

Зміст

Softserve -- співорганізатор та генеральний спонсор.....	5
"Техніка для бізнесу" - одна з найбільших ІТ компаній регіону.	6
Про досвід використання вільного програмного забезпечення на кафедрі «обчислювальна математика і програмування» Донецького національного технічного університету.....	7
Програма для обробки результатів ЛЛС-спостережень як ВПЗ.....	10
Розроблення електронного деканату для системи управління дистанційним навчанням MOODLE (на прикладі Львівської комерційної академії).....	11
Використання середовища VUE для формування інформатичних компетентностей учнів.....	14
Програмне забезпечення керування телескопом ЛЛС-станції “Львів-1831”.....	17
ВІКІ-портал Кіровоградського державного педагогічного університету.....	19
Використання вільно-поширюваного ПЗ математичного призначення в університеті.....	21
Вільне програмне забезпечення чи "CLOUD COMPUTING" у навчальному процесі?.....	25
Використання вільного програмного забезпечення в системі дистанційної освіти. 28	
Концепція вільного програмного забезпечення в освітньому менеджменті.....	31
Впровадження ВПЗ в проектних інститутах будівельної галузі України.....	33
Система підтримки прийняття рішень з вибору мобільної платформи на програмних засобах з відкритим кодом.....	36
Інформаційна технологія управління навчальним навантаженням у вищих навчальних закладах.....	38
Використання програми GANTTPROJECT для побудови календарних графіків при розробці ПВР.....	41
Разработка док-панели в виде плагина расширения COMPIZ-FUSION.....	44
До питання вивчення проблеми впровадження вільного програмного забезпечення в загальноосвітніх школах.....	46
Теоретичні та методологічні основи розробки автоматизованих систем передачі знань	48
Вільнопоширюване програмне забезпечення курсу «Нові інформаційні технології» для студентів спеціальності «Біологія».....	51
Опрацювання результатів спостережень проходження Венерою диска Сонця 8 линя 2004 року та розрахунки обставин майбутніх астрономічних явищ.....	54
Вільна система управління мережею NOC PROJECT	57

Про досвід використання офісного пакету OPENOFFICE.ORG в курсі інформатики для економічних і юридичних спеціальностей ВЗО.....	59
Використання ВПЗ для підтримки вивчення математичної логіки.....	61
PINGUINO и персональний генератор ентроpii.....	63
Віртуальний лабораторний практикут у процесі вивчення фізики.....	66
Проблеми популяризації вільного і відкритого програмного забезпечення в Україні	69
Використання відкритих програмних засобів в процесі навчання статистичним дисциплінам.....	71
Построение практикумов по программированию периферийных устройств и архитектуре ЭВМ на базеGNU/LINUX.....	73
Вибір платформи дистанційного навчання.....	76
Інституційний репозитарій університету як середовище інтеграції наукового знання.....	78
Відкриті проекти в області авіоніки.....	80
Вільнопоширювані системи комп'ютерної математики в освіті та науці.....	81
Створення спеціалізованих рішень на базі продуктів компанії D-LINK.....	84
Використання комп'ютерних математичних систем у професійній підготовці майбутнього вчителя математики.....	85
Вільне ПЗ для підготовки наукових текстів та презентацій.....	88
Використання контрольної-діагностичної програми iTEST у ході моніторингу якості процесу навчання старшокласників.....	91
Використання бібліотеки класівв GEANT4 в ОС LINUX при розробці програмного забезпечення для моделювання процесів взаємодії випромінювання з речовиною.....	94
Використання системи Moodle для організації контролю майбутніми вчителями-гуманітаріями.....	96
Комплекс програм для лазерних спостережень штучних супутників Землі.....	99
Компьютеризация процесса обучения	102
Досвід викладання редактора GIMP при вивченні курсу «Комп'ютерна графіка та дизайн».....	104
Міграція на вільне програмне забезпечення.....	107
Відкриті ліцензії і стандарти.....	109
Розрахунок фотоіонізаційних моделей світіння небулярного газу від ОС LINUX UBUNTU 10.10 та WINDOWS 7.....	112
RunaWFE - свободная система управления бизнес-процесами и административными регламентами.....	115

Використання OpenWrt як основи вбудованого програмного забезпечення маршрутизаторів.....	118
sK1 Project: Перезавантаження (перспективи розвитку sK1 і UniConverto).....	120
Создание общедоступной образовательной среды школ РФ на основе свободного программного обеспечения, 2007-2010. Взгляд участника проекта.....	122
Програмний комплекс моделювання процесу функціонування космічних систем	125
Використання вільного програмного забезпечення у професійній підготовці майбутніх інженерів.....	127
Інтелектуальна власність, впрз та світова економіка ноосферне та індустріальне виробництво: конкурентні підходи.....	130
Про досвід використання ОС LINUX у навчальному процесі Львівського національного медичного університету імені Данила Галицького.....	134
Використання вільного програмного забезпечення в навчанні і наукових дослідженнях у Львівському національному університеті імені Івана Франка.....	135
Помилки адресації в електронних мережах та методи їх корегування.....	137
Використання Скретч на уроках інформатики.....	140
Використання вільного програмного забезпечення для створення програми керування інформаційним автоматом.....	142
Тестування в Moodle як елемент менеджменту якості освіти: перший досвід.....	147
LINUX та VIRTUALBOX у навчанні абстрактних понять теорії операційних систем.....	149
Стан українського FOSS-сегменту на кінець 2010 року.....	152
Розробка системи спектральної діагностики димової плазми.....	154
Особливості програмного забезпечення в електронному навчанні.....	156
Обеспечение учебного процесса по курсу «Информатике и ИКТ» в образовательных учреждениях в условиях внедрения СПО.....	159
Інформаційно-аналітична система управління навчальним процесом ВНЗ на базі MOODLE.....	162
Використання CMS JOOMLA! та LCMS MOODLE у ВНЗ.....	165
Моделювання задач електротехніки у XCOS.....	168
Локалізація системи MOODLE.....	170
З досвіду використання вільного програмного забезпечення при вивченні інформатики у виші.....	172
Програмный комплекс для расчета процессов переноса водорода в металлах....	174
Розробка і використання web-інтерфейсів для роботи з системами комп'ютерної математики	176

Застосування вільного програмного забезпечення для дистанційного навчання у вищих навчальних закладах.....	178
Реалізація розподілених обчислень на основі грид-платформи з відкритим кодом VOINC.....	181
Реалізація високопродуктивної обчислювальної системи на базі ОС LINUX.....	184
Використання MACROMEDIA SHOCKWAVE PLAYER під час вивчення атомної та ядерної фізики студентами технічних спеціальностей.....	185
Про комп'ютерний супровід викладання геометрії.....	188
The Open Data, Open Society report.....	191
Why and how FOSS scripting languages are important at school and at home.....	192
How to survive the GNU AUTOTOOLS.....	193
Otwarta infrastruktura przetwarzania rozproszonego berkeley – wspomaganie międzynarodowych obliczeń naukowych.....	193

SoftServe -- співорганізатор та генеральний спонсор

Компанія SoftServe є однією з найуспішніших компаній Східної Європи в галузі розробки програмного забезпечення. Починаючи із 1993 року, SoftServe вдало здійснив понад 2 000 проектів. Нашими клієнтами стали понад сто компаній із усього світу, серед яких відомі бренди США, Канади, Західної Європи. Представництва SoftServe діють у Києві, Москві, а також у Бостоні, Лос-Анджелесі та Флориді (США). Компанія SoftServe має свої розробницькі центри у Львові, Дніпропетровську, Рівному, Івано-Франківську, Чернівцях, Севастополі та Манілі (Філіппіни), де працює понад 1600 професіоналів у різних сферах ІТ.

Формування дружньої команди у компанії таких масштабів – нелегка справа. Це відзначили міжнародні асоціації трудових ресурсів:

– 2007 рік - SoftServe зайняв 7 місце у світовому рейтингу “Десять лідерів у сфері розвитку людських ресурсів” від Global Services 100.

– 2010 рік - Компанія ввійшла до категорії Rising Star у рейтингу

– 2010 SoftServe "Найкращий Роботодавець України - 2010" серед ІТ компаній за версією Hewitt Associates і HR Center.

Наша компанія сприяє кар’єрному та особистісному розвитку кожного, хто потрапив у команду.

SoftServe став першою ІТ-компанією в Україні, яка відкрила власний навчальний заклад – SoftServe University (навчання за програмами Soft-Serve IT Academy, Microsoft IT Academy, школи іноземних мов та здійснює сертифікацію програмістів за міжнародними стандартами).

Наша компанія активно долучається до розвитку ІТ-інфраструктури в Україні і є співзасновником асоціації IT Ukraine. Серед офіційних партнерів SoftServe є такі компанії, як Oracle, Cisco, HP тощо.

У компанії діють системи якості, які відповідають міжнародним стандартам ISO і CMMI.

Головна ознака компанії – динамічність. Саме завдяки такій характеристиці ми завжди відповідаємо вимогам мінливого ринку і залишаємось найкращими!

"Техніка для бізнесу" - одна з найбільших ІТ компаній регіону.

Вона першою серед західноукраїнських компаній відкрила напрямок системної інтеграції в регіоні та надає послуги з ІТ-консалтингу, що забезпечує клієнтам максимальну віддачу від використання сучасних технологій, комп'ютерного обладнання та програмного забезпечення.

"Техніка для бізнесу" сприяє розвитку бізнесу своїх клієнтів: впроваджує найновіші інформаційні технології, використовує професіоналізм та досвід фахівців. Професійна підтримка гарантійного та післягарантійного обслуговування комп'ютерної й офісної техніки на базі власного сервісного центру – ще одна з численних переваг.

Напрямки діяльності ТзОВ «Техніка для бізнесу»:

- проектування та виконання складних високотехнологічних ІТ- проектів,
- аутсорсінг мереж та серверного обладнання,
- гуртова торгівля комп'ютерною, цифровою та офісною технікою;
- комплексні інтегровані ІТ та телекомунікаційні послуги для бізнесу: постачання комп'ютерів, офісної оргтехніки, офісних та відомчих АТС, проектування, встановлення, монтаж структурованих кабельних систем, організація корпоративних мереж через Internet/Intranet,
- сертифіковане виробництво комп'ютерів Technic-PRO.

У 2009 компанія здобула сертифікат, котрий засвідчує, що система управління якістю стосовно виробництва комплектів базових комп'ютера та сервера «Technic-Pro», відповідає вимогам ДСТУ ISO 9001:2009.

«Надійні рішення – гарантований успіх» - саме це гасло є визначальним у діяльності компанії.

Про досвід використання вільного програмного забезпечення на кафедрі «обчислювальна математика і програмування» Донецького національного технічного університету

Алексєєв Є.Р., Чеснокова О.В., Чеснокова М.А.

Донецький національний технічний університет, EAlekseev@gmail.com

Представлений досвід впровадження вільного програмного забезпечення на кафедрі ОМіП ДонНТУ, проаналізовано переваги використання вільних програм в університетах та проблеми, що виникають при цьому. Запропоновано шляхи впровадження вільного програмного забезпечення.

Багато років в університетах СНД пропрієтарні програми «де-факто» вважалися стандартом інформаційної підготовки студентів. Однак, в останнє десятиліття ситуація стала змінюватися, чому сприяв бурхливий розвиток вільного програмного забезпечення (ВПЗ). Сучасні дистрибутиви вільних операційних систем (Linux Mint, Ubuntu Linux, Alt Linux, Edu Mandriva) можуть використовуватися як в якості настільної операційної системи (ОС) на комп'ютері вченого, викладача та студента, так і в якості ОС в лабораторіях університетів. До переваг ОС сімейства Linux слід віднести невисокі апаратні вимоги, стабільність, безпеку, величезну кількість програм різного профілю. Досвід використання авторами сучасних дистрибутивів ОС ALT Linux, Debian та Ubuntu на комп'ютерах випуску кінця ХХ століття говорить про те, що для організації навчального процесу краще застосовувати ОС сімейства Linux, ніж ОС Windows.

Однак, крім самої операційної системи, важливо підібрати сучасне програмне забезпечення (ПЗ) для навчального процесу та наукових досліджень, яке буде працювати не тільки в навчальних лабораторіях, але і на комп'ютерах студентів і викладачів. Вибір як ПЗ кросплатформних програм дає змогу не прив'язуватися до конкретної ОС при організації навчального процесу. Кафедра ОМіП ДонНТУ вже не один десяток років займається комп'ютерною підготовкою студентів загальноінженерних та економічних спеціальностей на молодших курсах. Для навчання студентів використовуються наступні основні класи прикладних програм: офісні програми; програми для роботи в Інтернеті; компілятори; математичні програми.

Як кросплатформні вільні офісні програми можна використовувати OpenOffice.org, dia і ряд інших. Вільні програми сімейства Mozilla і багатопротокольную програму обміну миттєвими повідомленнями Pidgin можна рекомендувати як програми для роботи в Інтернеті.

Під час навчання майбутніх інженерів особливу роль відіграють засоби розробки програм і математичні програми, які дають змогу розв'язувати практичні і дослідницькі завдання.

Якщо у навчанні програмуванню використовують Basic, Pascal, C (C++), то які є альтернативи традиційним пропрієтарним засобам розробки?

Для програмування мовою Basic можна використовувати Gambas або

OpenOffice.org Calc. Застосування OpenCalc дає змогу отримати потужний засіб для розв'язання інженерних та економічних завдань: табличний процесор і візуальна об'єктно-орієнтована мова OpenOffice.org Basic.

Якщо навчання проходить на Pascal, то можна використовувати Free Pascal та Lazarus, а також спеціалізований текстовий редактор Geany і один з компіляторів Free Pascal та Gnu Pascal. На кафедрі ОМіП написаний підручник з програмування на базі Free Pascal та Lazarus [1], виданий у Москві та Донецьку.

Для програмування мовою C (C++) як компілятор можна використовувати gcc (g++), а в ролі середовища програмування в ОС Linux - текстовий редактор Geany, а при використанні ОС Windows - dev-cpp. Якщо завданням навчання є підготовка професійних програмістів, то як середовище програмування доцільно використовувати візуальне кросплатформне середовище QtCreator.

Особливу роль для підготовки інженерів відіграють математичні програми. Вони використовуються під час вивчення загальноосвітніх курсів: математики, інформатики, ТОЕ, опору матеріалів тощо, а також в багатьох спеціальних предметах для розв'язання реальних завдань.

В якості вільних математичних програм для університетів можна запропонувати Scilab, Maxima, Octave.

На кафедрі ОМіП ДонНТУ розроблено курс інформатики на базі пакету Scilab, в якому навчання програмуванню і розв'язанню інженерних задач проходить в єдиному середовищі Scilab. На базі цього курсу написана книга «Scilab: Решение инженерных и математических задач»[2].

У цьому навчальному році один з авторів веде практичні заняття з курсу «Вища та прикладна математика», в якому крім класичного розв'язання математичних завдань «на папері», використовуються і вільні математичні програми Scilab, Maxima і Octave. Для вивчення розділу «Лінійна алгебра та аналітична геометрія» як основний пакет використовується Scilab. Для вивчення математичного аналізу використовується програма Maxima. В якості навчальних посібників з вільних математичних програм використовуються [2] і відмінна книга по Maxima [3] Є.А. Чічкарева з Приазовського Технічного Університету. У завершальній частині курсу, присвяченій чисельному розв'язанню математичних завдань планується спільно використовувати Maxima і Scilab.

Варто відзначити, що розглянуті програмні засоби є не тільки засобами навчання студентів, але й повноцінними програмами для наукових досліджень і розв'язання прикладних завдань.

При всіх перевагах використання вільного забезпечення, впровадження ВПЗ в університетах проходить складно. Спробуємо об'єктивно оцінити переваги і недоліки використання вільних програм.

Переваги:

1. Університет отримує сучасну захищену і стабільну операційну систему з великою кількістю якісних програм, призначених, як для навчання студентів, так і для дослідницької діяльності. При цьому заощаджуються бюджетні кошти.

2. Університет відходить від диктату великих ІТ-компаній монополістів.
3. Викладач отримує можливість навчати студента, а не “натягувати” його на розв’язанні однотипних завдань у певних пакетах.
4. Студент отримує можливість працювати індивідуально, швидко опановувати досліджуваним матеріалом, творчо мислити і експериментувати.

Проблеми:

1. Необхідність постійного вивчення нового ПЗ викладачами, студентами та співробітниками університетів.
2. Недостатня кількість не тільки методичної, але і взагалі літератури, її потрібно писати.
3. При використанні вільного програмного забезпечення в освіті дуже складно буде готувати середнього фахівця («мишкоклацальника») і виживати середнім студентам і середнім викладачам.

Для впровадження ВПЗ в закладах освіти необхідне рішення на державному рівні. З іншого боку університетам не треба чекати, а починати готувати фахівців з вільного програмного забезпечення.

Прийшов час відкрити кафедру «Вільне програмне забезпечення», яка стала б з одного боку методичним центром з впровадження та використання ВПЗ, а з іншого боку стала б цілеспрямовано готувати фахівців з цього напрямку ІТ-галузі.

Література

1. Е. Р. Алексеев, О.В. Чеснокова, Т.В. Кучер. Free Pascal и Lazarus: учебник по программированию. – М.: ALT Linux, 2010. – 438с.
2. Е. Р. Алексеев, О.В. Чеснокова, Е.А. Рудченко. Scilab: Решение инженерных и математических задач. – М.: Бином, 2008. – 260с.
3. Е.А. Чичкарёв. Компьютерная математика с Maxima: Руководство для школьников и студентов. URL: <http://www.altlinux.org/Books:Maxima> (Дата обращения 2.01.2011).

Програма для обробки результатів ЛЛС-спостережень як ВПЗ

Апуневич С.Є., Апуневич С.В., Білінський А.І., Благодир Я.Т.

*Астрономічна обсерваторія, Львівський національний університет імені Івана Франка, apusbird@gmail.com, slr1831@ukr.net
Кафедра фізики Львівський національний університет ветеринарної медицини та біотехнологій, sofiya.apuneych@gmail.com*

Представлено варіант реалізації програми попередньої обробки результатів спостережень штучних супутників Землі на ЛЛС "Львів". Нова програма є відкрита в сенсі відкритих вихідних текстів, що дає можливість максимально адаптувати її до потреб ЛЛС-станції та моментально реагувати на запити Міжнародної служби лазерної локації. Такі особливості програми дозволять без особливих проблем налаштувати її на іншій ЛЛС-станції. Програма працює під операційною системою Linux та написана мовою С.

Лазерна локація штучних супутників Землі (Satellite Laser Ranging, SLR) -- один з найважливіших методів геодезичних та геофізичних досліджень. Спостереження в лазерній локації полягає у вимірюванні часу, протягом якого надкороткий лазерний імпульс подорожує від наземної станції до супутника та у зворотному напрямі (віддалеметрія). Станція лазерної локації Астрономічної обсерваторії Львівського національного університету імені Івана Франка належить до Міжнародної служби лазерної локації під кодом "LVIL-1831".

Для первинного опрацювання результатів спостережень використовують стандартний алгоритм Ермонсо (Herstmonceaux), який зазначає порядок отримання зі спостережень і теоретичних передбачень для орбіт супутників (ефемерид) результатів у вигляді "нормальних точок", які пересилають у міжнародні центри опрацювання даних. З огляду на зміни у форматах вхідних та вихідних даних, зміни і модернізації приймальної апаратури виникла потреба розробки власного програмного комплексу, що складається з системи керування телескопом у реальному часі, системи реєстрації та первинного опрацювання.

Для первинного опрацювання даних автори розробили програму під кодовою назвою NoPo. У ній використано новітні методи інтерполяції та апроксимації, а також методи аналізу часових рядів, статистичної фільтрації. Програма є вільним програмним забезпеченням, реалізована мовою С для платформи на базі ОС Linux.

Література

С. Апуневич, А.Білінський, С. Апуневич, Я.Благодир. Опрацювання результатів спостережень штучних супутників Землі на лазерно-локаційній станції "Львів-1831" // Теоретична електротехніка. Збірник наукових праць. Випуск 60. 2009. ст.104-109.

Розроблення електронного деканату для системи управління дистанційним навчанням MOODLE (на прикладі Львівської комерційної академії)

Артеменко В.Б.

Львівська комерційна академія, artem@iac.lviv.ua

Розглядаються на підґрунті Open Source підходи до розроблення електронного деканату, що є модулем типу «блок» для такої системи управління дистанційним навчанням (Learning Management Systems - LMS) як Moodle. Запропонована методологія адаптації налаштування параметрів е-деканату на засадах агент-орієнтованого моделювання. Агент-орієнтована модель (АОМ) розробляється для виявлення тенденцій виробництва та розповсюдження знань у віртуальному середовищі одного з вищих навчальних закладів за допомогою пакету вільного програмного забезпечення SWARM.

Інформаційно-комунікаційні технології (ІКТ) впливають практично на всі аспекти нашого життя. Зокрема доступність Інтернету широким верствам населення сприяє поширенню дистанційного навчання (ДН) у сфері освіти і корпоративного навчання.

У доповіді розглядаються досвід роботи та пілотні проекти, спрямовані на розроблення системи дистанційного навчання (СДН) Львівської комерційної академії (ЛКА). В академії використовується проектна модель організації ДН [1]. Одним із її результатів є створення Веб-центру ЛКА на основі платформи підтримки дистанційного навчання Moodle [2].

Розвиток Веб-центру ЛКА передбачає використання Open Source-проєкту Free Dean's Office (Електронний Деканат) російської компанії ТЗОВ "Відкриті Технології" [3]. Проєкт «Електронний Деканат» (е-деканат) спрямований на розробку інформаційної системи з відкритим вихідним кодом для підтримки управління навчальним процесом в освітніх організаціях будь-якого рівня.

Е-деканат є модулем типу «блок» для СДН Moodle, тому встановлюється поверх встановленої системи Moodle. Цей модуль дозволяє оперувати такими об'єктами: «Дисципліна», «Академічна група», «Семестр» («Навчальний рік»), «Навчальний план на семестр», «Навантаження викладача», «Поточна успішність і відвідуваність», «Підсумкові оцінки з дисциплін», «Розклад занять», «Договори на навчання» і т.д. Більше того, "Електронний Деканат" управляє навчальним процесом у СДН Moodle, позбавляючи методистів від рутинної роботи по підписці-відписці слухачів і збору підсумкових оцінок у відомості.

Перед використанням е-деканату його необхідно налаштувати на роботу з Веб-центром академії, ввівши таку інформацію: структура ЛКА, навчальні програми, навчальні дисципліни (включаючи вибіркові), дані про контингент слухачів, організацію навчального процесу з урахуванням роботи (взаємодії) викладачів (як авторів, так і тьюторів дистанційних курсів) і студентів та ін.

Налаштування параметрів е-деканату може бути адаптоване до діючого

у Веб-центрі навчального процесу на засадах агент-орієнтованого підходу [4].

Агент-орієнтовані моделі (АОМ) є новим засобом для добування знань. Серед найбільш популярних прикладних пакетів для побудови на принципах Open Source (вільного програмного забезпечення) АОМ є SWARM: колекція програмних бібліотек, які написані групою дослідників Інституту Санта Фе (Santa Fe Institute) [5].

Мета даної доповіді – розглянути нові підходи до створення АОМ, яка спрямована на виявлення тенденцій виробництва та розповсюдження знань на підґрунті формалізації комунікативних взаємодій між 3-ма типами агентів е-навчання (e-learning): авторами дистанційних курсів (ДК), тьюторами та студентами. Пропонуються підходи до розробки АОМ засобами SWARM.

Специфікація задачі побудови (формалізації) АОМ передбачає створення віртуального світу, в якому описуються комунікативні взаємодії таких типів агентів: А1 — автори ДК; А2 — тьютори, що супроводжують навчальний процес у віртуальному навчальному середовищі; А3 — студенти, учасники ДК.

Основні припущення моделі полягають у наступному:

- з точки зору прийняття рішень, агенти в цій моделі рухаються у двовимірному просторі і в них кінцевий горизонт бачення;
- будь-які агенти з'являються у Веб-центрі випадковим чином, з різними рівнями ймовірності, мають кінцевий термін життя;
- мета автора певного дистанційного курсу – виробити якомога більше знань і передати його тьютору, метою тьютора є поширити знання серед якомога більшого числа студентів, а мета студентів – спожити якомога більше знань;
- в початковий момент часу є певна кількість агентів усіх видів, які рухаються у випадковому напрямку.

Описуються поведінка та формальна модель комунікативних взаємодій агентів е-навчання.

Відповідно до технологій SWARM під час побудови моделі, що описує взаємодію аналізованих агентів, конструктор ModelSwarm виконує наступні ключові завдання:

- створює об'єкти, які використовуються в моделі;
- створює групу повідомлень, які надсилаються пакетованим списком дій для необхідного виконання кожним агентом;
- забезпечує перехід у віртуальний світ, в якому взаємодіють між собою безліч агентів аналізованих трьох типів.

SWARM забезпечує велику кількість інструментів для створення користувацького графічного інтерфейсу, що дозволяє візуалізувати агентів в їх віртуальному світі і проводити з моделлю комп'ютерні експерименти. Обидві можливості надаються інтерфейсом Swarm GUI, який має свою власну вбудовану панель управління GUI з п'ятьма кнопками: START, STOP, NEXT, SAVE і QUIT. Ці кнопки контролюють процес комп'ютерної імітації за допомогою клікання користувачем на них

мишкою (можуть натискатись і програмно).

Необхідно зазначити, що результати проведених експериментів указують на формування чітко виражених груп, майже в кожній з яких представлені всі типи агентів. Більшість студентів «тягнуться» до авторів і тьюторів ДК. Саме при такій структурі кількість виробленого та переданого знання прагне до теоретично максимального значення.

Отож, розглянуті в даній доповіді підходи щодо впровадження проекту «Електронний Деканат» у системі дистанційного навчання Moodle на засадах Open Source та агент-орієнтованого моделювання комунікативних взаємодій учасників е-навчання за допомогою технологій пакету SWARM дозволяють зробити наступні висновки та узагальнення.

1) Агент-орієнтоване моделювання комунікативних взаємодій ключових учасників електронного (або дистанційного) навчання можна розглядати як інструмент, що забезпечує проведення досліджень у сфері дистанційних освітніх технологій з метою підтримки адаптації налаштування параметрів е-деканату.

2) Експерименти з прототипом створюваної АОМ дозволяють зробити висновки щодо: раціональності підходів до формування такої структури навчальних груп студентів у віртуальному освітньому середовищі, при якій кількість виробленого та переданого знання прагне до теоретично максимального значення; потреби у перепроектуванні дистанційних курсів та поліпшення їх якості на підставі розроблення та впровадження адаптивних механізмів взаємодії ключових агентів е-навчання.

3) Надалі наші цілі передбачають розвиток АОМ, яка могла б врахувати всі представлені у формалізованому описі комп'ютерної моделі правила поведінки аналізованих агентів.

Література

1. Артеменко В.Б. Дистанційні технології та курси: створення і використання в освітній діяльності: Монографія / В.Б. Артеменко, Л.В. Ноздріна, О.Б. Зачко [за заг. ред. В.Б. Артеменка]. – Львів: Видавництво Львівської комерційної академії, 2008. – 297 с.
2. Сайт Веб-центру Львівської комерційної академії. URL: <http://virt.lac.lviv.ua>
3. Сайт Компанії ООО "Открытые технологии". URL: <http://www.open-technology.ru>.
4. Артеменко В.Б. Моделирование взаимодействия участников электронного обучения с использованием агент-ориентированного подхода // УСиМ. – 2010. – №2. – С.51-56.
5. Site Santa Fe Institute. URL: <http://www.santafe.edu>.

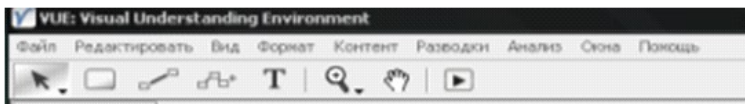
Використання середовища VUE для формування інформатичних компетентностей учнів

Барна О.В.

Чортківський інститут підприємництва і бізнесу ТНЕУ, ol_vas_shevchuk@i.u

Наведено перелік інформатичних компетентностей, які можна розвивати засобами середовища Visual Understanding Environment. Описано засоби середовища для побудови карт знань. Розглянуто приклад побудови інформаційної моделі розв'язування компетентнісного завдання.

Ефективне використання широкого кола засобів, які реалізуються на основі сучасних інформаційно-комунікаційних технологій пов'язується сучасними дослідниками із формуванням інформатичної компетентності всіх учасників освітнього процесу. За визначенням М. Голованя [1] інформатична компетентність – це інтегративне утворення особистості, яке інтегрує знання (про основні методи інформатичні та інформаційні технологій), уміння (використовувати комп'ютера і технологій зв'язку), здатності (представляти повідомлення і дані у зрозумілій для всіх формі) і виявляється у прагненні, здатності і готовності до ефективного застосування сучасних засобів інформаційних та комп'ютерних технологій для вирішення завдань у професійній діяльності і повсякденному житті, усвідомлюючи при цьому значущість предмета і результату діяльності. Формування інформатичної компетентності передбачає розвиток універсальних навичок критичного мислення, зокрема вміння спостерігати та робити логічні висновки, використовувати інформаційні моделі, аналізувати ситуацію, розуміти загальний зміст повідомлення та його прихований смисл. В якості інструменту, який сприяє становленню, розвитку та закріплення зазначених навичок, можна запропонувати середовище «візуального розуміння» Visual Understanding Environment (VUE), розроблене Tufts University Academic Technology. Середовище VUE розроблене з метою підтримки навчання, планування, презентування та дослідження. Його завантаження здійснюється зареєстрованим користувачем із сайту <http://vue.tufts.edu> [2]. Застосування простих інструментальних засобів та основної візуальної граматики, яка складається із вузлів та зв'язків дозволяє відображати зв'язки між поняттями, ідеями та цифровим вмістом шляхом побудови карти знань. Створені карти можуть бути завантажені на сервер VUE-ресурсів для їх подальшого використання (в тому числі і спільного), зберігатись на локальних ресурсах та імпортуватись у зображення різних форматів.



Вікно середовища достатньо просте: в центрі вікна – робоче поле,

назване Empty Map (порожня карта), у верхній частині – рядок меню та панель інструментів, що містить наступні інструменти - зліва-направо вибір об'єкта, інструмент вузол, зв'язок, швидке макетування, інструмент тексту, масштабування, переміщення, відтворення:

Для відкриття додаткової панелі налаштування параметрів об'єктів викликають Палітру форматування (мал.3) із меню Вікна, яка містить розділи стиль та форми фігур, колір та заливка фігур, тип та товщина лінії, початок і кінець лінії, блок форматування тексту (тип шрифту, розмір, колір, накреслення, виділення, вирівнювання, списки).

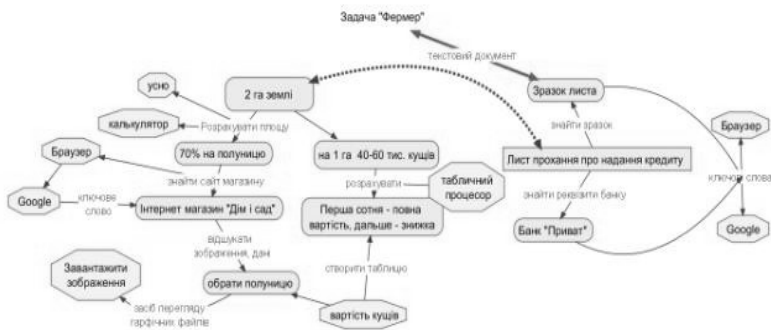


Різні форми вузлів (прямокутник, овал, ромб, шестикутник, трикутник тощо) та стилі їх оформлення (колір фону, рамки, формат тексту) забезпечують не тільки подання об'єктів різної природи, а, що головне, дозволяють автору карти знань сформувати у її користувачів візуальне сприйняття акцентів складових того чи іншого поняття. Важливим є також можливість будувати іменовані зв'язки між вузлами карти, застосовуючи при цьому прямі та криві лінії різної товщини, типу та напрямку. Це забезпечує побудову магістралей, що сприяє фокусуванню на вмісті з одночасним збереженням інформаційного контексту (вигляду карти).

Засоби середовища VUE достатньо ефективні при розв'язуванні компетентнісних завдань з інформатики, тобто таких завдань, які не передбачають наявності чітко визначеної моделі (у вигляді конкретних формул чи законів, які слід застосувати), прописаних вхідних даних та результатів. Компетентнісні задачі з інформатики можна розглядати як комплексні задачі прикладного характеру, для яких обов'язковим є застосування сучасних ІКТ як засобу розв'язування, надання різнорівневої допомоги та критеріїв оцінювання як кінцевого результату, так і способів його отримання [3]. В задачах такого типу прямим продуктом є свідоме засвоєння знань та умінь формувати стратегію розв'язування компетентнісних задач, планувати процес розв'язування, контролювати його правильність та оптимальність, виявляти та виправляти помилки.

Розглянемо для прикладу побудову інформаційної моделі розв'язування компетентнісного завдання наступного змісту: Фермер, вивчивши попит на ягідні культури в своєму регіоні, планує 70 % своєї земельної ділянки площею 2 га відвести на вирощування полуниці. Для цього він збирається звернутися до керівництва банку «Приват» з листом про надання йому відповідного кредиту, надавши обґрунтування свого прохання. Попередньо йому потрібно розрахувати кількість кущів певного сорту полуниці та потрібну суму на закупку розсади за умови, що за першу сотню кущів

покупець платить повну вартість, за кожну наступну сотню вартість зменшується на 1% від попередньої вартості. Розсаду фермер планує закупити в Інтернет-магазині «Дім і сад» з розрахунку 40-60 тис. кущів на 1 га. Побудована модель (мал. 4) складається із вузлів трьох типів: прямокутник із заокругленими кутами – для визначення вхідних даних завдання, прямокутник – для подання результату виконання завдання, восьмикутник – для запису засобів розв’язування завдання.



