління і контролю навчального процесу, призначені для викладачів.

### **Література**

1. "Армянский" Linux предназначен для самого широкого круга пользователей. URL <http://www.itel.am/ru/news/3214>.
2. About Edubuntu. URL <http://www.edubuntu.org/about>.
3. About UberStudent. URL <http://www.uberstudent.org/mod/resource/view.php?id=4>.
4. Baltix GNU/Linux — operacinė sistema jūsu kompiuteriui. URL <http://baltix.akl.lt>.
5. DoudouLinux. The computer they prefer! URL <http://www.doudoulinux.org/web>.
6. EduBOSS 2.0 Release notes. URL <http://bosslinux.in/eduboss>.
7. Educa.Lernstick. URL <http://lernstick.educa.ch>.
8. Le projet AbulEdu. URL <http://www.abuledu.org>.
9. Linux Caixa Mágica. URL [http://www.caixamagica.pt/pag/documentacao/CM17/ManualCxM17\\_KDE.pdf](http://www.caixamagica.pt/pag/documentacao/CM17/ManualCxM17_KDE.pdf).
10. Manual de LliureX Infantil. URL [http://lliurex.net/home/files/live1109/documentacion/ManualInfantil\\_es.pdf](http://lliurex.net/home/files/live1109/documentacion/ManualInfantil_es.pdf).
11. MAX: Madrid\_Linux. URL <http://www.educa2.madrid.org/web/max>.
12. Molinux Educación. URL [http://www.molinux.info/index.php?option=com\\_content&view=article&id=233&Itemid=65](http://www.molinux.info/index.php?option=com_content&view=article&id=233&Itemid=65).
13. O Linux Educacional. URL <http://linuxeducacional.c3sl.ufpr.br>.
14. Poseidon Linux. URL <http://www.en.poseidonlinux.org>.
15. Skolelinux. About us. URL <http://www.slx.no/en/about-us>.
16. Suriyan. URL <http://code.google.com/p/suriyan>.
17. Ubuntu Guardabarranco llega a Nicaragua. URL <http://barricada.com.ni/2010/01/06/ubuntu-guardabarranco-llega-a-nicaragua>.
18. What is Qimo? URL <http://www.qimo4kids.com/what-is-qimo>.
19. Zorin OS 5 Educational. URL <http://zoringroup.com/blog/2011/08/14/zorin-os-5-educational>.

**Організація ІНТЕРНЕТ-доступу до віддалених баз даних**  
Столярчук О.В., Шувар Р.Я., Продивус А.М.

*Львівський національний університет імені Івана Франка, факультет електроніки, лабораторія комп'ютерних інформаційних технологій, [stolyarchuk@electronic.wups.lviv.ua](mailto:stolyarchuk@electronic.wups.lviv.ua)*

The possibility of usage of free software PHP, MySQL is considered in the contribution to provide access to remote databases via Internet by http-protocol.

В багатьох випадках виникає потреба використання спеціалізованих баз даних для автоматичної обробки і збереження інформації. Як правило, використовують ряд таких баз даних, незалежних одна від іншої. Для передачі інформації між ними використовують різноманітні звіти або конвертори. З розвитком локальних мереж, підключенням їх до мережі Інтернет, створенням внутрішніх корпоративних мереж, стало можливим з будь-якого робочого місця отримати доступ до інформаційних ресурсів. При цьому виникає потреба в уніфікованому інтерфейсі до різних систем керування базами даних, використання якого не вимагало б наявності різного програмного забезпечення. Однак, при спробі використати існуючі системи керування базами даних виникають проблеми пов'язані з сумісністю робочих місць. Кожна з них вимагає наявності свого програмного забезпечення на клієнтській машині. Тобто до кожної з цих систем потрібно налаштувати окремий модуль. Деякі виробники СКБД пропонують свої унікальні і досить потужні засоби підключення до віддалених баз даних. Однак, як правило ці продукти є комерційними і дорогими а тому не кожна організація може собі їх дозволити.

В роботі запропоновано методика розробки уніфікованого веб-інтерфейсу доступу до віддалених баз даних. Використання протоколу http, зумовлено його універсальністю й підтримкою на різноманітних платформах та операційних системах, можливістю використання в локальних і глобальних мережах. Для його використання не потрібно жодного додаткового програмного забезпечення, крім браузера.

Для роботи з базами даних запропоновано використання СКБД MySQL, оскільки вона є вільна, високошвидкісна, стійка, легка у використанні і є ідеальним рішенням для малих і середніх баз даних. Вихідні тексти сервера компілюються на багатьох платформах. Найбільш повно можливості сервера СКБД MySQL проявляються на серверах з Unix-подібними операційними системами, що в більшості своїй є вільними.

При розробці веб-інтерфесів для перетворення даних у гіпертекстовий вигляд в роботі використовується мова програмування PHP. З її переваг над такими мовами, як Perl і C слід відмітити можливість створення HTML-документів зі вставками команд PHP. PHP-скрипти виконуються на боці сервера. Ця мова доступна для більшості операційних систем. PHP підтримується на більшості сучасних веб-серверів. Використовуючи її, ми отримаємо універсальний продукт, сумісний із різними операційни-

ми системами та веб-серверами. Під час розробки можна використувати як процедурне, так і об'єктно-орієнтоване програмування або їх поєднання. PHP, крім цього, дає змогу виводити будь-які текстові дані, в тому числі XML-файли. Однією з головних переваг PHP є підтримка нею практично всіх СКБД, у тому числі MySQL.

З використанням цих програмних продуктів розроблено методику створення веб-інтерфейсів до віддалених баз даних, на основі якої розроблено ряд проектів. Веб-інтерфейси надають користувачеві доступ до тієї чи іншої бази даних чи до декількох одночасно. Причому реалізовано різні інтерфейси для різних користувачів з різним рівнем доступу до бази даних: з адміністративними правами та правами користувача. Створено модель вводу-виводу запитів користувача через веб-інтерфейс до бази даних, що надає всі необхідні інструменти для наповнення, модифікації чи вибірки даних в зручному для сприйняття вигляді. Розроблені веб-інтерфейси легко модифікуються для доступу до різних за структурою баз даних.

Запропоновану методику використано при розробці адміністративних та користувацьких веб-інтерфейсів доступу до спеціалізованих баз даних факультету електроніки Львівського національного університету імені Івана Франка. Вони легко модифікуються для різноманітних завдань та баз даних. Одним з таких проектів є впроваджена на факультеті система управління розкладом занять студентів. Роботу користувацької частини системи можна переглянути в мережі Інтернет за адресою <http://www.electronics.wups.lviv.ua/rozk/>.

### ***Переваги використання вільного ПЗ в організації наукової діяльності Університету***

*Ткаченко Н.О.*

*Донецький національний технічний університет, [nsot@library.dgtu.donetsk.ua](mailto:nsot@library.dgtu.donetsk.ua)*

Article describes advantages and disadvantages of free software for example software D-Space. Have presented the experience to create university repository in Donetsk National Technical University.

Сучасні засоби комунікацій та Інтернет-технології з кожним роком здобувають все більше прихильників по всьому світу. Вже неможливо уявити собі сучасну людину, що не має комп'ютера або мобільного телефону та не користується мережею Інтернет у повсякденному житті. Зручність, мобільність та легкий доступ до різноманітної інформації додають всесвітньому павутинню неоціненні переваги.

Університети, як головні помічники освіти та науки, також активно розвиваються у цьому напрямку. Створення та підтримка університетських веб-сайтів, надання корисної інформації на сайті про наукові

досягнення та діяльність інституту, є важливим завданням університету. А виникнення міжнародних рейтингів, що оцінюють присутність ВУЗів в інформаційному просторі, свідчить про актуальність цієї проблеми.

Окрім освітньої діяльності, важливим завданням кожного інституту є оприлюднення результатів наукових досягнень, сприяючи тим самим, розвитку науки. Але, підвищення в останні роки вартості передплати на наукові журнали, обмеження авторських прав, викликало обурення в науковому суспільстві. Саме тому, організована у місті Будапешті зустріч, що закінчилася підписанням Будапештської Ініціативи відкритого доступу, стала вибуховою подією XXI ст.

Оцінивши важливість самої ідеї відкритого доступу до інформації, у 2009 році на території України був заснований проект «**ElibUkr: створення Центрів знань в університетах України**», засновниками якого є **три університети** - Національний університет «Києво-Могилянська академія», Чернівецький національний університет ім. Ю. Федьковича та Харківський національний університет ім. В. Каразіна.

Відзначивши для себе переваги цього проекту, Донецький національний технічний університет увійшов до його складу у вересні 2009 року. Порадившись зі своїми колегами, керівництво ДонНТУ прийняло рішення з організації репозитарія на базі бібліотеки університету [2].

Розробка проекту-положення про архів (принципи та політики роботи архіву, його завдання та мета) було початковим завданням при створенні E-archive DonNTU. Для вирішення виникаючих проблем вивчався досвід інших організацій, проводилися консультації з учасниками проекту.

Окрім організаційних питань, не менш важливими були технічні, головним з яких став вибір програмного забезпечення на базі якого буде функціонувати архів. При порівнянні переваг та недоліків комерційного та відкритого ПЗ вибір було зроблено на користь останнього, завдяки його популярності та доступності [3]. За даними DOAR у світі використовується більше 10 програмних забезпечень для організації репозитаріїв, більшість з яких є відкритими ПЗ (рис. 1).

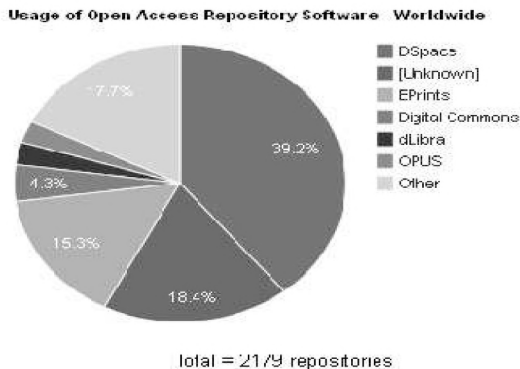


Рис. 1 – статистика використання ПЗ для організації репозитаріїв



Наведена діаграма показує розподілення популярного ПЗ у процентному співвідношенні. З неї видно, що найпопулярнішими на сьогоднішній день є D-Space та E-Print – вільне програмне забезпечення з відкритим кодом. В якості платформи для організації E-archive DonNTU також було обрано D-Space, а саме версія D-Space 1.5. Суттєву роль у цьому зіграло і те, що 80 % українських архівів створено саме на цій платформі.

D-Space – вільне програмне забезпечення з відкритим кодом, яке створено на базі Массачусетського технологічного університету, написано на [Java](#) та JSP. До основних його переваг можна віднести:

1. Популярність серед організацій. Більше 1000 організацій по всьому світу використовують D-Space у своїй діяльності, серед яких підприємства, бібліотеки, наукові установи та інше.

2. Відкритий код. Дозволяє користувачам змінювати, модифікувати або додавати початковий код згідно зі своїми потребами.

3. Підтримка цифрового контенту. Для роботи з D-Space використовуються всі відомі види цифрового контенту – від текстових до аудіо та відео файлів.