Мал. 4

На відміну від засобу створення карт знань, який також розповсюджується за ліцензією відкритого коду FreeMind, у середовищі VUE можна додати до карти декілька «батьківських» вузлів. У нашому випадку їх є два: вузол «Дано» та вузол «Отримати». Перехід від окремих складових вузлів містить мітки: опис дій чи пояснень, які забезпечують перехід за вказаним напрямком.

Демонстрування побудованої карти за допомогою режиму відтворення доповнює інструментарій середовища: у режимі перегляду вибір одного із вузлів карти веде за собою зміну вікна перегляду – перехід на вищий рівень деталізації, що задає пов’язані вузли, можливі переходи, демонструє засоби а також надає інструменти для повернення до загальної карти.

Таким чином, застосування середовища VUE відкриває нові засоби для організації навчальної діяльності учнів, що ґрунтується на компетентнісному підході.

Література

1. Головань М. Інформатична компетентність: сутність, структура і становлення. // Інформатика та інформаційні технології в навчальних закладах. – 2007.- № 4. – с.62 – 69.
2. Сайт VUE. – [Електронний ресурс] // Режим доступу: <http://vue.tufts.edu>.
3. Морзе Н.В., Кузьмінська О.Г. Компетентнісні задачі з інформатики. - Науковий часопис НПУ імені М.П. Драгоманова. Серія №2. Комп’ютерно-орієнтовані системи навчання: Зб. наукових праць. / Редада. – К.: НПУ імені М.П. Драгоманова, №6 (13), 2008.

Програмне забезпечення керування телескопом ЛЛС-станції “Львів-1831”

Білінський А.І., Мартинюк-Лотоцький К.П.

Астрономічна обсерваторія Львівського національного університету імені Івана Франка, slr1831@ukr.net, langure@mail.ru

У доповіді представлено результати розробки консольного програмного забезпечення на мові програмування С у операційній системі Linux для керування телескопом ТПЛ-1М. Відповідне програмне забезпечення проходить етап випробовування та кінцевої доводки при лазерних та координатних спостереженнях низькоорбітальних супутників і зір.

У АО ЛНУ ім. Івана Франка, у рамках переходу на вільне програмне забезпечення (ПЗ), розробляють нове програмне забезпечення для керування телескопом ТПЛ-1М [1, 2] у Unix-подібних системах. Метою такої роботи є створення ПЗ, яке б дало можливість оперативного вдосконалення програм для: використання іншої системи реєстрації (напр. SR620), модернізації програмно керованих вузлів телескопа, вдосконалення алгоритмів керування телескопом (карта похибок, спостереження супутника, зір, та інше), забезпечення простого безпечного обміну інформації між комп'ютерами/програмами (TCP/IP, smb).

Специфіка апаратного інтерфейсу наявної плати управління телескопів серії ТПЛ-1М [3] вимагає операційної системи (ОС) реального часу. За таку систему вибрано Open RTLinux [4] з кількох причин, що демонструють її перевагу серед інших аналогів, зокрема MSDOS: Open RTLinux – система жорсткого реального часу, відкритість коду, велика кількість програмного забезпечення, яке поширюється по GPL, мікроядро ОС запускає ядро Linux, що дає у руки потужний інструмент (система Linux). Керування телескопом ТПЛ-1М здійснюється за допомогою комп'ютера: Pentium200МГц, 96МБ ОЗУ, використовується IRQ 15; мікроядерна ОС Open RTLinux 3.2 на ядрі 2.4.29, основна операційна система Linux (Debian 3.1).

Структурно нове ПЗ керування телескопом [1, 5] складається з двох частин: модуля реального часу `tpl_driv.o` та інтерфейсу користувача `main_tpl.c`. Безпосередній зв'язок з телескопом здійснюється через модуль реального часу, плату управління та відповідні порти вводу/виводу [3]. За його допомогою реалізуються операції реального часу (реакції на 10 Гц апаратні переривання): синхронізація часу від синхронметра Ч7-37 та GPS приймача, керування роботою крокових двигунів повороту телескопа по осях висоти та азимуту, часова затримка дзеркал-комутаторів, запит на запуск лазера, формування часових воріт для реєструвальної системи між сигналами старту і стопу.

Програма-інтерфейс написана для консолі з використанням стандартних бібліотек: Ncurses – ввід/вивід з клавіатури на екран, організація окремих вікон (стану, позиції телескопа та ін.), організація меню; GPM – забезпе-

ння роботи з мишею; Socket, SMB – підтримка файлової системи і мережі.

Вдосконалення меню вибору потрібного проходження супутника, відображення відповідних даних на екрані, розділення вікон перегляду даних про вибране проходження (рис. 1) та гідуювання супутника на окремі, забезпечило оператору кращий контроль над проведенням спостережень.

```

Wed Sep 16 17:14:24 2009      A= 94.0000 h=20.0000  InitCommand  Klaser OFF
|                               0                X                |
3
Illumination: Umbra
Satellite COSPAR ID is 9502101 : 2009-09-16
Start 19:00:03.0; A= 94.389d H=20.061d R= 11.673ms
      Culmination: A= 59.925d H=25.902d R= 10.108ms
Stop 19:04:11.0; A= 25.432d H=20.006d R= 11.725ms
Satellite position A= 94.3883 h=20.0614 i satpos= 0

[ S ] Set start position: A= 94      H=20
                        TPL limbs A= 94
[ +/- ] Change START time
[ G ] Satellite TRACKING
[ F10,Q ] Exit to main MENU

DT= 6339sec

```

Рис. 1. Вікно перегляду даних про проходження супутника.

Проведено випробувальні ЛЛС спостереження вибраних супутників, гідуювання зір, а також координатні спостереження низькоорбітальних супутників [6] у автоматичному режимі без втручання оператора. Результати цих робіт показали доцільність використання для керування ПЗ у ОС Linux та RTLinux. На даний час продовжується розробка програмного забезпечення для покращення точності супроводу супутників та зір телескопом ТПЛ-1М, а також розпочато розробку нового ПЗ для керування чотириохвісним монтуванням ЛД-2 [2].

Література

- [1] А.Билинский, Б.Мелех Управление работой лазерного дальномера в системе RTLinux//Проблемы управления и информатики №2(2005), с.103
- [2] Астрономічна обсерваторія ЛНУ ім. І.Франка <http://astro.franko.lviv.ua>
- [3] Pavenis A. SLR telescope support software. – Riga : Astronomical Observatory University of Latvia, 1997. – 79 p.
- [4] Wind River : RTLinuxFree <http://www.rtlinuxfree.com>
- [5] Білінський А.І., Мартинюк-Лотоцький К., Благодир Я.Т. Автоматизація ЛЛС-спостережень на станції «Львів» // Бюлетень Українського центру визначення параметрів обертання Землі.- К.:ВАІТЕ, 2008.- №3, с.7-10.
- [6] Козырев Е.С., и др. Совместный проект НИИ "Николаевская астрономическая обсерватория" и Львовской астрономической обсерватории по позиционным наблюдениям ИСЗ // Материалы международной конференции «Околоземная Астрономия-2007», Нальчик.-2008, с.326-331.

ВІКІ-портал Кіровоградського державного педагогічного університету

Болілій В.О., Копотій В.В.

Кіровоградський державний педагогічний університет імені Володимира Винниченка (basilb@mail.ru, vika_kopoty@meta.ua)

В даній роботі представлено досвід використання серверу MediaWiki та вікі-технологій для розв'язування різних освітніх задач; описано організацію Вікі-порталу педагогічного університету.

Викладачі Кіровоградського державного педагогічного університету в травні 2008 року започаткували проект Вікі-портал КДПУ [1]. Для реалізації проекту була обрана платформа Вікі-Вікі на базі вільного серверного програмного забезпечення MediaWiki.

Середовище MediaWiki має переваги над іншими веб-ресурсами: можливість багатократно правити текст; облік змін, що були внесені до змісту сторінки, та можливість повернутися до попередньої версії; сторінка обговорень до кожної статті, де відвідувач може залишити свої коментарі.

Використання Вікі-порталу КДПУ викладачами у навчальному процесі призвело до утворення віртуального освітнього середовища. Оскільки Вікі-портал є доступний через інтернет, то розмістити свої роботи на сторінках середовища студенти можуть з будь-якого комп'ютера під'єданного до мережі. Викладач, маючи доступ до інтернету, може у будь-який час проводити моніторинг стану робіт студентів. Сторінка обговорень стає місцем, де можна вести дискусію. На сторінках Вікі-порталу викладачам та студентам надається можливість сумісної роботи над різними проектами.

Кожний зареєстрований користувач MediaWiki має власну сторінку користувача, яка може виконувати функції електронного портфолію, де студент збирає посилання на свої роботи. Викладач та інші студенти легко можуть переглянути матеріали, оцінити їх, прокоментувати.

Головна сторінка Вікі-порталу КДПУ містить шість основних розділів: Конкурси, Аудиторіум, Проекти, Педагогічна практика, Методична скарбничка, Публікації. Розділ «Конкурси» призначений для проведення он-лайн конкурсів. Розділ «Аудиторіум» призначений для підтримки вивчення різних навчальних дисциплін. У цьому розділі викладачі розміщують завдання для студентів, навчальні матеріали, проводять он-лайн консультації. Розділ «Проекти» містить поки що три проекти, які у даний час реалізуються зі студентами. Розділ «Педагогічна практика» призначений для підтримки студентів під час проходження педагогічної практики. Розділ «Методична скарбничка» містить Вікі-посібники, Вікі-книги та збірку конспектів уроків зі шкільного курсу інформатики. Розділ «Публікації» призначено для розміщення посилань на публікації викладачів та студентів.

З усіх розділів найбільш наповненим є «Аудиторіум». У ньому

міститься 14 різних навчальних дисциплін. Звичайно, кожен викладач використовує Вікі-середовище на різних етапах навчання і з різною метою [2].

В цілому залучення веб-сервісу MediaWiki має переваги над традиційним навчання: у майбутніх вчителів формуються навички роботи у середовищі MediaWiki та уявлення про можливі шляхи його використання у власній педагогічній діяльності; враховуються індивідуальні особливості студентів в усіх формах і методах педагогічного процесу; роботи студентів зберігаються вигляді гіпертексту і утворюють вільні інформаційні ресурси українською мовою, користуватися якими можна з будь-якого комп'ютера підключеного до інтернету [3]. Це сприяє широкому використанню студентами своїх наробок під час підготовки до занять, проходження педагогічної практики та власної педагогічної діяльності; підвищується мотивація студентів та покращується якість виконаних робіт; процес оцінювання стає більш прозорим та об'єктивним; студентами опановуються нові організаційні форми навчальної діяльності [4,5].

Література

1. Вікі-портал КДПУ // url: <http://wiki.kspu.kr.ua>.
2. Болілий В.О., Копотій В.В., Котяк В.В. Використання MEDIAWIKI при організації самостійної роботи студентів // Самостійна робота студентів та її інформаційно-методичне забезпечення: проблеми, досвід, методика: Методичний вісник. – Випуск 2. – Кіровоград: ПБВ КДПУ ім. В. Винниченка, 2009. – С. 56-64.
3. Присяжнюк О.В., Копотій В.В. Перспективи та проблеми впровадження інтернет-сервісу MEDIAWIKI в навчальний процес ВУЗу // Матеріали Міжнародної науково-практичної конференції «Сучасні інформаційні технології в управлінні та професійній підготовці операторів складних систем». – Кіровоград: Видавництво ДЛАУ, 2009. – С. 10-13.
4. Алексєєва С.І., Копотій В.В. Використання можливостей Вікі-середовища у системі підготовки майбутніх вчителів // Розвиток інтелектуальних умінь і творчих здібностей учнів та студентів у процесі навчання математики: Матеріали Всеукраїнської науково-методичної конференції (3-4 грудня 2009 р., м. Суми). – Суми: Вид-во СумДПУ імені А.С.Макаренка, 2009. – С. 191-192.
5. Присяжнюк О.В., Копотій В.В. Проблеми оцінювання студентських робіт при впровадженні в освітній процес інтернет-сервісу MEDIAWIKI // Науковий часопис НПУ імені М.П. Драгоманова. Серія №5. Педагогічні науки: реалії та перспективи. – Випуск 20: збірник наукових праць / за ред. В.П. Сергієнка. – К.: Вид-во НПУ імені М.П. Драгоманова. 2010. – С. 118–123.

Використання вільно-поширюваного ПЗ математичного призначення в університеті

Бугасць Н.О.

НПУ ім. М.П. Драгоманова, natashika777@gmail.com

Розглянуто особливості використання вільно поширюваного програмного забезпечення математичного призначення в університеті, а також розв'язування математичних задач засобами вільно поширюваних систем комп'ютерної математики.

Завдяки стрімкому розвитку комп'ютерної техніки і програмного забезпечення, комп'ютерні технології проникли в усі сфери людської діяльності. На сьогоднішній день неможливо уявити висококваліфікованого спеціаліста в будь-якій галузі, який не володіє інформаційними технологіями. В зв'язку з цим у викладачів університетів виникає проблема вибору програмного забезпечення для комп'ютерної підготовки студентів.

Особливе місце серед програмного забезпечення на фізико-математичних факультетах університетів займають програми математичного призначення, які дедалі ширше впроваджуються у процес навчання математичних дисциплін.

В Україні вітчизняними науковцями створені педагогічні програмні засоби, такі як Gran (Жалдак М.І.), DG (Раков С.А.), Extremum (Триус Ю.В.) та інші, які призначені для підтримки навчання математики. Проте, використання даного програмного забезпечення не може бути застосоване до багатьох розділів математики (лінійної алгебри, теорії чисел, аналітичного інтегрування та ін.).

Серед універсального математичного програмного забезпечення для аналітичних (символьних) перетворень найбільш відомі і широко розповсюджені системи комп'ютерної математики (СКМ) Derive, Maple, Mathematica, Mathcad, Matlab та ін. Ці програми є потужним інструментом для вченого, викладача, аспіранта чи студента. Їх використання дозволяє, оперуючи аналітичним записом даних, автоматизувати складні обчислення, дати наочну геометричну інтерпретацію абстрактних понять на основі використання інформаційних моделей в навчанні.

Однак всі ці комерційні програми досить дорогі. Тому більшість вищих навчальних закладів України не спроможні придбати достатню кількість ліцензій математичних пакетів для організації навчання математики.

Справжньою альтернативою закритим, пропріетарним (від "property" – право власності, володіння) програмам є вільно поширюване програмне забезпечення, яке схоже за функціональністю, але є некомерційним і відкритим.

Як зазначається в [2], використання вільно поширюваних програм є більш природним для фундаментальної науки, ніж комерційних, оскільки модель, яка використовується у вільно поширюваному програмному забезпеченні – це модель відкритості і загальнодоступності всіх напрацювань.

Очевидно, ці ж властивості притаманні і результатам наукової діяльності. Використовуючи таку схожість підходів, можна фактично розглядати розширення функціональності вільно поширюваних програм або додаткові бібліотеки, які можуть створюватися в процесі наукових досліджень, як невід'ємну частину результатів таких досліджень. І ці результати можуть використовуватися і розповсюджуватися на розсуд користувача без огляду на обмеження, накладені ліцензіями задіяного програмного забезпечення. Придбавши закрите програмне забезпечення, користувач отримує лише обмежене право користуватися ним, починаючи від неможливості вільно (і законно) передавати саме таке програмне забезпечення разом з напрацьованими і аж до можливих патентних позовів від компанії-виробника програмного забезпечення у випадку розповсюдження саморобних додатків бібліотек до нього.

Таким чином, використання для навчальних цілей саме вільно поширюваного програмного забезпечення – це реальна можливість і для ВУЗу, і для студентів, і для викладачів мати в своєму розпорядженні легальні копії такого програмного забезпечення без істотних грошових затрат.

Серед вільно поширюваних математичних пакетів можна виділити такі багатofункціональні програми як Maxima (символьні і чисельні обчислення), Sage (веб-СКМ, символьні та чисельні розрахунки), Scilab (виконання наукових та інженерних обчислень), Scidavis (побудова графіків, аналіз даних) та багато інших.

Ряд особливостей, врахування яких дозволяє рекомендувати систему комп'ютерної математики Maxima у вітчизняній освіті, виділяє С.О. Семєріков [1]:

1. система повністю відкрита, ліцензійно чиста і безоплатна;
2. система незалежна від використовуваної операційної системи й апаратної платформи;
3. сорокарічний досвід вдосконалення системи призвів до появи в ній повністю налагоджених, швидких та оптимізованих алгоритмів;
4. система невелика за розміром та невимоглива до апаратних ресурсів.
5. в системі надається широкий вибір інтерфейсів.

Розглянемо приклад розв'язування задачі з математичного аналізу за допомогою програми Maxima.

Задача. Знайти точки екстремуму і точки перегину функції

$$\int_0^x (t-1)(t-3)^4 dt$$

Обчислюємо інтеграл, знаходимо аналітичний вираз даної функції $y(x)$. Знаходимо першу похідну $y_1(x)$ та другу похідну $y_2(x)$.

(%i1) `y: integrate((t-1)*(t-3)^4,t,0,x);`

$$(\%o1) \quad \frac{5x^6 - 78x^5 + 495x^4 - 1620x^3 + 2835x^2 - 2430x}{30}$$

```
(%i2)  y:expand(y);
      define(y(x),y);
      y1:diff(y,x)$
```

$$(\%o3) \quad y(x) := \frac{x^6}{6} - \frac{13x^5}{5} + \frac{33x^4}{2} - 54x^3 + \frac{189x^2}{2} - 81x$$

$$(\%o5) \quad y1(x) := x^5 - 13x^4 + 66x^3 - 162x^2 + 189x - 81$$

$$(\%o7) \quad y2(x) := 5x^4 - 52x^3 + 198x^2 - 324x + 189$$

```
define(y1(x),y1);
y2:diff(y1,x)$
define(y2(x),y2);
```

Будуємо графік функції $y(x)$:

```
(%i4)  plot2d([y], [x,-0.5,5],[plot_format, gnuplot]);
```

Знаходимо стаціонарні точки:

```
(%i8)  solve(y1,x);
```

$$(\%o8) \quad [x = 1, x = 3]$$

$$(\%o9) \quad [x = \frac{7}{5}, x = 3]$$

```
solve(y2,x);
```

```
(%i10)  y1(0);y1(1.5);y1(4);
```

```
(%o10)  -81
```

```
(%o11)  2.53125
```

```
(%o12)  3
```

Отже, $x=1$ — точка мінімуму.

Обчислюємо значення функції в точці мінімуму:

```
(%i13)  y(1);
```

```
(%i14)  %,numer;
```

$$(\%o14) \quad -\frac{793}{30}$$

$$(\%o15) \quad -26.433333333333333$$

Встановлюємо, що $x=7/5$ та $x=3$ — точки перегину:

```
(%i13)  y2(1);y2(1.5);y2(4);
```

```
(%o13)  16
```

```
(%o14)  -1.6875
```

```
(%o15)  13
```

Таким чином, Maxima є альтернативою комерційним програмам і може успішно використовуватися як один з універсальних програмних засобів навчання інформатики та курсу вищої математики в університетах.

Для розв'язування задач обчислювальної математики, проведення аналітичних розрахунків, графічних досліджень слід звернути увагу на математичний пакет Sage, в якому об'єднано багато математичних пакетів в одному веб-середовищі. [3]

Незважаючи на все більшу популярність вільно поширюваного про-

грамного забезпечення і на певні кроки в напрямку його впровадження в освіті, успіх у цій справі залежить від готовності наукових і педагогічних працівників перебудувати навчальний процес, який в даний час орієнтований на використання пропрієтарних програмних продуктів і перейти на вільно поширювані програми. Тому необхідно забезпечити таку методичну підтримку використання некомерційних програм, яка підтвердить, що сучасне вільно поширюване програмне забезпечення математичного призначення може конкурувати з комерційним і може бути застосоване для розв'язування освітніх завдань при навчанні інформатичних та математичних дисциплін.

Література

1. Семеріков С.О. Фундаменталізація навчання інформатичних дисциплін у вищій школі: Монографія./Науковий редактор академік АПН України, д.пед.н., проф. М.І. Жалдак – К.: НПУ ім. М.П. Драгоманова, 2009 –340 с.
2. Тихон Тарнавский. Махіма – максимум свободы и символьных вычислений. [Електронний ресурс] – режим доступу: <http://lug-mgn.ru/ftp/Программы/Математика/Махіма/Статьи%20Т.Тарнавского/81-maxima.pdf>.
3. Шокалюк С.В. Основи роботи в SAGE/За редакцією академіка АПН України М.І. Жалдака. К.: НПУ ім. М.П. Драгоманова, 2008. – 64 с.

Вільне програмне забезпечення чи "CLOUD COMPUTING" у навчальному процесі?

Войтович І.С.

*Національний педагогічний університет імені М.П. Драгоманова,
igor_voitovich@ukr.net*

Описано основи функціонування і використання „хмар” та вільного програмного забезпечення. Розкрито переваги і недоліки “cloud computing” та вільного програмного забезпечення з позиції застосування їх у навчальному процесі. Внесено рекомендації щодо перспектив застосування навчального та прикладного програмного забезпечення у навчальному процесі.

Удосконалення апаратної частини персональних комп'ютерів (ПК) стимулює розвиток програмної частини, і навпаки. Це змушує навчальні заклади постійно оновлювати апаратну частину ПК та купувати нове програмне забезпечення (ПЗ). Однак часто ні на одне, ні на друге коштів немає. Це змушує шукати нові шляхи вирішення цієї проблеми в умовах стрімкого розвитку технологій та комунікацій.

Так, одним із перспективних нововведень у цьому напрямку вбачаємо “cloud computing”. „Револьюційні „cloud computing” позбавлять бізнес і держустанови від проблем з залізом і софтом. Майбутнє комп'ютерних мереж - за онлайнними документами і сервісами, віддалений доступ до яких надається як інтернет-послуга” [2]. Мова йде про технології, які дозволяють користувачам використовувати зовнішні, розташовані за межами їх персональних комп'ютерів, безмежні обчислювальні ресурси, щоб виконувати внутрішні завдання.

“Cloud Computing” — технологія обробки даних, в якій програмне забезпечення надається користувачеві як Інтернет-сервіс [5]. Користувач має доступ до власних даних, але не може управляти і не повинен думати про інфраструктуру, операційну систему (ОС) і власне ПЗ, з яким він працює. Згідно документу IEEE (Institute of Electrical and Electronics Engineers), опублікованому в 2008 році, „Cloud computing— це парадигма, в рамках якої інформація постійно зберігається на серверах в мережі інтернет і тимчасово кешується на клієнтській стороні, наприклад на ПК, ігрових приставках, ноутбуках, смартфонах тощо.

Для „cloud computing” основним припущенням є нерівномірність запиту ресурсів з боку клієнта(ів). Для згладжування цієї нерівномірності для надання сервісу між апаратним забезпеченням і middleware використовується віртуалізація серверів. Сервери, що виконують програми, віртуалізуються і балансування навантаження здійснюється як засобами ПЗ, так і засобами розподілу віртуальних серверів за реальними.

„Cloud computing” передбачає [6]:

- послуги на вимогу;
- широкий доступ до мережі;
- об'єднання ресурсів;

- гнучкий розподіл ресурсів;
- вимірювання послуг.

Оплата у “хмарах” відбувається за фактом надання послуг. Користувач оплачує лише таку кількість ресурсів і сервісу, яка йому необхідна. „Cloud computing” виявляються в 2-3 рази дешевшими, ніж розробка і застосування ліцензійного ПЗ. Наприклад, Google Apps забезпечує безкоштовний доступ до свого ПЗ в режимі онлайн, що відбувається за допомогою Інтернет-браузера тоді як ПЗ і дані зберігаються на серверах Google. Служба підтримує декілька веб-додатків зі схожою функціональністю як у традиційних офісних пакетів, і включає: Gmail, Google Calendar, Google Talk, Google Docs і Google Sites. Також, у „хмарах” вже зараз можна знайти аналоги більшості прикладних ПЗ і проводити практичні заняття зі студентами та учнями з комп’ютерних дисциплін, не задумуючись над тим, чи ліцензійне ПЗ встановлене у аудиторії, як встановити і запустити нові версії ПЗ, якщо апаратна частина ПК не відповідає програмним вимогам, тощо.

Разом із позитивними моментами хочемо спинитись і на негативних:

- не всі дані можна довірити сторонньому провайдеру в інтернеті, тим більше, не тільки для зберігання, але ще й для обробки;
- не кожен додаток дозволяє зберегти хоча б на флешку проміжні етапи обробки, а також фінальний результат роботи, адже онлайнві результати зручні не завжди;
- потрібний постійний і надійний доступ в Інтернет.

Концепція „cloud computing” неодноразово піддавалася критиці з боку спільноти вільного програмного забезпечення і, зокрема, з боку Річарда Столлмана: „Використовувати веб-додатки для своїх обчислювальних процесів не слід, наприклад, тому, що ви втрачаєте над ними контроль. І це не краще, ніж використовувати будь-яку програму. Робіть свої обчислення на своєму комп’ютері, використовуючи програми, які поважають вашу свободу. Якщо ви використовуєте будь-яку програму на чужому веб-сервері, ви стаєте беззахисними. Ви стаєте іграшкою в руках того, хто зробив це ПЗ” [1].

Термін „вільне ПЗ” ввів Річард Столмен, засновник проекту GNU, для опису ПЗ, котре можна безперешкодно використовувати, вивчати та змінювати і котре може копіюватись та поширюватись у змінній чи незмінній формі без будь-яких обмежень, з тим щоб інші користувачі також мали всі перелічені права. Щоб ПЗ вважалось вільним, воно повинно поширюватись під однією з ліцензій, котра закріплює за користувачем вищеописані права, та з вільнодоступними джерельними кодами [3].

Будь-хто може продавати копії вільного програмного забезпечення, використовувати його в комерційних чи інших цілях. Проте, оскільки кожен, хто володіє вільним програмним забезпеченням, може передавати його будь-кому безкоштовно, то в цілому вільне ПЗ є безкоштовним.

Так, зокрема Міністерство освіти і науки, молоді і спорту України має намір створити та впровадити систему паралельного використання закри-

того та відкритого ПЗ. Зі слів М.Стріхи, це зумовлене тим, що відкрите ПЗ розвивається більш швидкими темпами [4].

Водночас, при розповсюдженні вільного ПЗ на українському ринку необхідно враховувати той факт, що більшість програмних продуктів, які використовуються у державних організаціях та навчальних закладах, мають іншомовний інтерфейс (російський та англійський). Також слід звернути увагу на те, що крім переваг (дешевизна, надійність та стабільність в роботі, захист інформації та прав доступу, відсутність вірусів), недоліком сьогодні є й велике відставання в розробці прикладних ПЗ, які б повністю задовольняли потребам будь-якого українського користувача - від великої корпорації до окремої людини, що стримує її розповсюдження.

Що ж вибрати сучасному вчителю, викладачу, учню, студенту? Є народна мудрість: „Перед тим, як руйнувати старий будинок, треба побудувати новий”. Тобто, якщо хтось користується ліцензійним ПЗ– нехай продовжує ним користуватися, якщо ж немає можливості придбати дороге ПЗ – потрібно знайти альтернативу. Чи це буде вільне ПЗ, чи це буде „cloud computing” – вибір користувача, виходячи з особливостей та умов його роботи. Ми ж, освітяни, в свою чергу повинні показати цю альтернативу та забезпечити можливість навчитись користуватись і вільним ПЗ, і „cloud computing”, разом з тим створюючи необхідне методичне та дидактичне забезпечення.

Література

1. Bobbie Johnson Cloud computing is a trap, warns GNU founder Richard Stallman.- Дата перегляду: 10.01.2011.- Режим доступу: <http://www.guardian.co.uk/technology/2008/sep/29/cloud.computing.richard.stallman>
2. Баллмер Стив. Дорога в облака // Кореспондент.- №45.- 2010 Дата перегляду: 06.12.2010.- Режим доступу: <http://blogs.korrespondent.net/opinions/1142792-doroga-v-oblaka>
3. Вільне програмне забезпечення.- Дата перегляду: 05.01.2011.- Режим доступу: [http://uk.wikipedia.org/wiki/Вільне програмне забезпечення](http://uk.wikipedia.org/wiki/Вільне_програмне_забезпечення)
4. Міносвіти переходить на відкрите ПЗ Дата перегляду: 06.01.2011.- Режим доступу: <http://www.mon.gov.ua/main.php?query=monitoring>
5. Облачные вычисления.- Дата перегляду: 25.12.2010.- Режим доступу: [http://ru.wikipedia.org/wiki/ Облачные вычисления](http://ru.wikipedia.org/wiki/Облачные_вычисления)
6. Романченко Владимир Облачные вычисления на каждый день: Аналитика - 3DNews - Daily Digital Digest.- Дата перегляду: 27.12.2010.- Режим доступу: http://www.3dnews.ru/editorial/cloud_computing

Використання вільного програмного забезпечення в системі дистанційної освіти

Воронкін О.С.

Луганський державний інститут культури і мистецтва, alex.voronkin@gmail.com

У роботі розглядаються актуальні питання поширення вільного програмного забезпечення та перспективні безкоштовні Internet-сервіси для базової підтримки дистанційної форми навчання. Робиться висновок про необхідність прийняття цільової програми по впровадженню відкритого програмного забезпечення у сфері державного управління, в органах державної влади, місцевого самоврядування та навчальних закладах України.

Останнім часом у багатьох країнах світу намітилася тенденція використання вільного програмного забезпечення (ВПЗ) у всіх сферах людської діяльності. Цьому посприяло й те, що рівень і якість ВПЗ стали набагато вищими чим кілька років назад.

І. Операційні системи

На сьогодні дві третини операційних систем (ОС), які встановлені на комп'ютерах українських користувачів й у комп'ютерних класах навчальних закладів – це продукція компанії Microsoft, виробника закритих програмних продуктів а, саме ОС Windows XP – 7. Маючи велику кількість державних контрактів, Microsoft фактично є монополістом та впливає на рішення щодо вибору програмного забезпечення у державному секторі, це дозволяє здійснювати масові маркетингові протидії просуванню ВПЗ шляхом розповсюдження реклами на свою продукцію.

Тим не менш існує широкий спектр конкурентоспроможних програм, в яких права користувача необмежені на установку, запуск, а також вільне використання, вивчення й змінювання (удосконалювання) коду. Найбільше розповсюдження отримали ОС сімейства Linux. Завдяки можливості модернізації існує безліч дистрибутивів, а саме: RedHat Linux (США), SuSe Linux (Німеччина), ALT Linux та ASP Linux (Росія) та ін. На відміну від ОС сімейства Windows аналогічні версії Linux потребують значно менші вимоги до системних ресурсів комп'ютера. Наприклад дистрибутив Xubuntu 9.0 працює на комп'ютері із процесором 500 МГц і 128 Мб ОЗП. Такі ОС легко модифікуються для роботи з конкретними вимірювальними приладами. Заслуговує уваги ОС Scientific Linux, що була розроблена фахівцями з Європейського Центра ядерних досліджень CERN і Національної лабораторії ім. Е. Фермі FermiLab.

До переваг використання таких ОС слід віднести: 1) доступність в Internet мережі для завантаження, копіювання й використання, 2) всі матеріальні витрати зводяться лише до витрат на адміністрування, 3) для цих ОС існує велике число вільних прикладних програм, 4) ОС Linux мають високий захист від зараження вірусами в мережі Internet, 5) ОС мають високу швидкість виконання операцій, 6) спеціальні сервера – репозитарії, надають відповідне прикладне ВПЗ [1].

II. *Офісне програмне забезпечення*

До відкритих програм для офісної роботи варто віднести офісні пакети Open Office і Lotus Symphony, що аналогічні пакету Microsoft Office, а також графічні редактори (наприклад dia), словники (наприклад StarDict), програми для перегляду документів у форматах pdf і djvu.

Самим розповсюдженим офісним пакетом є Open Office, його версії існують для ОС Linux і ОС Windows. Open Office працює як з власним форматом документів, так і з документами Microsoft Office. Реалізовано експорт у формат наукових статей LaTeX і pdf-формат.

Для роботи в Internet мережі найбільш часто використовують Mozilla Firefox, Mozilla SeaMonkey та Thunderbird, Pidgin, Kompozer й ін.

Існують програми для вирішення практично всіх прикладних задач – математичних розрахунків (наприклад SciLab [2] і Maxima), програмування (Geany) та ін.

III. *Moodle у навчальному процесі*

Moodle – модульне об'єктне-орієнтоване динамічне навчальне середовище, система управління навчанням (learning management system, LMS). У країнах СНД подібні платформи називають також системами дистанційної освіти (СДО). Moodle є ВПЗ з ліцензією GPL, яка завдяки своїм функціональним можливостям успішно конкурує з комерційними LMS. Дозволяє створювати, а потім управляти ресурсами інформаційно-освітнього середовища, має зручний інтерфейс, html-редактор, підтримує таблиці, схеми, графіки, відео, форум, чат, глосарій, flash, wiki та ін. Різні версії Moodle існують для роботи в ОС Windows, Linux та FreeBSD.

З 2008 по 2010 рр. за допомогою LMS Moodle автором були розроблені: 1) електронний підручник «Мікроелектронні підсилювачі вимірювальних пристроїв»; 2) дистанційний курс «Фізичні принципи утворення, розповсюдження, реєстрації та аналізу звукових коливань»; 3) дистанційний курс «Мікроелектронні підсилювачі спеціального призначення». Електронні ресурси успішно апробовані, використовуються в навчальному процесі декількох ВНЗ України та розміщені в мережі Українського інституту інформаційних технологій в освіті НТУУ «КПІ», Східноукраїнського національного університету ім. В. Даля та системи Moodle Харківського національного університету радіоелектроніки.

З листопада 2010 р. розпочав роботу інформаційно-освітня портал «Технології дистанційної освіти», розроблений автором на сервісі uCoz (<http://www.tdo.at.ua>) [3] з метою: інформування щодо розвитку та впровадження дистанційної освіти в Європі, Україні та Луганській області; систематизації й аналізу сучасного стану інформаційної інфраструктури; обговорення нових наукових і практичних результатів використання інформаційних технологій та мереж в науці й освіті; апробації новітніх підходів до реалізації перспективних освітніх технологій; пошуку сумісних напрямків досліджень.

IV. *Безкоштовні Internet-сервіси в базових технологіях СДО*

В 2005 році з'явився соціальний сервіс Web 2.0, що надав можливість будь-якому користувачу в Internet створювати особисті ресурси та

надавати до них доступ без особових знань мов програмування. Появу терміна Web 2.0 прийнято пов'язувати із статтею Тіма О'Рейлі «What Is Web 2.0» [4].

До перспективних проєктів, які можуть використовуватися як базові в СДО слід віднести: 1) сервіси для публікації презентацій у вигляді слайд-шоу (наприклад Slideshare та Issuu) [5, 6]; 2) соціальні мережі (Facebook, Livejournal, ВКонтакте та ін.); 3) мережеві журнали для ведення блогів (blogger, twitter [7]); 4) web-платформи для проведення інтерактивних відеозанять і вебінарів в віртуальних класах (наприклад Dimdim, WiZiq [8] та ін.); 5) електронні пошти; 6) енциклопедії знань (Вікіпедія, Knol [9]); 7) віртуальні дошки; 8) програми IP-телефонії.

Існує ВПЗ і у сфері культури, так уваги заслуговує віртуальний музей по історичній, культурній і природній спадщині країн Центральної Азії – проєкт Museolog, який був розроблювався ЮНЕСКО з 1998 р.

Необхідно зазначити, що для повноцінного та широкого використання ВПЗ з відкритими ліцензіями в Україні необхідно: 1) прийняття цільової програми впровадження відкритого програмного забезпечення у сфері державного управління, в органах державної влади та органах місцевого самоврядування, навчальних закладах; 2) цілеспрямовані та зацікавлені дії уряду; 3) ініціативні дії програмістів; 4) формування попиту на використання цих програм.

Література

1. Никитенко С. Н. Организация учебного процесса в медицинском ВУЗе на основе операционной системы ASP Linux / С. Н. Никитенко, О. Г. Горшков, С. М. Тетюра, О. Н. Лахно // Актуальные вопросы теоретической и прикладной биофизики, физики и химии: матер. VI междунар. науч.-техн. конф. БФФХ- 2010, Севастополь, 26–30 апр. 2010 г. – Севастополь. – Том 1, 2010. – С. 400-401.
2. Офіційний сайт системи Scilab. – Режим доступу: <http://www.scilab.org>.
3. Інформаційно освітянський портал «Технології дистанційної освіти». – Режим доступу: <http://www.tdo.at.ua>.
4. O'Reilly T. What Is Web 2.0? Design Patterns and Business Models for the Next Generation of Software / Т. O'Reilly. – Режим доступу: <http://www.oreilly.de/artikel/web20.html>.
5. Сервіс Slideshare. Presentations, documents and professional videos. – Режим доступу: <http://www.slideshare.net>.
6. Платформа Issuu. – Режим доступу: <http://www.issuu.com>.
7. Журнал Twitter (blog). – Режим доступу: <https://twitter.com>.
8. Education online WiZiq. – Режим доступу: <http://www.wiziq.com>.
9. Internet енциклопедія Knol. – Режим доступу: <http://knol.google.com>.

Концепція вільного програмного забезпечення в освітньому менеджменті

Галушко О.І.⁽¹⁾, Галушко І.М.⁽²⁾, Веселова С.І.⁽¹⁾

(1) Придніпровська державна академія будівництва та архітектури,

(2) Національна металургійна академія України, olena_galushko@ukr.net

Розглянуто загальну сутність контролінгу, доцільність та переваги впровадження контролінгу у вищих навчальних закладах. Підкреслено важливість розробки програмного забезпечення для систем контролінгу. Особлива увага приділена значенню вільного програмного забезпечення, перевагам його використання в освітньому менеджменті взагалі та у розробці програм для систем контролінгу зокрема.

Україна стала на шлях розвитку інформаційного суспільства. Постановою Кабінету Міністрів України утворено Міжгалузеву раду з питань розвитку інформаційного суспільства. За мету поставлено збільшення частки інформаційних комунікацій і більш глибоку інтеграцію у світовий інформаційний простір.

В системі освіти набуває актуальності проблема інтеграції інформаційних і педагогічних технологій. Для вирішення зазначеної проблеми в роботі [1] пропонується використання контролінгу.

Узагальнено контролінг можна визначити як цілеорієнтовану систему планування і контролю, яка забезпечує інтеграцію, системну організацію та координацію фаз процесу управління [2]. Тобто контролінг надає можливість керівникам усіх рівнів відстежувати досягнення встановлених цілей та співвідношення запланованих та фактичних значень показників. Це дозволяє підвищити ефективність управлінського процесу як в оперативному режимі, так і в стратегічній перспективі.

Оскільки контролінг є сучасним та ефективним управлінським інструментом, його використання в освітньому менеджменті дозволить підвищити ефективність організації навчання та конкурентоспроможність вищих учбових закладів. Впровадження елементів контролінгу в учбовий процес дозволить, щонайменше, вирішити дві задачі [1]:

1. Збільшити ефективність навчального процесу шляхом оптимального керування інформаційними потоками в системі “викладач-студент-деканат”;
2. Сформувані у студентів системний підхід до управління та розвинути практичні навички використання автоматизованих систем управління.

Переважною ознакою контролінгу є цілковите сполучення, тобто інтеграція інформаційних потоків. Ефективна організація інформаційних потоків неможлива без належного програмного забезпечення. Пріоритетним в цьому напрямку є, на нашу думку, використання вільного програмного забезпечення. Адже в Україні існують програми з контролінгу (зокрема, модуль “Бюджетування і контролінг” ERP системи “ІТ-Підприємство”; програмні продукти з контролінгу корпорації “Інталев” та інші), але всі вони, по-перше, орієнтовані на промислові підприємства та коме-

рційні організації, а по-друге, проекти з їх впровадження є дуже дорогими, як і вартість ліцензій.

Вільне програмне забезпечення (ВПЗ) отримує все більш широке розповсюдження в державних закладах різних країн: за результатами опиту Маастрихтського університету 49% респондентів у 12 країнах використовують ВПЗ. Найбільш важливими критеріями вибору програмного забезпечення є: потужність, безпека та низькі ліцензійні відрахування [3]. Роль вільного програмного забезпечення в державних установах країн СНД, зокрема в системі вищої освіти, також зростає.

До головних переваг вільного програмного забезпечення (ВПЗ) в системі освіти можна віднести:

1. Більш дешево, ніж в комерційних продуктів, вартість впровадження та технологічної підтримки, що робить ВПЗ конкурентоспроможним в ринкових умовах;
2. Гнучку модифікація та можливість доробки програмних продуктів відповідно до вимог користувачів, що є важливим фактором з огляду на різноманітність дисциплін, що викладаються, та багатопрофільність вищих навчальних закладів;

Невибагливість до існуючих ресурсів, що є також важливим фактором в умовах обмеженого фінансування державних вищих навчальних закладів.

Більш висока швидкість реакції на вимоги зовнішнього середовища сприяє постійному оновленню продуктів ВПЗ та швидкій модернізації систем, що є характерною ознакою інформаційного рівня розвитку суспільства та підкреслює відповідність концепції вільного програмного забезпечення вимогам часу. Наведені особливості ВПЗ розширюють межі інформаційного простору, завдяки цьому вільне програмне забезпечення є перспективним для використання в освітньому менеджменті взагалі та при розробці програмних продуктів для системи контролінгу зокрема.

Література

1. Галушко Е.И., Галушко И.М., Веселова С.И. Использование контроллинга при подготовке специалистов в сфере информационных технологий // Збірник наукових праць VIII Міжнародної науково-практичної конференції “Новітні комп’ютерні технології”. - Київ-Севастополь, 2010. - с.100-101.
2. Майер Э. Контроллинг как система мышления и управления / Перевод с нем. Ю.Г. Жукова, С.Н. Зайцева; под ред. С.А. Николаевой. - М.: Финансы и статистика, 1993. - 92с.
3. Методические рекомендации по внедрению и использованию свободного программного обеспечения в образовательных учреждениях Российской Федерации. – М: Министерство образования и науки России, 2010. - 656 с.

Впровадження ВПЗ в проектних інститутах будівельної галузі України

А.В. Гірник, А.Ф.Неминуца

Державний науково-дослідний інститут автоматизованих систем в будівництві,
dndiasb@gmail.com

Дослідження вільного програмного забезпечення, призначеного для створення різних розділів проекту в проектних інститутах будівельної галузі України.

Вже всі українські підприємства усвідомили, що використання не-ліцензійного пропріетарного програмного забезпечення (ПЗ) несе в собі ризики як щодо безпеки інформації, так і щодо можливих штрафних санкцій з боку правоохоронних органів.

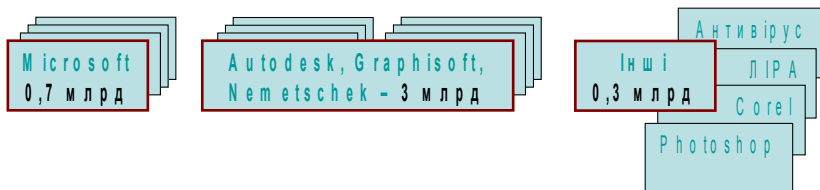
Асоціація виробників програмного забезпечення Business Software Alliance (BSA) — веде активну діяльність в Україні щодо заборони нелегального використання програмних продуктів своїх членів, серед яких такі компанії, як Adobe, Apple, Autodesk, Corel, Microsoft, Symantec тощо.

За даними останніх (опубліковано 11.05.2010) щорічних досліджень рівня комп'ютерного піратства компанії IDC (в партнерстві з BSA) кількість нелегального програмного забезпечення, встановленого на персональних комп'ютерах в Україні в 2009 році, виросло на 1% (до 85%).

Завдяки активній діяльності BSA правоохоронними органами відкрито вже кільканадцять кримінальних справ стосовно порушників авторських прав на комп'ютерні програми.

На сьогодні ліцензії на господарську діяльність у сфері будівництва мають близько 30 тис. організацій. З них близько 6 тис. на проектування.

За даними Асоціації "Українське об'єднання проектних організацій" вартість заходів з легалізації (закупівлі ліцензій на програмне забезпечення) в одній тільки будівельній галузі України сягає 4 млрд.грн. Причому це на 95% імпорт. Приблизна структура програмного забезпечення наведена на рисунку нижче. Як видно, найбільш проблема легалізації вдаряє по проектних організаціях, що експлуатують вартісні пакети САПР.



Одним із шляхів зниження рівня піратства в галузі є перехід на вільне програмне забезпечення.

Визначальними чинниками при впровадженні вільного програмного забезпечення, за нашим досвідом, є наступні:

– відповідність функціональності програмного забезпечення стану проект-

них технологій в будівельній галузі на даний час та в розвитку;

– забезпечення сумісності документів з іншими (пропріетарними) програмними продуктами, що використовуються, читання та коригування напрацьованих документів, в тому числі в застарілих форматах; відслідковування змін цих форматів;

– максимальна наближеність до стандартного інтерфейсу користувача пропріетарних пакетів, що використовується сьогодні на більшості робочих місць, щоб уникнути тривалого перенавчання у процесі впровадження.

Переважає більшість робочих місць в проектному інституті оснащена системою автоматизації проектування (САПР). На 95% місць - це AutoCAD.

Багатократні спроби перевести проектувальників на будь-яку іншу САПР неодмінно зазнавали невдачі, за рідкісними виключеннями. Можливо це пов'язане з тим, що середній вік співробітників у проектних інститутах сягає пенсійного.

Серед відомих DWG-сумісних САПР, на жаль, фактично всі призначені для Windows. Тому, швидше за все, вона в ближньому майбутньому утримає першість серед операційних систем в проектному інституті. Цього буде вимагати і сумісність різних САПР та мережеве використання AutoCAD.

Хоча, наприклад, перша вітчизняна САПР БудКАД чудово працює під Wine. Ми перевіряли її на Ubuntu 10. Під Linux запущена також ще одна сумісна САПР — BracsCAD (Бельгія). Можливо, що розвиток САПР в цьому напрямку продовжиться.

Вітчизняна САПР БудКАД створена за ініціативою Асоціації "Українське об'єднання проектних організацій" з метою вирішення проблеми легалізації програмних засобів в проектних організаціях. Розробку першої версії завершено в I половині 2010 року. Розробник - базова організація з інформаційних технологій Міністерства регіонального розвитку та будівництва України – Державний науково-дослідний інститут автоматизованих систем у будівництві (ДНДІАСБ).

БудКАД має інтерфейс українською, російською та англійською мовами, які можна перемикає через меню. Довідкова система поки що російською.

САПР БудКАД створена на базі платформи IntelliCAD від консорціуму ІТС (IntelliCAD Technology Consortium), який на корпоративних засадах розробляє та підтримує базову платформу. Програмні продукти, створені на цій платформі, широко відомі у світі і поставляються в 80-ти країнах, в тому числі в США, Європі, Японії.

Перша версія БудКАД призначена для 2D-креслень проектування, що за даними Асоціації проектних організацій достатньо для близько 80% робочих креслень.

Офісний пакет OpenOffice.org знайшов широке поле застосування в проектних організаціях галузі. З прийняттям в Росії стандарту OpenDocument як державного, процес його впровадження, як нам здається, піде ще більш інтенсивно.

Проте слід зауважити, що інколи виникають несумісності передачі даних між пакетами САПР та OpenOffice.org. Тому в якості резервних кілька пакетів MS Office тримають навіть ті організації, де OpenOffice.org впроваджено на всіх робочих місцях.

В цьому питанні, на жаль, негативну роль відіграють державні органи влади. Самі найчастіше не маючи ліцензованого програмного забезпечення, викладають свої документи в форматі MS Office. Можливо приклад Росії надихне і наших законодавців прийняти для обов'язкового користування державними органами відкритий формат документів.

Кілька слів про вільний простір для вільного програмного забезпечення в цій галузі:

- архітектурна частина проекту: не виявлено вільних програм; для подачі проекту замовнику можливе використання GIMP та Blender;
- будівельне конструювання: не виявлено вільних DWG-програм;
- будівельні розрахунки: напевне, неможливе використання вільних програм через підвищену відповідальність в цій сфері;
- проектування та розрахунки інженерних систем будівель: не виявлено вільних програм;
- автоматизований випуск кошторисів: не виявлено вільних програм.

Зрозуміло, що надто важко проаналізувати всі існуючі програми в цій сфері щодо їх застосування в проектному інституті. Але можна стверджувати, що жодна з існуючих вільних програм не набула якого-небудь помітного визнання серед наших проектувальників для розробки вказаних вище частин проекту.

Ми розглянули тільки програмні засоби безпосередньої виробничої діяльності проектною організацією. Суміжні програмні продукти загальні для значної частини застосувань і виходять за рамки цієї роботи.

Література

1. Гірник А.В. Вітчизняна САПР БудКАД як засіб легалізації програмного забезпечення. // Новітні комп'ютерні технології. Матеріали VIII Міжнародної науково-технічної конференції. - Київ - Севастополь: 2010. - с. 9 - 11.
2. Гірник А.В., Неминуца А.Ф. Боротьба з нелегальним використанням програмного забезпечення. Матеріали VIII Міжнародної науково-технічної конференції. - Київ - Севастополь: 2010. - с. 12 — 14.

Система підтримки прийняття рішень з вибору мобільної платформи на програмних засобах з відкритим кодом

Д.А. Гірник

*Міжнародний науково-навчальний центр інформаційних технологій та систем
НАН та МОН України, den.girnyk@gmail.com*

Дослідження системи підтримки прийняття рішень з вибору мобільної платформи для визначеного класу задач, що використовує відкрите програмне середовище LMS Moodle.

Сьогодні мобільні технології швидко розвиваються та проникають практично в усі галузі діяльності людини: економіку, науку, освіту.

Наряду з цим, розширюється модельний ряд та функціональність мобільних терміналів, спектр підтримуваного системного та прикладного програмного забезпечення, систем стільникового зв'язку, управління сканерами штрих-кодів, методів віддаленого доступу до територіально розподілених баз даних та інтернет-ресурсів тощо.

Застосування мобільних технологій стає все більш складним. Вибір мобільної технології для конкретних застосувань стає нетривіальним завданням. Людина, що приймає рішення з вибору для конкретного класу задач, повинна мати належні знання у сферах, не пов'язаних із самою задачею застосування мобільних технологій. Підтримка рішень щодо вибору мобільних технологій для конкретних застосувань на сьогодні відсутня. Тому актуальною проблемою є розроблення систем підтримки прийняття рішень з вибору мобільних технологій для визначеного класу задач.

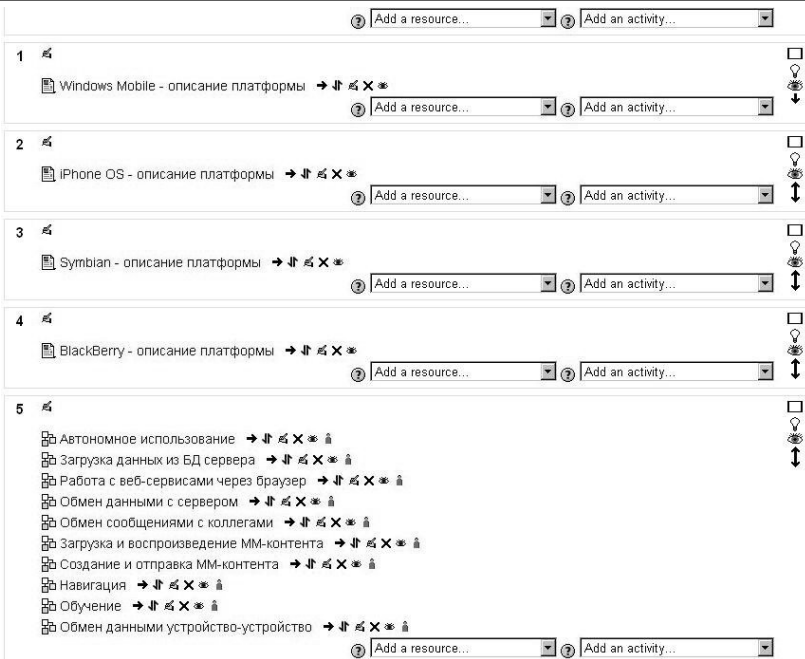
Для визначення вимог до мобільних платформ, які можна застосувати для підтримки різних видів діяльності, досліджені приклади конкретних застосувань (кейси) мобільних технологій в наступних сферах: складські системи, будівництво, технічна інвентаризація, мобільні торгівля та логістика, довідкові системи, системи дистанційної освіти, перекладачі, медичний моніторинг індустрія туризму та відпочинку та інші. Визначені основні класи задач для досліджених 28 застосувань, що вирішуються на мобільних платформах, та вимоги до платформи з урахуванням особливостей предметної області [1].

Розроблений алгоритм прийняття рішень при виборі мобільних технологій на основі визначених класів задач та вимог до апаратних та програмних характеристик і до функціональності системи розробки власного програмного забезпечення.

Основні компоненти системи підтримки прийняття рішень щодо вибору мобільної платформи реалізовані на програмному забезпеченні з відкритим кодом у вигляді веб-ресурсу. Використана LMS Moodle.

Система містить короткі курси навчання користувачів характеристикам та можливостям вибраних мобільних платформ, необхідні для подальшого аналізу альтернатив при виборі.

Копія сторінки адміністрування системи.



Література

1. Гірник Д.А. Дослідження класів задач, що вирішуються на мобільних платформах // Новітні комп'ютерні технології. Матеріали VIII Міжнародної науково-технічної конференції. - Київ - Севастополь - с. 162 - 165.
2. Pérez I.J., Cabrerizo F.J., Herrera-Viedma E. A Mobile Decision Support System for Dynamic Group Decision-Making Problems, in IEEE Transactions on Systems, Man and Cybernetics, Part A: Systems and Humans, Nov. 2010, vol 40, pp. 1244 – 1256.
3. Raghavan V., Rundensteiner E.A. Progressive result generation for multi-criteria decision support queries, IEEE 26th International Conference on Data Engineering (ICDE), March 2010, pp. 733 – 744.

Інформаційна технологія управління навчальним навантаженням у вищих навчальних закладах

Гриценко В.Г.

Черкаський національний університет імені Богдана Хмельницького,
(grycenko@ukr.net)

У роботі розглянуті проблеми створення та способи застосування інформаційно-комунікаційних технологій до управління навчальним навантаженням у вищих навчальних закладах.

Навчальний план є базовим документом, що визначає зміст професійної підготовки фахівця. В ньому реалізуються основні принципи відбору предметів, їх систематизація, регламентується об'єм навчальних дисциплін, навантаження студента за періодами навчання, форми підсумкового контролю знань. На структуру і зміст навчального плану впливають ОКХ і ОПП за відповідними напрямками і спеціальностями, інструктивні матеріали МОН України [1], вимоги підприємств і організацій - безпосередніх замовників фахівців, тощо.

В умовах переходу до трирівневої системи вищої освіти та постійного зростання кількості спеціальностей, які запроваджують вищі навчальні заклади, виникає проблема контролю якості розробки навчальних планів та їх подальшого використання у процесі формування робочого навчального плану, розподілу навчального навантаження між кафедрами і викладачами. Використання традиційних способів стало трудомістким і довготривалим, а отже, виникла необхідність в дослідженні можливостей вирішення зазначених проблем шляхом використання інформаційно-комунікаційних технологій.

Теоретичний аналіз наукових праць, ознайомлення з практичним досвідом роботи університетів щодо окресленої проблеми дали змогу виявити низку суперечностей між вимогами сьогодення до управління навчальним процесом та забезпеченістю ВНЗ відповідними сучасними засобами та методикою їх використання [2].

Визначені суперечності надають можливість сформулювати проблему організаційно-педагогічного забезпечення цілісності процесу управління формуванням та реалізацією навчального навантаження у ВНЗ в умовах використання інформаційно-комунікаційних технологій.

Останнім часом у багатьох ВНЗ України почали впроваджувати локалізовані засоби автоматизації окремих напрямів управлінської діяльності (системи обліку документів та кореспонденції, системи обліку кадрів, системи управління бібліотеками, системи бухгалтерського обліку, тощо). Але неможливість синхронізації дій між різними системами, відмінності у представлені інформації, і як наслідок, складність передачі даних між ними, породжують нові проблеми, пов'язані з надлишковою інформацією, неоперативністю отримання потрібних даних, фрагментарністю виконання єдиних процесів. Все це зводить нанівець намагання суб'єктів освітньої

діяльності перекласти її рутинні процеси на засоби інформаційно-комунікаційних технологій.

Найбільш вдалим рішенням у цій ситуації є використання сучасних мережових технологій і єдиного сховища даних, що надає можливість реалізувати простий механізм інтеграції інформації в єдиний інформаційний ресурс вищого навчального закладу і забезпечити можливість сумісного використання інформації з урахуванням механізму розмежування доступу (з метою захисту даних) багатьма користувачами: адміністрацією ВНЗ, викладачами, студентами різних форм навчання, абітурієнтами. Це надає можливість досягти високого рівня цілісності даних і створюються умови для реалізації комплексної автоматизованої системи управління навчальним процесом університету з наступними функціями:

- Розробка (модернізація) навчальних планів за напрямками та спеціальностями і робочих навчальних планів на наступний навчальний рік;
- Розрахунок штатів професорсько-викладацького складу університету;
- Розподіл навчального навантаження викладачів кафедр;
- Формування розкладу занять навчальних груп в університеті;
- Облік проміжної (рейтинг) і підсумкової успішності студентів;
- Формування додатку до диплома.

1 - 3 із зазначених функцій нами було досліджено, спроектовано та реалізовано в АІСУ “Навчальне навантаження”, що складається з підсистем:

- “Навчальний план”;
- “Робочий навчальний план”;
- “Розрахунок навчального навантаження”;
- “Навчальне навантаження кафедри”;
- “Навчальне навантаження викладача”.

Автоматизована інформаційна система управління навчальним навантаженням створювалась включно засобами мови програмування PHP та СУБД MySQL і має відкритий об’єктно-орієнтований програмний код.

Система “Навчальне навантаження” реалізує дворівневу модель архітектури клієнт-сервер, в якій клієнт звертається до послуг серверу. Клієнтська частина (діалогові компоненти, засоби візуалізації) генерується на сервері і передається користувачеві у HTML (мал. 1.), XLS або PDF форматі. Усі компоненти управління даними: операції з базами даних і файлові операції, бізнес логіка і логіка управління даними розміщуються на сервері.

У системі також реалізовано інтерфейс обміну даними з іншими системами у форматі XML та JSON, що значно спрощує і прискорює процес її інтеграції з іншими системами управління навчальним процесом вищого навчального закладу.

ДИСЦИПЛІНИ - редагування

Цикл дисциплін		Цикл гуманітарної та соціально-економічної підготовки						
Назва дисципліни		Іноземна мова (за професійним спрямуванням)						
Статус дисципліни		Специфічна						
		Специфічна Нормативна Факультативна Альтернативна						
Екз.	Зал.	Курс. роб.		Лек.	Лаб.	Практ.	Семестр	
<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	0	0	36	1	
<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	81	0	0	36	
<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	81	0	0	36	
<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	81	0	0	36	

Мал. 1. Веб інтерфейс редагування параметрів дисципліни навчального плану

Наш практичний досвід реалізації зазначених функцій та впровадження системи в навчальний процес дозволив вирішити низку проблем:

- дублювання дій і документів;
- ухвалення оперативних і адекватних рішень щодо уніфікованого формування навчальних планів, робочих навчальних планів, розподілу навчального навантаження між викладачами, тощо;
- підвищення якості навчання за рахунок аналізу між предметних зв'язків та використання наступності у викладанні дисциплін, варіативності формування індивідуального плану навчання студента, нормування тижневого навчального навантаження студента;
- зниженню витрат на організацію і управління навчальним процесом.

Література

1. Болубаш Я.Я. Організація навчального процесу у вищих закладах освіти: Навч. посібник для слухачів закладів підвищення кваліфікації системи вищої освіти / Я.Я.Болубаш; — К.: ВВП «КОМПАС», 1997.— 64с.
2. Львов М.С., Співаковський О.В., Щедролосьєв Д.Є. Інформаційна система управління вищим навчальним закладом як платформа реалізації управління академічним процесом.- // Комп'ютер у школі та сім'ї, №2,3,4, К. - 2007 р.

Використання програми GANTTPROJECT для побудови календарних графіків при розробці ПВР

Грицук Ю.В., Меліхов О.І.

Донбаська національна академія будівництва і архітектури, м. Макіївка, *yuri.grit-suk@gmail.com*

В роботі розглянуто можливість використання програми GanttProject (поширюється на умовах GNU General Public License) в навчальному процесі підготовки інженерів-будівельників, а саме при розробці проектів виробництва робіт.

Проект виробництва робіт (ПВР) – це основний документ, який регламентує організацію виробництва будівельних робіт відповідно до технологічних правил, вимог до охорони праці, екологічної безпеки і якості робіт. ПВР у системі організаційно-технологічної підготовки будівельних робіт є основним документом, що встановлює порядок інженерного устаткування і облаштованості будівельного майданчика, забезпечує моделювання будівельного процесу, прогнозування можливих ризиків і визначає оптимальні строки будівництва.

Згідно з вимогами нормативних документів, до складу проекту виробництва робіт до числа інших документів включаються:

- календарний графік виробництва робіт або комплексний сітковий графік, у якому встановлюється послідовність і строки виконання робіт з максимально можливим їх суміщенням;
- графіки руху робочих кадрів по об'єкту.

Таким чином, календарний графік визначає послідовність і строки здійснення будівництва, реконструкції, модернізації. Відповідно до календарного графіка будівництва розробляються календарні плани-графіки потреби в робочих кадрах і матеріально-технічних ресурсах.

Зазвичай, на основі прийнятих трудомісткостей окремих процесів, знаючи зв'язок між ними і послідовність їх виконання, побудову календарного плану можна виконувати вручну або за допомогою систем автоматизованого проектування і креслення (наприклад, AutoCAD, BtoCAD, nanoCAD). Однак, у такому випадку виконання оптимізації календарного плану (або, що відбувається частіше – виправлення помилок) досить ускладнене.

Для більш зручного розв'язання таких завдань, як побудова і оптимізація календарного плану, існують чимало програм та програмних комплексів, серед яких є платне (Microsoft Project), умовно безкоштовне, (OpenProj, обмеження можливості виводу і конвертації звіту), а також вільно розповсюджене безкоштовне програмне забезпечення (GanttProject).

Слід відмітити, що пошук і використання безкоштовних аналогів програмних продуктів є однією з задач, що стоїть перед Центром комп'ютерних та інформаційних технологій Донбаської національної академії будівництва і архітектури.

Отже, GanttProject — програма, призначена для планування проектів на основі побудови діаграм Ганта і діаграм типу PERT. Програма розроблена мовою Java. Поширюється на умовах GNU General Public License (ліцензія надає користувачеві право копіювати, модифікувати і поширювати (у тому числі, на комерційній основі) програму (що за замовчуванням заборонено законом про авторські права), а також гарантує, що і користувачі всіх похідних програм отримують перераховані вище права). Підтримується імпорт/експорт документів Microsoft Project, підтримується виведення звітів у різних форматах, включаючи HTML і PDF.

В роботі програма була використана для розв'язання наступного навчального завдання: потрібно побудувати календарний графік і графік потреби в робочих кадрах при виконанні робіт з обладнання даху в ході реконструкції житлового будинку серії 5-515 методом надбудови мансардного поверху.

Передбачалось, що в розпорядженні будівельної організації є всі необхідні для роботи кадри, що мають необхідні розряди і спеціальності. Прізвища робочих були задані довільно.

В результаті, за допомогою GanttProject був побудований календарний графік (Рис.1).

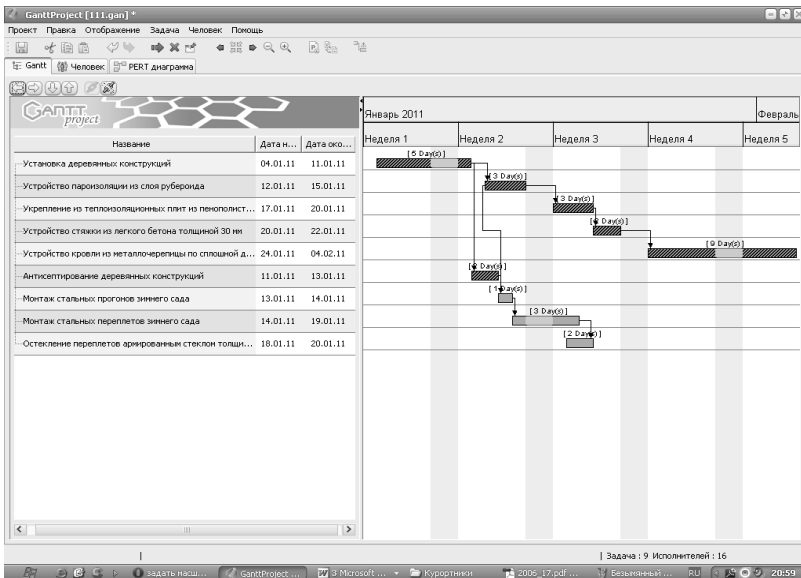



Рис.1. Календарний графік провадження робіт з обладнання покрівлі (навчальний приклад).

Однією з переваг програми є те, що в GanttProject можливе достатньо гнучке налаштування візуалізації графіка, зокрема відображення на графіку різної інформації. У нашому випадку – це тривалість робіт у днях.

Критичний шлях за графіком показаний штрихуванням ().

За критичним шляхом можна визначити, що роботи з обладнання покрівлі з обліком вихідних займають рівно місяць (з 04.01.11 по 04.02.11).

Далі, шляхом присвоєння певним видам робіт бригад необхідного складу, був побудований графік руху робочих кадрів по об'єкту (Рис.2).

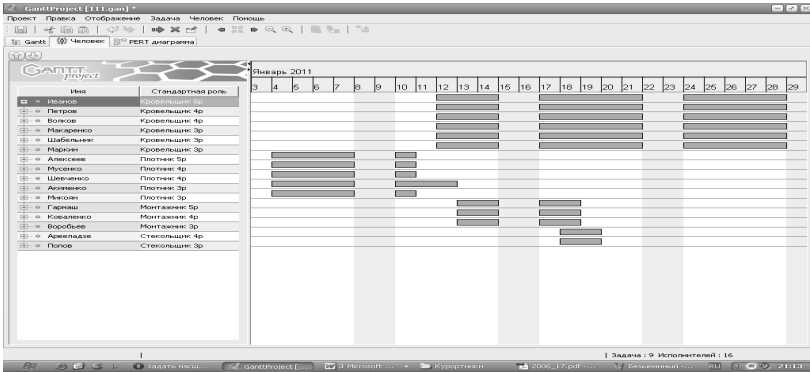


Рис.2. Графік руху робочих кадрів по об'єкту при виконанні робіт (навчальний приклад).

Якщо до списку робітників додати необхідні матеріали або необхідні машини і механізми, то є можливість побудувати, відповідно графік надходження на об'єкт будівельних конструкцій, виробів, матеріалів і устаткування або графік руху основних будівельних машин по об'єкту. На жаль у версії 2.0.10 ці графіки не відособлені у вигляді окремих вкладок.