4. Налаштування платформи під свої потреби:

- можливість налаштування інтерфейсу для користувачів. D-Space надає два види інтерфейсу для користувачів – JSP та XML;
- можливість налаштування метаданих. Платформу налаштовано на формат метаданих Дублінського ядра, але присутня можливість зміни формату;
- зміна та додавання полів для перегляду та пошуку інформації;
- можливість вибору мови, D-Space доступний більше ніж на 20 мовах.

5. Сумісність з відомими операційними системами - Linux, Windows та інше.

Окрім позитивних моментів існують і деякі проблеми, що виникають при налаштуванні платформи під свої потреби. Для коректної роботи репозитарію ДонНТУ спеціалістам університету довелося вирішувати багато технічних проблем, серед яких були: формування вірного посилання на документ, неможливість реєстрації в архіві будь-якого користувача, отримання коректної статистики. Порадившись з колегами інших університетів та досконально дослідивши переваги нової версії D-Space, співробітники університету у травні 2011 року прийняли рішення про перехід на версію D-Space 1.7, яка має більше можливостей.

У 2012 році вийшла нова версія платформи - D-Space 1.8, яка поки що не локалізована для вітчизняних організацій. Взагалі серед основних недоліків існуючого вільного програмного забезпечення для вітчизняних користувачів можна зазначити декілька головних:

- 1) проблема локалізації ПЗ;
- 2) відсутність підтримки зі сторони виробників.



Але не дивлячись на вказані недоліки, позиції вільного ПЗ на інформаційному ринку зміцнюються з кожним роком, головним чином, завдяки його безкоштовності та підтримці відомих організацій світу. Таким чином, використання університетами вільного програмного забезпечення, є важливою складовою для успішної організації освітньої та наукової діяльностей.

### **Література:**

1. Воропаева, В.Я. Внедрение новых информационно-компьютер-ных технологий в библиотеке ДонНТУ/ В.Я. Воропаева// Деятельность библиотеки высшего учебного заведения в свете модернизации высшего образования: материалы междунар. науч.-практ. конф., г. Севастополь, 25-26 ноября 2010 г.- Севастополь: Изд-во СевНТУ, 2010. - С.113-117.
2. Ткаченко, Н.А. Возможности университетского репозитария для представления научных достижений ученых ДонНТУ в мировом информационном пространстве / Н. А. Ткаченко // Сучасна бібліотека в інноваційному освітньому просторі: матеріали наук.-практ. конф., м. Луганськ, 17 - 18 берез. 2011 р. / ред. кол. : Ж. В. Марфіна, О.В. Бикова, К.О. Шарамет. – Луганськ: Вид-во ДЗ «ЛНУ імені Тараса Шевченка», 2011. - С.192 – 197.
3. Воропаева В.Я. Сравнительный анализ использования открытого и коммерческого ПО в библиотеке ДонНТУ[Электронный ресурс] /В.Я. Воропаева, Н.А.Ткаченко, А.Ю.Клочко; Донец. нац. техн. ун-т. Электронный архив.- Режим доступа:
4. <http://ea.donntu.edu.ua:8080/jspui/handle/123456789/1518>

### **Використання вільно-поширюваного ПЗ для створення систем управління навчальним процесом у ВНЗ**

Триус Ю.В., Герасименко І.В.

Черкаський державний технологічний університет, [tryusyv@gmail.com](mailto:tryusyv@gmail.com), [gerasi-menkoinna@mail.ru](mailto:gerasi-menkoinna@mail.ru)

У роботі розглядаються питання створення та основні напрями використання систем управління навчальним процесом у ВНЗ, розроблених на базі вільно поширюваного програмного забезпечення. На формування і розвиток особистості найбільше впливає середовище, в якому вона живе, навчається, працює. Тому сьогодні для ВНЗ важливою і актуальною проблемою є проблема створення такого високотехнологічного інформаційно-комунікаційного освітньо-наукового середовища, в якому студент знаходиться щодня в процесі всього періоду навчання у вищій школі.

Серед інноваційних технологій, на основі яких у ВНЗ повинно створюватися нове навчальне середовище, де студенти можуть отримати доступ до навчальних матеріалів у будь-який час та в будь-якому місці, є технології електронного (дистанційного, мобільного) навчання.

Одним із засобів створення інформаційно-комунікаційного освітньо-наукового середовища ВНЗ є вільно поширювана система управління навчальним контентом MOODLE – модульне об'єктно-орієнтоване динамічне навчальне середовище. На рис. 1 наведена статистика використання системи MOODLE в світі [3].



Рис. 1. Статистика використання системи MOODLE в світі

Перехід ВНЗ на вільно поширюване програмне забезпечення в останній час усе частіше виникає як у колах користувачів на рівні виконавчої та законодавчої влади. Це пов'язано з посиленням контролю за ліцензійною чистотою використання програмного забезпечення, завдання переходу на ліцензійне програмне забезпечення, яке не потребує істотних вкладень і забезпечує достатню функціональність.

Статистика, наведена на сайті [3], свідчить про те, що у першій десятці сайтів за кількістю навчальних курсів (рис. 2) знаходяться сайти на основі MOODLE, а за кількістю користувачів сайт LMS MOODLE-SENA SOFIA Plus є лідером – 1 149 319 осіб.

Сайт	Користувачів	Курсів
AulaXXI	225,546	104,248
VDU Moodle	3,566	59,920
Minha UFMG	62,887	56,385
Campus Virtual	24,615	44,561
LMS Moodle – SENA SOFIA Plus	1,149,319	40,455
Ming Chuan University portal(銘傳大學入口網站)	75,882	33,362
東海大學 數位教學平台	44,371	31,515
RTU E-Studiju sistēma (private)	42,314	30,678
UFMG Virtual	77,595	30,654
CALIFORNIA STATE UNIVERSITY, FULLERTON	119,267	29,302

Рис. 2. Статистика першої десятки сайтів по числу курсів

Як результат, склалася світова спільнота, зацікавлених у використанні відкритих ресурсів в освіті і нагромаджено певний досвід такого використання. Створено певні Internet-ресурси, через які має координуватися ця робота. Це говорить про достатньо великий інтерес світової педагогічної громади до даної теми та перспективи щодо впровадження відкритого програмного забезпечення в українській освіті.

Тому при створенні в ЧДТУ інформаційно-аналітичної системи контролю і оцінювання навчальних досягнень студентів (ІАС КОНДС) використовувалося лише вільно поширюване програмне забезпечення: система електронного навчання ЧДТУ створена на основі MOODLE, підсистеми статистичної обробки результатів контролю і підсистеми підтримки прийняття рішень розроблені на основі технології AMP (web-сервер *Apache*, база даних MySQL, мова програмування PHP) [1].

Основним засобом, через який відбувається інформаційне наповнення підсистеми статистичної обробки результатів контролю і підсистеми прийняття рішень, є система електронного навчання (СЕН), яка призначена для підтримки навчального процесу студентів різних форм навчання (денної, заочної, дистанційної), організації їх самостійної роботи, а також для проведення різних видів контролю та оцінювання навчальних досягнень студентів у автоматизованому режимі.

СЕН має потужні засоби для створення електронних навчальних курсів (ЕНК) і організації автоматизованого контролю навчальних досягнень студентів [2]. Система електронного навчання доступна в мережі Internet керівництву університету, викладачам і студентам у відповідності до прав доступу до інформаційних ресурсів і підсистем ІАС КОНДС [4].

Особливістю підсистеми підтримки прийняття рішень (СППР) (англ. Decision Support System, DSS), яка розробляється, є те, що, вона призначена для прийняття рішень не лише керівниками ВНЗ (ректор, проректори), керівниками навчальних підрозділів ВНЗ (начальник навчального відділу, декан, завідувач кафедри), а також викладачами і, навіть, студентами, з метою удосконалення навчального процесу та ґрунтується на даних, що одержуються з підсистеми статистичної обробки результатів контролю навчальної діяльності студентів ВНЗ.

В результаті реалізації проекту створено інформаційно-аналітичну систему контролю та оцінювання навчальної діяльності студентів ВНЗ в умовах кредитно-модульної системи навчання, яка розміщена в корпоративній мережі ЧДТУ і доступна керівництву університету, викладачам і студентам у відповідності до прав доступу до інформаційних ресурсів і підсистем.

### **Література:**

1. *Інформаційно-аналітична система контролю та оцінювання навчальної діяльності студентів ВНЗ: Монографія / А.А. Тимченко, Ю.В. Триус, І.В. Стеценко, Л.П. Оксамитна, В.М. Франчук, Г.О. Заспа, Д.П.*

- Туницький, О.В. Тьорло, І.В. Герасименко. – Черкаси: МакЛаут, 2010. – 300 с.
2. Використання системи електронного навчання MOODLE для контролю і оцінювання навчальної діяльності студентів ВНЗ: методичний посібник / Ю.В. Триус, І.В. Стеценко, Л.П. Оксамитна, В.М. Франчук, І.В. Герасименко / За ред. Ю.В. Триуса. – Черкаси: МакЛаут, 2010. – 200 с.
  3. Система управління навчанням MOODLE. [Електронний ресурс] – Режим доступу: [www.MOODLE.org](http://www.MOODLE.org).
  4. Система електронного навчання ЧДТУ. [Електронний ресурс] – Режим доступу: <http://ias.cdtu.edu.ua/MOODLE19-test>.

## **Перехід університетів України на вільне програмне забезпечення**

Тверезовська Н.Т., Липська В.В.

Національний університет біоресурсів і природокористування України, [tverezovskaya@gmail.com](mailto:tverezovskaya@gmail.com), [lipaska73@gmail.com](mailto:lipaska73@gmail.com)

This paper analyses the prospects of the use of free software at the National University of Life and Environmental Sciences of Ukraine/ It examines the current state of free software use as well as the negative factors hindering the implementation of free software in teaching and learning processes.

Перехід від пропрієтарного програмного забезпечення до вільного у численних сферах його застосування (у тому числі в освіті) є загальносвітовим трендом. Використання вільного програмного забезпечення виступає виразником демократичного поступу, свободи, відкритого суспільства та суспільства знань, а також права вивчати та ділитись своїми доробками з іншими. Розглянемо на прикладі Національного університету біоресурсів і природокористування перехід на вільне програмне забезпечення, його можливості, перспективи, переваги тощо.

Підтримку системи інформаційного забезпечення зазначеного університету покладено на два структурні підрозділи – **центр інформаційно-комунікаційного та програмного забезпечення**, підпорядкований першому та **центр інформаційно-комунікаційних та дистанційних технологій навчання**, підпорядкований директору Українського ННІ інформаційного і телекомунікаційного забезпечення агропромислової та природоохоронної галузей економіки.

Головні завдання центру інформаційно-комунікаційного та програмного забезпечення полягають у:

- впровадженні інформаційно-комунікаційних технологій в систему адміністративного управління університетом;
- забезпеченні роботи телекомунікаційної системи;
- експлуатації та обслуговуванні комп'ютерної техніки;
- впровадженні та експлуатаційного супроводу програмного забезпечення;
- розробці та супроводу Web-ресурсів.