У якості висновку можна сказати, що розглянута програма дозволяє в достатньому обсязі (як в навчальних, так і в виробничих цілях) виконати побудову календарних планів, що особливо цінне завдяки її доступності.

Література

1. ДБН А.3.1-5-96. Управление, организация и технология. Организация строительного производства. – К.: Державний комітет України у справах містобудування і архітектури, 1996.
2. Посібник з розробки проектів організації будівництва і проектів виконання робіт (до ДБН А.3.1-5-96 «Організація будівельного виробництва»). Частина 1. Технологічна та виконавча документація. – К., 1997.
3. ДБН А.2.2-3-2004. Состав, порядок разработки, согласования и утверждения проектной документации для строительства. – К.: Госстрой Украины, 2004.
4. <http://ganttproject.blogspot.com/search/label/faq>
5. <http://www.ganttproject.biz/>

Разработка док-панели в виде плагина расширения COMPIZ-FUSION

Дёмин В.В, Костюк Д.А

Брестский государственный технический университет, spas.work@gmail.com

Описывается концепция док-панели с динамическим отображением миниатюр окон, построенная в виде плагина расширения для оконного менеджера CompiZ. Концепция содержит ряд нововведений, основанных на оптимизации работы пользователя с группами окон. Выполнена частичная реализация предложенных нововведений для текущей девелоперской версии CompiZ Fusion.

Задача манипулирования окнами в ситуации, когда их совокупная площадь существенно превышает разрешение устройства вывода, возникает почти одновременно с графическим интерфейсом пользователя. Сегодняшние мощности компьютера позволяют использовать для управления окнами различные графические эффекты. Одним из таких решений является размещение в док-панели миниатюрных изображений окон, находящихся за пределом видимой части рабочего пространства. Впервые полноценная док-панель, совмещавшая ярлыки быстрого запуска с апплетами и мини-окнами, обозначавшими запущенные приложения, была реализована в ОС NeXTSTEP ещё в 1989 году; однако мощности компьютеров в то время не позволяли использовать полноценные миниатюры окон, а полноценное динамическое отображение миниатюр (live-preview) стало появляться только в последнее десятилетие, с развитием аппаратно-ускоренных оконных интерфейсов.

Современные аппаратно-ускоренные менеджеры окон используют возможности библиотеки OpenGL для передачи вычислительной нагрузки графическому акселератору. Одно из свойств OpenGL – объекты фреймбуфера – дает оконному менеджеру эффективный доступ к окнам неактивных приложений. Для приложения объекты фреймбуфера выглядят как обычные окна, а для оконного менеджера — как текстуры, которыми можно управлять с помощью обычных команд отрисовки мультитекстур. Одним из наиболее популярных композитных менеджеров, CompiZ Fusion, выполняет отрисовку по схеме [1], изображенной на рис. 1.

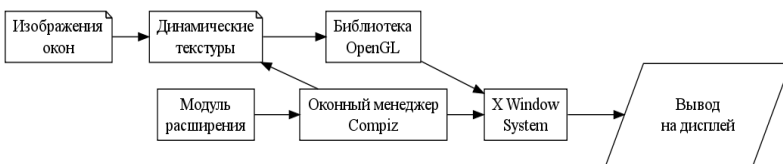


Рисунок 1 — Взаимодействие оконного менеджера со средствами аппаратно-ускоренной графики

Благодаря экономному использованию памяти и аппаратному ускорению Compiз стал популярен среди пользователей и разработчиков. На его основе существует целая инфраструктура плагинов, улучшающих управление окнами, оптимизирующих рабочее пространство и просто украшающих рабочие столы пользователей.

За время существования Compiз предпринималось несколько попыток создания плагина, реализующего док-панель. На сегодняшний день ни одна из них не достигла необходимой степени завершенности для включения в основной набор плагинов, а авторы не выдерживали характерной для данного оконного менеджера гонки изменений в API. Тем не менее, обсуждения на форумах портала compiz.org показывают высокий спрос пользователей на реализацию в Compiз функционала док-панели.

Нами предложена концепция модифицированной док-панели с управлением на основе миниатюр окон в виде модуля расширения Compiз [2, 3]. Предлагаемый подход к управлению окнами базируется на следующих положениях:

- панель содержит миниатюры минимизированных окон;
- благодаря аппаратному ускорению отрисовки панели миниатюры отображают в реальном времени все изменения, происходящие в минимизированных окнах;
- перетаскивая миниатюры в панели, пользователь может группировать окна для одновременного управления ими.

Группы выделяются более плотным расположением миниатюр в панели. Для переключения групп окон используется эффект Cover Flow.

Модель группирования мини-окон является эффективным инструментом управления окнами. Окна, отображаемые на экране в естественном масштабе и не имеющие миниатюр в панели, считаются текущей группой. Переключение между группами позволяет выполнять средствами единственной док-панели одновременную миниатюризацию и разминиатюризацию нескольких окон по аналогии с пейджером виртуальных рабочих столов — текущая группа перемещается в панель в виде миниатюр, а ее место занимают разминиатюризированные окна соседней группы.

Помимо группового переключения окон панель оказывается удобной также для группового запуска приложений (что позволяет перекрыть функционал ярлыков запуска стандартной док-панели). Панель мини-окон может содержать для этого специальный доклет — псевдогруппу с пиктограммой, отвечающей за историю работы оконного интерфейса. Выбор ее в списке мини-окон приводит к отображению на экране групп окон, формировавшихся пользователем ранее [4].

При реализации предложенного функционала в виде модуля расширения за основу был взят тестовый плагин dock, доступный как git репозиторий по адресу: git://anongit.compiz-fusion.org/users/warlock/dock. На момент подготовки данной публикации реализация концепции выполняется для Compiз Fusion версии 0.9 и находится на стадии переключения

групп окон. Код разработки доступен по адресу <http://gitorious.org/dock>.

Литература:

1. Дёмин В. Разработка аппаратно-ускоренных графических интерфейсов на базе оконного менеджера Comriz // Материалы Конференции «Linux Vacation / Eastern Europe 2010». Тематическое приложение к журналу «IT Бел». Июнь 2010 г. - С. 5-6.
2. Костюк Д.А., Демин В.В. Модель мини-окон с динамическим отображением в аппаратно-ускоренном графическом интерфейсе // Вестник БрГТУ. - 2009, №5 (59): Физика, математика, информатика. - С. 71-74
3. Дёмин В.В. Реализация модуля расширения для миниатюризации окон приложений в оконном менеджере Comriz // Сборник конкурсных работ студентов и магистрантов. - Брест: БрГТУ, 2010. - С. 250-255.
4. Diomin V.V., Kostyuk D.A. Grouped windows focus switching with variable scale factor // Computer Science & Engineering 2010 (CSE-2010): proceedings of the IV International Academic Conference of Young Scientists. November 25-27 2010, Lviv, Ukraine. – P. 32-33.

До питання вивчення проблеми впровадження вільного програмного забезпечення в загальноосвітніх школах

Дегтярьова Н.В.

аспірант інституту інформатики Національного педагогічного університету імені М.П. Драгоманова, nelya-d@yandex.ru

В роботі розглядається необхідність вивчення продуктів вільного програмного забезпечення в загальноосвітніх навчальних закладах в умовах компетентнісного підходу в навчанні. Розглядаються питання зміни парадигм навчання як фактор розвитку самостійності учня.

На даний час стан впровадження в школах вільного програмного забезпечення є досить актуальним. Досить тривалий час обговорюються переваги та недоліки застосування вільного програмного забезпечення саме в середніх навчальних закладах. Переваги пояснюються з точки зору безкоштовності та безпечності розповсюдження, високої швидкості розробки нових програмних продуктів, відкритість кодів програм. До недоліків відносять меншу популярність, відсутність підтримки розробників, високі вимоги до рівня фахівців, що впроваджують програмне забезпечення [1,2]. Запровадження освітніх інновацій, неперервність освіти, інтеграція до європейського освітнього простору вимагає від майбутнього фахівця окрім вміння застосовувати отримані знання, навички також вчитися протягом життя, вірно застосовувати накопичений життєвий та професійний досвід. Ми розглядаємо навчання учнів користуватись альтернативним програмним забезпеченням, а саме вільним, з точки зору зміни парадигм навчання.

Результатами навчання в загальноосвітній школі на сьогодні повинні

бути знання, вміння, навички, здібності та компетентності молодій людині. В зв'язку зі змінами в освіті, науці змінюються і парадигми навчання. Так, наприклад, цінність навчання полягає в подальшій самореалізації людини, мотивація — у учня в отриманні задоволення від результатів навчання, а вчитель повинен бути зацікавленим в розвитку школяра, зміщується акцент в бік самостійності як навчання, так і оцінювання. [2,26]. Головною проблемою використання вільного програмного забезпечення в школі найчастіше є небажання вчителя опанувати нову сферу, його консерватизм. Адже при виконанні навчальної програми з інформатики (рівень стандарту), наприклад, 9 та 10 класів при вивченні програмного забезпечення загального призначення, а саме текстового процесору, практичні роботи можна виконувати не тільки в одному програмному продукті, але й застосовувати паралельно два процесори, розбиваючи завдання на дві частини. Таким чином, вчитель не лише знайомить учнів з двома альтернативними програмами, але й виховує незалежну, впевнену в своїх можливостях, готову до конкуренції особистість.

Література

1. М.Карпенко, М. Кияк Перспективи та можливості впровадження вільного програмного забезпечення в навчальних закладах та державних установах України [Електронний ресурс]: Режим доступу: <http://old.niss.gov.ua/Monitor/june2009/15.htm>
2. Л.В. Кожитов Инновации в образовании: монография / Л.В. Кожитов, С.Г. Емельянов, В.А. Дёмин [и др.]; Юго-зап. гос. ун-т. Курск, 2010. - 610с.

Теоретичні та методологічні основи розробки автоматизованих систем передачі знань

Федорук П.І., Дутчак М.С.

Прикарпатський національний університет імені Василя Стефаника
marichkadu@gmail.com

У даній публікації описано алгоритм і програмну реалізацію механізму адаптації навчального заняття до можливостей конкретного студента. Програмний модуль використовує результати проходження занять студентами і визначає ймовірності досягнення кожного із станів засвоєння за певну кількість спроб, рекомендовану складність заняття та оптимальну кількість спроб для досягнення необхідного рівня засвоєння.

Підвищення якості навчання, інтенсифікація навчального процесу і перехід на нові технології в даний час неможливі без впровадження в процес навчання різного роду автоматизованих навчальних систем (АНС), зокрема адаптивних дистанційних технологій навчання.

Впровадження адаптивних дистанційних технологій навчання стримується через відсутність якісного нового навчально-методичного забезпечення і його програмної підтримки.

Так як метою побудови адаптивних систем дистанційного навчання є підвищення якості освіти, то необхідно так комплектувати навчальне заняття, щоб його зміст був найбільш сприятливий для засвоєння поданого матеріалу конкретно взятим студентом.

Для реалізації механізму адаптації навчального заняття до можливостей даного конкретного студента розроблено алгоритм і написано програмний модуль. Даний програмний продукт написаний з використанням мов програмування Perl (модуль CGI) і HTML та створено базу даних для зберігання результатів проходження занять, використовуючи СУБД PostgreSQL.

Досліджувана складна система (заняття) включає скінченну множину підсистем, які являють собою навчальні блоки. Кожен блок характеризується рівнем складності, змістом матеріалу, методикою викладу матеріалу, відведеним часом, та інші. Із набору альтернатив складових занять формується навчальне заняття.

Етап навчання визначається рівнем знань студента, а складність заняття повинна відповідати ступеню засвоєння нових знань. Для визначення необхідних характеристик студента потрібно в перший день курсу провести два тести по кожному з критеріїв: початкові знання студента тематики курсу та ступінь засвоєння нової інформації. [2].

Нехай кожне із занять відноситься до одного із трьох рівнів складності і характеризується ступенем засвоєння. Заняття як найнижчого, так і найвищого ступеня складності можна пройти на високий рівень засвоєння, але в абсолютному значенні ці величини, звичайно, не будуть еквівалентними.

Значення ступеня засвоєння вимірюється в десятибальній шкалі. Стану E_1 відповідає інтервал $[0...5)$ – низький рівень, стану E_2 інтервал

[5...9) – середній, стану E_3 [9...1] – високий (Рис. 1)

Адаптивна система									
Результати:		Складність 1			Складність 2			Складність 3	
Номер	Прізвище	Спроба 1	Спроба 2	Спроба 3	Спроба 1	Спроба 2	Спроба 3	Спроба 1	Спроба 2
1	Абрам Оксана	7	8	9	6	8	8	5	6
2	Андрюк Василь	7	6	7	7	8	2	5	5
3	Бойчук Василь	8	9	10	7	9	9	1	5
4	Васильок Олег	6	7	7	4	5	6	2	3
5	Воржеват Оксана	8	9	9	7	7	6	5	5

Рис. 1. Результати проходження занять.

Початкові ймовірності перебування студента в i -му стані засвоєння визначається з початкового тестування чи обчислюється як відношення кількості занять пройдених на цей рівень (k_i), до загальної кількості пройдених занять за певний період:

$$p_i(0) = \frac{k_i}{k_1 + k_2 + k_3}$$

Ймовірності переходів із одного стану в інший визначаються з відношення кількості студентів, які здійснили перехід із i -го стану в j -тий за n -тою спробою до загального числа студентів, що перебували у i -му стані:

$$p_{ij}^n = \frac{S_{ij}^n}{S_i^n}$$

Можливість досягнення j -го стану за n спроб $p_j(n)$, $n, j = 1, 2, 3$ дорівнює: $p_j(n) = \sum_i (0) p_{ij}^n$ (сума добутків умовних ймовірностей) [1, 3].

Виходячи із цих ймовірностей, визначається складність пропонованого даному студенту заняття і достатня кількість спроб. Якщо з достатньою ймовірністю студент пройде заняття третьої складності на потрібний рівень, то йому рекомендується ця складність, інакше перевіряється заняття складності E_2 і, якщо ця складність не підходить, то пропонується заняття складності E_1 ті, тобто пропонується максимально допустима складність заняття. Потім для заняття цієї складності рекомендується кількість спроб (Рис.2).

Отже, розроблено механізм адаптації, який відповідно до рівня знань студента визначає, з якого етапу розпочинати вивчення курсу, відповідно до ступеня засвоєння рекомендує складність пропонованого студенту заняття, а відповідно до ступеня сприйняття рекомендовану кількість спроб.

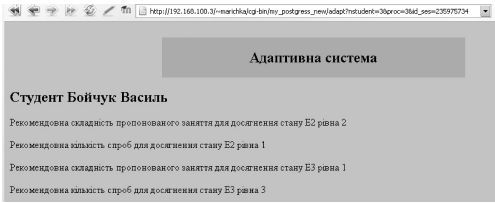


Рис. 2. Рекомендовані складності занять та кількості дозволених спроб.

Для реалізації механізму адаптації написано програмний, який використовує результати проходження занять студентами, обробляє їх, і як результат повертає:

- ймовірності перебування студента у кожному із рівнів засвоєння нового матеріалу;
- перехідні ймовірності із одного стану засвоєння нового матеріалу в інший в залежності від складності заняття і номеру спроби;
- можливості досягнення кожного із станів засвоєння за певну кількість спроб;
- рекомендовану складність заняття та оптимальну дозволена кількість спроб для досягнення необхідного рівня засвоєння.

Створений механізм адаптації спрямований на підвищення якості навчання і раціональніше використання ресурсів систем навчання.

Створений адаптивний модуль може бути застосований не тільки в системах дистанційного навчання, але і при здійсненні традиційного процесу навчання для допомоги вчителю чи викладачу у побудові навчальних занять.

Література

1. П.Федорук, М. Дутчак. Автоматизація процесу квантування лекційного матеріалу в адаптивних системах дистанційного навчання. // Інноваційні комп'ютерні технології у вищій школі. Матеріали 2-ї науково-практичної конференції. – Львів, 2010 р. – С.14-18.
2. Андреев А.А. Введение в дистанционное обучение. ч.II М.: МЭСИ, 2005.
3. Ю. Рыжиков "Вычислительные методы" изд. ВНУ, 2007 г., 400 стр.

Вільнопоширюване програмне забезпечення курсу «Нові інформаційні технології» для студентів спеціальності «Біологія»

Єфименко В.В.

Національний педагогічний університет імені М.П.Драгоманова, priief@ukr.net

В доповіді описано вивчення курсу «Нові інформаційні технології» для студентів спеціальності «Біологія», яке передбачає формування знань, умінь, навичок при підготовці учителя біології загальноосвітньої школи. Базовими для даного навчального курсу є знання, уміння і навички, набуті студентами після закінчення загальноосвітніх навчальних закладів.

На сьогоднішній день вже цілою низкою країн визнано доцільність використання вільного ПЗ в державному секторі та сфері освіти. До таких належать, зокрема, країни Західної Європи, Японія, Китай, Росія, Індія, країни Скандинавії та інші.

Використання вільнопоширювального програмного забезпечення виступає виразником демократичного поступу, свободи, відкритого суспільства та суспільства знань, а також права вивчати та ділитись своїми доробками з іншими. Тому при створенні наповнення курсу «Нові інформаційні технології» для студентів спеціальності «Біологія» було орієнтовано на ознайомлення студентів як з комерційним так і вільнопоширюваним програмним забезпеченням.

Були використані літературні джерела, в яких подається сучасне висвітлення проблем навчання інформаційних технологій [4,6,9], джерела з вивчення операційних систем і прикладного програмного забезпечення [3,5,7,10], дисертаційні дослідження про роль інформаційних технологій в системі фахової підготовки вчителя біології [1,2,11], освітні інтернет-ресурси [8].

Вивчення дисципліни «Нові інформаційні технології» за робочим планом спеціальності «Біологія» заплановано на 3-му курсі в осінньому семестрі. Всього на вивчення дисципліни дається 108 годин, з них лекції - 22 години, лабораторні роботи - 20 годин, самостійна робота - 48 годин, індивідуальна робота - 16 годин. Завершується вивчення диференційованим заліком.

Запропоновано такий зміст програми:

Модуль 1. Інформаційні технології.

1. Дані та інформаційні процеси. Інформаційна система.
2. Операційні системи.
3. Системи опрацювання графічних даних.
4. Системи опрацювання текстів.
5. Електронні таблиці.
6. Бази даних.

Модуль 2. Системи Multimedia, комп'ютерні мережі та ППЗ

7. Системи Multimedia. Системи створення електронних презентацій
8. Комп'ютерні мережі та Інтернет.

9. Програмні засоби навчального призначення. Технологія розв'язування задач з використанням засобів сучасних інформаційних технологій.

10. Програмні засоби для самостійного ознайомлення та індивідуальної роботи.

Сучасні програмні засоби з даної тематики:

Засоби загального призначення:

- операційні системи Windows7 та LINUX з графічним інтерфейсом KDE;
- графічний редактор OpenOffice Draw. Він є частиною пакету Open Office. Поряд з ним для ознайомлення розглянули способи створення векторної графіки в Sodipodi або Inkscape чи Dia. Широкі можливості обробки растрової графіки (зокрема й цифрового фото) надає GIMP. Його можливості мало в чому поступаються широко знаному Photoshop. Вивчення GIMP було винесено для самостійного опрацювання.
- для максимально "інтуїтивного" стилю роботи, найбільш зручним являється OpenOffice, який дозволяє створювати і текстові документи, і електронні таблиці, бази даних та презентації.
- навігація в мережі Інтернет проводилась за допомогою броузерів Opera та Mozilla.

Спеціалізовані програми:

- спеціалізована програма візуалізації молекул «RasMol V2.7.5 Molecular Visualisation Program», яка дозволяє відтворити на екрані молекули ДНК і т.п.;
- спеціалізовані програми, які дозволяють перекодувати послідовність амінокислот білку або нуклеотидів ДНК в музичні твори «PROM - Protein Music Composer» та «Gene2music»;
- програмний засіб «Neurosim for Windows» (симулятор роботи нейронів) - сімейство програм (by W.J.Heitler з The Gatty Marine Laboratory, Університет св. Андрея, Шотландія), що моделюють роботу як окремих нейронів, так і нейронних мереж.
- програми складання родоводів «GenoPro», «MyHeritage Family Tree Builder», «Genezis», які дозволяють виконувати аналіз родоводів, створення учнями родоводів по умові завдань, наприклад, родовід для завдання, в якому треба розрахувати вірогідність народження хворих дітей в парі і т.п.

Всі програми вільнопоширювані, але з англійським інтерфейсом. Безкоштовних вітчизняних розробок не вдалось знайти.

Під час написання завдань для вивчення стандартного пакету програм добирався матеріал, який відповідає спеціалізації «Біологія».

Зокрема, при вивченні графічних редакторів пропонується зобразити схеми лактози, глюкози, мальтози, целюлози, схему життєвого циклу пивних дріжджів, схему будови клітини бактерії, схематичне подання іонних каналів в мембрані клітини тощо. При вивченні електронних таблиць пропонується створити таблицю, в якій внесені результати вимірів довжини 30 листків вишні, та щоб виявити закономірності модифікаційної мінливості, необхідно провести статистичне опрацювання даних,

одержаних в результаті дослідження. При вивченні баз даних запропоновано створити базу даних ссавців, з фото та зазначенням роду, виду та ареалу проживання.

Література

1. Жалдак М. И. Система подготовки учителя к использованию информационной технологии в учебном процессе : дис. ... в форме науч. доклада доктора пед. наук : 13.00.02 / Жалдак М. И. ; АПН СССР; НИИ содержания и методов обучения. – М., 1989. – 48 с.
2. Беляков О.И. Использование средств новых информационных технологий для контроля знаний и умений учащихся по биологии: Автореф. дис...канд. пед. наук: 13.00.02 / РГПУ им. А.И. Герцена. - СПб., 2000. - 19с.
3. Єфименко В.В., Оніщенко С.М. Опрацювання табличних даних засобами Microsoft Excel. Лабораторний практикум: Навчальний посібник. –К.: Логос, 2005. – 167с.
4. Компетентнісний підхід у сучасній освіті: світовий досвід та українські перспективи: Бібліотека з освітньої політики / Під. заг. ред. О.В.Овчарук. – К.: «К.І.С.», 2004. – 112 с.
5. Лапінський В.В., Габрусев В.Ю., Бачинська Н.В. Основи операційних систем. – навч. Посібник, Тернопіль: Богдан, 2002, 78 с. 4.65 друк.арк
6. Рамський Ю.С. Зміни в професійній діяльності вчителя в епоху інформатизації освіти// Науковий часопис НПУ імені М.П. Драгоманова. Серія №2. Комп'ютерно-орієнтовані системи навчання: Зб. наук. праць/ Редрада. – К.: НПУ імені М.П. Драгоманова, 2007. – №5(12). – С. 10-12.
7. Рамський Ю.С., Цибко Г.Ю. Проектування і опрацювання баз даних.- К.,1998.- 84 с.
8. Список Інтернет-сайтів, присвячених біології та освіті (огляд літератури) [Електронний ресурс]. – Режим доступу : <http://www.informika.ru>
9. Титовец Т. Сущность информатизации естественно-научного образования в системе профессиональной подготовки учителя / Т. Титовец // Information communication technology in natural science education. Минск. – 2006. – С. 144–146.
10. Фурман О.А. Роль інформаційних технологій в системі фахової підготовки вчителя біології / О.А. Фурман // Наукові записки ТНПУ імені В. Гнатюка. Серія : Педагогіка. – 2008. – № 8. – С. 28–32.
11. Фурман О.А. Сучасні інформаційні технології / О.А. Фурман // Посібник для студентів біологічного факультету. – Кременець : РВЦ КОГПІ ім. Тараса Шевченка. – 2007. – 54 с

Опрацювання результатів спостережень проходження Венерою диска Сонця 8 липня 2004 року та розрахунки обставин майбутніх астрономічних явищ

Жолонко М.М.

*Черкаський національний університет ім. Богдана Хмельницького,
zholonko@ukr.net*

Описано можливості астрономічної програми Eclips, створеної для порівняння і передбачення особливостей руху світил при проходженні внутрішніх планет по диску Сонця, сонячних затемнень, а також сходу і заходу Сонця і Місяця з урахуванням рефракції. Програма розроблена в середовищі Matlab і може бути використана в дослідницьких і в навчальних цілях.

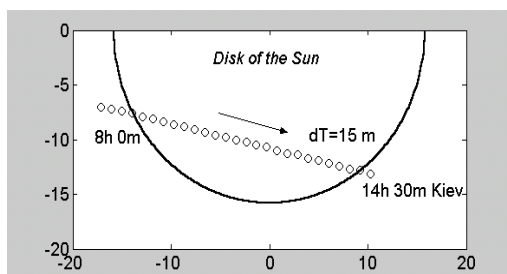
Сьогодні є широкі можливості через мережу інтернет отримати відкрите і зручне програмне забезпечення, що дозволяє докладно і точно дізнатися про розташування зірок, планет, комет та інших об'єктів на небесній сфері. Це - усілякі програми-планетарії. Вони також дають анімацію руху небесних тіл в будь-якому зручному для користувача темпі. Складніше йде справа з такими тонкими явищами, як докладні спостереження проходжень, затемнень, особливості сходу і заходу Сонця і Місяця (їх частин щодо лінії горизонту з урахуванням рефракції). Запропонована програма Eclips, написана в середовищі Matlab, на прикладі проходження Венерою диска Сонця ілюструє можливості створеного програмного забезпечення для навчальних та дослідницьких цілей, проблем навігації та подальшого розвитку інформаційних технологій. Результати розрахунку порівнюються з безпосередніми теодолітним спостереженнями, проведеного 8 червня 2004 р. у м. Черкаси.

Меркурій і Венера відносяться до нижніх планет, чії орбіти розташовуються усередині орбіти Землі. Періодично Земля і одна з планет опиняються по один бік від Сонця. При цьому система "Сонце-планета-Земля" утворює в космічному просторі або трикутник, або близьку прямий лінію. В останньому випадку земний спостерігач може побачити нижню планету на диску Сонця. Оскільки планети і Землю відділяють десятки мільйонів кілометрів, для неозброєного ока Венера на Сонце виглядає чорною точкою, а щоб помітити Меркурій, необхідно хоча б десятиразове збільшення. Проходження планет по диску Сонця відбуваються рідко і бувають видно на всьому зверненому до Сонця півкулі Землі. Приблизний час проходження планети по диску Сонця можна визначити з періодичності, з якою повторюються ці небесні явища. Проходження Меркурія по диску Сонця можливі з 6 по 14 листопада та з 6 по 10 травня. Якщо проходження Меркурія припадає на 10 листопада або 8 травня, планета перетинає сонячний диск по діаметру. Коли проходження припадає на сусідні дати, шлях планети - хорда. Ось, наприклад, дати спостережень і майбутніх проходжень Венери з вказівкою тимчасових інтервалів між сусідніми подіями: 7 грудня 1631; 4 грудня 1639; 6 червня 1761; 4 червня 1769; 9 гру-

дня 1874; 6 грудня 1882; 8 червня 2004; 6 червня 2012; 11 грудня 2117; 8 грудня 2125. Листопадові проходження Меркурія відбуваються через 4750,733 діб або через 2550,0 діб. Останнє проходження Меркурія по диску Сонця, видиме в Європі, відбулося 6,292 листопада 1993 року. Таким чином, нескладно підрахувати чергування листопадових проходжень Меркурія в доступному для огляду майбутньому: 9,025 листопада 2006 року, 11,758 листопада 2019, 13,491 листопада 2032 і 7,491 листопада 2039 ...

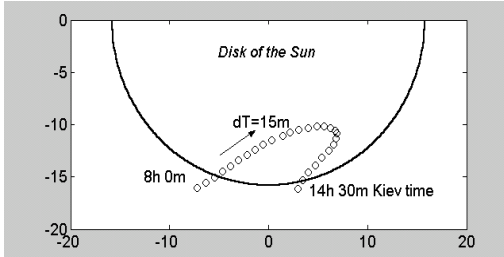
Травневі проходження Меркурія відбуваються через 12050,98 діб і потім через 4751,3 діб. Знаючи, що останнє травневе проходження сталося 9,52 травня 1970 року, слід очікувати, що вони знову наступлять: 7,5 травня 2003 року, 9,8 травня 2016 і 7,78 травні 2049 ... Так, наприклад, перше в ХХІ столітті проходження Меркурія по диску Сонця сталося 7 травня 2003 року. Проходження Венери по диску Сонця 8 червня 2004 виявилось сприятливим для спостережень в Європейській частині і в Західному Сибіру. На Далекому Сході Сонце зайшло за обрій до закінчення явища. На американському континенті, навпаки, - при сході Сонця проходження вже підходило до завершення. Проходження Венери по диску Сонця відбувається тоді, коли планета Венера перебуває точно між Сонцем і Землею, закриваючи собою крихітну частину сонячного диска. При цьому планета виглядає з Землі як маленьке чорне цятку, що переміщається за Сонцем. Проходження схожі з сонячними затемненнями, коли наша зірка закривається Місяцем, але хоча діаметр Венери майже в 4 рази більше, ніж в Місяця, під час проходження вона виглядає приблизно в 30 разів менший за Сонце, так як знаходиться значно далі від Землі, ніж Місяць. Спостереження цього явища дозволяють досить точно визначити відстань від Землі до Сонця (астрономічну одиницю) методом паралакса.

8 червня 2004 року проходження Венери в Черкасах можна було спо-



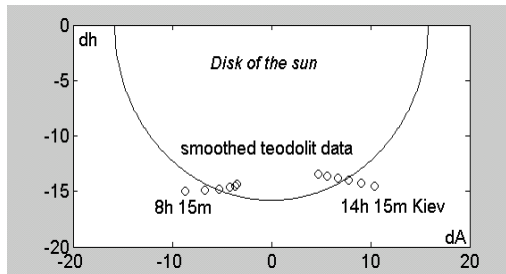
стерігати в умовах практично ідеальної видимості. На рисунку наведені результати теодолітних спостережень для початкової стадії проходження і його другої половини, оброблені програмою Eclips для безпосередньо спостерігаються в горизонтальних координатах висот і азимутів Сонця і Венери.

Розрахункові поправки до геоцентричних моментів початку проходження (T1 - перший контакт), кінця входження Венери в диск Сонця (T2), останній контакт (T4) і самі моменти для точки спостереження в цілому узгоджуються з передбаченими В.В. Михальчуком (Одеський АК на 2004 рік) для м.Черкаси, час київський: T1 = 8ч 19м 49с; T2 = 8ч 39м 0с; T3 =



14ч 3м 11с; T4 = 14ч 22м 10с, а також з даними NASA, Фред Еспенак: (схід Сонця 4ч48м) T1 = 8ч 19м 18с; T2 = 8ч 38м 55с; T3 = 14ч 2м 50с; T4 = 14ч 22м 13с.

Повну реконструкцію за допомогою програми Eclips проходження Венери ілюструє рис. 2 (різниці висот і азимутів), а також рис.3, де побудова виконана у небесних координатах (пряме сходження-відміна). Відповідність точності зазначених вище моментів часу реально спостережуваним залежить від використовуваних для цього даних програми-планетарію. З неї з дискретним кроком бралися дані положення на небесній сфері Сонця і Венери і будувалися їх різниці. Програма Eclips дозволяє передбачити особливості будь-якого майбутнього проходження Венери і Меркурія, затемнення Місяця, а також спостерігати схід або захід Сонця і



Місяця (з урахуванням рефракції).

Вільна система управління мережею NOC PROJECT

Зайка Д. С.

старший мережевий інженер, ТОВ ТРК Соніко-связь, Донецьк, zaika@so.net.ua

До останнього часу в області універсального програмного забезпечення для управління комп'ютерними мережами передачі даних існували тільки пропрієтарні рішення. Існуюче раніше вільне ПЗ виконувало, зазвичай, специфічні задачі - управління обладнанням, збір та обробка статистики, моніторинг і не намагалося вирішити проблему управління мережами в комплексі. Однак проект - NOC Project, який з'явився всього кілька років тому, але активно розвивається, вже зараз готовий задовольнити потреби в управлінні мережею багатьох операторів зв'язку, провайдерів, а також організацій, що активно експлуатують комп'ютерні мережі.

Протягом останніх кількох десятків років пропрієтарному ПЗ для універсального управління мережами практично не було альтернативи. Існуючі рішення від великих компаній - CiscoWorks, HP OpenView, CA Spectrum і менш відомі, наприклад - Zyxel NetAtlas, при достатній універсальності завжди були досить дорогими і, при цьому, розрахованими, в першу чергу, на обслуговування обладнання власного виробництва і обмеженої кількості партнерів. Відомі вільні системи такі як OpenNMS, Nagios, Zabbix та інші - є, в першу чергу, засобами моніторингу мережі і мають значно менший функціонал, ніж пропрієтарні. Тим не менш, існує вільний проект, розрахований на універсальне застосування в якості системи управління мережею, назва проекту - NOC Project (www.nocproject.org). Незважаючи на те, що даний проект досить молодий, йому всього кілька років, він дуже швидко розвивається і вже кілька років активно використовується в безлічі організацій, що займаються обслуговуванням комп'ютерних мереж. Я для себе відкрив цей проект більше 2-х років тому, і він уже тоді міг автоматизувати значну частину завдань (до цього нормально не вирішених) з обслуговування достатньо великої домашньої мережі в місті з більш ніж мільйонним населенням. Проект реалізований мовою Python, і має BSD-подібну ліцензією, а з недавнього часу розробники стали активно залучати спільноту не тільки до написання коду, а й до внесення коду в репозиторій проекту (mercurial-based), а також до управління проектом в цілому.

В останньому релізі продукту реалізовані наступні можливості (я назву найбільш значущі для мене і, відповідно, маю досвід впровадження та експлуатації яких, в порядку, в якому вони впроваджувалися на моїх підприємствах):

1. Configuration Management (управління конфігураціями обладнання). Ця частина проекту здійснює операції з локальними репозиторіями даних, здійснюючи збір параметрів конфігурації з устаткування, роботу з зонами DNS, а також з даними, необхідними для координації взаємодії автономних систем в публічних базах даних таких як RipeDB.
2. Address Space Management (управління адресним діапазоном IPv4, IPv6).

Дуже гнучка і зручна база даних для обліку адресного простору, що включає концепцію VRF, засоби пошуку й інструменти автоматизації DNS зон, пов'язаних з IP адресами.

3. Virtual Circuit Management (управління віртуальними мережами). Знову ж таки вкрай гнучка і зручна база для обліку віртуальних з'єднань (dot1q VLANs, FR DLCI, static MPLS labels, Q-in-Q і т. д.). Крім того існують інструменти для створення зв'язків з об'єктами Address Space Management, що також полегшує облікову роботу.
4. Knowledge Base - база знань, інструмент для ведення документації на підприємстві. Окрім самої KB існує ще й досить корисний інструмент - Reference Books, який представляє собою таблиці для зберігання довільних довідкових даних, які гнучко настраюються.
5. DNS - інструмент для управління DNS зонами, заснований на використанні репозиторіїв.
6. Peering Management - система для автоматизації роботи з публічними базами даних і автоматизації конфігурації граничних маршрутизаторів.
7. Fault Management - система моніторингу, працює з обладнанням, використовуючи SNMP-traps і syslog. Може дуже гнучко налаштовуватися з використанням класифікаторів подій і кореляцій між подіями.

На даний момент проект підтримує величезний перелік обладнання, при цьому профілі для нових пристроїв з'являються досить регулярно, а існуючі постійно поліпшуються. Процес створення нових профілів досить простий, тому для користувача проекту, чиє обладнання не підтримується, не складе великих труднощів додати свої пристрої до списку підтримуваних. Крім цього, проект включає в себе безліч невеликих корисних інструментів для вирішення найрізноманітніших задач, досить згадати різні калькулятори для специфічних для мережного інженера розрахунків таких як полісінг, можливість створення графів топологій зв'язності мережевих пристроїв з використанням різних алгоритмів, створення звітів по SAM таблицях комутаторів та інше.

Незважаючи на досить тривалий досвід роботи з даним проектом, багато можливостей проекту я згадати не можу тому, що сам не використовував або просто упустив. Проте, поза сумнівом, що великій кількості мережевих фахівців, які активно працюють з комп'ютерними мережами, даний проект буде дуже корисний. А, враховуючи високу швидкість розвитку проекту, в недалекому майбутньому його можливості зможуть охопити переважну більшість задач, що вирішуються NOC інженером.

Література:

1. NOC Project [Головна сторінка проекту](http://www.nocproject.org) - <http://www.nocproject.org>
2. OpenNET [Релиз системы операционного управления и контроля сетевой инфраструктуры NOC 0.6](http://www.opennet.ru/opennews/art.shtml?num=29117) - <http://www.opennet.ru/opennews/art.shtml?num=29117>

Про досвід використання офісного пакету OPENOFFICE.ORG в курсі інформатики для економічних і юридичних спеціальностей ВЗО

Злобін Г. Г.

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

В доповіді розглядається можливість використання офісного пакету OpenOffice.org в курсі інформатики для студентів вищих закладів освіти

Сьогодні неможливо уявити випускника вищого закладу освіти (ВЗО), який не вмів працювати з текстовими документами, електронними таблицями та презентаціями на ПЕОМ. Кількість офісних пакетів, якими можна скористатись з цією метою, немала: Microsoft Office; Corel WordPerfect Office; IBM Lotus SmartSuite; IBM Lotus Symphony; iWork; OpenOffice.org; ThinkFree Office; Лексикон 5.1.

До офісних пакетів, які встановлюються на локальні робочі станції, слід додати онлайнві офісні пакети: Zoho Office Suite; Google Docs; Microsoft Office 365; Oracle Cloud Office; ThinkFree Office Online; OpenGoo.

В зв'язку з цим важко передбачити, з яким офісним пакетом зустрінеться випускник ВЗО на своєму робочому місці.

Разом з тим, після перегляду навчальних програм і підручників з інформатики для студентів вищих закладів освіти (ВЗО) може скластись враження, що у світі є лише одна операційна система Microsoft Windows і єдиний офісний пакет Microsoft Office [1-3]. Почасти це зумовлено політикою видавництва (українських і російських), які масово друкують підручники і навчальні посібники з описом лише пропріетарного програмного забезпечення, почасти політикою Міністерства освіти і науки України, яке далі розмов про впровадження вільного програмного забезпечення в навчальний процес не йде. Свою частку в замовчування факту існування конкурентно спроможного вільного програмного забезпечення вносять викладачі ВЗО, які не бажають освоювати нові програмні засоби і не переймаються ліцензійністю програм, використовуваних в навчальному процесі.

В умовах існування великої кількості альтернативних офісних пакетів орієнтація більшості викладачів ВЗО на Microsoft Office знижує професійну компетентність випускників. Для забезпечення готовності випускників ВЗО до використання будь-якого сучасного офісного пакету потрібно перейти від вивчення інтерфейсу Microsoft Office (тим більше, що Microsoft починаючи з Microsoft Office 2007 докорінно змінив попередній інтерфейс) до формування алгоритмів виконання основних операцій з офісним пакетом:

- створити новий документ — Файл (File) - Новий (New) або відповідний інструмент в панелі інструментів;
- відкрити існуючий документ — Файл (File) - Відкрити (Open) або від-

повідний інструмент в панелі інструментів;

➤ зберегти створений документ — Файл (File) — Зберегти/ Зберегти як (Save) або відповідний інструмент в панелі інструментів;

➤ вставити в текст малюнок — Вставка (Insert) — Малюнок (Picture) — From File...;

➤ вставити в текст таблицю — Таблиця (Table) — дії з таблицею і її елементами;

➤ і т.д.

Для формування таких алгоритмів достатньо розглянути на лекційних заняттях повністю основний офісний пакет, за аналогією — один стаціонарний і один мережевий офісний пакет. На лабораторних заняттях в основному офісному пакеті виконуються усі завдання, передбачені програмою курсу, в додаткових офісних пакетах — відкриття і редагування документів, створених в основному офісному пакеті. Такий підхід автор використав у Львівському банківському інституті (2003-2006 рр.) та Львівському державному університеті внутрішніх справ (2008-2009 рр.):

основний офісний пакет — Microsoft Office XP;

додатковий офісний пакет — OpenOffice.org.ukr.

Microsoft Office XP був основним пакетом лише тому, що цей пакет вже був встановлений в комп'ютерних лабораторіях цих ВЗО, а от OpenOffice.org довелося встановлювати автору доповіді.

На превелике здивування автора в обох ВЗО були встановлені російськомовні Microsoft Office XP, а у Львівському державному університеті внутрішніх справ завідувач кафедри інформатики навіть робив спробу заборонити використовувати україномовне програмне забезпечення в навчальному процесі.

На думку автора, найкращим претендентом на роль основного офісного пакету є OpenOffice.org.ukr, оскільки він достатньо добре українізований та вільнопоширюваний, як додатковий офісний пакет можна використовувати Microsoft Office 365 (якщо він буде доступним в Україні і, бажано, безоплатно для навчальних закладів). Найкращим претендентом на роль мережевого офісного пакету є Google Docs.

Література

1. Рзаев Д. О., Шарапов О. Д., Ігнатенко В. М., Дибкова Л. М. Інформатика та комп'ютерна техніка: Навч.-метод. посібник для самост. вивч. дисц. — К.: КНЕУ, 2002. — 486 с.
2. Доценко В.І., Сілкова О.В. Медична інформатика: Навч.- посібник. — Полтава: 2005. — 164 с.
3. Інформатика. Комп'ютерна техніка. Комп'ютерні технології: Підручник. — К.: Каравела, 2003. — 464 с.

Використання ВПЗ для підтримки вивчення математичної логіки

Люха В.А.

Полтавський національний педагогічний університет ім. В.Г. Короленка
wtg2702@rambler.ru

Описано основні функції авторської програми «Логічний калькулятор» створеної для підтримки вивчення основ математичної логіки.

На сьогоднішній день процес освіти досить важко уявити без використання комп'ютерних технологій. Проте, на жаль, більшість програм, які при цьому використовуються, являються платними або безкоштовно дозволяють використовувати лише частину своїх можливостей.

Можливість навчатися з використанням комп'ютерів повинна давати людям свободу та розширення прав і можливостей, а не використовуватися приватними компаніями для впровадження своєї монополії через освіту.

Вільне програмне забезпечення дає можливість учням вивчати, як воно працює. Багатьом юнакам цікаво розібратися, як працює комп'ютер та його програми. Саме у юному віці майбутні хороші програмісти мають отримати ці знання. Щоб навчитися добре писати програми, учні повинні вивчити багато текстів програм та написати багато власних. Вони мають читати і розуміти програми, які люди справді використовують. Їм буде, дуже цікаво побачити код програми, із якою вони реально працюють. Вільне програмне забезпечення дає дітям можливість розвиватись, стимулюючи їх досліджувати і вивчати. Натомість власницьке програмне забезпечення ставить перепону до знань словами: «Знання, яке ви хочете – таємниця. Вивчення забороняється!» Вільне програмне забезпечення заохочує кожного вчитися.

Враховуючи ці спостереження нами було вирішено створити ряд абсолютно вільних програм, які б допомогли викладачам у навчанні студентів. Однією з перших програм у цьому циклі для підтримки вивчення основ математичної логіки була створена авторська навчально-тестувальна програма «Логічний калькулятор», яка призначена для розвитку умінь і навичок будування таблиць істинності за заданою формулою алгебри висловлень та організації підсумкового контролю.

Основною функцією програмного продукту «Логічний калькулятор» є побудова таблиць істинності для довільної формули алгебри висловлень від двох до п'яти висловлювальних змінних, та п'яти логічних операцій: заперечення, кон'юнкції, диз'юнкції, імплікації та еквівалентності. Слід зауважити, що програма виводить на екран не тільки остаточний результат обчислення, а й послідовність логічних операцій та вивід проміжних результатів обчислення для кожної з них. Це дозволяє легко організувати перевірку самостійного заповнення користувачем як всієї таблиці істинності так і її проміжних результатів, що дозволяє користувачеві зрозуміти

сам процес побудови таблиці та підказує порядок виконання дій. Для мотивації роботи з програмою була введена система тестування з широкими набором налаштувань (вибір режиму тестування, максимальний час заповнення, вибір максимальної оцінки за тестування), за допомогою якої користувачеві виставляється бали за правильність введення даних.

Програма була створена так, щоб з нею було максимально просто працювати. Так «Логічний калькулятор» має зручний та інтуїтивно зрозумілий інтерфейс. Доступ до кожної функції програми можна здійснити з робочої області. Інтерфейс для введення формул досить схожий на стандарті інтерфейси, які використовуються в калькуляторах і тому не викликає у користувача складностей з введенням формул алгебри висловлень, окрім того програма контролює правильність введення формули і не дозволяє здійснювати синтаксичні помилки. Програмний продукт дозволяє також зчитувати формули збережені в звичайних текстових файлах, та при необхідності дописувати в текстові файли ті формули, які використовуються. З нововведень також можна виділити те, що програма дозволяє працювати у трьох режимах роботи: обчислення, тренування та тестування. Причому кожен режим має окремий набір функцій. Так, наприклад, у режимі тренування чи тестування користувач не зможе автоматично побудувати таблицю істинності. Окрім того в програмному продукті реалізовано захист від подвійного відкриття програми, що унеможлиблює розв'язування тестувальних завдань за допомогою цієї програми.

Отже «Логічний калькулятор» дозволяє:

- будувати таблицю істинності для довільної формули алгебри висловлень від n висловлювальних змінних, де $2 \leq n \leq 5$;
 - виводити проміжні результати обчислень;
 - перевіряти правильність заповнення таблиць істинності як по окремих логічних операціях, так і всю таблицю в цілому;
 - виставляти відповідні бали за правильність заповнення таблиць;
- Характерними рисами цієї програми є:
- простота у роботі з програмою, контроль коректності вхідних даних;
 - зручний та інтуїтивно зрозумілий інтерфейс;
 - можливість роботи у декількох режимах;
 - можливість використання формул збережених у txt-файлах.

Слід зауважити, що дане програмне забезпечення планується оновлюватися, розширювати свій функціонал враховувавши при цьому як побажання викладачів так і студентів. Як бачимо, знання мови програмування значно спрощує навчальний процес для вчителів та викладачів. Адже, непотрібно купувати пакети програмного забезпечення, що значно сприяє суттєвому заощадженні бюджетних коштів та підвищенні мотивації до вивчення предмету. При розробці власного програмного продукту, чи розбираючись в тому як працює існуючий та вносячи в нього свої корективи, можна оптимізувати програму саме під ті цілі, які специфічні для конкретного навчального закладу.

PINGUINO и персональный генератор энтропии

Касьяник В.В.

Брестский государственный технический университет, val.tut@gmail.com

Проблема получения качественных случайных числовых последовательностей в настоящее время решается созданием различных сложных генераторов псевдослучайных чисел, однако с ростом вычислительных мощностей, повышается вероятность взлома даже таких алгоритмов. Создание достаточно простого и дешевого персонального генератора высококачественных случайных последовательностей чисел на основе свободной платформы Pinguino позволит каждому получать случайные числа для защиты своей информации или проведения квантовых исследований. На первый взгляд проблема не так уж остра, однако уже сейчас ощущается недостаток качественных последовательностей, что вызывает их активную продажу и покупку. В данной работе предлагается создание технологии производства качественной энтропии и случайных последовательностей на основе свободного программного и аппаратного обеспечения.

В последнее время широкое развитие и различное применение получили так называемые Arduino-совместимые конструкторы электронных устройств. Это универсальные контроллеры, достаточно простые в использовании и разработке, но отлично подходящие для проектирования сложных электронных устройств, тесно взаимодействующих с окружающей физической средой. Ардуино-совместимые платформы, предназначенные для «physical computing» с открытым программным кодом и открытым аппаратным обеспечением позволяют пользователю соединить виртуальность персонального компьютера и окружающую реальность.

Ардуино-совместимых платформ на сегодняшний день огромное количество. Они варьируются по функциональности, производительности, типу используемого оборудования и т.д. Они находят применение в домашнем моддинге, подходят для встраивания в автомобиль, для проведения научных исследований. Нами был рассмотрен вариант применения одной из таких свободных платформ – Pinguino – для создания простого персонального генератора энтропии (источник энтропии).

Источники энтропии используются для накопления энтропии, с последующим получением из неё начального значения, необходимого генераторам случайных чисел (ГСЧ) для формирования абсолютно случайных чисел. Отличие от генератора псевдослучайных чисел (ГПСЧ) в том, что ГПСЧ использует единственное начальное значение, откуда и получается его псевдослучайность, а ГСЧ всегда формирует случайное число, имея в начале высококачественную случайную величину, предоставленную различными источниками энтропии.

Случайные числа имеют много применений в криптографии, в частности, для создания криптографических ключей, паролей. Например, на некоторых системах на платформе UNIX можно считать данные с устройства /dev/audio без помещённого у микрофона источника звука,

либо считать лишь низкий уровень фонового шума. Такие данные по существу являются случайным шумом, хотя им не стоит доверять без некоторой проверки. Физические процессы являются одними из самых надежных для получения последовательностей случайных чисел, поэтому применение Ардуино-подобной платформы является наиболее простым и доступным для создания источника энтропии.

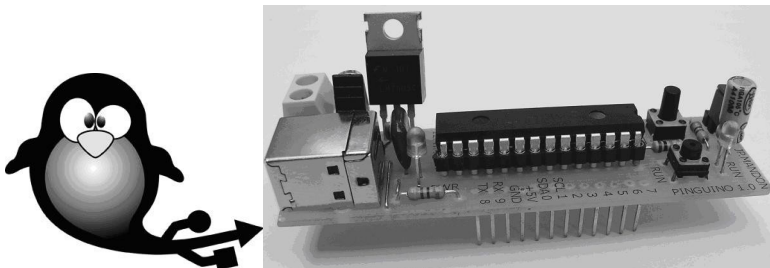


Рис. 1 – Логотип проекта и Pinguino-плата

Pinguino – это ардуино-подобная вычислительная платформа на основе микроконтроллера PIC, которая была создана французским исследователем из лаборатории робототехники Жан-Пьером Мэндоном и группой HackingLab. Целью проекта являлось создание совместимой с ардуино вычислительной платформы, обладающей теми же свойствами кроссплатформенности, простоты, открытости кода и схематических решений. Однако большинство atmel-вариантов ардуино-платформ не имеют нативной поддержки USB-интерфейса, что затрудняет легкое подключение таких устройств к компьютеру и соответственно, не является легким в использовании. Используемый в Pinguino микроконтроллер PIC 2550 поддерживает USB-интерфейс и последовательный интерфейс на уровне чипа, что позволяет уменьшить размеры платы и кода для связи с компьютером.

Для получения самой физической энтропии с помощью Pinguino, к нему необходимо подключить датчик инфракрасного излучения (простейшая схема с излучателем и приемником) и акселерометр. Вибрации дадут возможность акселерометру накапливать случайную величину, причем более дешевый вариант датчика дает более качественный физический шум. Отражение инфракрасного излучения от объектов реального мира сильно зависит от физических свойств среды – оптической проводимости, отражающей поверхности, наличия других источников излучения, например ламп накаливания. Приемник должен обладать возможностью получения аналогового сигнала. Аналогичным образом можно применять достаточно простую схему высокоточного измерения температуры по формуле Найквиста [5].

В заключение стоит отметить, что на базе платформы Ардуино был создан необычный генератор энтропии на основе капиллярного течения жидкости [4]. Однако, предложенный в данной работе вариант отличается простотой и доступностью исходных деталей, малыми размерами платы

Pinguino, комбинированием различных датчиков для получения энтропии.

Со временем ценность качественных генераторов энтропии будет только расти, достаточно вспомнить известный афоризм Роберта Р. Кавью: «генерация случайных чисел слишком важна, чтобы оставлять её на волю случая».

Литература

1. Веб-сайт проекта Pinguino. Режим доступа: <http://www.hackinglab.org/index.html>, 9.01.2011
2. Блог проекта Pinguino. Режим доступа: <http://jpmandon.blogspot.com/>, 9.01.2011
3. Группа рассылки на сайте Google. Режим доступа: <http://groups.google.fr/group/pinguinocard>, 9.01.2011
4. Генератор энтропии на базе Ардуино. Режим доступа: <http://www.circuitlake.com/usb-hourglass-sand-timer.html>, 9.01.2011
5. Высокоточное измерение температуры по формуле Найквиста. Режим доступа: <http://www.shematic.net/page-17.html>, 9.01.2011

Віртуальний лабораторний практикум у процесі вивчення фізики

Качан В.М., Харченко О.В.

Черкаський національний університет імені Богдана Хмельницького (vasilij1987@mail.ru)

У роботі розглянуті проблеми створення та способи застосування віртуального лабораторного практикуму створеного з використанням Java-апплетів та JQuery додатків.

У багатьох країнах світу, і в Україні зокрема, практична реалізація дистанційного навчання починалася з гуманітарних спеціальностей. У той же час спроби впровадити елементи дистанційного навчання у навчальний процес за фізико-математичними та інженерними спеціальностями донедавна викликали немало труднощів, що пов'язані в основному з необхідністю виконання практичних завдань з використанням приладів та обладнання, зокрема, реалізації лабораторного практикуму.

Розвиток інформаційних технологій зумовив появу поняття “віртуальний лабораторний практикум”, в основі якого лежить імітаційне комп'ютерне моделювання. Основні способи використання віртуального лабораторного практикуму в навчальному процесі:

у якості комп'ютерного “тренажера” для підготовки до виконання практикуму в реальній лабораторії (у цьому випадку програми комп'ютерного і фізичного експериментів зазвичай однакові);

у якості доповнення до реального практикуму, що передбачає такі комп'ютерні експерименти, які з різних причин (технічних, організаційних, фінансових, тощо) не можуть бути реалізовані на фізичному обладнанні.

Використання віртуального лабораторного практикуму у якості комп'ютерного “тренажера” надає можливість студентові краще підготуватися до виконання фізичного експерименту, глибше з'ясувати досліджувані ефекти, набути навичок роботи з вимірювальними приладами (у випадку, якщо віртуальний практикум включає комп'ютерні моделі вимірювальних приладів, близькі за своїми властивостями до властивостей реальних приладів). Звичайно такий підхід можна рекомендувати для студентів заочно-дистанційної форми навчання, оскільки він не тільки сприяє кращому засвоєнню матеріалу, що вивчається, але і дозволяє скоротити тривалість виконання практикуму в реальній лабораторії.

У випадку використання віртуального лабораторного практикуму у формі доповнення до реального практикуму, він має орієнтуватися на виконання досліджень підвищеного рівня складності або досліджень, які потребують обладнання, що відсутнє в університеті. Вказаний підхід реалізований в Черкаському національному університеті імені Богдана Хмельницького для студентів фізико-математичних та технічних спеціальностей.

За технологіями створення віртуальних лабораторних практикумів можна вирізнити наступні основні варіанти:

1. Віртуальні лабораторні практикуми з використанням універсальних пакетів програм, що забезпечують можливість виконання широкого спектру практичних завдань. Універсальні пакети містять величезні бібліотеки елементів, які надають можливість створення віртуальних інтерфейсів фізичних приладів і лабораторних установок.

2. Віртуальні лабораторні практикуми на основі спеціалізованих наочно-орієнтованих пакетів програм. Зокрема, пакетів програм створених для моделювання електронних схем, систем, призначених для моделювання і аналізу фізико-хімічних процесів, тощо. Так само як і у попередньому випадку, програмне забезпечення даного класу є універсальним середовищем, призначеним для розв'язання прикладних задач користувача.

3. Віртуальні лабораторні практикуми на основі Java-апплетів та JQuery додатків. На відміну від попередніх варіантів, де користувач (викладач) зазвичай працює в режимі графічного програмування, процес створення Java-апплетів та JQuery додатків є набагато більш трудомістким і потребує програмування коду. Проте, дана технологія має і певні переваги, особливо коли мова йде про віртуальний лабораторний практикум, що буде використовуватись дистанційно через глобальну мережу.

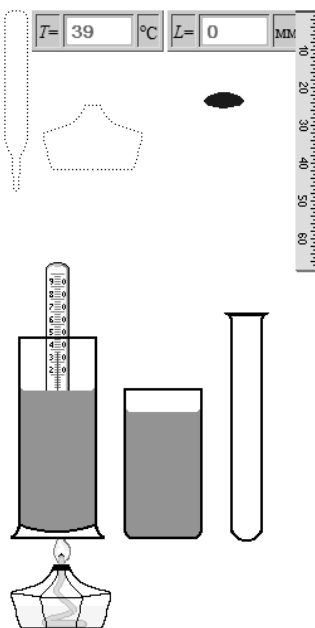


Рис.1. Фрагмент віртуальної лабораторної роботи з фізики (нагрівання рідини)

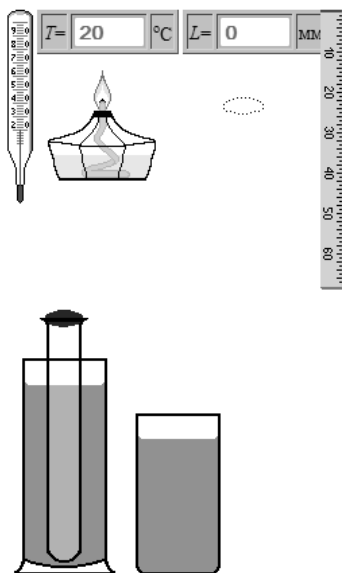


Рис.2. Фрагмент віртуальної лабораторної роботи з фізики (нагрівання повітря в трубці до температури рідини)

Наша робота націлена на створення віртуального лабораторного практикуму з використанням Java-апплетів та JQuery додатків (рис. 1, рис. 2). Основна мета цієї роботи полягає в тому, щоб надати можливість студентам віддалено готуватись до виконання реальних лабораторних робіт. Безумовно, у зв'язку з тим, що на виконання віртуальних лабораторних робіт витрачається значно менше часу порівняно з виконанням реальних, з'являється можливість за короткий сформувані у студентів уміння і навички планувати процедуру виконання експерименту і обробку його результатів, що значно полегшує виконання реального експерименту.

Проблеми популяризації вільного і відкритого програмного забезпечення в Україні

Колос В.В.

Луганський Державний інституту культури і мистецтв, kvvlgua@gmail.com

В доповіді розглядаються проблеми популяризації вільного і відкритого програмного забезпечення на Україні та пропонуються шляхи їх можливого вирішення.

Як показує практика, досить складно, а іноді й неможливо переконати користувача у перевагах використання вільного та відкритого програмного забезпечення (ВВПЗ) і причин цьому кількя.

На сьогоднішній день на Україні у сфері освіти середньої та вищої школи домінує Microsoft. Починаючи зі шкільної лави учнів привчають до таких програмних продуктів як MS Windows і MS Office. У результаті після закінчення навчального закладу учнями суспільство отримує нових прихильників пропріетарних програмних продуктів, головним аргументом яких при виборі програмного забезпечення є знайомий зі шкільних років інтерфейс користувача. Навіть вартість продукту при цьому відходить на другий план.

При цьому суспільство не тільки набуває економічну і технологічну залежність, а й створює передумови щодо її зміцнення.

Популяризації ВВПЗ в сфері освіти перешкоджає саме освітнє середовище. Багато хто вірить, що саме знання конкретних програм, а не розуміння загальних концепцій допоможе їх вихованцям в подальшій роботі. Це в свою чергу створює інформаційний вакуум, одним з результатів якого є незнання альтернатив для заміни звичного програмного продукту і підсвідоме підштовхування суб'єкта суспільством до порушення авторських прав.

Відсутні в достатній кількості підготовлені фахівці, здатні одразу перейти на альтернативні програмні продукти.

Сфера освіти є одним з основних джерел популятизації ВВПЗ, так як саме вона готує майбутні покоління користувачів. Між тим переконати користувача у перевагах ВВПЗ в процесі підготовки дуже важливо, адже це майбутні архітектори, інженери, медики, педагоги і перш за все батьки, які будуть передавати свої знання й досвід власним нащадкам.

Крім вище перерахованих проблем слід зауважити, що сфера освіти дуже тісно пов'язана з іншими сферами людської діяльності. Для більш активного та успішного впровадження ВВПЗ в навчальному процесі необхідні спільні проекти впровадження так само в державних структурах і бізнесі, бо сфера освіти орієнтується на вимоги до випускників на ринку праці.

Вирішення цих проблем - досить довгий процес, практично не здійснений, як показує міжнародна практика, без участі держави [1, 2, 3, 4, 5, 6]. Необхідна програма переходу принаймні бюджетної сфери на ВВПЗ.

Посилення контролю за дотриманням авторських прав, у свою чергу дозволить стимулювати перехід на ВВПЗ в сфері бізнесу, для якої найбільш вагомим показником при прийнятті рішень є фінансовий.

Початок цьому шляху вже покладено. Кабінет міністрів України 23 грудня 2009 року прийняв концепцію переходу державних органів на програмне забезпечення з відкритим кодом. Програма розрахована на реалізацію протягом 4 років по закінченню яких з'явиться можливість перевести на використання ВВПЗ органи державної влади, в майбутньому зберігши при цьому 80 - 87% бюджетних коштів, які витрачаються на придбання та підтримку пропріетарного програмного забезпечення сьогодні [7].

Література

1. Міністерство освіти Франції і Mandriva будуть спільно просувати GNU / Linux і вільне ПЗ http://mandriva.ru/novosti/mandriva_press_02_06_2008/
2. Досвід застосування вільного програмного забезпечення у світі <http://anotherlevel.profi.net.ua/seminar/linux.html>
3. Проект переходу німецького парламенту на вільне ПЗ <http://ru.wikipedia.org/wiki/BundesTux>
4. Перу переходить на вільне ПЗ <http://linux.perm.ru/modules.php?name=News&file=article&sid=143>
5. Японські школи перейдуть на Linux <http://lenta.ru/news/2007/03/08/japanlinux/>
6. Даш вітчизняне ПЗ в державних установах! <http://vz.ru/society/2006/3/30/28079.html>
7. Розпорядження від 23 грудня 2009 р.. N 1588-р <http://zakon1.rada.gov.ua/cgi-bin/laws/main.cgi?nreg=1588-2009-%F0&chk=XX7MfyrCSgky4ELIZiRmAKgzH14vAs80msh8Ie6>

Використання відкритих програмних засобів в процесі навчання статистичним дисциплінам

Коркуна Т.Й.

Самбірський технікум економіки та інформатики t.korkuna@gmail.com

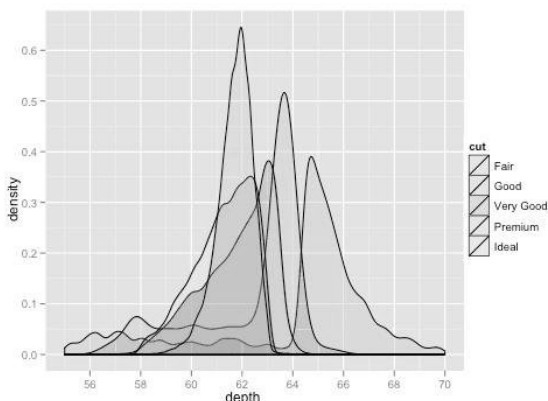
На прикладі мови статистичної обробки R розглянуте відкрите програмне забезпечення для навчання статистичним дисциплінам.

На сьогоднішній день у сфері статистичних та соціологічних досліджень провідне положення займають комерційні професійні програмні пакети SPSS Statistics корпорації IBM [1] та S (Bell Labs) компанії Mathsoft [2].

Аналогом вказаних пакетів є мова R з відповідними додатками, що розширюють графічний інтерфейс користувача. R - це мова програмування для статистичної обробки даних і роботи з графікою, і в той же час - це вільне програмне середовище з відкритим вихідним кодом, що розвивається в рамках проекту GNU[3].

Зручним стандартним графічним інтерфейсом для мови програмування R є програма RKWard. Розробники RKWard намагалися поєднати міць R з простотою використання, подібно до комерційних статистичних пакетів. Програма використовує для роботи бібліотеку від середовища KDE. Автори RKWard взяли курс на якомога більш повну інтеграцію функцій R в графічне середовище: присутні переглядач поточного оточення (environment), редактор даних, менеджер пакетів, що вміє не тільки встановлювати їх, але і стежити за оновленнями, довідкова система. Існує також чимало кросплатформних графічних інтерфейсів для R, наприклад, Jgr на умова ліцензії GNU.

Подібність інтерфейсів користувача та технології статистичних операцій комерційних і вільних пакетів статистичного аналізу дозволяє вибрати для навчання статистичним дисциплінам в якості інструментарію мову R з відповідними надбудовами та додатками. Такий підхід надає можливість зняти всі застереження комерційних виробників, вільно користуватися про-



грамним забезпеченням, виконувати коди з комерційних пакетів на мові S, охопити в процесі навчання як комерційні, так і вільні пакети програм.

В цьому році мова R увійшла до списку переможців конкурсу кращого відкритого програмного забезпечення року «InfoWorld Bossie Awards 2010: The best open source application development software».

Вивчення курсу економічної статистики та галузевих статистик, особливо в умовах дистанційної освіти, припускає застосування графічного інструментарію для наочного відображення результатів статистичних виборок та розрахунків [4].

Для вирішення цих питань для мови R створений графічний пакет Ggplot2, який реалізує граматику графіків, подібну комерційному пакету GGRAPH в SPSS Statistics [5]. За допомогою цього пакету можна створювати всі види графічних статистичних презентацій: стовпчикові та кругові, дво- та тривимірні діаграми, графіки тощо, навіть будувати дещо “екзотичні” залежності, показані на прикладі.

Література

1. Raynald Levesque and SPSS Inc. Programming and Data Management for IBM SPSS Statistics 19. A Guide for IBM SPSS Statistics and SAS Users. Copyright SPSS Inc. 1989, 2010. – 458 pp.
2. Vanables W.N., Ripley B.D. Modern Applied Statistics with S. Fourth Edition. Springer Science+Businnes Media, New York, 2002 – 498 pp.
3. John M. Chambers. Software for Data Analysis. Programming with R. Springer Science+Businnes Media, New York, 2008 – 514 pp.
4. Коркуна Т.Й. Web-інструментарій графічного представлення результатів автоматизованого оброблення статистичної інформації // Новітні комп'ютерні технології. Матеріали VIII Міжнародної науково-технічної конференції. - Київ - Севастополь - с. 153 — 155.
5. Paul Murrell. Drawing Diagrams with R. The R Journal, 1(1):15-21, May 2009, p.p. 15-21.
6. Christopher R. Bilder, Boan Zhang, Frank Schaarschmidt, and Joshua M. Tebbs. binGroup: A Package for Group Testing. The R Journal, 2(2):56-60, December 2010, p.p. 56-60.

Построение практикумов по программированию периферийных устройств и архитектуре ЭВМ на базе GNU/LINUX

Костюк Д.А., Жук А.М.

Брестский государственный технический университет, dmitrykostiuk@gmail.com

Рассматриваются возможности и особенности применения ОС Linux для изучения архитектуры, взаимодействия с периферийными устройствами и системного программирования на уровне узлов ЭВМ. Оцениваются подходы к решению данной задачи на разных уровнях абстракции, а также практические вопросы проведения соответствующих лабораторных практикумов.