Технічні ресурси системи інформаційно-комунікаційного забезпечення Київського територіального центру на сьогодні нараховують близько 3000 персональних комп'ютерів, які під'єднані до локальної мережі університету, біля 2-х десятків серверів різного призначення, оптоволоконну мережу, яка з'єднує 15 навчальних корпусів та 14 студентських гуртожитків, локальні мережі в усіх навчальних корпусах і студентських гуртожитках, три аудиторії, обладнанні засобами для проведення відеоконференцій (фірм Polusom та Sony). Доступ до сервісів Інтернету в університеті став невід'ємним елементом його інфраструктури. НУБіП України забезпечує якісним доступом до Інтернету і його сервісів усі підрозділи університету та студентські гуртожитки.

Нині викладачі мають десятки або навіть сотні вільних програм, які допомагають їм у викладанні різноманітних дисциплін в університеті. Так вільна програма для дистанційного навчання, Moodle (<http://moodle.org>), допомагає викладачам можуть створювати курси, публікувати мультимедійні документи, готувати онлайн-уроки, адмініструвати wiki і форум, задавати домашні завдання, відправляти повідомлення, відслідковувати як студенти користуються курсом тощо.

SWOT- аналіз переходу НУБіП України на безкоштовне програмне забезпечення GNU GPL

Strengths – сильні сторони	Weaknesses – слабкі сторони
<p>Значна економія коштів (<math>2000 * (500 \text{ windows} + 500 \text{ office} + 500 \text{ antivirus}) = 3000000</math> +інший софт</p> <p>Юридично все законно, без потенційних проблем з захистом авторських прав (при залишенні на піратському програмному забезпеченні можуть виникнути проблеми з відділком МВС у боротьбі з економічними злочинами та з позовами до суду з недотримання авторських прав на ліцензійне програмне забезпечення)</p>	<p>Потреба у додатковому навчанні працівників та студентів, аспірантів</p> <p>Тимчасова втрата оперативності робіт за ПК пов'язана із невмінням працювати на новому програмному забезпеченні</p> <p>Затрати людино-годин пов'язані із переустановленням програмного забезпечення</p> <p>Неможливість 100% забезпечення безкоштовного програмного забезпечення (існують унікальні платні програми – але ця потреба</p>

	тільки для обмеженого кола ПК) Нестабільність у роботі окремих програм та відсутня технічна підтримка окремих програм
<b>Opportunities – можливості</b>	<b>Threats – загрози</b>
<p>Широкий спектр безкоштовних програм</p> <p>При налагодженні даної системи роботи з безкоштовним ПЗ, можна надавати платні консультації іншим ВНЗ чи іншим підприємствам по переходу на безкоштовне ПЗ</p> <p>Створення платних курсів по навчанню роботі з ПК на новому програмному забезпеченню</p>	<p>Супротив працівників НУБіП, небажання переходити на безкоштовне програмне забезпечення</p> <p>Небажання навчатись роботі з новим ПЗ</p> <p>Можливість переходу окремих програм в майбутньому на платну основу</p>

#### Перехід на автоматизацію документообігу

<b>Strengths – сильні сторони</b>	<b>Weaknesses – слабкі сторони</b>
<p>Покращення оперативності донесення інформації</p> <p>Відслідковування за виконанням наказів та розпоряджень</p> <p>Зниження кількості паперових документів, що в свою чергу відобразиться на економії паперу та канцтоварів</p> <p>Можливість відслідкувати слабкі штатні одиниці в університеті, без яких університет може обійтись</p> <p>Можливість залучати в систему відокремленні структурні підрозділи (інститути, технікуми, коледжі, навчально-дослідні господарства, тощо</p>	<p>Затрати на придбання додаткових серверів (20000-40000 грн)</p> <p>Затрати на придбання програмного забезпечення (50000-100000 грн для Microsoft Office SharePoint Server</p> <p>Затрати на налаштування працездатності системи</p> <p>Труднощі переходу на електронний документообіг працівників старшого віку, які не вміють працювати з ПК</p> <p>Необхідність постійно працювати із ПК навіть для тих осіб, чия робота напряму непов'язана із ПК</p>



	Потреба у оновленні значної частини комп'ютерів
Opportunities – можливості	Threats – загрози
<p>Можливість взяти людей на виконання цього проекту на громадських засадах (залучивши аспірантів та докторантів відповідного напрямку)</p> <p>Покращення інформаційної безпеки, що вимагається міжнародними стандартами серії ISO 27000 із інформаційної безпеки</p> <p>Можливість віддаленого доступу до документів з будь-якої точки планети де є інтернет</p>	<p>Небажання персоналу працювати з електронними документами</p> <p>Збої у роботі комп'ютерних мереж</p> <p>Хакерські атаки</p> <p>Загримки у роботі під час руху електронного документу при відсутності окремих ключових фігур (поїздки у відрядження, виїзди та інші причини відсутності)</p>

### ***Автоматизовані системи створення та супроводу пакетів у дистрибутиві ALT LINUX. Крок у майбутнє.***

*Власенко І.Ю.*

*ALT Linux Team, [viy@altlinux.org](mailto:viy@altlinux.org)*

Automated packaging systems for a Linux distro. Building a new future. Automated packaging systems are a new approach to address a constant lack of resources and interdistributive development issues that plague modern linux desktop.

В даний час очікування користувачів Linux перевершують ті можливості, що можуть надати всі основні дистрибутиви Linux. У перших рядах критики практики сучасного дистрибутивобудування такі постаті, як Лінус Торвальдс та Інго Молар.

В доповіді стверджується, що криза, що охопила сучасне дистрибутивобудування, пов'язана з кризою традиційної розподіленої моделі розробки дистрибутивів, що нагадує феодалізм. Як за часів здорового феодалізму король чи уряд розподіляв земельні маєтки за службу, так і у традиційній розподіленій моделі розробки дистрибутивів майнтайнер на час виконання своїх обов'язків отримував пакети у власність. Їх колектив являв собою щось на зразок феодальної республіки чи монархії, залежно від дистрибутиву.

В попередню епоху розвитку ВПЗ ця модель була достить ефективною, проте постійний ріст кодової бази ВПЗ виявив суттєвий недолік такої моделі - від майнтайнера вимагається виконання всіх ролей супроводження, пов'язаних з пакетом:

- має бути знавцем політик пакетування самих різних підсистем, до яких належать різні файли, що входять в його пакети
- має стежити за оновленнями і оновлювати пакети
- має виправляти помилки в програмному забезпеченні, що пакетується.
- бажано, щоб приймав участь у розробці програмного забезпечення, що пакетується.
- має тестувати програмне забезпечення, що пакетується.
- бажано, щоб був користувачем програмного забезпечення, що пакетується.

Проте ріст кодової бази призвів до того, що ймовірність, що у одній людині перетнуться всі ці якості, стає все меншою. Крім того, потреба на таких людей є у кожному дистрибутиві, тому потреба на них зростає з ростом числа пакетів нелінійно. Відповідно, ріст числа традиційних майнтайнерів, "феодальних баронів", все більше і більше відстає від росту наявної кодової бази ВПЗ.

Виходом з кризи буде настання "нових часів": злам "феодальних" кордонів, розподіл ролей (спеціалізація), промислова революція у супроводі пакетів.

Зараз дистрибутиви ALT Linux є лідерами у впровадженні автоматизованих систем супроводу пакетів. `geroscop`, `cronbuild`, `autoports`, `autoimports` --ці системи ще роблять перші кроки, але їх вже можна назвати аналогом парового двигуна та ткацького верстата, що почали промислову революцію.

### **«Хмарні» обчислення як основа формування персональних навчальних середовищ**

*Воронкін О. С.*

*Луганський державний інститут культури і мистецтв, [alex.voronkin@gmail.com](mailto:alex.voronkin@gmail.com)*

The trends and perspectives of "cloud" computing are analyzed. The usage of such computing allows effectively involve informational resources into teaching process, whereas improving the level of study. "Cloud" technologies were used during author's distant course «Introduction to the physics of sound» for the pupils of 10-11 grades of general educational institutions in November-December 2011.

Серед основних тенденцій розвитку ринку інформаційних технологій в минулому році, що мають пряме відношення до вільного програмного

забезпечення, слід відзначити «хмарні обчислення», мобільні та соціальні технології [1].

Парадигма «хмарних» обчислень

«Хмарні» обчислення є перспективним напрямком, який швидко розвивається й динамічно поширюється. «Хмарні» обчислення (англ. cloud computing) – це парадигма, в рамках якої інформація постійно зберігається на серверах в Інтернет-мережі і тимчасово кешується на клієнтській стороні (персональних комп'ютерах, мобільних пристроях, ігрових приставках тощо). Таким чином користувач хоча й має доступ до власних даних, але не може управляти інфраструктурою, операційною системою, а програмне забезпечення, з яким він працює, надається як відповідний сервіс. Дані зберігаються в розподіленому вигляді і періодично архівуються [2, 3].

«Хмарні» обчислення є результатом синтезу цілого ряду технологій та підходів. Основою таких обчислень є інфраструктура, на яку накладається платформа, а поверх – програмне забезпечення.

1. *Модель надання апаратних ресурсів як сервісу (англ. Infrastructure as a Service, IaaS). Керування всією інфраструктурою здійснює постачальник сервісів, а споживач керує тільки операційною системою та встановленими застосунками. В якості прикладів слід навести Amazon Elastic Compute Cloud й Simple Storage Service, Google App Engine й Aptana Cloud.*
2. *Модель надання платформи як сервісу (англ. Platform as a Service, PaaS). Створюється можливість оренди платформи, яка звичайно включає операційну систему та прикладні сервіси. Прикладом можуть бути Windows Azure й Google Apps.*
3. *Модель надання програмного забезпечення як сервісу (англ. Software as a Service, SaaS). Створюється можливість оренди застосунків. Програмне забезпечення як сервіс включає IaaS та PaaS. Прикладом можуть бути Google Docs й Business Productivity Online Suite.*

Виділяють й інші моделі – дані як сервіс, робоче місце як сервіс, комунікація як сервіс та інші, але на сьогоднішній день намітилася тенденція до інтеграції всіх існуючих моделей (Everything as a service, EaaS).

### **Мобільні технології**

В розвитку мобільних технологій помітним є збільшення динаміки світового ринку продуктів та послуг навчання. Відбувається інтеграція електронного навчання з мобільним. Ціни на мобільні пристрої стрімко падають, а освітні можливості зростають. Це стало можливим завдяки довготривалому автономному енергозабезпеченню, підтримці мережевих протоколів, гнучким методам оновлення програмного забезпечення, росту швидкості передачі даних і використання хмарних платформ [1].

В 2010 році за участю компаній Intel, Nokia і Linux Foundation вийшла операційна система MeeGo, а в 2011 році розпочалася інтеграція проектів

MeeGo і LiMo в єдину платформу Tizen. Google і Samsung займаються розробкою мобільних платформ на базі GNU/Linux, а в Китаї розроблюється власна «хмарна» операційна система Aliyun.