Исторически практикумы по архитектуре вычислительных систем и их сопряжению с периферийным оборудованием для студентов-программистов строятся на базе 16-битной платформы Intel и ОС DOS. Переход к более современным 32- и 64-битным платформам осложнен рядом дополнительных факторов, в первую очередь повышенным уровнем абстракции от оборудования и изоляцией прикладных процессов [1]. Поскольку прикладной процесс не имеет полного доступа к аппаратным ресурсам, становится проблематичным писать простые учебные программы в стиле DOS, свободно взаимодействующие с устройствами.

GNU/Linux позволяет частично решить проблему доступа к системным ресурсам эскалацией привилегий прикладной программы для доступа к специальным механизмам, встроенным в ОС; однако для полноценного практического ознакомления с аппаратной архитектурой студенту необходимо научиться создавать драйвера устройств в виде модулей ядра. В последнем случае частью учебного процесса становится исследование исходного кода стандартных драйверов, позволяющее лучше понять архитектуру как программной, так и аппаратной подсистемы, и выгодное с точки зрения приобретения востребованной на рынке квалификации.

Взаимодействие с аппаратными узлами из пространства прикладных программ при наличии привилегий оказывается даже более простой задачей, чем в DOS. Написание драйверов для ядра Linux — также одна из наиболее простых задач в сравнении с разработкой драйверов для других современных ОС, как в отношении требуемой квалификации, так и по наличию в свободном доступе и распространенности необходимых пособий.

Традиционно при изучении архитектуры ЭВМ на базе 16-битной архитектуры программирование устройств может выполняться двумя способами: более простым, с использованием прерываний (системных вызовов) BIOS, и более сложным, через порты ввода/вывода соответствующих микроконтроллеров.

Доступ к функциям BIOS в GNU/Linux является наиболее спорным. Согласно общедоступной информации, ядро не использует их в своей работе; однако предоставляемые BIOS обработчики прерываний техниче-

ски доступны и могут использоваться при написании модулей ядра — по крайней мере, наименее разрушительные из них. Например, можно относительно безопасно использовать функции BIOS, читающие системные данные из области CMOS, или взаимодействующие с контроллером клавиатуры. Однако нецелесообразно уделять данным методам существенного внимания в курсе, поскольку либо их практическая польза ограничена, либо они небезопасны на действующей системе (как в случае функций доступа к дискам, используемых в первичном загрузчике ОС).

Альтернатива — доступ к устройствам через порты ввода/вывода — в ряде случаев оказывается не намного сложнее (как упоминавшееся выше взаимодействие с контроллером клавиатуры) и может относительно безопасно выполняться на двух уровнях абстракции:

- файл виртуального устройства `/dev/port` отображает пространство портов и при наличии необходимых прав доступа позволяет пользовательскому процессу выполнять чтение и запись в порты средствами файлового ввода/вывода (`open`, `close`, `read`, `write`);

- доступ к портам на уровне модуля ядра может быть выполнен классическими ассемблерными инструкциями `in` и `out`, непосредственно из ассемблерного кода модуля ядра либо из ассемблерной вставки в тексте программы на C.

Для ряда устройств, обычно не изучаемых в традиционных 16-битных практикумах, разумно воспользоваться более высокой абстракцией, предоставляемой виртуальными файловыми системами `/dev/` и `/proc/`. Это дает оправданную экономию, позволяя хотя бы попробовать работать с теми объектами, для которых в противном случае в рамках практикума не нашлось бы учебных часов. В простейшем случае, доступ к файловой системе `/proc/` позволяет ознакомиться с более полной конфигурацией оборудования (в сравнении с весьма ограниченными возможностями, предоставляемыми традиционным для классического курса анализом области CMOS). В качестве более сложного примера можно упомянуть файловый доступ к устройствам `/dev/dsp` и `/dev/mixer` (выполняемый все теми же системными вызовами `read`, `write`, а также `ioctl`), который при всей простоте реализации позволяет изучить принципы работы со звуковой картой и одновременно получить навыки разработки программ, анализирующих сигналы, принимаемые аудио-трактом. Аналогична ситуация с изучением взаимодействия по шине USB.

Точкой схождения двух уровней абстракции может быть написание модуля ядра, связанного с созданием собственного файла устройства в каталоге `/dev/`. Подобная работа проясняет, как осуществляется резервирование конкретных аппаратных ресурсов (портов, прерываний) и связь действий над ними с соответствующими операциями на уровне файловой абстракции.

Конкретный перечень задействованных в курсе системных и периферийных устройств зависит от выделенных часов и от предшествующей подготовки студентов в области системного программирования под Linux. Необходимо как минимум знание студентами особенностей работы в

консоли GNU/Linux, владение языком С и хотя бы основами ассемблера.

При организации практикума требуется учитывать и тот факт, что необходимая эскалация привилегий накладывает дополнительные ограничения на рабочую среду, в которой должны выполняться практические задания. Для запуска модулей ядра или доступа к узлам вычислительной системы из пространства пользователя студенты нуждаются в правах администратора, т.е. получают возможность легко нарушить целостность и работоспособность ОС. Проведенные эксперименты показывают [2], что решения на базе виртуализации для платформы x86 либо не обеспечивают необходимую точность эмуляции устройств в нестандартных режимах их использования (QEMU, VirtualBox), либо обладают недостаточными для комфортной работы производительностью и функционалом (эксперимент показывает точную работу эмулятора Vochs при цене 20-кратного снижения производительности по сравнению с нативным выполнением кода). Поэтому целесообразной альтернативой виртуализованным окружениям представляется использование специализированного LiveCD-подобного дистрибутива.

Литература

1. Костюк Д., Жук А. Перевод обучения программированию на ассемблере на платформу GNU/Linux // Журнал Root@UA. №1, 2009. Киев, Украина. - С. 39-41.
2. Применение платформы GNU/Linux для изучения архитектуры ЭВМ. //Современные проблемы радиотехники и телекоммуникаций «РТ — 2010». Материалы 6-ой международной молодежной научно-технической конференции. 19 - 24 апреля 2010 г. - Севастополь, 2010. - С. 511

Вибір платформи дистанційного навчання

Коцаренко М.В., Бойко О.В.

*Львівський національний медичний університет ім. Данила Галицького
taxymts777@gmail.com*

Розглянуто вимоги, які висуваються до платформ дистанційного навчання, та основні особливості платформи Claroline.

Одним з важливіших напрямків у розвитку сучасної людини є дистанційна освіта, яка відкриває шлях до реалізації своїх можливостей та сприяє більш чіткому та координованому розв'язанню як повсякденних, так і фахових завдань.

Однією з основних структурних ланок безперервної системи освіти є вища школа, що зумовлює у державній політиці пріоритетність забезпечення можливостей підвищення якості підготовки студентів та постійного зростання професійного рівня викладачів вищих навчальних закладів.

Саме цим вимогам відповідає система дистанційного навчання, специфіка якої полягає у наявності доступного і якісного програмного забезпечення, що надає можливість постійного, легкодоступного спілкування слухачів (студентів) з викладачем.

Правильний вибір системи комп'ютерного навчання та програмного забезпечення у відповідності до специфіки та потреб навчального закладу є однією з основних передумов успішного результату при використанні системи дистанційного навчання.

При виборі програмного забезпечення необхідно враховувати багато факторів, насамперед розділення вмісту структури сайту та навчального курсу, а також можливість збереження важливих даних структури меню при оновленні контенту; наявність систем допомоги, контролю і безпечного способу передачі даних доступними каналами зв'язку з одночасним забезпеченням захисту від перехоплення інформації третіми особами.

З метою уникнення постійної залежності від розробника системи, при її виборі необхідно звернути увагу на сумісність з іншими e-learning рішеннями, чи хоча б з одним широко поширеним стандартом.

Ще однією необхідною вимогою є зручність у використанні, можливість легко знайти меню допомоги та швидко перейти від одного розділу дисципліни до іншого.

Прикладом такого програмного забезпечення можна назвати платформу Claroline. В порівнянні з іншими платформами для електронного навчання можна відзначити простоту в використанні, відсутність масивності інтерфейсу та зручність в експлуатації. Claroline здатна легко обробляти інформацію для великої кількості (до 20 000) користувачів. Вона сумісна з Linux, Mac і Windows. Claroline заснована на вільних технологіях, таких як PHP і MySQL, і використовує поточні стандарти SCORM і IMS / QTI для обміну вмістом.

Дана платформа повністю підтримується операційним середовищем

Linux і успішно використовується у Львівському національному медичному університеті ім. Данила Галицького для проведення комп'ютерного контролю знань студентів.

Використовуючи досвід впровадження в дистанційне навчання безкоштовного програмного забезпечення Clargline, можна рекомендувати для використання у вищих навчальних закладах.

Інституційний репозитарій університету як середовище інтеграції наукового знання

Кузьмінська О.Г.

*Національний університет біоресурсів та природокористування (НУБіП) України,
м. Київ, e-mail: Olena_K@bk.ru*

Матеріали доповіді присвячені питанням проектування та створення інституційного репозитарію в середовищі Eprints – мережного сервісу відкритого доступу для зберігання, систематизації, управління та поширення цифрових колекцій інтелектуальної продукції університету, а також поширення досвіду їх застосування та перспектив використання для вирішення завдань наукового, освітнього і прикладного характеру.

Університети й дослідницькі інститути в усьому світі активно планують і реалізують архіви власної наукової продукції. В умовах інформаційного суспільства інституційні репозитарії (ІР) мають значний потенціал як реальні показники якості університетів, наукової, соціальної та економічної значущості дослідницьких робіт і, таким чином, вияв статусу та суспільного значення університетів.

За даними Digital Repository Infrastructure Vision for European Research (DRIVER), з 27 країн Євросоюзу сім не мають створених репозитаріїв, в п'яти країнах – ІР на ранніх стадіях розробки, 15 країн Європи мають різного рівня ІР. Безумовними лідерами щодо впровадження ІР є Великобританія, Німеччина та Нідерланди. Все більше українських університетів стають центрами серйозних наукових досліджень і визначають себе як науково-освітні інституції, де у якості одного з пріоритетних напрямів визначено створення та розповсюдження власних ресурсів університетських громад України через об'єднану систему університетських репозитаріїв та відкритих електронних журналів з метою інтеграції у світову академічну спільноту.

Правильний добір програмного забезпечення (ПЗ), яке б максимально повно задовольнило потреби навчальних і наукових організацій, при мінімумі прямих і непрямих витрат – один з аспектів рішення завдання розбудови інституційного репозитарію університету. Аналіз структури та ступеня використання ІР, в тому числі 13 українських, свідчить, що найбільш використовуваним ПЗ для побудови інституційних репозитаріїв є ПЗ Dspace та Eprints. Останнє, за даними OpenDOAR (каталог репозитаріїв відкритого доступу), відповідає і сучасним європейським тенденціям.

Аналіз можливостей та прикладів використання в бібліотеках різних навчальних закладів зумовив вибір ПЗ Eprints для створення архіву (репозитарію) фондів наукової продукції НУБіП України та адміністрування доступу до нього. Підтвердженням правильності добору програмного забезпечення є те, що Eprints вільно поширюється під ліцензією GNU General Public License, реєстрація провайдера даних в OAI також безкоштовна.

Визначення навчально-наукових потреб студентів та науково-педагогі-

чних працівників університету щодо створення, зберігання та поширення наукової продукції зумовило розробку моделі інституційного репозитарію та розширення стандартного рубрикатора системи та створення системи додаткових рубрикаторів локальної версії відкритого архіву як дослідного зразка репозитарія.

При створенні дослідного зразка відповідно до розробленої структури репозитарію було визначено типи ресурсів (колекції магістерських робіт, електронні аналоги (копії) друкованих видань навчально-методичного призначення (навчально-методичні матеріали, робочі зошити тощо) та навчально-методичних комплексів дисциплін, розроблених викладачами та співробітниками університету для організації навчального процесу) та рівні доступу (обмеженого доступу - до матеріалів користувацького фонду, доступних через локальну мережу університету, а саме описів та презентацій магістерських робіт та навчально-методичних матеріалів, та закритого (депозитарій) – матеріали доступні лише для зареєстрованих користувачів репозитарію: повнотекстові бази магістерських робіт НУБіП України.

Для опанування студентами та науково-педагогічними працівниками університету навичок опису, депонування та використання матеріалів репозитарію необхідне системне поєднання нормативно-методичного забезпечення та інструктивної бази. Для забезпечення ефективного функціонування такої системи було розроблено настанови користувача та депонента, а також розроблено та проведено цикл навчально-методичних семінарів. Забезпечення збереження авторських прав власників наукової продукції здійснюється згідно розробленого Положення про інституційний репозитарій магістерських робіт.

Створення та використання дослідного зразка інституційного репозитарію надає широкі можливості для управління значними обсягами даних та їх опрацювання, а також сприяє підвищенню якості наукової діяльності університету.

Відкриті проекти в області авіоніки

Лаврущенко О.М.

aleksandr.lavrushchenko@gmail.com

Рішення на основі відкритої моделі розробки проникають в усі сфери людської діяльності. Причому, в першу чергу в технічній області, у зв'язку з посиленням динамічності розвитку технологій та спрощенню доступу до інформації, завдяки вибуховому розвитку комунікацій.

Проекти з відкритою моделлю розробки зараз можна зустріти в будь-якій галузі з технічним спрямуванням. у зв'язку із збільшеною динамічністю посиленням динамічності розвитку технологій та спрощенню доступу до інформації, завдяки вибуховому розвитку комунікацій.

Не є винятком і авіоніка. У доповіді описано стан відкритих проектів в області авіоніки. Необхідно відразу обмовитися, що необхідно чітко відокремлювати відкриті проекти та проекти з відкритими проектними даними поширюваними під ліцензіями на зразок GPL, CC, і т.д.

У авіамоделної практиці основними використовуваними апаратами є літаки, вертольоти і мультиротори. Останнім часом найбільш активно розвивається сегмент мультироторних апаратів через їх технологічну простоту, стабільність, порівняно невелику вартість, доволі велику вантажопідйомність і природньо — інерційність спільноти користувачів. Адже чим більше моделістів використовують мультиротори, тим більше є щодо них інформації і тим нижчий поріг входження для новачків.

Саме розробкам апаратів даного типу приділено основну увагу в доповіді. Відзначено його можливості і слабкі сторони.

У доповіді також висвітлено історію розвитку відкритих проектів в області авіоніки. Описані найбільш значущі зарубіжні проекти і дано короткий огляд стану даної галузі в Україні та Росії.

Коротко описана базова методологія проектування мультироторних апаратів, дано порівняльні характеристики найбільш популярних моделей. Крім того описаний власний проект кваддротора що знаходиться в стадії розробки.

Вільнопоширювані системи комп'ютерної математики в освіті та науці

к.ф.-м.н. Лазурчак І.І., к.пед.н. Кобильник Т.П.

*Дрогобицький державний педагогічний університет імені Івана Франка
lazurchak@mail.ru; taras2408@mail.ru*

Пропонується вивчення вільного математичного програмного забезпечення у вигляді пакету Maxima, що розповсюджується під ліцензією GPL і є доступною як користувачам операційних систем Linux, так і користувачам Windows

Оснащення закладів середньої та вищої освіти сучасною комп'ютерною технікою та відповідним прикладним програмним забезпеченням є безумовно позитивним явищем. Проте для ефективного використання комп'ютерної техніки в процесі навчання не тільки інформатики, а й інших дисциплін, зокрема фізико-математичного циклу, цього недостатньо. Існує досить багато досліджень з впровадження та застосування інформаційних технологій, зокрема систем комп'ютерної математики (СКМ), в навчальний процес. Однією з перешкод на шляху ефективного використання СКМ є недостатній обсяг знань з природничих дисциплін, практичних умінь та навичок роботи студентів з математичними пакетами. Усунення цієї перепони є однією з цілей вивчення дисципліни „Системи комп'ютерної математики” для студентів фізико-математичних та інженерних факультетів педагогічного університету.

Використання СКМ надає можливість виокремити два напрямки:

1. застосування СКМ при вивченні дисциплін фізико-математичного циклу та професійна значущість СКМ для вчителів математики, фізики та інших фундаментальних дисциплін, включаючи галузь знань – системні науки та кібернетика;
2. використання СКМ при вивченні дисциплін комп'ютерних напрямів для підготовки студентів інженерних галузей наук.

Пропонується за основу навчання вибрати СКМ Maxima. За походженням Maxima належить до однієї з перших СКМ. У системі Maxima прийнятий такий же принцип нумерації версій, як і в операційній системі Linux: номер складається з трьох чисел, розділених крапками, причому номери з непарним середнім числом відповідають так званим development-версіям (в розробці), з парним – stable (стабільним). Стабільність однієї вітки і статус „в розробці” іншої тут означає не стільки стабільність чи нестабільність роботи програми, скільки стабілізацію самого процесу розробки: у development-гілці нова молодша версія може мати нові функції та нові інтерфейси, у стабільній же молодші версії будуть містити тільки виправлення помилок.

Вибір системи Maxima за основу навчання обумовлений тим, що:

- система Maxima є вільнопоширюваною;
- система надзвичайно легка для опанування (Maxima оснащена

системою меню, є україномовний інтерфейс);

- СКМ Maxima є дуже зручною для виконання символічних перетворень.

Система Maxima серед математичних пакетів володіє досить широкими можливостями при виконанні символічних обчислень. Це, по суті, єдина з вільнопоширюваних відкритих систем, яка не поступається комерційним СКМ Mathematica та Maple. Система Maxima розповсюджується під ліцензією GPL і є доступною як користувачам операційних систем Linux, так і користувачам Windows.

За походженням Maxima належить до однієї з найперших СКМ. Розвиток системи Maxima бере свій початок з 60-х років ХХст., коли з'явилася програма з назвою Macsyma, де реалізовувались всі найновіші (на той час) технології в галузі комп'ютерної математики. Пізніше ці ідеї були покладені в основу лідерів ринку математичних систем – Mathematica та Maple.

Проект Macsyma був заснований Енергетичним Управлінням США (Department of Energy, DOE). Створювали його в Масачусетському Технологічному Інституті (Massachusetts Institute of Technology, MIT) на основі мови Lisp, яка вважалася на той момент найбільш придатною для створення систем символічних обчислень.

Спочатку система Macsyma була закритим комерційним проектом. Вільний доступ до проекту став можливим завдяки професору Вільяму Шелтеру (William Schelter), який домігся від DOE отримання коду Macsyma та його публікації під ліцензією GPL з назвою Maxima.

Мінімум, що потрібно для того, щоб почати роботу зі системою Maxima в будь-якому розповсюджуваному Linux-дистрибутиві, це пакет maxima. Цей пакет містить насправді мінімум: консольну версію програми з необхідними бібліотеками та кілька демо-файлів. Консольна версія забезпечує доволі бідні візуальні можливості: всі математичні формули будуються звичайними текстовими символами в кілька рядків дисплею, а зображення графіків відображаються в окремому вікні (причому продовження роботи можливе тільки після його закриття). Проте за рахунок цього різко зменшуються вимоги до технічних характеристик комп'ютера – система Maxima в консольному варіанті здатна працювати навіть на таких комп'ютерах, які сьогодні й за комп'ютери ніхто не вважає. Для системи Maxima розроблено кілька графічних інтерфейсів: xmaxima, emaxima, imaxima та інші. Робота в будь-якому з цих інтерфейсів системи Maxima відбувається в діалоговому режимі.

Після вивчення студентами системи Maxima студентам пропонується ознайомитися з математичним середовищем SAGE [1] – вільним програмним забезпеченням для алгебраїчних, тригонометричних та геометричних перетворень, за допомогою яких можна розв'язувати задачі з алгебри, диференціального та інтегрального числення, теорії чисел, комбінаторики, аналітичної та диференціальної геометрії, теорії графів, математичного моделювання, методів обчислень, теорії кодування, паралельних та розподілених систем тощо.

Вивчення СКМ на інформатичних спеціальностях у педагогічному уні-

верситеті доцільно починати не раніше, ніж на другому курсі навчання, коли студенти вже вивчили елементи дискретної математики, математичного аналізу, лінійної алгебри та аналітичної геометрії, а також прослухали курс „Алгоритмізація” і знайомі хоча б з однією мовою програмування.

Пропонований курс „Системи комп’ютерної математики” призначений для студентів інформатичних спеціальностей. По-перше, коло вибраних ними інтересів передбачає використання комп’ютера як предмету, так і засобу навчання. Успіх в майбутній професійній діяльності залежить від того, наскільки володіють вони знаннями, вміннями та навичками роботи за комп’ютером, наскільки вони здатні оволодіти новими програмними засобами. Систематичне вивчення інформаційних технологій, зокрема систем комп’ютерної математики, сприяє формуванню у студентів ставлення до комп’ютера і як до засобу розв’язування професійних задач.

По-друге, у студентів відзначається підвищений інтерес до таких інформаційних технологій як системи комп’ютерної математики. Такі студенти отримують більш глибокі знання не тільки з математичних дисциплін, але й з інформатики. Як правило, у них нема психологічного бар’єру перед використанням складних програмних засобів. Навпаки, їх притягують створені на високому професійному рівні програми, і вони помічають унікальні можливості використання таких систем.

Даний курс, призначений для вивчення СКМ, буде сприяти:

– розширенню та поглибленню знань студентів як з інформатики, так і з математичних дисциплін;

– оволодінню студентами уміннями розв’язувати задачі різноманітного характеру за допомогою систем комп’ютерної математики;

Мета курсу: вивчення систем комп’ютерної математики та можливостей застосування математичних пакетів до розв’язування практичних задач, що передбачає:

– розуміння проблематики дисципліни та можливість використання СКМ для доведення тверджень теоретичного (аналітичного) характеру;

– розуміння методології розробки алгоритму від математичної ідеї до формулювання алгоритму та вміння застосувати цю методологію;

– вміння здійснювати обґрунтування та оцінку складності алгоритму за часом виконання і необхідної пам’яті.

Література

Корольський В.В. Інноваційні інформаційно-комунікаційні технології навчання математики: Навчально-методичний посібник / Корольський В.В., Крамаренко Т.Г., Семеріков С.О., Шокалюк С.В; за ред. М.І. Жалдака. – Кривий Ріг: Видавничий дім, 2009. – 334 с.

Створення спеціалізованих рішень на базі продуктів компанії D-LINK

Лецинський О.О.

D-Link International PTE Ltd. oleschinsky@dlink.ua

ВПЗ стало тим трапліном, завдяки якому виробники апаратних засобів змогли досягти масовості та знизити вартість розробки. Спочатку не всі розуміли сутність ВПЗ, тому по-перше результати розробки спрямовувались лише в один бік, по-друге такі здобутки не сприймалися як належить. Через деякий час непорозуміння були вирішені, але все ще існувала деяка ізольованість груп розробників у великих компаніях та незалежних розробників ВПЗ. Останні роки всі учасники цього процесу зрозуміли, що без активного обміну знаннями та здобутками в рамках розробки ВПЗ втрачають всі. В рамках цього заходу компанія D-Link знайомить з перспективами побудови рішень не лише розробників embedded-сегменту, але й для інших категорій.

Використання комп'ютерних математичних систем у професійній підготовці майбутнього вчителя математики

І.В. Лов'янова

*м. Кривий Ріг, Криворізький державний педагогічний університет
lira7-1-8@mail.ru*

Комп'ютерні математичні системи є ідеальним засобом, який дає змогу розширити математичну практику. Сьогодні найбільшої популярності набули Derive, Mathcad, Matlab, Maple, Mathematica, Maxima. Актуальним є питання створення і накопичення банку комп'ютерно-орієнтованих задач для нормативних курсів з усіх математичних дисциплін. У даній роботі розглядаються психолого-педагогічні аспекти проблеми використання комп'ютерних математичних систем; використання системи комп'ютерної математики Maxima з метою автоматизації процесу обчислення інтегралів під час розв'язування звичайних диференціальних рівнянь першого порядку; використання програми Mathematica для візуалізації отриманих результатів розв'язування задач математичної фізики.

Мета розвитку особистості студента потребує специфічного підходу до відбору змісту освіти. Сьогодні системи знань, умінь, навичок недостатньо для будови змісту навчання математичних дисциплін у вузі. У цьому процесі поруч із засвоєнням інформації, фактів має бути присутній пошук, процес формування знань, правил, алгоритмів, формул, тощо. Комп'ютерні математичні системи (КМС) є ідеальним засобом щодо надання умов для такого пошукового процесу, оскільки дають змогу розширити математичну практику.

Сьогодні є досить багато КМС, серед яких найбільшої популярності набули Derive, Mathcad, Matlab, Maple, Mathematica, Maxima.

Слід зазначити, що однією з перших КМС була Macsyma (потім Maxima). Maxima серед КМС відзначається досить широкими можливостями її використання при виконанні символьних обчислень.

Розвиток Maxima бере свій початок з 60-х років ХХст., коли появився продукт з назвою Macsyma, де реалізовувались всі найновіші (на той час) технології в галузі комп'ютерної математики. Пізніше ці ідеї були покладені в основу лідерів ринку математичних систем – Mathematica та Maple.

Сильні сторони Maxima – розвинений апарат лінійної алгебри та диференціальних рівнянь. Система орієнтована на прикладні розрахунки і не призначена для теоретичних досліджень в галузі математики. У зв'язку з цим у програмі відсутні або зменшені розділи, присвячені теоретичним методам (теорія чисел, теорія груп, математична логіка і т.п.). Головною перевагою Maxima перед іншими універсальними системами є те, що користувач має змогу аналітично та чисельно розв'язувати велику кількість різних типів рівнянь у частинних похідних. Maxima має кілька видів графічних інтерфейсів (xmaxima, emaxima, imaxima та ін.). За допомогою системи Maxima можна генерувати коди описів програм мовами Fortran та C, включаючи оператори управління, підпрограми

subroutine та function.

Сучасний розвиток комп'ютерних технологій, орієнтованих на створення інтегрованих пакетів multimedia-технологій, привів до появи нового рівня математичних систем, серед яких найвідомішими є пакети Maple фірми Maple Waterloo Inc та Mathematica фірми Wolfram Research Inc.

Maple – універсальний математичний пакет, в якому передбачено засоби для аналітичних перетворень, чисельних методів розв'язування задач, комп'ютерної графіки.

Maple, Mathematica – це системи комп'ютерної математики, орієнтовані на виконання аналітичних обчислень на будь-якому рівні. Досить розповсюджені у практиці виконання наукових досліджень. Ці КМС є одними з найпотужніших систем, орієнтованих на символічні обчислення.

Розв'язання навчальних задач є специфічним видом навчальної діяльності, який успішно застосовується у методиці усіх вузівських математичних дисциплін. Сьогодні, коли з'явилась можливість використання ЕОМ і сучасного програмного забезпечення для комп'ютерного супроводу всіх видів занять у вузі, актуальним є питання створення і накопичення банку комп'ютерно-орієнтованих задач для нормативних курсів з усіх математичних дисциплін. Наведемо приклад, так студентами усіх спеціальностей фізико-математичного факультету педагогічного університету вивчається дисципліна «Диференціальні рівняння», значне місце у розв'язуванні диференціальних рівнянь займає процес інтегрування функцій, оскільки фундаментальні методи інтегрування вивчалися студентами у курсі математичного аналізу, ми вважаємо за можливе, використовуючи КМС Maxima, автоматизувати процес обчислення інтегралів під час розв'язування звичайних диференціальних рівнянь першого порядку.

Розглянемо психолого-педагогічні аспекти проблеми використання КМС. З одного боку, описаний вище підхід дає можливість більше уваги приділити суто диференціальним рівнянням, а не витратити час на міркування студентів щодо інтегрування функцій, яке для більшості студентів є складним процесом і відповідно знижує їх інтерес до вивчення дисципліни. З іншого боку, використання КМС на заняттях з математичних дисциплін, зокрема «Диференціальні рівняння», демонструє міжпредметні зв'язки, сприяє вмотивованому вивченню фундаментальних дисциплін, забезпечує діяльнісний та особистісно-орієнтований підходи до процесу навчання.

Значне місце займає вміння використовувати КМС і у підготовці майбутнього вчителя. Вчитель математики, передусім, має бути кваліфікованим спеціалістом у своїй предметній області, тобто повинні уміти формулювати цілі навчання, ставити задачі, розробляти алгоритми розв'язування цих задач і ефективно використовувати нові інформаційні технології (НІТ) у своїй професійній діяльності. Підготовка майбутнього вчителя до використання НІТ має здійснюватися не тільки на заняттях з дисциплін методичного циклу, а в першу чергу шляхом використання КМС на заняттях з фундаментальних дисциплін.

В силу абстрактності більшості понять, які вивчаються в алгебрі,

математичному аналізу інших дисциплінах слід відмітити можливості КМС у процесі унаочнення отриманих даних. Так, у процесі формування умінь розв'язувати задачі математичної фізики доводиться інтегрувати знання з різних розділів фізики, математичного аналізу, диференціальних рівнянь. Як показує досвід, студенти доволі часто досить формально сприймають задачі такого класу, з одного боку, це пов'язано з необхідністю використання складного математичного апарату, з іншого, нерозумінням того, яке практичне застосування мають ці знання. Тому, на нашу думку, при розв'язуванні задач, що описують динамічні процеси, не обмежуватись отриманням аналітичного розв'язку, а використовувати комп'ютерні технології для візуалізації отриманих результатів.

Найбільш ефективно за допомогою комп'ютера можна реалізувати принцип наочності під час розв'язування задач математичної фізики з використанням програми Mathematica, що дозволить змодельовати фізичні процеси, які важко представити уявно. Використовуючи цей програмний засіб студенти мають можливість проводити візуалізацію отриманих результатів, усвідомлювати отримані результати, аналізувати та досліджувати їх, що, в свою чергу, поліпшує якість знань студентів з дисципліни.

Актуальність досліджуваних питань, достатньо великі можливості КМС у вивченні математичних дисциплін визначають напрямки подальших досліджень у цій області, зокрема навчання математичним дисциплінам з використанням КМС потребує залучення таких засобів навчання, як: комп'ютерні навчальні програми, комп'ютеризовані підручники і задачі, електронні підручники, розробка психолого-педагогічного і методичного супроводження використання КМС у навчанні студентів вищих закладів світу.

Вільне ПЗ для підготовки наукових текстів та презентацій

Лутфулін М.В., Моторний М.І.

*Полтавський національний педагогічний університет ім. В.Г. Короленка
M.Lutfullin@i.ua*

Обґрунтовується оптимальність вибору LaTeX'a як основного програмного пакету для підготовки наукових публікацій різного типу та презентацій для лекцій і доповідей науковцями математичних та інших спеціальностей.

На складність і важливість безпомилкового друку наукових та навчальних математичних текстів вказував ще у 1945 р. О.М. Крилов [1]. Зокрема він рекомендував авторам притримуватись такої позиції: "Я пишу ту коректуру, в якій більше помилок набору немає, нехай вона буде хоч десята...". Також Крилов вказував на недопустимість "економії паперу" при наборі дрібними і нечіткими літерами, що приводить до напруження зору читача і погіршення сприйняття ним матеріалу.

Сучасна техніка видавництва виключила деякі етапи підготовки публікації, зокрема набір друкарських форм металевими літерами, але залишаються незмінними вимоги відсутності щонайменших помилок у формулах та грамотної верстки. З іншого боку комп'ютерна поліграфія дала можливість займатись видавничою діяльністю людям, що не мають спеціальної підготовки в цій сфері, трапляються видавці, що довіряють редагування і верстку непрофесіоналам та не надсилають авторам навіть однієї коректури. Сумні результати можна бачити на полицях книгарень.

Відсутність помилок у більшості випадків можна забезпечити тим, що комп'ютерний "рукопис" роботи набирає найбільш компетентна в ній людина — її автор. Подальше оформлення друкованого видання називається версткою і передбачає виконання композиційних, гігієнічних і стилістичних вимог, які в цілому визначають зручність користування виданням. Програмне забезпечення класу WYSIWYG вимагає високої кваліфікації верстальника і значного обсягу його роботи. Навряд чи можна вимагати від автора уміння виконувати її.

Метою даної публікації є привернути увагу авторів та потенційних користувачів до можливостей програмного пакету LaTeX. Історія його розвитку розпочалась на початку 1970-х років, коли відомий американський математик і програміст Дональд Кнут, переглядаючи гранки другого тома своєї книги "The Art of Computer Programming" зрозумів, що у жодній з існуючих на той час систем комп'ютерної типографії неможливо отримати необхідну якість видання. Розв'язуючи цю проблему Д. Кнут створив програму для верстки TeX та програму для створення шрифтів METAFONT.

У перших доповідях про TeX Д. Кнут відзначав такі його переваги: він призначений для використання авторами, які точно знають, що вони хочуть отримати; його вхідні і вихідні данні однакові для комп'ютерів з різною архітектурою; це вільний програмний продукт.

У 1984 р. світ побачив створений Леслі Лампортом набір макропакетів

LaTeX, який базується на TeX і робить використання його більш зручним, тому зараз більшість користувачів цих програм готує публікації саме в LaTeX. За більш ніж 20 років програми розвивались і вдосконалювались, не змінюючи початкових принципів. Зараз існують їх реалізації на всіх апаратних платформах і операційних системах.

Завдяки його перевагам LaTeX фактично став світовим стандартом для підготовки математичних і технічних текстів до публікації в наукових видавництвах. Вхідні формати файлів дозволяють готувати публікації на багатьох десятках мов і використовувати найрізноманітніші кодові таблиці. Про переваги винайдено Д. Кнутом і випробуваного часом принципу набору математичних формул свідчить той факт, що його стали використовувати у свіжих версіях редакторів формул в пакетах OpenOffice та MS Office.