Веб-соціалізація та формування персональних навчальних середовищ

**Інтернет-мережа** надає як інформаційні, так і комунікаційні послуги користувачам. Завдяки соціалізації з'явилися віртуальні співтовариства – соціальні мережі, які дозволяють створювати власні профілі й спілкуватися з іншими учасниками віртуально. Соціальні мережі безупинно інтегруються з багатьма зовнішніми сайтами, сервісами та мобільними пристроями. З'явилися принципово нові можливості передачі знань, відбувається трансформація методів навчання із розвитком ІТ.

Один із нових альтернативних підходів до неперервного навчання оснований на теорії конективізму та формуванні персональних навчальних середовищ (ПНС) [4]. Реалізується мережева педагогіка співробітництва, яка вимагає більш активної участі слухачів в організації власного навчання (учасник самостійно управляє навчанням, змістом, процесом і контролем, визначає інструменти й потоки даних, які для нього є інформативними). Справа в тому, що більшість інструментів та Інтернет-сервісів не створювалися спеціально для освітнього процесу, а тому вони можуть використовуватися вибірково в залежності від спеціалізації та переконань того чи іншого педагога або слухача [5]. Тому ПНС може бути організовано цілим рядом соціальних сервісів веб 2.0 – веб 3.0.

#### **«Хмарні» обчислення в освіті**

В освітніх установах України «хмарні» сервіси первісно використовувалися як безкоштовні хостинги поштових служб, а інші інструменти «хмарних» обчислень практично не використовувалися. І тільки відносно недавно в педагогічному процесі розпочали використовувати ІТ-пропозиції від Google, Microsoft і Amazon. Компанія Zoho створила цілий пакет онлайн-офісних додатків: текстовий і табличний процесори, редактор презентацій (<http://www.zoho.com>). Існують й інші онлайн-сервіси офісних додатків, наприклад: 1) текстові редактори iNetWord, J2E, Writeboard, ThinkFree та ін.; 2) табличний сервіс EditGrid ([www.editgrid.com](http://www.editgrid.com)); 3) редактор презентацій Sliderocket ([www.sliderocket.com](http://www.sliderocket.com)); 4) графічні редактори **Lunapic** ([www.lunapic.com/editor](http://www.lunapic.com/editor)) і **Pixlr Editor** ([pixlr.com](http://pixlr.com)); 5) редактор діаграм і блок-схем Diagram ([www.diagram.ly](http://www.diagram.ly)); конструктор сайтів Ucoz; 6) платформи для проведення вебінарів WizIq ([www.wiziq.com](http://www.wiziq.com)), Quatla ([www.quatla.com](http://www.quatla.com)), Webinar ([webinar.ipo.kpi.ua](http://webinar.ipo.kpi.ua)) та ін.; 7) «хмарні» операційні системи **Cloudo** ([www.cloudo.com](http://www.cloudo.com)), **Glide OS** ([www.glideos.com](http://www.glideos.com)) та ін.

Досвід проведення відкритого дистанційного курсу з фізики звуку

Автором у листопаді–грудні 2011 року для учнів 10-11 класів загальноосвітніх навчальних закладів було проведено дистанційний курс «Вступ до фізики звуку». В курсі прийняло участь 7 слухачів з України та США. Курс було розраховано на 6 лекційних занять, 1 семінарське занят-

тя та підсумкове заняття, які проводилися у формі вебінарів. Активно проявляли себе тільки 2 слухача. Інформація про проведення кожного із занять публікувалася у розділі новин інформаційно-освітнього порталу «Технології дистанційної освіти» ([www.tdo.at.ua](http://www.tdo.at.ua)), створеного в конструкторі сайтів Ucoz, відповідне повідомлення розміщувалося в Twitter-блозі з наступним кросспостингом в соціальній мережі Facebook. Для проведення вебінарів використовувалася платформа WizIq, а для демонстрацій фізичних експериментів використовувалася ретрансляції відео-фрагментів з сервісу Youtube. Зв'язок з учасниками підтримувався електронною поштою. Презентації всіх занять розміщені на сервісі Slideshare.

### **Література**

1. Аронова Л. Исследование российского рынка СПО. Тенденции развития рынка СПО в России. Ч.3 [Електронний ресурс] / Л. Аронова, И. Матюшонко. – Режим доступу : [http://www.opennet.ru/docs/RUS/fss\\_history3](http://www.opennet.ru/docs/RUS/fss_history3).
2. Колесников А. В. Применение «облачных» вычислений в программах стационарного и дистанционного обучения [Електронний ресурс] / А. В. Колесников, С. А. Деревянко, Е. В. Ромашка // Вісник Східноукраїнського національного університету імені Володимира Даля. – Луганськ, 2011. – № 3 (157). – Режим доступу : [http://www.nbu.gov.ua/portal/Soc\\_Gum/VSUNU/2011\\_3/Kolesnikov.pdf](http://www.nbu.gov.ua/portal/Soc_Gum/VSUNU/2011_3/Kolesnikov.pdf).
3. Войтович І. С. Вільне програмне забезпечення чи «CLOUD COMPUTING» у навчальному процесі? / І. С. Войтович // Foss Lviv–2011 : тези міжнародної науково-практичної конференції. – Львів : ЛНУ ім. І.Франка, 2011. – С. 25–27.
4. Воронкін О. С. Основи використання інформаційно-комп'ютерних технологій в сучасній вищій школі : навч. посіб. / О. С. Воронкін. – Луганськ : Вид-во ЛДІКМ, 2011. – 156 с.
5. Воронкін О. С. Персональні навчальні мережі в системі дистанційної освіти / О. С. Воронкін // New information technologies in education for all: learning environment : збірник праць VI міжнар. конф. ІТЕА–2011, м. Київ, 22–23 листопада 2011 р. – Київ, 2011. – С. 202–208.

## ***Library for advanced functions in algorithms, data structures and AI implemented in C/C++ - Olib***

*Wszeborowska A. and Swierczewski L.*

*Computer Science and Automation Institute, College of Computer Science and Business Administration in Łomża, olcia90.prv@wp.pl, luk.swierczewski@gmail.com*

Developers of one of most popular languages used for scientific computations which is C, have a lot of libraries at their disposal. Many of those allows usage of modern platforms equipped with advanced multicore processors. Among them is BLAS (Basic Linear Algebra Subprograms) and LAPACK (Linear Algebra PACKage). They contain mainly high performance procedures operating on matrix and vectors.

A lot of high quality functions have been implemented in FORTRAN language, which, despite its age, is still used by developers in many science facilities. completely new direction in informatics are graphic accelerators which can be fully programmed and use for any computation, not limited to rendering tasks. Support for GPUs is provided by CUBLAS (CUDA Basic Linear Algebra Subroutines) library, which supports nVidia GPUs. Optimized for modern GPUs is also LINPAC benchmark, which measures the speed of solving dense system of linear equations. It is being used to test fastest computers on Earth and to classify them on the TOP500 list. The following work presents conception of Olib library binding together capabilities of BLAS, CUBLAS and LAPACK and also allows usage of wide spectrum of implemented algorithms both from cryptography and genetic algorithms. It is completely free and provided on GNU GPL license, it also allows to use classic and multicore CPUs and GPUs.

Olib is a library written primarily for Linux OS. In other systems (eg. Windows, BSD, Solaris) minor compatibility issues with the code may occur. efficient methods was implemented that can be divided on:

- Linear algebra,
- Discrete Mathematics,
- Cryptography,
- Numerical methods,
- Artificial intelligence

In the field of cryptography both historical ciphers (among others. Caesar cipher, *ślótkowy*, Vigenère'a) as well as those recent (RSA, DES, 3DES, AES, SHA, MD5) were implemented. Artificial intelligence is related mainly to Genetic Engine, which gives it the possibility to use genetic algorithms to solve the unsolvable equations in an analytical way. Genetic Engine has great possibilities of defining crossover and mutation operators, and also other characteristics such as methods of selection or type of initial population.

Whole, for several computing architectures, was optimized. This division is presented in Figure 1. In one program, the programmer can use both functions



- using multi-core processors, as well as those which supports graphics accelerators. In the test phase, is also handling CELL processors that are integrated in the PlayStation 3 consoles and IBM Blade computing stations. These processors have a very large computing capabilities, IBM officially announced the discontinuation of its development. How easily can guess few people will try to learn a technology that has no future.

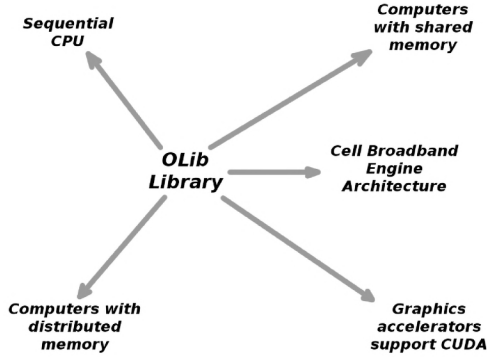


Fig. 1. Architectures supported by Olib Library.

With this library, in some cases, a significant reduction in run-time is possible. Example of acceleration achieved with the GPU while multiplication of matrix is presented in Figure 2.

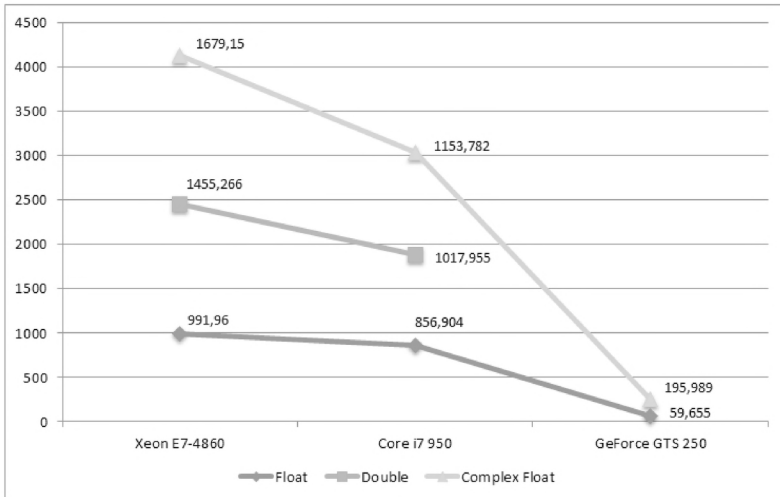


Fig. 2. Achieved performance increase (decrease in execution time measured in seconds) during matrix multiplication using GPU compared to CPU

Presented library does not have the capacity as large as any other software of this type, which are often developed for many years. In many cases, a very good results can be achieved - especially when a programmable graphics accelerator or a multi-core processor are available. More information about Olib can be found on the homepage [5].

### **References:**

1. Brian W. Kernighan, Dennis M. Ritchie (1988). *The C Programming Language*. Prentice Hall, Inc
2. [http://en.wikipedia.org/wiki/Genetic\\_algorithm](http://en.wikipedia.org/wiki/Genetic_algorithm)
3. Zbigniew Czech (2010). *Wprowadzenie do obliczeń równoległych*. Wydawnictwo Naukowe PWN
4. Jason Sanders, Edward Kandrot (2011). *CUDA by Example: An Introduction to General-Purpose GPU Programming*. Addison-Wesley
5. <http://www.goldbach.pl/olib>

## **Досвід впровадження відкритого програмного забезпечення в стартапах компанії Медіа-Ключ**

Зелінський О.Л

Компанія Медіа-Ключ, [zelan@media-kluch.ua](mailto:zelan@media-kluch.ua)

Experience of implementing open source software company startups Media Key. The issue of introducing open source software. Open source software in Ukraine. Solutions based on open source software.

Проблематика впровадження будь-якого програмного забезпечення в тому числі і ВПЗ актуальна практично завжди для різного роду систем. Зазвичай в процесі впровадження розв'язуються не тільки технічні завдання, а й ряд специфічних завдань, пов'язаних з учасниками переходу і майбутніми користувачами програмного продукту. Більш того, враховуючи існуючу ситуацію з легалізацією ВПЗ в Україні, особливої уваги потребує питання про юридичне оформлення переходу на ВПЗ і про розробку супутньої документації. Перед інженерами у компанії «Медіа-Ключ» стояли всі перераховані вище завдання з урахуванням специфіки одного з стартапів. Робота в проєкті вимагала роботи з графічними редакторами, 3D-моделями та програмним забезпеченням, пов'язаних з генерацією звітів та моніторингом проєктів. Враховуючи задані вимоги, був проведений економічний розрахунок і порівняльний

аналіз можливих технічних рішень. Перше рішення базувалося на платформі ОС Windows, друге на базі ВПЗ.

У разі використання СПО і таких програм як Gimp, Inkscape, Blender, автоматично виникала додаткова задача пов'язана з перенавчанням діючих співробітників, які є фахівцями з роботи в середовищі Photoshop, 3Dmax, CoralDrow. В результаті, на початковому етапі проекту було прийнято рішення про організацію інфраструктури з використанням ВПЗ, а частина завдань пов'язаних з розробкою складних 3D-моделей і графічних макетів по мірі їх надходження передавати зовнішнім розробникам. Як основна операційна система на комп'ютерах використовувався дистрибутив Linux Ubuntu 10.04 розроблений компанією Canonical, який задовольняв велику частину вимог і полегшив перехід, враховуючи стереотипи користувачів при роботі з інтерфейсом ОС. Перехід виконувався в два такти. У першому такті на комп'ютерах користувачів ОС Linux Ubuntu 10.04 встановлювалася як альтернативна система, яка функціонувала разом з WindowsXP і в критичних випадках користувач міг повернутися в звичне середовище і виконати необхідні йому завдання. Принаймні це створювало позитивний психологічний ефект. Випадки повернення на WindowsXP фіксувалися і враховувалися у другому такті. Повний перехід на ВПЗ проводився з урахуванням побажань користувачів і додаткових технічних рішень.

Також, в результаті впровадження ВПЗ, відповідно до положення Всесвітньої конвенції про авторське право, договору Всесвітньої організації інтелектуальної власності про авторське право, Конституції України, Цивільного кодексу України, закону України «Про авторське право і суміжні права» був розроблений пакет юридичних документів, що дає змогу використовувати Вільне програмне забезпечення та вести необхідну звітність у складі:

- наказ про затвердження правил використання комп'ютерних програм;
- правила використання комп'ютерних програм;
- акт вводу в експлуатацію;
- картка обліку комп'ютерної техніки;
- документально оформлене закріплення комп'ютерної техніки за користувачем.

### ***Література***

1. Рекомендації щодо забезпечення правомірності використання комп'ютерних програм вільного користування  
<http://sips.gov.ua/ua/rozcompprofree.html>
2. Рекомендації щодо забезпечення правомірності використання комп'ютерних програм у діяльності суб'єктів господарювання  
[http://sips.gov.ua/ua/soft\\_recommendations.html](http://sips.gov.ua/ua/soft_recommendations.html)

**Использование свободного ПО для реализации функций  
контроля при обучении студентов специальности  
«Математика. Информатика» в педагогическом ВУЗе**  
Зенько С.И., Ненартович М.В.

Учреждение образования «Белорусский государственный педагогический университет имени Максима Танка», [sergey.zenko@tut.by](mailto:sergey.zenko@tut.by)

This paper describes the basic training and analytical-methodical control functions of students, activity. The examples of use of open source software («Geany», «Advanced Grapher», «MiniTestSl») are offered. It describes the main characteristics of open source software used in monitoring during the process of studying such subjects as "Technology programming and algorithmic methods", "Elementary Mathematics and some practice on problem solving," "Methods of informatics teaching."

В настоящее время переход на свободное и открытое программное обеспечение (СПО) является актуальным. Вопросы организации контроля за усвоением знаний, умений и навыков студентов в процессе изучения ими учебных дисциплин в вузе не менее актуальны. Для повышения эффективности осуществления этой деятельности можно использовать имеющийся потенциал свободного и открытого программного обеспечения.

Среди функций контроля мы выделяем две группы: обучающие и аналитико-методические. К обучающим функциям контроля деятельности студентов отнесены предупреждающая, стимулирующая, коррекционная, развивающая и воспитательная.

Предупреждающая функция направлена на создание ситуаций по актуализации знаний, необходимых для предостережения студентов от возможных ошибок через повторение, указание, которые присутствуют в ходе изучения программного материала и в процессе контроля.

Стимулирующая функция состоит в обеспечении положительных мотивов при изучении дисциплин, закреплении знаний, проверке их усвоения и создании благоприятных условий для проявления активности.

Коррекционная функция предполагает активизацию умственной деятельности студентов для накопления положительного опыта в схожих, более простых ситуациях и для «локального прорыва».

Развивающая функция направлена на интеллектуальный рост обучаемых, развитие внимания, памяти, мышления, речи, овладение студентами рациональными приемами учебно-познавательной деятельности.

Воспитательная функция реализуется через приучение студентов к систематической работе и формирование у них в процессе приобретения знаний моральной ответственности за результаты собственной деятельности.

К аналитико-методическим функциям контроля деятельности студентов отнесены диагностическая, дифференцирующая и методическая.

Диагностическая функция предполагает осуществление систематического анализа уровня усвоения учебного материала для получения объективной информации о причинах, которые не позволяют студентам достигнуть необходимого уровня усвоения материала.

Дифференцирующая функция нацелена на предоставление преподавателю достоверной информации для деления студентов на подгруппы.

Методическая функция связана с осуществлением преподавателем педагогической рефлексии собственной деятельности, самоанализа эффективности различных организационных форм и методических приемов.

Использование свободного программного обеспечения позволяет реализовывать выше описанные функции контроля при обучении студентов на математическом факультете БГПУ.

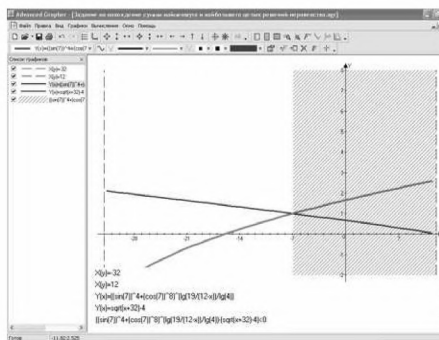
Свободное программное обеспечение (СПО) — это программное обеспечение, распространяемое на условиях так называемых открытых или свободных лицензий. Например, лицензии GNU General Public License (GNU GPL), на базе которых свободно распространяются такие программы как «Geany», «Advanced Grapher», «MiniTestSl».

Преимуществами СПО является доступность и бесплатность, кросс-платформенность, открытость исходного кода. Это позволяет, безболезненно расширять аппаратную базу; использовать данное ПО самостоятельно на домашних компьютерах, несмотря на различные виды операционных систем: Windows, Linux, Mac OS, Solaris и т. д. Открытые исходные тексты приложений позволяют изучать их внутреннюю структуру, а также создавать свои дополнительные модули. Такой подход позволяет повышать функциональность программного обеспечения. Все это положительно сказывается на качестве учебного процесса.

Оправдано использование программы Advanced Grapher в процессе изучения дисциплины «Элементарная математика и практикум по решению задач». Advanced Grapher – это мощная и простая в использовании программа для построения графиков и их анализа. Поддерживает построение графиков функций вида  $Y(x)$ ,  $X(y)$ , в полярных координатах, заданных параметрическими уравнениями, графиков таблиц, неявных функций (уравнений) и неравенств. В программе присутствуют вычислительные возможности: регрессионный анализ, нахождение нулей и экстремумов функций, точек пересечения графиков, нахождение производной, первообразной и т.д. Имеется возможность печати, сохранения и копирования графиков в виде рисунков.

Например, при анализе результатов со студентами решения задания для самостоятельной работы «Найти сумму наименьшего и наибольшего целых решений неравенства» можно использовать программу Advanced Grapher. С ее помощью определяют перечень необходимых знаний и умений, которые позволят прийти к правильному результату: основные свойства функций (логарифмической, показательной и степенной); теорема о корне; способы решения неравенств; область определения

различных функций; формулы сокращенного умножения; преобразования тригонометрических выражений; арифметические действия над целыми числами. Построение графиков функции и определение промежутка решения неравенства (рис.1) позволяет найти правильный ответ — пять. Применение программы Advanced Grapher позволяет реализовывать стимулирующую, коррекционную, развивающую и воспитательную функции контроля.



Текстовый редактор Geany обладает большой производительностью. В нем реализован «вкладочный» интерфейс. Предусмотрена подсветка синтаксиса, которая полезна обучаемым в данной сфере студентам нашего факультета. Программа обладает удобным «классификатором». К примеру, в левой колонке отображает все используемые классы и идентификаторы. Базовый функци-

онал программы можно расширить большим количеством модулей, которые можно найти на официальном сайте.

Все эти возможности можно использовать в обучении дисциплинам «Технологии программирования и методы алгоритмизации», «Методика преподавания информатики». Если в первом случае речь может идти об отработке синтаксиса и формирования навыков создания и реализации типовых и творческих задач, то во втором случае – с методической точки зрения – анализ кода программы на присутствие типичных ошибок, путей их исправления и разработки соответствующей системы методических превенций, которые необходимо сформировать у будущего педагога.

Программа MiniTestSL предназначена для компьютерного тестирования знаний учащихся в процессе обучения. При контроле знаний используются различные задания: выбор одного правильного ответа; множественный выбор; ввод прямого ответа (текст, число, дата); установка правильной последовательности; установка соответствия.

Для обеспечения качественного компьютерного тестового контроля существует гибкая система настроек: многоязыковая поддержка интерфейса программы; сохранение результатов тестирования в базе данных для дальнейшей обработки; не требуется установка. При этом обеспечивается тестирование знаний, как с удаленного компьютера, так и со сменного носителя. Программа проверки знаний предназначена для компьютерного тестирования знаний по различным предметам в процессе обучения.

Работа с данной программой на занятиях различных дисциплин («Элементарная математика и практикум по решению задач», «Технологии



программирования и методы алгоритмизации», «Методика преподавания информатики» и т.д.) позволяют реализовывать практически все, выше-описанные функции контроля.