Існують розширення (додаткові пакети), що дозволяють використовувати LaTeX для роботи не тільки з математичними текстами, наприклад ХуМTeX, MusiXTeX, призначені для роботи з хімічними формулами, музичними партитурами; TikZ & PGF дає широкі можливості для виконання графічних побудов. Такі пакети Beamer дозволяють підготувати якісні презентації для доповіді на науковій конференції або лекції, при цьому автор використовує підготовлений для друку текст і не зв'язаний особливостями офісних пакетів.

Відзначимо, що багато програм комп'ютерної алгебри можуть зберігати результати у форматі TeX. Вихідними форматами можуть бути зокрема такі поширені формати як .ps (PostScript) і .pdf (Portable Document Format). Існує ряд сторонніх програм-конверторів, як вільних, так і комерційних, які в значній мірі розв'язують проблеми сумісності з такими форматами, як .html, .rtf, .doc тощо.

Використання LaTeX дає автору можливість основну увагу зосередити на змісті роботи і її структурі, а оформлення і виконання багатьох допоміжних операцій, які вимагають багато часу і зусиль, перекласти на програму. В результаті користувач, навіть не будучи спеціалістом з комп'ютерної верстки, отримує друковану роботу (або її електронний варіант) найкращої якості при мінімальних витратах часу і зусиль. За принципом роботи LaTeX є спеціальною мовою розмітки, тобто вхідний файл є текстовим файлом, який містить як звичайний текст (українською або іншою мовою) так і спеціальні команди, що починаються символом `\` або `@`.

Звичайний текст буде відображений на виході зверстаним відповідно до передбачених вимог. Всі об'єкти, які не є гладким текстом, описуються за допомогою команд. Наприклад, команди `\section{..}`, `\subsection{..}` розпочинають новий розділ та підрозділ книги; `\tableofcontents` наказує програмі вставити зміст роботи, який генерується автоматично.

Також автоматично програма може нумерувати формули, теореми, рисунки, таблиці тощо і створювати посилання на ці об'єкти за присвоєними їм унікальними мітками. Схожим чином організовується робота із посиланнями на літературу. При написанні наукової роботи автор неодноразово

переставляє, видаляє, додає шматки тексту, доповнює і упорядковує список літератури, при цьому автору непотрібно слідкувати за тим, яким виявиться номер формули в остаточному варіанті, всі посилання будуть правильні. Якщо помилково двом об'єктам дати однакові мітки, то при компіляції система попередить про необхідність скоректувати їх.

На наш погляд навіть лише згадані можливості програми настільки полегшують роботу при підготовці складних наукових текстів, що на неї варто звернути увагу всім, кому таку роботу доводиться виконувати.

Для підготовки бібліографічних списків можна використовувати додатковий пакет BibTeX, який дозволяє інформацію про бібліографічні джерела зберігати у окремих базах даних і представляти її у різних видах в залежності від стандартів та вимог, які можуть бути різними, наприклад у різних видавництвах.

На нашу думку, основними перешкодами для більш широкого застосування LaTeX, є упереджена думка про те, що працювати з ним дуже складно. На це Лесли Лампорт відповів: "It's easy to use — if you're one of the 2% of the population who thinks logically and can read an instruction manual. The other 98% of the population would find it very hard or impossible to use" (Він простий у застосуванні — якщо ви належите до 2% людей, що мислять логічно і можуть прочитати інструкцію. Інші 98% вважатимуть його використання дуже складним або неможливим) [2].

Об'єм цієї публікації не дозволяє розглядати більше прикладів використання LaTeX, але в цьому і немає необхідності, адже існує достатньо літератури як для початківців так і досвідчених TeXників, і як сам програмний продукт так і література є у вільному доступі (див. [3-4]).

Література

1. Крылов А.Н. О печатании ученых трудов / А.Н. Крылов // Вестник Академии наук. — 1945. — №3. — С. 111 (см. также в кн.: Мои воспоминания, Л., 1984)
2. How LaTeX changed the face of Mathematics: An E-interview with Leslie Lamport // Mitteilungen der Deutschen Mathematiker-Vereinigung. — 2000. — №1. — P. 49-51. (see also www.tug.org/TUGboat/Articles/tb22-1-2/tb70lam-p.pdf)
3. TeX в Ярославском государственном университете // [Електронний ресурс] / режим доступу: <http://www.tex.uniyar.ac.ru>
4. TeX в Институте математики и механики УрО РАН // [Електронний ресурс] / режим доступу: <http://tom.imm.uran.ru/new/teximm.htm>

Використання контрольно-діагностичної програми iTEST у ході моніторингу якості процесу навчання старшокласників

І.С.Макаренко, П.В.Мерзлікін

Криворізький державний педагогічний університет, aspirantirina@ukr.net, linux-oid@i.ua

Автоматизоване оцінювання навчальних досягнень учнів – один із найважливіших напрямків застосування інформаційно-комунікаційних технологій у навчальному процесі. В роботі розглядається відкритий програмний пакет iTest як інструмент для автоматизації моніторингу якості процесу навчання старшокласників.

Проблема оцінювання якості процесу навчання, нині є однією з найбільш актуальних у сучасній вітчизняній педагогіці, зокрема вимірювання рівня навченості старшокласників, їх готовність до написання ЗНО з різних дисциплін. Слід зауважити, що в цьому випадку не менш важливим є питання розвитку подальших досліджень в області моніторингу якості процесу навчання, як частини (підсистеми) моніторингу якості освіти. Адже саме моніторинг забезпечує динамічне, неперервне та системно-організоване відстеження процесу навчання старшокласників у будь-який момент часу.

Порівняльний аналіз відомих методів вимірювання рівня знань, що використовуються у сучасній педагогіці свідчить про те, що тестування найбільшою мірою задовольняє критерію якості при визначенні рівня навченості, характеризується найвищою оперативністю. Окрім того, технологічність вважається однією з найважливіших властивостей тестів, що дозволяє повністю автоматизувати процес навчання за індивідуальними програмами та кардинально його вдосконалити. Широке впровадження тестування у педагогічну практику зумовлено також незалежністю результатів тестування від суб'єктивної оцінки екзаменатора. Тестування на відміну від усіх інших методів контролю, найбільшою мірою придатний для комп'ютерної реалізації.

Можна стверджувати, що моніторинг якості процесу навчання, зокрема автоматизоване оцінювання навчальних досягнень учнів – один із найважливіших напрямків застосування інформаційно-комунікаційних технологій у навчальному процесі.

У практиці тестування прийнято розрізняти тести комп'ютеризовані і тести комп'ютерні. Перші є адаптованими до середовища комп'ютера варіантом реалізації відомих тестів, а другі визначально розроблюються з урахуванням комп'ютерних технологій і можливостей обчислювальної техніки.

Метод комп'ютерного тестування має певні переваги і певні недоліки порівняно з бланковим тестуванням. Головною перевагою є отримання результату тестування одразу після його закінчення. Основним недоліком є необхідність тестованому мати мінімальні навички роботи за комп'юте-

ром та орієнтуватися в інтерфейсному середовищі тестової програми, а також потреба мати досить велику базу якісних тестових завдань для забезпечення закритості тесту і валідності його змісту.

Нині відома чимала кількість програм з конструювання тестів, що мають сприяти інтенсифікації та ефективності процесу навчання з одночасним підвищенням продуктивності праці усіх його учасників. Але більшість з них розповсюджуються на комерційній основі. Як наслідок, такі програмні продукти не завжди доступні для сучасних загальноосвітніх навчальних закладів в умовах обмеженого бюджету. Щодо відкритих рішень, то найбільш розповсюджений програмний пакет для створення тестів Keduca в даний момент практично не розвивається.

Ми пропонуємо у ході моніторингу якості процесу навчання застосувати контрольню-діагностичну програму iTest [1], що розповсюджується на умовах ліцензії GPL v2. Програмний пакет iTest написаний з використанням бібліотеки Qt 4. Це означає, що програма підтримує багато поширених платформ (GNU Linux, Apple Mac OS X, інші UNIX-подібні системи, Microsoft Windows). З іншого боку, використання бібліотеки Qt 4 дещо підвищує системні вимоги. Проте, з огляду на технічне оснащення сучасних шкіл, системні вимоги є прийнятними.

Програма iTest складається з серверної та клієнтської частин. Серверна частина включає засоби управління базою даних та екзаменаційний сервер, клієнтська частина призначена для проходження тестів учнями.

Всі питання зберігаються в базі даних. Кожне питання належить до певної групи питань та має свій рівень складності. Також ведеться статистика для кожного питання (наприклад, відсоток правильних відповідей), що дозволяє коректувати питання, виходячи з цих даних, а також визначити теми, з яких в учнів виникають проблеми.

Крім того, в базі даних зберігаються дані про учнів. Можна слідкувати за динамікою якості знань як окремого учня, так і всього класу на певному проміжку часу.

Протягом тесту вчитель може слідкувати за проходженням тестів учнями в “прямому ефірі”, бачити, які варіанти відповіді обираються. Існує можливість одразу після проходження тесту роздрукувати результат. Підтримується гнучке налаштування системи оцінювання.

Описані можливості даного програмного продукту роблять його зручним для моніторингу якості процесу навчання.

Але, не зважаючи на ряд переваг, використання iTest не завжди може бути доступним для вчителів, не спеціалістів з інформатики з огляду на відсутність доступної україномовної документації. Тому нами був здійснений переклад інструкції користувача, а також її доповнення, виходячи з нашого досвіду впровадження програмного пакету в загальноосвітніх навчальних закладах.

Крім того, був розроблений пакет тестових завдань з ключових тем фізики для курсу 10-11 класу, з урахуванням усіх правил конструювання тестів.

Таким чином, використання критеріально-діагностичної програми iTest

у ході моніторингу якості процесу навчання старшокласників, дозволить продовжити впровадження інформаційно-комунікативних технологій у навчально-виховний процес, сприятиме підвищенню об'єктивності оцінювання знань учнів, що в свою чергу впливатиме на зростання рівня їх мотивації до навчання. Окрім того, широке впровадження комп'ютерного тестування у шкільну практику означає економію робочого часу вчителя, який він зможе витратити на самовдосконалення своєї педагогічної майстерності, а статистична звітність, яку надає iTest, дозволить яскраво демонструвати результати навчальних досягнень учнів усім зацікавленим особам.

У перспективі ми вбачаємо подальше вдосконалення програми iTest, шляхом розширення її функціональних можливостей.

Література

<http://itest.sourceforge.net/>

Використання бібліотеки класів GEANT4 в ОС LINUX при розробці програмного забезпечення для моделювання процесів взаємодії випромінювання з речовиною

Малихіна Т.В.

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

Представлений приклад використання вільного ПЗ в науково-дослідній роботі кафедри електроніки і управляючих систем факультету комп'ютерних наук Харківського національного університету. Розроблено програму з використанням вільного ПЗ - бібліотеки класів Geant4 в ОС Linux.

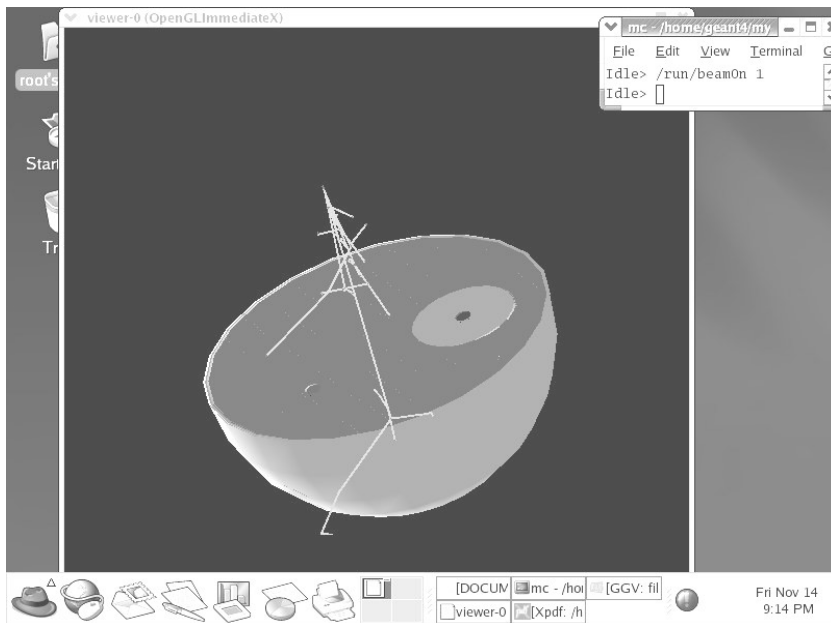
Проект Geant4 розробляється міжнародною колаборацією, у складі якої працюють вчені Європейської організації з ядерних досліджень (CERN), вчені з Японії, США, Канади та інших країн [1]. Geant4 є вільним ПЗ. Областю застосування Geant4 є, в основному, наукові дослідження у фізиці високих енергій (BaBar, ATLAS, ALICE і ін), космічних дослідженнях (проекти ESA), медичних наукових розробках, екологічних проєктах тощо.

Geant4 - бібліотека класів для комп'ютерного моделювання процесів взаємодії випромінювання з речовиною, що використовує метод Монте-Карло. 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 і простежування частинок; відгук детектора; моделювання події; управління треком і т.д. [2]. Бібліотека класів Geant4 має великий набір службових функцій, а також є можливість використовувати модуль візуалізації.

На факультеті комп'ютерних наук ХНУ імені В. Н. Каразіна розроблена комп'ютерна програма, що дозволяє проводити віртуальний фізичний експеримент для визначення оптимальних параметрів установки, призначеної для локації точкового джерела гамма-випромінювання. Програма розроблена в ОС Linux на мові C++ і використовує Geant4. Завданням, яке вирішує дане моделювання, є виявлення оптимальних геометричних параметрів установки для ефективного визначення напрямку приходу гамма-кванта в систему, що складається з кулястого алюмінієвого поглинача радіусом 50 мм, в якому знаходиться сферична порожнина радіусом 15 мм. Як детектори в даній системі використовуються кристали CdTe у формі куба з ребром 5 мм.

Програма має два режими роботи: пакетний та інтерактивний з графічним інтерфейсом. Інтерактивний режим з графічним інтерфейсом призначений для візуалізації моделі установки і траєкторій частинок для найбільш повного уявлення просторового розподілу гамма-квантів під час проходження через установку. Модуль візуалізації використовує графічну бібліотеку OpenGL. На малюнку представлена візуалізація фрагмента

установки, що використовується при комп'ютерному моделюванні.



Пакетний режим роботи програми необхідний для моделювання з набором великий статистики подій. У пакетному режимі програма з метою економії ресурсів не використовує модуль візуалізації, але записує у вихідний файл результати моделювання.

У результаті моделювання установки показано, що установка може бути використана для локації точкових джерел гамма-випромінювання на півплощині, тобто, в діапазоні кутів від 0 до 180 градусів.

Більш докладно геометричні параметри установки та результати моделювання опубліковані в роботі [3].

Література

1. <http://geant4.web.cern.ch/geant4/collaboration/index.shtml>
2. <http://geant4.web.cern.ch/geant4/applications/index.shtml>
3. Т.В.Мальхіна, В.В.Марущенко, А.В.Сакун, Н.Г.Стерведов. Комп'ютерное моделирование установки для локализации источника гамма-излучения. Вестник ХНУ. Серия МИА., № 925, 2010, вып. 14, С. 132-139.

Використання системи Moodle для організації контролю майбутніми вчителями-гуманітаріями

Маркова Є.С.

Бердянський державний педагогічний університет, markova_es@mail.ru

В тезах розглядається питання комп'ютерного тестування і підготовки майбутніх вчителів-гуманітаріїв до використання системи Moodle для організації контролю.

Активне впровадження в усі сфери діяльності людини сучасних інформаційних технологій потребує вирішення проблеми підготовки спеціалістів гуманітарного профілю, які здатні ефективно працювати в умовах становлення інформатизації системи освіти. У зв'язку з цим неодмінною ознакою високого професіоналізму вчителя-гуманітарія є оволодіння комп'ютерними методами оцінювання навчальної діяльності учнів. Інформатизація освіти та інтенсивний технологічний розвиток засобів навчання призвели також до змін у вимогах до організації контролю навчальної діяльності. Наразі набули поширення комп'ютерні форми контролю знань, що потребують більшої формалізації та технологічності засобів контролю.

Комп'ютеризація під час проведення контролю знань є ефективним засобом, що відкриває нові можливості в рефлексії учбової діяльності, які дозволяють по-новому організувати навчальний процес та стимулювати мотивацію навчання, розвивати творчі здібності учнів.

Комп'ютерне тестування – це засіб, що дозволяє з мінімальними витратами часу викладача об'єктивно перевірити знання великої кількості учнів. Добре спланований графік тестувань є гарним стимулом, що спонукує студентів до систематичної роботи протягом семестру. Перевагою комп'ютерного тестування є автоматична перевірка результатів і виключення впливу людського фактора.

З огляду на вищесказане вважаємо за необхідне під час навчання базового курсу інформаційних технологій ознайомлювати студентів гуманітарних факультетів педагогічних ВНЗ з методами розробки тестових завдань та використання системи Moodle для здійснення тестування.

Moodle (модульне об'єктно-орієнтоване динамічне навчальне середовище) – це інструментальне середовище для розробки як окремих Online-курсів, так й освітніх Веб-сайтів. В основу проекту покладена теорія соціального конструктивізму і її використання для навчання.

Moodle пропонує широкий спектр можливостей для повноцінної підтримки процесу навчання в дистанційному середовищі - різноманітні способи подання навчального матеріалу, перевірки знань і контролю успішності. Moodle поширюється вільно, як програмне забезпечення з відкритим вихідним кодом (відповідно до [GNU Public License](#)). Система Moodle пройшла сертифікацію відповідності стандарту SCORM 1.2

За допомогою модуля Тести у системі Moodle можна створювати і задавати тести багатьох типів:

1. питання в закритій формі (з однією чи багатьма правильними відповідями);
2. питання відкриті, що потребують короткої відповіді (слово чи фразу);
3. питання з альтернативними відповідями (так/ні);
4. випадкові питання;
5. числові питання;
6. вкладені питання;
7. питання на відповідність;
8. опис;
9. нарис, есе;
10. випадкові питання у відкритій формі на відповідність.

Тести можуть допускати кілька спроб відповіді. Кожна спроба автоматично перевіряється і викладач може схвалювати відповіді, надати свій коментар або правильний варіант відповіді. Цей модуль включає і пропонує інструменти, що забезпечують можливість оцінювання. Час виконання тесту також може бути обмежений або ні. Може бути вбудований описовий текст і графіка чи посилання на мультимедійні об'єкти.

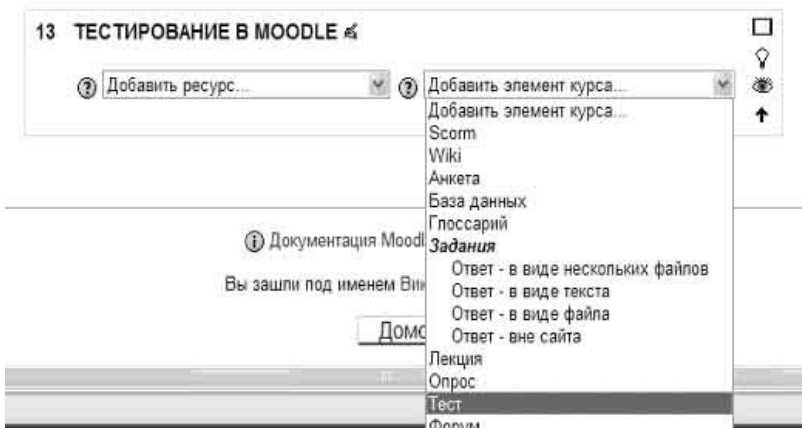


Рис. 1. Створення тесту у системі Moodle

Система Moodle надає широкий спектр можливостей для побудови тестів різного роду:

- кількість спроб, які налагоджуються, проходження тесту;
- тимчасові затримки, що налагоджуються, між спробами;
- вибір методу оцінювання (у випадку декількох спроб): вища/нижча оцінка, перша/остання спроба;
- перемішування як самих питань у тесті, так і варіантів відповідей;
- навчальний режим: студент зможе відповісти на запитання кілька разів у рамках однієї спроби. Можливе нарахування штрафних балів за кожну неправильну відповідь;
- налаштування режиму перегляду результатів: що (свою відповідь, бали,

коментарі, всі відповіді, загальний коментар до всього тесту) і коли (відразу після спроби, пізніше, але перше ніж тест буде закритий, після того як тест буде закритий) зможе побачити студент;

- коментарі до всього тесту залежно від отриманої оцінки;
- коментарі для кожного варіанта відповіді;
- коментар для кожного питання;
- конструювання тесту на основі випадкового вибору питань із категорій.

Отже, тест у системі Moodle – інтерактивний елемент курсу, який дозволяє викладачу створювати набори тестових завдань різних типів для здійснення різних видів контролю: проміжного, тематичного, модульного та підсумкового [1; 2].

Використання системи Moodle дає можливість використовувати тест не тільки на етапі контролю, а й на етапі формування та закріплення компетенцій, установивши можливість декількох спроб та різні стратегії оцінювання – з урахуванням попередніх спроб чи ні. Проходячи тестовий тренінг на етапі самостійної підготовки до контрольного тестування, учень має можливість сам критично оцінити свої успіхи та промахи при вивченні даного матеріалу. Тому є актуальним дослідження питання підготовки майбутніх вчителів у педагогічних ВНЗ до використання системи Moodle для організації тестового контролю.

Література

1. Андреев А.В. Практика электронного обучения с использованием Moodle [Текст] / А.В. Андреев, С.В. Андреева, И.Б. Доценко. – Таганрог: Изд-во ТТИ ЮФУ, 2008. – 146 с.
2. Болубаш Н.М. Створення тестів для інформаційно-освітньої системи на базі електронної платформи Moodle: Навчальний посібник [Текст] / Н.М. Болубаш; під ред. О.П. Мещанінова. – Миколаїв: Вид-во МДГУ ім. Петра Могили, 2008. – 68 с
3. Moodle – <http://www.moodle.org>

Комплекс програм для лазерних спостережень штучних супутників Землі

Мартинюк-Лотоцький К.П., Білінський А.І.

Астрономічна обсерваторія Львівського національного університету імені Івана Франка langure@mail.ru, slr1831@ukr.net

Приведено результати розробки програмного забезпечення з використанням засобів операційної системи Linux. Отримано низку програм та скриптів, що об'єднані задачею організації потоку даних для проведення лазерно-локаційних спостережень штучних супутників Землі у Астрономічній обсерваторії ЛНУ імені Івана Франка.

Лазерна локація супутників (ЛЛС) полягає у прямому вимірюванні часу прольоту пікосекундного лазерного імпульсу від станції до спеціалізованого супутника та у зворотньому напрямку – таким чином реєструється подвійна відстань до об'єкта. До складу ЛЛС станції входять такі системи: телескоп, лазерний передавач, приймач випромінювання, система обробки і реєстрації сигналів, служба точного часу і частоти, метеостанція, відповідне комп'ютерне та програмне забезпечення.

ЛЛС станцію «Львів-1831» [1] введено у дію у 1998 році, а у 2002 році вона увійшла у міжнародну службу лазерної локації супутників ILRS [2] та в національну мережу Українського центру визначення параметрів обертання Землі Головної астрономічної обсерваторії Національної академії наук України.

Один з напрямків досліджень – розробка ПЗ для керування телескопом, результати якої представлено у «Програмне забезпечення керування телескопом ЛЛС-станції «Львів-1831»». Також розробляється програмне забезпечення для керування багатофункціональним частотоміром SR620 через послідовний порт `/dev/ttyS0` – для цього розроблено блок узгодження сигналів та спряження з комп'ютером керування телескопом, який видає імпульси часових воріт при спостереженні супутника.

Друга частина – це роботи з автоматизації керування системами станції та забезпечення пасивного потоку даних, тобто без втручання оператора. Відповідне програмне забезпечення теж реалізовується у системі Linux, що забезпечує простий та безпечний доступ до даних та програм. Структурна схема програмного забезпечення, що забезпечує потік даних наведено на рис. 1, контроль за потоком даних можна здійснювати з віддаленого терміналу (напр. через ssh, веббраузер).

Таким чином створене програмне забезпечення реалізовує такі функції (рис. 1):

- отримання первинних даних для проведення спостережень та приведення їх до потрібного формату [3];
- первинна обробка отриманих результатів спостережень [3, 4];
- ведення баз даних метеорологічних параметрів та результатів спостережень, створення архіву самих файлів [5, 6];

- оперативна передача результатів спостережень у Європейську базу даних мережі ILRS та в УЦПОЗ ГАО НАНУ;
- гнучкий контроль процесів через веббраузер або з віддаленого терміналу через ssh.

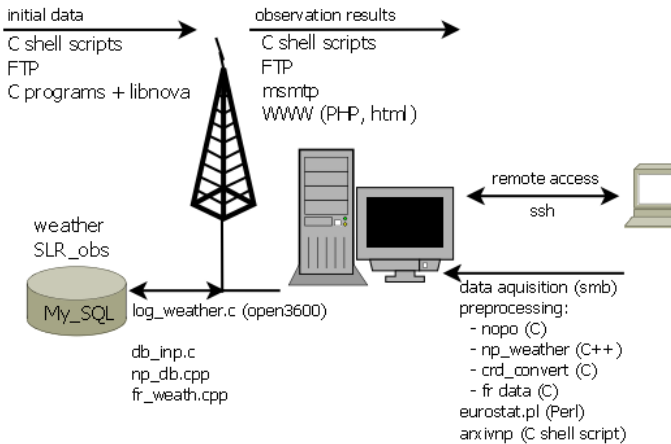


Рис. 1. Діаграма ПЗ для лазерних спостережень.

Отже, вдосконалене та нове розроблене ПЗ забезпечило більш оптимальну роботу оператора, скоротило час діалогового режиму роботи, що дало можливість підвищити ефективність ЛЛС спостережень. Також результати роботи дали можливість приймати участь в інших наукових та прикладних дослідженнях з використанням телескопа ТПЛ-1М [7].

Література

1. Blagodyr, Ja., Bilinsky, A., Martynyuk-Lototsky, K., et al. "Overview and Performance of the Ukrainian SLR Station "Lviv-1831"" // *Artificial Satellites*, Vol.42(1), pp. 9 - 15, 2007. DOI: 10.2478/v10018-007-0014-4.
2. Pearlman, M.R., Degnan, J.J., and Bosworth, J.M., "[The International Laser Ranging Service](#)" // *Advances in Space Research*, Vol. 30, No. 2, pp. 135-143, July 2002, DOI:10.1016/S0273-1177(02)00277-6.
3. Білінський А.І., Мартинюк-Лотоцький К., Благодир Я.Т. Автоматизація ЛЛС-спостережень на станції «Львів» // Бюлетень Українського центру визначення параметрів обертання Землі.- К.:ВАІТЕ, 2008.- №3, с.7-10. Доступний з: http://www.mao.kiev.ua/EOP/UKRAINE/books/bulletin_3.pdf
4. С. Апунович Опрацювання результатів спостережень штучних супутників Землі на лазерно-локаційній станції «Львів 1831» / С. Апунович, А. Білінський, С. Апунович, Я. Благодир // [Теоретична електротехніка : збірник наукових праць](#). – Львів : Видавничий центр ЛНУ ім. І. Франка, 2009. – [Вип. 60](#). – С. 104-110.

5. А. Білінський Використання цифрової метеостанції WS 3600 на станції лазерної віддалеметрії «Львів 1831» / А. Білінський, К. Мартинюк-Лотоцький, Н. Вірун, С. Апуневич, Я. Благодир, Е. Вовчик // [Теоретична електротехніка : збірник наукових праць](#). – Львів : Видавничий центр ЛНУ ім. І. Франка, 2009. – Вип. 60. – С. 111-117.
6. Білінський А.І. База даних результатів ЛЛС спостережень станції “Львів-1831” / Білінський А.І., Мартинюк-Лотоцький К.П // Бюлетень Українського центру визначення параметрів обертання Землі. – К. : Компанія ВАПТЕ, 2009. – №4. – С. 7-11.
7. Козырев Е.С., Сибирякова Е.С., Шульга А.В., Апуневич С.В., Билинский А.И., Благодыр Я.Т., Вирун Н.В., Вовчик Е.Б., Логвиненко А.А., Мартинюк-Лотоцкий К.П. Совместный проект НИИ «Николаевская астрономическая обсерватория» и Львовской астрономической обсерватории по позиционным наблюдениям ИСЗ // Материалы международной конференции «Околосемная Астрономия-2007», Нальчик.-2008, с.326-331.

Компьютеризация процесса обучения

Матвеева Л.М., Назмутдинов Ф.Ф.

Башкирский государственный университет, г. Уфа, Россия, MatveevaLM@mail.ru,
mmx@mail.ru

Актуальна проблема развития и внедрения в учебный процесс современных обучающих технологий. Целесообразно использование компьютерной техники с установкой разнообразных операционных систем и программ: Ubuntu, Office Free и др. Представлен опыт использования программы Lazarus в учебном процессе по специализации физическая гидродинамика (ФГД). Перспективно начинать обучение детей современным техническим методам программирования со школьного возраста, используя пакеты СПО - свободного программного обеспечения.

В настоящее время становится актуальным использование СПО в учебном процессе в ВУЗе. Это связано с тем, что появляется большое количество свободно распространяемых программ, которые по функционалу мало отличаются от широко известных, проверенных временем и хорошо зарекомендовавших себя программ: Windows, MSOffice, Delphi и др.

В физико-техническом институте и на некоторых факультетах Башкирского государственного университета обучение информатике, с включением программирования, и ряду других учебных дисциплин, происходит на компьютерах с установленными операционными системами: Ubuntu, и др.

Такие учебные предметы как «компьютерные технологии», «численные методы в гидродинамике» осваиваются студентами с использованием программы Lazarus, обработки материалов практических исследований при помощи встроенных функций системы Excel, программной среды Turbo Pascal, Delphi. В рамках этих сред практически осуществляется компьютерное моделирование изучаемых физических явлений и процессов с целью наглядной демонстрации присутствующих в них закономерностей. Кроме того осуществляется использование комплексов программ СПО на основе операционной системы Linux.

Так, студенты - физики IV курса при выполнении лабораторных работ получают задания, включающие составление расчётных программ с расшифровкой идентификаторов, выбором различных параметров, характеризующих физические процессы, моделирование физического процесса. Все решения поставленных задач выдаются в различных математических представлениях. Полученные результаты анализируются, на их основе производится сопоставление реального физического эксперимента с расчётным – косвенным экспериментом, проведённым по формулам и уравнениям.

При переходе учебного заведения на использование СПО возникали опасения по поводу того, что нынешние выпускники вузов в работе на промышленных предприятиях будут иметь дело с давно распространенными программными продуктами. Однако, беседы, с уже работающими молодыми специалистами, показали, что знания и навыки, полученные в

процессе обучения студентами в вузах, в том числе и в Башкирском государственном университете свободно могут быть перенесены и на другие программные комплексы.

Чтобы не случались подобные ситуации, мы считаем, что целесообразно уже в современной российской школе развивать у детей все таланты и прививать компьютерную грамотность, являющуюся технократической компонентой в образовании. развивать различные навыки, базирующиеся на образном, абстрактном мышлении, и работающие на основе логики. Очевидно, что для обучающейся молодежи в профильных (естественнонаучного направления), а также и в общеобразовательных классах (с гуманитарным уклоном) учебных заведений: школ, лицеев, ПТУ, техникумов, вузов будут полезны знания основ технических предметов, включая курс информатики, азы программирования, знакомство с компьютерными языками и т.д.

Известно, что как в обучении, так и в реальной жизни для человека очень многие действия и решения разнообразных задач выполняются "на стыке наук". Грамотного инженера и талантливое гуманитария можно воспитать и подготовить лишь в современной культурной среде.

В настоящее время в средних школах и других учебных заведениях, а также в вузах Башкирии робко берет начало практика использования в учебном процессе СПО, как возможной альтернативы традиционным программным средствам, применяемым в рамках операционных систем семейства Windows.

Мы надеемся, что операционная система СПО, наряду с другими программными проектами, будет шире внедряться в процесс обучения молодого поколения в школах, средних учебных заведениях, а также в вузах.

Таким образом, будет обозначен приоритет подготовки будущих инженеров, и других специалистов так необходимых для реализации инновационного пути развития нашей страны.

Досвід викладання редактора GIMP при вивченні курсу «Комп'ютерна графіка та дизайн»

Матвієнко Ю.С.

Полтавський національний педагогічний університет імені В.Г.Короленка
wasiwews2009@gmail.com

В тезах доповіді описується досвід викладання графічного редактора GIMP при вивченні курсу «Комп'ютерна графіка та дизайн» студентами спеціальності «Інженери-програмісти». Як показала практика застосування даного редактора студентами, вони оволоділи в повному обсязі основами роботи в GIMP, які б стали базою для подальшого вдосконалення своїх вмінь та навичок.

Графічний редактор GIMP викладається в рамках курсу «Комп'ютерна графіка та дизайн» після вивчення математичних основ комп'ютерної графіки як програмний засіб основних алгоритмів растеризації. Даний редактор обраний не випадково. Зупинюся на основних аспектах, які стали визначальними під час вибору редактора.

GIMP – кросплатформне програмне забезпечення для редагування зображень (GIMP- GNU Image Manipulation Program). Редактор GIMP придатний для розв'язку безлічі завдань по зміні зображень, включаючи ретуш фотографій, об'єднання й створення зображень. Програма GIMP багатofункціональна. Її можна використовувати як простий графічний редактор, як професійний додаток для ретушування фотографій, як мережну систему пакетної обробки зображень, як програму для рендерингу зображень, як перетворювач форматів зображення тощо.

GIMP спроектований розширюваним, тобто за допомогою доповнень він здатний реалізовувати будь-які можливі функції. Передовий інтерфейс для розробки сценаріїв дозволяє легко автоматизувати виконання будь-яких завдань різного рівня складності. Однією з сильних сторін GIMP є його доступність із багатьох джерел для багатьох операційних систем.