## ***Using Puppet to reduce Time of Deployment***

*V. Zhhuta*

*UK2 Limited t/a VPS.NET [vitaliyz@vps.net](mailto:vitaliyz@vps.net)*

Deployment, Configuration hundreds of UNIX hosts(including OSX and recently MS Windows) with Puppet. Manage a host throughout its lifecycle: from initial build and installation, to upgrades, maintenance, and finally to end-of-life, when you move services elsewhere.

Puppet[1] is an open source configuration management tool. It is written in Ruby and released under the GPL until version 2.7.0 and the Apache 2.0 license after that. Puppet and Puppetlabs was founded by Luke Kanies in 2005.

Server configuration management (CM) is big news in the IT world these days. Rightly so, because Linux automation, DevOps and CM tools like Puppet and Chef[2] can save you an enormous amount of time and money and help you build a really reliable and automated Linux infrastructure.

If you're a sysadmin, or anyone else who manages a bunch of servers, CM tools can help you create patterns or recipes which you can use to build lots of identical servers, or cloud instances, or re-use in different places and for different applications. Automating Linux servers is a snap with Puppet. Puppet can manage thousands of servers as easily as just one or two - but let's start with one or two

### ***References***

1. <http://projects.puppetlabs.com/projects/puppet>

2. <http://www.opscode.com/chef/>

## ***Мюнхен за рік заощадив 4 млн євро завдяки переходу на Linux: хроніка подій***

*Г. Злобін*

*Львівський національний університет імені Івана Франка,  
[zlobin@electronics.wups.lviv.ua](mailto:zlobin@electronics.wups.lviv.ua)*

The chronology of transfer to free software by city administration of Munich is considered in the report.

У зв'язку з прийняттям Державної цільової науково-технічної програми використання в органах державної влади програмного

забезпечення з відкритим кодом на 2012-2015 роки викликають великий інтерес приклади успішного впровадження ВПЗ в органах влади. Одним із таких прикладів є впровадження ВПЗ в Мюнхені - "Крістіан Уде (Christian Ude), мер Мюнхена (Німеччина), повідомив, що його місту вдалося лише за останній рік заощадити більше чотирьох мільйонів євро за рахунок міграції на вільне програмне забезпечення: операційну систему GNU / Linux і офісний пакет OpenOffice.org.". На жаль, на наш лист із пропозицією виконавцям проекту LiMux надіслати доповідь з описом цього проекту відповіді від них ми так і не отримали. Разом з тим з цей досвід для України є досить цікавим, що й змусило мене впорядкувати інформацію про LiMux із доступних джерел.

У 2005 р. міська рада Мюнхена прийняла рішення про відмову від операційної системи Microsoft Windows на настільних ПЕОМ на користь Debian GNU/Linux. Адміністрація міста заявила, що планує перевести 14000 ПЕОМ під управління безкоштовного Linux-дистрибутива від проекту Debian за допомогою консалтингових компаній Gonicus і Softcon. Петер Хофманн (Peter Hofmann), керівник проекту міграції Мюнхена на Linux, який отримав кодову назву LiMux, зауважив, що отримав величезне число запитів на участь в тендері, і це, як він вважає, доводить, що перехід на Linux на настільних ПЕОМ в комерції вже аж ніяк не незвичайне рішення.

6 вересня 2005 р. прийнято рішення про відтермінування проекту на рік. Тепер перехід 14 тисяч робочих місць з Windows NT 4.0 на Linux і з Microsoft Office 97 і 2000 на OpenOffice.org заплановано на 2006 р. Першим на Linux і OpenOffice.org буде переведений департамент мера міста. Це відбудеться в середині 2006 року, однак не всі 250 комп'ютерів департаменту будуть переведені одночасно. Одна з причин затримки полягає у необхідності додаткового етапу дослідної експлуатації, який планується на перше півріччя 2006 року. «На останньому етапі планування стало ясно, що період дослідної експлуатації важливіше, ніж ми припускали, і повинен тривати довше», сказав Петер Гофман.

Команда міграції планує встановити дослідні ПЕОМ з Linux і OpenOffice.org в кожному департаменті. Після закінчення періоду дослідної експлуатації департаменти будуть переводитися на ВПЗ в один або два етапи. «Деякі відділи розпочнуть з OpenOffice на Windows, а інші відразу будуть переводитися на OpenOffice на Linux, - сказав Гофман. - Це залежить від їх інфраструктури. Наприклад, якщо у відділі використовується невелика кількість простих макросів і шаблонів [Office], але багато складних додатків, його легше спочатку перевести на OpenOffice на Windows».

Співробітники міської адміністрації по-різному реагують на новину про перехід на Linux. За словами П. Гофмана, «деякі бояться, що ніщо не буде працювати, а інші налаштовані оптимістично».

У 2006 р. після річної затримки адміністрація Мюнхена почала, нарешті, перехід робочих місць на Linux - на рік пізніше, ніж планувалося. 100 користувачів мерії вже переведені на Debian GNU / Linux, а до середини 2009 року передбачається охопити проектом 80% ПЕОМ. Шлях адміністрація німецького міста виявився нелегким: вперше про свій намір перейти на Linux вона оголосила в 2003 році, а перший етап планувалося почати в 2005 році. Однак проект LiMux кілька разів відкладався через дебати про патенти на програмне забезпечення, тривалі переговори за контрактом і продовження на 12 місяців пілотної фази проекту.

«Тепер випробування закінчилися. Ми виправили помилки і усунули деякі проблеми, - повідомив в понеділок заступник керівника групи клієнтської Linux Флоріан Шісл (Florian Schiessl). - Все, що ми планували зробити на першій стадії, вже працює ». За його словами, перевести на ВПЗ всіх користувачів неможливо, але 80% до кінця 2008 - середини 2009 року буде переведено.

Перехід буде здійснюватися в один або в два етапи - з проміжною стадією, коли частина з 14 тисяч ПЕОМ буде працювати на OpenOffice під Windows. Рішення про необхідність проміжної стадії буде прийматися на рівні відділів. Debian GNU / Linux була обрана за рекомендацією зовнішніх консультантів, які підготували проект на базі саме цієї конфігурації.

Мюнхен - не єдине європейське місто, в якому співробітники адміністрації переводяться на Linux. У липні на ВПЗ вирішила перейти адміністрація Відня; аналогічні плани є і у норвезького Бергена.

30 березня 2012 р. мер міста Мюнхена Крістіан Уде (Christian Ude) повідомив про успішне завершення проекту переходу міської адміністрації на ВПЗ. Завдяки цьому Мюнхен за рік заощадив 4 мільйонів євро.

Цей фінансовий показник виглядає ще цікавіше у відносних масштабах, оскільки він становить близько третини всього бюджету ІТ-департаменту міста. Проект LiMux - цікавий приклад економії завдяки переходу на вільне ПЗ, в якому вдалося не тільки знизити витрати завдяки відсутності ліцензійних відрахувань і використання більш дешевого устаткування, але і зменшити навантаження на службу технічної підтримки (кількість дзвінків знизилася з 70 до 46 в місяць).

## **Аналіз використання ВПЗ в закладах освіти України**

**Г. Злобін**

*Львівський національний університет імені Івана Франка,*

*[zlobin@electronics.wups.lviv.ua](mailto:zlobin@electronics.wups.lviv.ua)*

The report is on the matters of international scientific-practical conferences FOSS Lviv-2011,2012. The status of usage of free software in educational institutions of Ukraine is analyzed.

Незважаючи на позитивний досвід використання вільного програмного забезпечення (ВПЗ) в освіті у країнах ближнього та далекого зарубіжжя в Україні досі не прийнято відповідної концепції. Водночас зусиллями ентузіастів у закладах освіти України ВПЗ все ж таки використовують. Через відсторонену позицію Міністерства освіти й науки молоді та спорту України немає докладної інформації про досвід використання ВПЗ в освіті. Завдяки тому, що з 2001 р. у Львівському національному університеті імені Івана Франка започаткувалась серія міжнародних науково-практичних конференцій “FOSS Lviv-20xx”, з’явилась можливість проведення аналізу використання ВПЗ у вищих закладах освіти України. Доповіді, подані на ці конференції можна згрупувати за такими напрямками:

Дистанційне навчання

Цій тематиці присвячено чотирнадцять доповідей:

- «Розроблення електронного деканату для системи управління дистанційним навчання MOODLE» - Артеменко В.Б., Львівська комерційна академія
- «Вибір платформи дистанційного навчання» - Коцаренко М.В., Бойко О.В., Львівський національний медичний університет ім. Данила Галицького
- «Використання контрольно-діагностичної програми iTest у ході моніторингу якості процесу навчання старшокласників» - Макаренко І.Є., Мерзлікін П.В., Криворізький державний педагогічний університет
- «Використання системи Moodle для організації контролю знань майбутніми вчителями-гуманітаріями» - Маркова Є.С., Бердянський державний педагогічний університет
- «Тестування в Moodle як елемент менеджменту якості освіти: перший досвід» - Сергієнко В.П., Сліпухіна І.А., національний педагогічний університет ім. М.П. Драгоманова
- «Особливості програмного забезпечення в електронному навчанні» - Жарких Ю.С., Лисоченко С.В., Сусь Б.Б., Третяк О.В., Київський національний університет ім. Т. Шевченка

- «Інформаційно-аналітична система управління навчальним процесом ВНЗ на базі Moodle» - Триус Ю.В., Черкаський державний технологічний університет
  - «Використання CMS JOMLA! та LCMS MOODLE у ВНЗ» - Франчук В.М., національний педагогічний університет ім. М.П. Драгоманова
  - «Локалізація системи MOODLE» - Франчук В.М., національний педагогічний університет ім. М.П. Драгоманова
  - «Застосування вільного програмного забезпечення для дистанційного навчання у вищих навчальних закладах» - Захарченко В.М., Шапо В.М., Одеська національна морська академія
  - «Використання вільно-поширюваного програмного забезпечення для створення систем управління навчальним процесом у ВНЗ -Триус Ю.В., Герасименко І.В. Черкаський державний технологічний університет
  - «Системи дистанційного навчання» - Іванов В. Г., Гвозденко М. В., Кошева Н. А., Національний університет «Юридична академія України імені Ярослава Мудрого»
  - «Соціальні сервіси в дистанційному навчанні» - Артеменко В.Б., Львівська комерційна академія
  - «Використання відкритого програмного забезпечення для організації дистанційного навчання в технічному університеті» - Кривенко О.В., Чичкар'юв Є.А., Державний вищий навчальний заклад «Приазовський державний технічний університет»
- Використання систем комп'ютерної математики  
Дев'ять доповідей можна зарахувати до математичної тематики:
- «Використання вільно-поширюваного ПЗ математичного призначення в університеті» - Бугаєць Н.О., Національний педагогічний університет ім. М.П. Драгоманова
  - «Вільно-поширювані системи комп'ютерної математики в освіті і науці» - Лазурчак І.І., Кобильник Т.П., Дрогобицький державний університет ім. І. Франка
  - «Про досвід використання вільного програмного забезпечення на кафедрі «Обчислювальна математика і програмування» Донецького національного технічного університету» - Алексеев Є.Р., Чеснокова О.В., Чеснокова М.А., Донецький національний технічний університет
  - «Використання комп'ютерних математичних систем у професійній підготовці майбутнього вчителя математики» - Лов'янова І.В., Криворізький державний педагогічний університет
  - «Моделювання задач електротехніки у XCOS» - Філь І.М., Донецький національний технічний університет
  - «Розробка і використання web-інтерфейсів для роботи з системами комп'ютерної математики» - Чичкар'юв Є.А., Приазовський державний технічний університет

- «Про комп'ютерний супровід викладання геометрії» - Яхненко І.В., Лутфулін М.В., Полтавський національний педагогічний університет ім. В.Г. Короленка
  - «Застосування вільних математичних пакетів до розв'язання задач теоретичної механіки» - Періг О. В., Кисіль А. В., Матвеев І. А., Міхенко Д. Ю., Донбаська державна машинобудівна академія
  - «Підтримка шкільної математики вільно-поширюваним ПЗ» - Семеніхіна О.В., Друшляк М.Г., Сумський державний педагогічний університет імені А. С. Макаренка
- Загальні питання використання ВПЗ в освіті
- Цій тематиці присвячено дев'ятнадцять доповідей:
- «Використання вільного програмного забезпечення в навчанні та наукових дослідженнях у Львівському національному університеті імені Івана Франка» - Апуневич С.Є., Злобін Г.Г., Рикалюк Р.Є., Шувар Р.Я., Львівський національний університет ім. І. Франка
  - «Використання вільного програмного забезпечення в системі дистанційної освіти» - Воронкін О.С., Луганський державний інститут культури і мистецтв
  - «Вільно-поширюване програмне забезпечення курсу “Нові інформаційні технології” для студентів спеціальності “Біологія”» - Єфименко В.В., національний педагогічний університет ім. М.П. Драгоманова
  - «Використання вільного програмного забезпечення у професійній підготовці майбутніх інженерів» - Покришень Д.А., Дрозд О.П., Сподаренко І.Й., Чернігівський державний технологічний університет
  - «Про досвід використання ОС Linux у навчальному процесі Львівського національного медичного університету імені Данила Галицького» - Риковський П.А., Львівський національний медичний університет ім. Данила Галицького
  - «LINUX та VIRTUAL-BOX у навчанні абстрактних понять теорії операційних систем» - Спірін О.М., Свєрчевська О.С., Житомирський державний університет ім. І. Франка
  - «З досвіду використання вільного програмного забезпечення при вивченні інформатики» - Харченко В.М., Ніжинський державний університет ім. М. Гоголя
  - «Інформаційні технології в підготовці магістрів фізико-математичних спеціальностей педагогічного університету» - Лазурчак І.І., Кобильник Т.П., Грозовський В.В., Дрогобицький державний педагогічний університет імені Івана Франка
  - «Застосування вільного програмного забезпечення для підготовки спеціалістів зі створення суднових інформаційних систем і комп'ютерних мереж» - Шапо В. Ф., Шевченко Т. І., Одеська національна морська академія, Одеська державна академія будівництва та архітектури



- «Використання електронного навчального курсу математики як засобу підготовки до ЗНО» - В.М. Харченко, Л.В. Ваврикович, І.І. Карпенко, Ніжин НДУ ім. Миколи Гоголя, Ніжинський обласний педагогічний ліцей Чернігівської обласної ради
- «ВПЗ—можливість цікавого навчання від малечі до науковця» - Чоповський С. С., Львівський професійний ліцей залізничного транспорту
- «Використання ВПЗ для тестування апаратного забезпечення ПЕОМ в навчальному процесі факультету електроніки ЛНУ імені Івана Франка» - Батюк А.Я., Злобін Г.Г., Львівський національний університет імені Івана Франка
- «Використання технології віртуалізації в навчальному процесі факультету електроніки ЛНУ імені Івана Франка» - Бойко Я., Ванькевич Д., Злобін Г. Львівський національний університет імені Івана Франка
- «Віртуальний лабораторний практикум на базі GEANT4 в середовищі ОС LINUX для вивчення процесів взаємодії випромінювання з речовиною» -- Малихіна Т.В., Харківський національний університет імені В.Н.Каразіна
- «Огляд дистрибутивів LINUX, створених на базі DEBIAN/UBUNTU, що використовуються у сфері освіти» - Смалько О.А., Кам'янець-Подільський національний університет імені Івана Огієнка
- «Використання відкритого програмного забезпечення у науково-дослідних роботах і навчальному процесі кафедри комп'ютерних систем та мереж» - Луцків А.М., Тернопільський національний технічний університет імені Івана Пулюя
- «Про досвід використання офісного пакету OpenOffice.org.ukr в курсі інформатики для економічних і юридичних спеціальностей ВЗО» - Злобін Г.Г., Львівський національний університет ім. І. Франка
- «Досвід використання редактора Gimp при вивченні курсу “Комп'ютерна графіка і дизайн» - Матвієнко Ю.С., Полтавський національний педагогічний університет ім. В.Г. Короленка
- «Emacs Org-Mode у навчанні та науковій роботі» - Булка Ю.А., Львівська національна музична академія ім. Миколи Лисенка

Використання відкритих засобів програмування

Ця група нараховує чотири доповіді:

- «Використання відкритих програмних засобів в процесі навчання статистичним дисциплінам» - Коркуна Т.Й., Самбірський технікум економіки та інформатики
- «Построение практикумов по программированию и архитектуре ЭВМ на базе GNU/LINUX» - Костюк Д.А., Брестський державний технічний університет

- «Розрахунок фотоіонізаційних моделей небулярного газу в ОС LINUX UBUNTU 10.10 та WINDOWS 7» - Мелех Б.Я., Тишко Н.Л., Коритко Р.І., Львівський національний університет ім. І. Франка
- «Статический анализ ПО с помощью его промежуточных представлений и технологий с открытым кодом» - Зубов М.В., Пустыгин А.Н., Старцев Е.В., Челябинский государственный университет

Паралельні обчислення на EOM

У цій групі є п'ять доповідей:

- «Реалізація розподілених обчислень на основі ґрід-платформи з відкритим кодом BOINC» - Шийка Ю.Я., Шувар Р.Я., Львівський національний університет ім. І. Франка
- «Реалізація високопродуктивної обчислювальної системи на базі ОС LINUX» - Шувар Р.Я., Бойко Я.В., Львівський національний університет ім. І. Франка
- «Використання обчислювального кластера з використанням ОС Linux у вищому навчальному закладі» - Сідун Н. М., Чичкарьов Є. А., Приазовський державний технічний університет
- «Про досвід використання компілятора gcc та технології OpenMP при розв'язанні задач обчислювальної математики» - Алексеев Є.Р., Чеснокова О.В., Савченко М.А., Хохлова А.А. Донецький національний технічний університет
- «Виконання завдань розподіленої обробки зображення під управлінням системи CONDOR» - Шийка Ю.Я., Шувар Р.Я., Львівський національний університет імені Івана Франка

Розробка програмного забезпечення

Цій тематиці посвячено дев'ять доповідей:

- «Комплекс програм для лазерних спостережень штучних супутників Землі» - Мартинюк-Лотоцький К.П., Білінський А.І., Львівський національний університет ім. І. Франка
- «Розробка системи спектральної діагностики димової плазми» - Сподарець Д.В., Драган Г.С., Одеський національний університет імені І.І. Мечнікова
- «Використання вільного програмного забезпечення для створення програми керування інформаційним автоматом» - Злобін Г., Скляр В., Чмихало О., Шевчик В., Львівський національний університет ім. І. Франка
- «Використання бібліотеки класів GEANT4 в ОС Linux у розробці програмного забезпечення для моделювання процесів взаємодії випромінювання з речовиною» - Малихіна Т.В., Харківський національний університет ім. В.Н. Каразіна

- «Використання крос-платформного інструментарію розробки програмного забезпечення Qt для створення навчальних програм» - Рудий М.Ф., Львівський національний університет імені Івана Франка
- «Використання вільного ПЗ для управління телескопом» - Мартинюк-Лотоцький К.П., Львівський національний університет імені Івана Франка, астрономічна обсерваторія
- «Новий підхід до створення INITRAMFS» - Сітало В.С., Ясько М.М., Дніпропетровський національний університет імені Олеся Гончара
- «ALT LINUX как пример свободной платформы» - А.Е. Новодворский, ООО «Альт Линукс»
- «Автоматизовані системи створення та супроводу пакетів у дистрибутиві ALT LINUX. Крок у майбутнє» - І. Ю. Власенко, ALT Linux Team

Аналіз використання ВПЗ в закладах освіти

У цій групі є три доповіді:

- «Использование свободного программного обеспечения в университетах и исследовательских учреждениях Российской Федерации» - Алексеев Е.Р., Брагилевский В.Н. Донецкий национальный технический университет, Южный федеральный университет,
- «FREE/LIBRE SOFTWARE USAGE IN THE BELARUSIAN SYSTEM OF HIGHER EDUCATIONAL INSTITUTIONS Kostjuk D.A., Pynkin D.A. Brest State Technical Institute, 2Belarusian State University of Informatics and Radioelectronics
- Аналіз використання вільного програмного забезпечення в закладах освіти України - Г. Злобін, Львівський національний університет імені Івана Франка

Використання вільного апаратного забезпечення

Автоматизація фізичного експерименту засобами фірми Cypress Semiconductor - Бублик П. І., Ткач О.І., Злобін Г.Г., Львівський національний університет імені Івана Франка

Окремі доповіді

І нарешті, вісімнадцять доповідей, які не вдалося віднести до жодного із перерахованих вище напрямків:

- «Інформаційна технологія управління навчальним навантаженням у вищих навчальних закладах» - Гриценко В.Г., Черкаський національний університет ім. Б. Хмельницького
- «Використання програми GANTPROJECT для побудови календарних графіків при розробці ПВР» - Грицук Ю.В., Меліхов О.І., Донбаська академія будівництва і архітектури

- «Вільне ПЗ для підготовки наукових текстів і презентацій» - Лутфулін М.В., Моторний М.І., Полтавський національний педагогічний університет ім. В.Г. Короленка
- «Досвід впровадження відкритого програмного забезпечення в стартапах компанії Медіа-Ключ» - Зелінський О.Л., Компанія Медіа-Ключ
- «Поштовий сервер IREDMAIL в бізнесі». - Сейдаметов Г. С., Кримський інженерно-педагогічний університет
- «Обзор возможностей существующего открытого ПО для работы с музыкальными данными» - Чубарьян А.С., Федеральное государственное автономное образовательное учреждение высшего профессионального образования «ЮЖНЫЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»
- «Технологія AJAX у інформаційних системах автоматизованого збору та опрацювання даних». - Словінський О.В., Словінська О.Д., Житомирський державний університет імені Івана Франка
- «Спотворення спектрів сигналів внаслідок джиттеру частоти дискретизації» - О.М. Яремчук, С.П. Дзюба, м. Миколаїв, Чорноморський державний університет
- «Правові аспекти використання ІКТ у навчальних закладах» - Хачіров Т.С., Харківський національний педагогічний університет ім. Г.С. Сковороди
- «Перспективи розвіцця свабоднаго праграмнага забезпечэння у беларускіх “аблаках”» - Карагода А.І., Стальбоўская Н.І., Беларускі нацыянальны тэхнічны універсітэт
- «SCIENTIFIC INQUIRY BASED EDUCATION AND FOSS» - Levitchi A., Duca M., University of the Academy of Sciences of Moldova, Republic of Moldova
- «MOZILLA -- вчора, сьогодні, завтра» - Бабич Т.Ю., Mozilla Україна
- «Використання вільного програмного забезпечення при створенні науково-освітніх електронних бібліотек» - Прилуцка Н. С., Житомирський державний університет імені Івана Франка
- «Організація Інтернет-доступу до віддалених баз даних» - Столярчук О.В., Шувар Р.Я., Продивус А.М., Львівський національний університет імені Івана Франка
- «Досвід та преспективи використання вільного ПЗ у видавничо-поліграфічній галузі» - Піскозуб Й. З., Миклушка І. З. Піскозуб Л.Г., Українська академія друкарства
- «Дистрибутив ZENTYAL для побудови серверів для малого та середнього бізнесу» - Сало М., Ванькевич Д., Львівський національний університет імені Івана Франка
- «Переваги використання вільного ПЗ в організації наукової діяльності університету» - Ткаченко Н.О., Донецький національний технічний університет

- «Перехід університетів України на вільне програмне забезпечення» - Тверезовська Н. Т., Ліпська В. В., Національний університет біоресурсів і природокористування України

Отже можна констатувати як широкий спектр використання ВПЗ в українських закладах освіти від дистанційного навчання до розробки відкритого програмного забезпечення, так і широку географію використання ВПЗ від Луганська на сході до Львова на заході та від Чернігова на півночі до Одеси на півдні (див. Рис. 1).



***Статический анализ ПО с помощью его промежуточных представлений и технологий с открытым кодом***  
Зубов М.В., Пустыгин А.Н., Старцев Е.В.

*Челябинский Государственный Университет, p2008an@rambler.ru.*

This paper presents current existing approaches of using programs' representations for static analysis. This analysis can help with understanding and extending open-source software. The most common approaches were implemented in prototypes with open technologies. Also class-level detailed representation was developed and implemented in prototype to prove the assumption of more effective analysis on less detailed representation.

Статический анализ исходного кода с использованием методов обработки текста может быть достаточно неэффективным. На практике инструменты, выполняющие статический анализ, обычно формируют промежуточные представления исходного кода. Самое распространенное решение - использование абстрактного дерева разбора (AST) [1]. Его можно получать различными способами, в том числе инструментами генерации парсеров, такими как ANTLR[2] (Checkstyle[3] для языка Java), либо специфичными инструментами, такими как API компилятора ROSE (Compass/ROSE)[4], поддерживающего языки C, C++ и Fortran. В проекте Pylint[5] (для языка Python) для генерации AST используется отдельный свободно-распространяемый модуль logilab-astng [6], который может быть использован независимо от Pylint. Были разработаны 2 прототипа генератора промежуточных представлений, основанных на AST: первый - на основе свободного модуля logilab-astng[6] для языка Python, второй - на основе компилятора javac, входящего в состав открытой реализации средств разработки на Java - OpenJDK[7].



Принципиально другой вариант промежуточного представления - реляционное представление исходного кода анализируемого ПО. Для получения такого представления по грамматике исходного языка подготавливается схема реляционной базы данных, далее созданная по этой схеме база данных заполняется в соответствии с конструкциями исходного кода. В результате, процесс анализа сводится к выполнению различных запросов к этой базе. В современных инструментах используются SQL-подобные языки QUEL[8] (в OMEGA[9] для Ada, C и Pascal) и .QL [10] (SemmlCode[11] для Java). В настоящее время ведется создание прототипа построителя реляционного промежуточного представления для языка Java.

Концепция реляционного представления расширяется с помощью логических языков (Prolog в ASTLOG[12], Datalog в CodeQuest[13]). Эти языки гораздо больше подходят для анализа структур, которыми представляется исходный код, например, транзитивных замыканий. Запросы к базе знаний транслируются в SQL и выполняются на конкретной базе данных, либо же база данных может быть представлена в виде базы знаний.

Отдельно от реляционного представления выделяется направление статического анализа, просто основанного на выполнении запросов к исходному коду. Обычно такой подход не накладывает ограничений на внутреннее представление. Оно может быть как реляционным в системе CodeQuest[13], так и AST в ASTLOG[12]. Некоторые заявляют о потенциально любом представлении (PQL[14]). В основном, в концепции запросов к исходному коду используют логические языки как наиболее удобные и нацеленные на извлечение знаний. Однако в пользу удобства и гибкости существуют анализаторы, выполняющие запросы на естественном языке (английском)[15].

Зачастую промежуточное представление определяется целью анализа, как в случае с концепцией запросов. Так в свободном проекте BLAST [16] программа представляется виде автомата потока управления (CFA), описывающего набор состояний процедур и переходы между ними. Далее в соответствии с алгоритмом “абстракция и уточнение по контрпримерам” создаются абстрактные деревья достижимости (ART), которые описывают возможные пути распространения программы с той долей абстракции, которая используется на очередной итерации алгоритма.

Независимо от различных представлений можно выделить модель анализа с помощью моделей отражения[17]. Она основана на сравнении высокоуровневой модели программы, которую программист мыслит себе в процессе разработки, и реальной, полученной в результате. Этот подход не требует конкретного представления кода, и иногда может быть выполнен с использованием текстовой обработки. Так, в частности, он был реализован в одном из первых опытов применения подхода для изучения подсистемы виртуальной памяти NetBSD[17]. На основе этого подхода с помощью собственного генератора промежуточного представления для

Python, использующего logilab-astng[6], была построена и проанализирована высокоуровневая модель проекта с открытым исходным кодом на Python.

Промежуточные представления могут строиться как одноязыковые, так и мультязычные. Универсальные решения позволяют производить типовые процедуры анализа для различных языков, однако не все детали конкретного языка в этом случае могут быть представлены. В качестве прототипа универсального представления было реализовано представление уровня структуры классов для языков Python и Java на основе прототипов генерации частных представлений.

Помимо указанных можно предложить классификацию представлений по уровню детализации. Так в академическом проекте Bauhaus project[18] для языков Ada, C, C++, C# и Java имеются 2 представления - низкоуровневое InterMediate Language, описывающее конструкции языка на синтаксическом и семантическом уровнях, и Resource flow graphs, описывающее архитектурные особенности программной системы. В анализаторе, использующем PQL[14], выделяется 3 уровня анализа: модель глобального программного дизайна, модель структуры программы, модель детального программного дизайна. Для созданного прототипа универсального представления для Python и Java был выбран средний уровень детализации - уровень структуры и связи классов.

При работе с любым представлением всегда важно иметь возможность соотнести анализируемый объект или область программы с ее местом в исходном коде. Это достигается сохранением координат объектов в представлениях, как и было реализовано в универсальном представлении для Java и Python. Полученное высокоуровневое представление позволяет производить анализ на уровне классов и архитектуры проекта, что позволяет снизить накладные расходы на выполнение анализа по сравнению с более детализированным представлением. За счет универсальности категорий ООП, существующих во всех объектных языках[19], полученное представление является полным для своего уровня анализа и, в то же время, универсальным для объектно-ориентированных языков.

### **Литература**

1. Ахо А.В., Ульман Д.Д. / *Компиляторы. Принципы, технологии и инструментари.* / М.: Вильямс, 2008г.
2. ANTLR Parser Generator / [www.antlr.org](http://www.antlr.org)
3. Checkstyle – Checkstyle 5.5 / [checkstyle.sourceforge.net](http://checkstyle.sourceforge.net)
4. ROSE / [rosecompiler.org](http://rosecompiler.org)
5. Pylint (analyzes Python source code looking for bugs and signs of poor quality) / [www.logilab.org/857](http://www.logilab.org/857)
6. logilab-astng (Python Abstract Syntax Tree New Generation) / [www.logilab.org/856](http://www.logilab.org/856)
7. OpenJDK / [openjdk.java.net](http://openjdk.java.net)

8. Ingres 10.0 QUEL Reference guide / Ingres corp. 2010, [code.ingres.com/ingres/main/src/tools/techpub/pdf/QUELRef.pdf](http://code.ingres.com/ingres/main/src/tools/techpub/pdf/QUELRef.pdf)
9. M. Linton / *Queries and Views of Programs Using a Relational Database System* / [www.eecs.berkeley.edu/Pubs/TechRpts/1983/5296.html](http://www.eecs.berkeley.edu/Pubs/TechRpts/1983/5296.html)
10. O. de Moor, E. Hajiyeve, M. Verbaere / *Object-oriented queries over software systems* / ACM Press, 2007.
11. *Enabling tools for your project* | SemmleCode / [semml.com/solutions/enabling-tools-for-your-projects](http://semml.com/solutions/enabling-tools-for-your-projects)
12. R. Crew / *ASTLOG: A Language for Examining Abstract Syntax Trees* / Microsoft Research, 1997.
13. E. Hajiyeve / *CodeQuest - Source Code Querying with Datalog* / St. Anne's College, Oxford University, 2005.
14. S. Jarzabek / *Design of Flexible Static Program Analyzers with PQL* / IEEE Transactions on software engineering, vol. 24, no. 3, march 1998.
15. M. Kinmming, M. Monperrus, M. Mezini / *Quering Source Code with Natural Language* / 26th IEEE/ACM International Conference On Automated Software Engineering (ASE'2011).
16. MTC (Models and Theory of Computation): BLAST Project / [mtc.epfl.ch/software-tools/blast/index-epfl.php](http://mtc.epfl.ch/software-tools/blast/index-epfl.php)
17. G.C. Murphy, D. Notkin, K. Sullivan / *Software Reflexion models: Bridging the gap between source and high-level models* / Third ACM SIGSOFT Symposium on the Foundations of Software Engineering, pages 18–28, New York, ACM Press, 1995.
18. Project Bauhaus. *Software Architecture, Software Reengineering, and Program Understanding* / [www.bauhaus-stuttgart.de/bauhaus/index-english.html](http://www.bauhaus-stuttgart.de/bauhaus/index-english.html)
19. И. Грэхем / *Объектно-ориентированные методы. Принципы и практика* / М.: Вильямс, 2004 г.

Збірник наукових праць

МАТЕРІАЛИ ДРУГОЇ МІЖНАРОДНОЇ  
НАУКОВО-ПРАКТИЧНОЇ КОНФЕРЕНЦІЇ  
FOSS LVIV 2012

Під редакцією:  
Злобін Г.Г., Апуневич С.Є.,  
Апуневич С.В., Ванькевич Д.Є.

Підписано до друку 20.04.2012  
Формат 60x84/16. Папір офсетний. Цифровий друк.  
Гарнітура Times.  
Фіз. друк. арк. 10,125 Умов. друк. арк. 9,42  
Тираж 100.

Видавець:  
Приватний підприємець Сорока Тарас Богданович  
79060, м. Львів, вул. Наукова, 41/18  
Свідоцтво державного реєстру: серія ЛВ №17  
soroka@soroka.lviv.ua