GIMP входить до складу більшості дистрибутивів GNU/Linux. GIMP також доступний і для інших операційних систем на кшталт Microsoft Windows™ або Mac OS X™ від Apple (Darwin).

GIMP – вільне програмне забезпечення, що випускається під ліцензією GPL (General Public License). GPL надає користувачам право доступу до вихідного коду програм і право змінювати його. Будучи досить потужним продуктом, GIMP здатний стати незамінним помічником у таких областях, як підготовка графіки для Web сторінок і поліграфічної продукції, оформлення програм (малювання піктограм, заставок і т.п.), створення анімаційних роликів, обробка кадрів для відеофрагментів і побудова текстур для тривимірної анімації. Дуже корисна функція створення й обробки анімаційних роликів, що дозволяє накладати анімацію на об'єкт як текстуру і виконувати певні фінішні операції після рендерингу.

В результаті вивчення курсу одні студенти характеризували GIMP як доступний в Linux аналог Photoshop, інші наполягали на тому, що

принципово неможливо порівнювати ці дві програми, і відзначають, що їх інтерфейс і основна концепція значно різняться, а збігається, строго говорячи, тільки тип оброблюваних даних – растрові зображення. Це, звичайно, не зовсім вірно: редактори подібні як мінімум ще й тим, що обидва належать до професійних.

Загалом, після завершення вивчення, можна зробити висновки, що набори вбудованих інструментів у двох редакторах досить схожі, і студенту, який знав Photoshop, було нескладно почати роботу в GIMP. Але в освоєнні більш складних засобів досвід використання Photoshop не допоміг: модулі, що підключаються, GIMP організовані зовсім по-іншому.

Крім самого редактора студентам було запропоновано реалізувати індивідуальні дизайнерські роботи, використовуючи GIMP Paint Studio.

Gimp Paint Studio – це колекція пензлів і супутніх налаштувань інструментів (пресетів), призначена для прискореного виконання повторюваних завдань. GPS дозволяє уникнути необхідності щораз вручну набувувати параметри інструментів. Як виявилось більшість студентів, які до початку вивчення курсу чули про редактор GIMP навіть не здогадувалися про існування пресетів (предвстановлених налаштуваннях інструментів). Ця можливість рідко обговорюється в мережі Інтернет, але є досить корисною функцією GIMP.

Gimp Paint Studio орієнтований у першу чергу на художників та дизайнерів. Загалом кажучи, це не тільки колекція пензлів і пресетів, але й спроба зробити ще дещо. Автор цього пакета ([іспанський ілюстратор Ramon Miranda](#)) доводить, що GIMP придатний не тільки для редагування фотографій, але що GIMP також є відмінним інструментом для цифрових художників. За допомогою GIMP можна з нуля створювати зовсім нові витвори мистецтва.

В GIMP закладений величезний потенціал, який однак не повною мірою реалізований у налаштуваннях за замовчуванням. В GPS зібрані пресети й пензлі для художників. Предвстановлені налаштування дозволяють художникові замість того, щоб довго розбиратися в налаштуваннях редактора GIMP, відразу приступитися до роботи. Пресети названі за аналогією з інструментами художника в реальному світі, і художник замість ручного налаштування може, приміром, вибрати серед пресетів простий олівець заданої твердості й приступитися до створення свого чергового шедевра.

Крім пензлів і пресетів у пакеті GPS присутня також колекція палітр і градієнтів. Палітри також орієнтовані в першу чергу на цифрових художників. Приміром, у колекції присутні палітри із кольорами людської шкіри, кольорами людського волосся та ін.

В ході застосування даного пакету при виконанні лабораторних робіт та індивідуальних дизайнерських завдань студенти дійшли висновку про корисність та доцільність використання у власній професійній діяльності зв'язки GIMP+GPL. Крім цього велике значення має і етичний бік вивчення вільного графічного редактора, а не «краденого» пропрієтарного.

Як недолік варто зазначити брак методичного забезпечення курсу. В ході викладання було розроблено, апробовано та опубліковано навчально-

методичний комплекс «Комп'ютерна графіка та дизайн» для студентів фізико-математичного факультету спеціальності «Інженери-програмісти».

В результаті вивчення студентами курсу «Комп'ютерна графіка та дизайн» головний здобутком можна вважати психологічний злам, який відбувся при прийнятті студентами рішення використовувати для власних професійних потреб вільного програмного забезпечення.

Міграція на вільне програмне забезпечення

Машков В. В.

SoftServe Inc., mashckoff@gmail.com

Розглядаються етапи міграції з успадкової пропрієтарної системи на вільне ПЗ.

Перед багатьма організаціями постають питання повного чи часткового переходу на використання вільного ПЗ. Іноді здається, що питання, що постають, неймовірно важкі, кількість проблем росте кожного дня. І що з цим впоратись неможливо. Але це не так, це доведено історією. Наприклад, років 18 тому на більшості комп'ютерів домінувала ОС MS DOS, редактор "Лексикон" і Norton Commander. Потім з'явилася графічна надбудова - Windows 3.1, яка зажадала зміни підходів до вирішення проблем офісної автоматизації. Отже, подібні еволюційні процеси є природними для будь-якого типу програмного забезпечення. І єдина різниця в тому, що тепер у користувачів при прийнятті рішення про міграцію на нову програмну платформу з'явився вибір -- можливість використання вільного ПЗ. Причин використання багато - це можливість знизити вартість ліцензій або ж бажання підвищити безпеку системи чи потреба у відкритих стандартах. Згадайте коли до Вашого Office 95 потрапив документ у форматі Office 97 — скільки проблем виникло через несумісність форматів. Це лише один сумний приклад наслідків використання пропрієтарного ПЗ.

Як ілюстрацію візьмемо типову невелику фірму чи підрозділ: один виділений сервер і декілька робочих станцій. Історично склалося так, що тут найчастіше використовується пропрієтарне ПЗ.

Отже, прийшов день "ЧЕ" і виникла потреба у зміні використовуваних програмних рішень - з'явилась потреба міграції на нову систему. Такі міграції є двох типів - спішні і успішні. Для успіху, перш за все необхідно сформувати робочу групу. В яку в обов'язковому порядку, крім технічних фахівців, повинен входити організатор (це може бути керівник підрозділу або менеджер більш високого рівня), який вирішує питання не-технічного характеру. Непоганою ідеєю буде включити до складу рупи незалежного експерта, чия оцінка дасть змогу уникнути прорахунків на етапі проектування, тестування і впровадження рішення на базі ВПЗ.

Наступним кроком буде опис існуючої системи. Тут необхідно провести інвентаризацію існуючого апаратного та програмне забезпечення, а також описати модель (функціональність) діючої інформаційної системи.

Наступним етапом є опис функціональних можливостей проектованої системи, щоб уникнути переплати за надмірну функціональність і підвищену вимогу до апаратного забезпечення. Наприклад, потреба в електронних словниках може бути лише в деяких спеціалістів, а історично було встановлено скрізь. На цьому етапі нам стане в нагоді побудована на попередньому етапі модель діючої інформаційної системи, з якої ми можемо перенести вимоги до її функціональних можливостей і врахувати

їх при проектуванні нової системи.

Оскільки ми хочемо побудувати економічно привабливу систему, нам необхідно обов'язково розглянути питання сукупної вартості володіння. Тема розрахунку сукупної вартості володіння ТСО актуальна, але на жаль не існує усталених методик розрахунку. Я не буду заглиблюватися в тонкощі розрахунку, але зупинюся на основних елементах, які, на мій погляд, необхідно врахувати. Часто вартість володіння вільним ПЗ є нижчою за рахунок відсутності обов'язкових ліцензійних платежів і низькому ризику вірусних атак. Основна складова частина вартості ТСО у вільних рішеннях припадає на підтримку.

В більшості випадків частина функціональності клієнтів закривається вільними аналогами без проблем. Якщо немає вільного аналога, то одним з рішень буде використання емуляторів оточення (Wine), термінальних рішень до серверів Microsoft Windows з конкурентними ліцензіями або даний пропрієтарний продукт у його рідному середовищі.

Пілотний проект буде основою для ухвалення рішення про впровадження нової інформаційної системи на базі ВПЗ. Під час вибору об'єкта випробувань необхідно знайти золоту середину. По-перше, вибраний об'єкт повинен надати достовірні дані для оцінки. По-друге, проведення пілотного проекту не повинно критично впливати на ведення бізнесу. У будь-якому випадку проведення пілотного проекту є вигідним кроком. Його результати можна використовувати і на інших ділянках.

Як приклад розглянемо заміну використання макросів MS Office, які використовуються для обчислень в нашій фірмі. Отже, мета - заміна офісного пакету вільним аналогом. У результаті ми маємо отримати макроси, які реалізують необхідну функціональність. У процесі заміни необхідно перевірити правильність роботи макросів і з'ясувати, як користувачі оцінюють роботу офісних пакетів "до" і "після".

Ретельне планування - запорука успіху міграції. Під час планування враховуйте особливі випадки. Це може і складне-спеціалізоване ПЗ та недостатня підготовка персоналу, що може зажадати його навчання. І особливі вимоги з боку керівництва (планування міграції на період подачі в податкову адміністрації річного звіту -- погана ідея). Складіть детальний план, в якому мають бути описаними всі деталі, починаючи від необхідності створення резервних копій існуючих даних до розгортання системи і закінчуючи термінами навчання користувачів. Приділіть особливу увагу перенесення даних з успадкованої системи до нової. На етапі навчання користувачів приділіть особливу увагу тим, хто є прихильником старої системи. На підставі додаткових даних після проведення пілотного проекту виправте кінцеву вартість проекту міграції. Визначте відповідальних осіб та складіть докладний календарний план, вкажіть у ньому проміжні контрольні точки.

Залишається найпростіше - власне міграція. Після переходу на використання рішень на базі ВПЗ Ви можете отримати: кращу безпеку системи в цілому; покращену керованість робочими станціями; зменшення залежності від закритого ПЗ.

Відкриті ліцензії і стандарти

Машков В. В.

SoftServe Inc., mashckoff@gmail.com

В доповіді описується термінологія вільних і відкритих стандартів і технологій, вільного відкритого програмного забезпечення, показано відмінності від безкоштовного ПЗ.

Історично склалося, що з часів Радянського Союзу питанням ліцензій на програмне забезпечення не надавалось належної уваги. Це призвело до того, що не тільки користувачі ПЗ, але й ІТ-професіонали часто плутаються в термінах відкрите, вільне і безкоштовне програмне забезпечення, не розуміють важливості відкритих стандартів. Наслідком цього в країні ми бачимо безладне використання програмних продуктів, високий рівень використання неліцензійного ПЗ, використання нестандартних технологій — на рівні держави, у військовому відомстві, в освіті і в комерційних фірмах.

Для правильного стратегічного планування потрібно мати точне і коректне уявлення про термінологію в цих питаннях.

Спочатку розглянемо питання відкритих технологій. 9 вересня 2005-го року урядовці 13 країн¹ у звіті Світовому банку закликали держави застосовувати відкриті стандарти і інформаційні технології з метою сприяння економічному зростанню держав, ефективності управління та інноваціям. Вперше на світовому рівні було визначено поняття відкритих стандартів і технологій.

Ідея відкритих технологій і стандартів виходить з того, що для різних завдань необхідні різні системи - як системи загального призначення, так і спеціалізовані.

Складність побудови систем чи об'єднання існуючих полягає в тому, що потрібно забезпечити:

- взаємодію різнорідних систем в гетерогенній мережі;
- обмін даними між різними аплікаціями (додатками) на різних платформах;
- здатність до міграції прикладного ПЗ з однієї платформи на іншу (наприклад, шляхом перекомпіляції джерельних кодів);
- за можливістю однорідний інтерфейс користувача.

Ці завдання передбачається вирішувати за допомогою відкритих технологій і стандартів.

Відкрита технологія чи стандарт чи формат даних подібні до мови спілкування, яку всі розуміють однаково. Інтернет виник саме завдяки відкритим стандартам. Профінансувавши створення умов та реалізацію протоколу TCP/IP для Unix, агентство DARPA дозволило використовувати цей стандарт усім бажаним. Звідти з'явився Internet, що мільйони індивідуальних комп'ютерів, і мереж, розташованих по всьому світу.

¹<http://www.nytimes.com/2005/09/09/technology/09open.html>

На відміну від відкритих стандартів, закриті стандарти мають локальну область застосування і підтримуються обмеженим колом систем і фахівців (розробників). Наприклад, служба директорій Microsoft Active Directory (MSAD) побудована на відкритому протоколі Kerberos, але Microsoft реалізувала Kerberos таким чином, що MSAD є несумісною з реалізаціями інших постачальників.

Але чи можна назвати будь-який де-факто поширений стандарт відкритим? Давайте розглянемо формати даних Microsoft Office. Автор стандарту не надав в публічний доступ технічну документацію і таким чином заборонив іншим доступ до знання, як організовані ваші дані. Чи зможете ви отримати до них доступ через десять чи двадцять років? Інші продукти, і зокрема вільний офісний пакет OpenOffice.org, нерідко забезпечують досить якісне читання і запис цих документів — але закритість формату навряд чи дозволяє забезпечити повноцінну підтримку. Отже, першою важливою ознакою відкритого стандарту буде наявність у публічному доступі технічної документації, достатньої для цього стандарту.

Проте, лише наявності технічної документації недостатньо. Наприклад, формат OpenOffice.org 1 (.sx?) не був відкритим, попри непогане документування і вільний статус продукту. Цього замало. OpenOffice.org гілки 2 використовує за умовчанням OASIS OpenDocument XML format - міжнародний відкритий формат даних. Формат Open Document схвалений Єврокомісією як стандарт для міжнародного документообігу. Відкритий стандарт повинен відповідати одній умові - бути затвердженим (сертифікованим) як міжнародний стандарт однієї із міжнародних організацій, що здійснюють стандартизацію (ISO, IEEE, OASIS).

Але поглянемо на нашу аналогію з мовами. Важко уявити вимогу платити за право використання мови - такого не було навіть у "дикому" середньовіччі, хоча можна зрозуміти платню за навчання чи словники. І тоді третьою важливою ознакою слід назвати відсутність обмеження на використання стандарту/формату - правило RAND (Reasonable and Non-Discriminatory).

Отже, відкрита технологія чи стандарт (на дані, протоколи обміну, формати даних) мають відповідати наступним умовам:

- бути затвердженими (сертифікованими) як міжнародний стандарт в одній з міжнародних або державних організацій по стандартизації (ISO, IEEE, OASIS);
- технічна документація має перебувати у публічному доступі;
- немає інших ліцензійних обмежень використання цієї технології чи стандарту.

Ліцензії на програми.

Ліцензія на використання програмного забезпечення це набір правил, що визначають умови використання програми. Це - набір свобод для користувача, які автор надає кінцевому користувачу. Деякі ліцензії можуть лише надавати дозвіл користуватись програмою, деякі дозволяють поділитись програмою з друзями чи покращити цю програму.

ПЗ з відкритим кодом.

Політика відкритого коду (англ. open source) - принцип <<розподіленої>> розробки, що використовується у створенні програмного забезпечення. Це означає, що джерельний код доступний для завантаження і вивчення, але ліцензія може забороняти використання коду за певних умов (наприклад, для комерційної діяльності).

Перевага open source в тому, що є можливість оцінити якість програми, недолік в тому, що наявність коду не завжди означає можливість використання ПЗ.

Безкоштовне ПЗ

Безкоштовне програмне забезпечення (freeware) -- це програмне забезпечення, яке можна безкоштовно використовувати протягом певного (можливо, необмеженого) терміну і поширюване без вихідних кодів. Автори такого програмного забезпечення зазвичай хочуть <<дати щось спільноті>>, але хочуть також контролювати його подальшу розробку. Перевага такого ПЗ в тому, що користувач не зобов'язаний платити за це. Недоліки в тому, що неможливо змінити програму згідно ваших вимог і, якщо автори вирішують припинити розробку, програма починає "вмирати".

Вільне ПЗ з відкритим кодом

Вільне Відкрите програмне забезпечення, або ВВПЗ (англ. FOSS, FLOSS -free/libre/open source software) - це узагальнюючий термін для позначення двох видів програмного забезпечення:

- ПЗ з відкритим кодом та ліцензією, яка дозволяє вільно використовувати програму з будь-якою метою ("нульова свобода");
- дозволяє вивчати, як програма працює, і адаптувати її для своїх цілей ("перша свобода");
- дозволяє вільно поширювати копії програми — на допомогу товаришу ("друга свобода") і дозволяє вільно покращувати програму і публікувати свою поліпшену версію ("третя свобода").

Власницьке ПЗ

Власницьке забезпечення (закрите ПЗ, пропрістарне ПЗ, від англ. proprietary software) - це програмне забезпечення, на яке зберігається право закритої (пропріетарної) інтелектуальної власності. Правові обмеження викладаються у дозволах (ліцензіях) на використання закритого ПЗ і можуть містити обмеження щодо копіювання (право на відтворення) та модифікації такого програмного забезпечення, заборонену на доступ до коду, заборону на внесення будь-яких змін, використання більш ніж на одному комп'ютері, тиражування і розповсюдження, перепродаж тощо. Таким чином користувач отримує лише обмежене право користування ним. Власницьке ПЗ може бути безкоштовне.

Література:

- <http://www.uafoos.org.ua/>
- <http://portal.rada.gov.ua/>

Розрахунок фотоіонізаційних моделей світіння небулярного газу від ОС LINUX UBUNTU 10.10 та WINDOWS 7

Мелех Б.Я., Тишко Н.Л., Коритко Р.І.

Кафедра астрофізики Львівського національного університету імені Івана Франка

Наведено результати тестування на швидкість розрахунку типових фотоіонізаційних моделей світіння (ФМС) небулярних об'єктів з різною складністю структури та вмісту елементів під 64-бітними версіями ОС Ubuntu 10.10 (Linux) та Microsoft Windows 7 Ultimate. ФМС розраховувалися за допомогою відкритої програми Cloudy, яка була попередньо відкомпільована під обома вище згаданими ОС за допомогою компілятора GNU C++ 4.5.1. Усі моделі швидше розраховуються під ОС Ubuntu 10.10.

Під час дослідження газопилових комплексів виникає потреба детального розрахунку переносу випромінювання у небулярному середовищі. Найчастіше в астрофізичних задачах необхідно розрахувати методом ФМС небулярного газу [1,2] іонізаційні структури зон НІ (іонізований водень), оболонок планетарних туманностей навколо поодиноких центральних зір, чи гігантських зон НІ навколо компактних областей зореутворення. Цей метод описує випромінювання небулярного газу під дією жорстких ультра-фіолетових квантів з ядра (зорі, чи скупчення зір). На сьогоднішній день актуальною є задача пошуку оптимальної ФМС окремих ділянок протяжного небулярного об'єкту, шляхом відтворення відповідних спостережуваних спектрів. У таких випадках працюють в рамках звичайних геометрій — сферично-симетричної, плоскопаралельної, циліндричної. Тут ми будемо використовувати плоскопаралельну геометрію. Для розрахунку ФМС більшість астрофізиків використовує відкриту програму Г. Ферланда Cloudy (<http://www.nublado.org>) [2]. Тому наші тести ми будемо здійснювати саме на цій програмі.

У даній праці ми поставили собі за мету порівняти швидкість розрахунку ФМС типових небулярних середовищ під x64 ОС Ubuntu 10.10 та Windows 7 Ultimate на одному і тому ж комп'ютері. Тестовий ПК має наступну конфігурацію: CPU: Intel Core i3 550, 3200 MHz; Chipset: Intel Ixeh Peak H55, Intel Ironlake; RAM: Patriot Memory 2 Гб DDR3-1333; HDD: 500.1GB, SATA300, 7200rpm, 16MB Cache. Для компіляції програми Cloudy~08.00 під обома ОС нами був використаний компілятор GNU C++ версії 4.5.1. Для визначення реальних часів розрахунку тестових ФМС під ОС Ubuntu 10.10 використовувалася системна програма time, а під ОС Windows 7— функція gettime.

У Таблиці 1 подана структура рівнів збіжності програми Cloudy [2]. Тут наведено перелік функцій - солверів кожного рівня збіжності разом із статусами відповідних фізичних величин, значення яких визначаються в ітеративний спосіб. В останньому стовпчику подано для кожного рівня збіжності назви тих фізичних величин, які на ньому визначаються.

Таблиця 1. Структура рівнів збіжності програми Cloudy.

Routine	Ionization OTS	Electron density	T_{kin}	Pressure	Trace keyword
ConvPresTempEdenIoniz	Stable	Stable	Stable	Solve	pressure
ConvTempEdenIoniz	Stable	Stable	Solve		temperature
ConvEdenIoniz	Stable	Solve			eden
ConvIoniz	Solve				ioniz
ConvBase	Drive				

У випадку плоско-паралельної геометрії небулярне середовище розбивається на тонкі шари. Зовнішньою межею моделі при цьому може бути іонізаційний фронт (так звані іонізаційно-обмежені моделі), або ж фізична межа туманності (так звані густинно-обмежені моделі). Кожна з тестових ФМС запускалася п'ять разів. У Таблиці 2 наведено усереднені часи розрахунку тестових ФМС під обома вище згаданими ОС.

Таблиця 2. Часи розрахунку ФМС різного рівня складності під ОС Ubuntu 10.10 та Windows 7.

№	Опис тесту	Усереднений час виконання (мс, якщо не вказано)			
		Ubuntu 10.10	Win- dows 7	Δ	δ , %
1	Однозонна модель, стала електронна температура; H, He, O.	773	1366	593	77
2	Іонізаційно-обмежена модель; H, He, O.	8779	9784	1005	11
3	Іонізаційно-обмежена модель; H, He, C, N, O, Ne, Si, Ar, Fe.	6165	7082	917	15
4	Іонізаційно-обмежена модель; 30 хімічних елементів	9264	9955	691	8
5	Густинно-обмежена модель; H, He, O.	1894	2521	627	33
6	Густинно-обмежена модель; H, He, C, N, O, Ne, Si, Ar, Fe.	2299	2942	643	28
7	Густинно-обмежена модель; 30 хімічних елементів	3839	4514	675	18
8	Пошук однозонної ОФМС, стала електронна температура; H, He, O; оптимізатор: Subplex.	7224	7718	494	7
9	Як 8, однак оптимізатор: Phymir.	5900	6448	548	9
10	Пошук іонізаційно-обмеженої ОФМС; H, He, O; о-р: Phymir.	8хв	8хв29с	29с	6
11	Пошук густинно-обмеженої ОФМС; H, He, O; о-р: Phymir.	2хв41с	2хв48с	7.2с.	5

Тести на швидкість розроблені нами в порядку зростання складності моделі. У тесті №1 розраховувалася найпростіша однозонна модель з заданою електронною температурою. При заданій концентрації газу тиск у такому випадку відомий. Отже, тут відсутня робота двох зовнішніх солверів, а однозонність виключає повторення запусків відповідних солверів для наступних шарів. У тестах №2—4 розраховуються іонізаційно-обмежені моделі з різною кількістю хімічних елементів. Збільшення кількості хімічних елементів має два наслідки: 1) збільшується кількість рівнянь (час розрахунку зростає); 2) зростає оптична товщина туманності, що

приводить до зменшення іонізованої області (час розрахунку зменшується). Як видно з Табл. 2, збільшення кількості хімічних елементів від 3-ох до 9-ти під обома ОС дало домінацію другого наслідку, а від 9-ти до 30-ти — першого. У тестах №5—7 розраховуються густинно-обмежені моделі. Ці моделі мають однакову товщину оболонки. Це приводить до монотонного зростання часу розрахунку. У тестах №1—7 спектр іонізуючого випромінювання задавався чорнотільним з ефективною температурою $T_{\text{eff}}=40000\text{K}$. Концентрація водню приймалася сталою і рівною $n(\text{H})=95826\text{ см}^{-3}$. Іонізаційний параметр, який тут визначає кількість іонізуючих квантів, що входять в туманність, приймалася рівним $\log\Phi = -3$. Хімічний склад задавався рівним Сонячному. У тестах №8—11 ми підключаємо алгоритм пошуку оптимальної ФМС (ОФМС). Параметрами підгонки є інтенсивності випромінювання в діагностичних емісійних лініях, а вільними параметрами — T_{eff} та $n(\text{H})$. У тестах №8—9 на прикладі пошуку ОФМС з найпростіших ФМС ми порівнюємо різні алгоритми оптимізації. Бачимо, що під обома ОС швидше працює алгоритм `Phumir[3]`. У тестах №10—11 порівнюються також часи пошуку ОФМС з густинно-іонізаційно-обмежених ФМС відповідно. Густинно-обмежені ФМС мають менші розміри (кількість шарів), а отже розраховуються швидше.

З Табл. 2 видно, що ОС Ubuntu у всіх тестах випереджує ОС Windows. Виграш 77% у першому тесті можна пояснити кращою продуктивністю файлової системи EXT4 у порівнянні з NTFS, оскільки тут час розрахунку (ЧР) є співмірним з часом завантаження тесту в пам'ять. Зі збільшенням ЧР “відставання” Windows від Ubuntu зменшується. Якщо пересортувати тести в порядку зростання ЧР, то буде видно, що зменшення δ із збільшенням ЧР носить монотонний характер у випадку нехтування даними тестів з пошуком ОФМС (№8—11). Однак, різниця часів виконання тестів №10 та №11 у випадку обох ОС — значна, що збільшує достовірність зміни характеру поведінки δ із збільшенням ЧР. З іншого боку, для остаточної впевненості необхідно розрахувати більш довготривалі тестові задачі з пошуку ОФМС, що ми й плануємо зробити найближчим часом разом з розширенням парку ОС, апаратної архітектури ПК та компіляторів.

Програма Cloudy широко використовується як базова під час наукових досліджень та при підготовці студентів-астрофізиків у різних вищих навчальних закладах світу. Тому отримані нами результати тестування можна розцінювати як рекомендаційні щодо вибору оптимальної ОС для інтенсивних та довготривалих розрахунків ФМС небулярних середовищ.

Література

1. Головатый В. В. Современная фотоионизационная модель свечения планетарной туманности. Краткое описание / Головатый В. В., Мальков Ю. Ф. — К., 1991. — 41с. (Препр./ АН УРСР Ін-т теор.фізики; 91-66Р).
2. Ferland G. J. Hazy, a Brief Introduction to Cloudy.—2008—450Р.(
3. P.A.M. van Hoof Photo-Ionization Studies of Nebulae. Ph.D. thesis (Rijksuniversiteit Groningen. — 1997).

RunaWFE - свободная система управления бизнес-процессами и административными регламентами

Михеев А. Г.

Консалтинговая группа РУНА Miheev_A@runa.ru

RunaWFE — открытая, масштабируемая, ориентированной на конечного пользователя система управления бизнес-процессами и административными регламентами. Система платформонезависима (написана на Java), распространяется под LGPL-лицензией, разрабатывается Консалтинговой группой РУНА. Система RunaWFE помещена в репозиторий свободных программ Sisyphus и включена в специализированный дистрибутив AltLinux Workflow.

Системы управления бизнес-процессами и административными регламентами.

Процессную организацию управления организацией используют для повышения эффективности работы организации. Эффективность повышается за счет оптимизации бизнес-процессов коммерческой организации или административных регламентов ведомства, появления возможности быстрого изменения бизнес-процессов в ответ на изменение условий деятельности предприятия, а также за счет повышения производительности труда работников..

Некоторое время назад выполнение бизнес-процессов в организациях производилось в основном косвенным образом - через изменение должностных инструкций, организационной структуры предприятия, прямые указания руководителей. Однако степень автоматизации современных предприятий позволяет реализовывать прямое выполнение бизнес-процессов в компьютерной среде. В этом случае в организации появляется аналог производственного конвейера, от которого можно получить увеличение производительности труда, сравнимое с тем, которое было получено от внедрения конвейера на производстве. Повышение производительности труда достигается вследствие того, что данный механизм позволяет работникам выполнять поступившие задачи, не отвлекаясь на:

- Получение от других работников необходимой для выполнения задания информации
- Передачу результатов своего труда другим работникам
- Изучение должностных инструкций

Все необходимое возникает перед работником на экране компьютера.

Для решения данной задачи разработан специальный класс компьютерных систем – системы управления бизнес-процессами и административными регламентами. Такие системы рассматривают бизнес-процессы и административные регламенты как потоки элементов работ. Выполнение процесса или регламента можно представить в виде перемещений точек управления потоков работ по определенным маршрутам между исполнителями в соответствии с заданными правилами. Последовательность

выполнения элементов работ определяется схемой процесса, которую можно разработать в специальном редакторе. В узлах схемы система управления бизнес-процессами и административными регламентами раздает задания исполнителям и контролирует их выполнение.

Пример экземпляра выполняющейся регламентной схемы:



К программному обеспечению, реализующему взаимодействие с государственными органами (системе управления бизнес-процессами и административными регламентами), разумно предъявить следующие требования: 1. Государственные организации должны иметь возможность использования неограниченного количества экземпляров программного обеспечения без увеличения расходов при увеличении количества экземпляров; 2. Граждане должны иметь возможность использования такого программного обеспечения бесплатно; 3. В случае, если разработчик программного обеспечения по каким-либо причинам перестал устраивать государство, государство должно иметь возможность сменить разработчика, не потеряв при этом программное обеспечение

Этим требованиям удовлетворяет свободное программное обеспечение с открытым кодом. В настоящее время существует несколько свободных систем управления бизнес-процессами и административными регламентами. Одна из них - система Runa WFE разработана в России Консалтинговой группой Руна. Эту систему можно бесплатно скачать через интернет вместе с документацией и исходными кодами с портала разработчиков свободного программного обеспечения sourceforge.

Описание системы RunaWFE

Основная задача системы: раздавать задания исполнителям. Последовательность заданий определяется схемой бизнес-процесса, который менеджер или бизнес-аналитик может быстро изменять при помощи редактора бизнес-процессов.

Система состоит из следующих компонентов:

- RunaWFE-сервер
- Графический редактор процессов
- Клиент-оповещатель о поступивших заданиях
- Функции компонента «RunaWFE-сервер»:

- Работа с определениями и экземплярами процессов
- Работа со списками заданий
- Визуализация форм, соответствующих заданиям
- Работа с системой через web-браузер.
- Предоставление возможности работы с системой приложениям специального вида - ботам. (В частности, боты могут моделировать работу сотрудника предприятия).
- Авторизация и аутентификация пользователей
Функции компонента «Графический редактор процессов»:
- Редактирование графа процесса
- Создание и редактирование графических форм заданий
- Создание и назначение ролей
- Создание переменных
Функции компонента «Клиент-оповещатель о поступивших заданиях»:
- Оповещение о поступивших заданиях
- Работа с системой через специальное приложение-клиент

Возможные последовательности заданий бизнес-процесса определяют направленный граф: множество узлов, соединенных между собой линиями со стрелками - возможными переходами. Узлы бизнес-процесса могут быть двух типов — узлы, соответствующие шагам процесса (узлы-действия), и маршрутные узлы, иногда в литературе эти узлы называются — «вентили». По переходам перемещается точка управления (указатель на активный узел процесса), руководствуясь правилами в маршрутных узлах.

В узле-действии система дает задание исполнителю (сотруднику или информационной системе) и ждет ответа (сообщения, что работа выполнена). После ответа исполнителя точка управления движется по переходу к следующему узлу процесса.

Маршрутный узел соответствует разветвлению или разделению-слиянию точек управления. В таких узлах система выбирает на основании содержащихся в маршрутных узлах правил следующий узел (узлы), в который будет передано управление.

В выполняющемся бизнес-процессе одновременно может быть несколько точек управления. В соответствии с бизнес-логикой процесса точка управления в маршрутном узле может разделиться на несколько точек управления, также точки управления могут ждать друг друга в другом маршрутном узле и далее слиться в одну точку управления.

Система RunaWFE поддерживает web-сервисы, может использоваться в виде портлетов, а также интегрирована с ECM-системой Alfresco.

Литература

1. OnLine demo системы RunaWFE : http://wf.runa.ru/Online_Demo
2. Проект RunaWFE на портале sourceforge: <http://sourceforge.net/projects/runawfe>
3. Русскоязычный сайт проекта RunaWFE: <http://wf.runa.ru/rus>

Використання OpenWrt як основи вбудованого програмного забезпечення маршрутизаторів

Нек Т.

Львівський національний університет імені Івана Франка, mr.taras1k@gmail.com

Розглянуто використання збірки OpenWRT як основи вбудованого програмного забезпечення маршрутизаторів.

OpenWRT— це дистрибутив для вбудованих систем, заснований на GNU/Linux. Обладнання, на якому встановлюють OpenWRT, відповідає концепції SOC (System-on-chip), за якою всі компоненти ПЕОМ інтегруються в одну мікросхему.

Поява OpenWRT, як і решти альтернативних прошивок, стала можливою через те, що багато виробників маршрутизаторів використовували вільне програмне забезпечення як базове, що в свою чергу змушувало їх публікувати свої напрацювання за аналогічними правами.

Однією із ключових особливостей OpenWRT є використання файлової системи JFFS2, яка має можливість запису на flash-накопичувачі (і спеціально створена для них). Ця файлова система враховує специфіку перезаписування флеш-пам'яті.

OpenWrt реалізує багато тих же функцій, які є в стандартних прошивках для маршрутизаторів, а саме - послуги DHCP і шифрування бездротового з'єднання за допомогою WEP, Wi-Fi Protected Access, або WPA2. Окрім того в OpenWRT реалізовано велику кількість функцій, які можуть бути відсутніми в стандартних прошивках. До них відносяться:

- переадресація портів обминаючи NAT всередині локальної мережі;
- UPnP для динамічної переадресації портів;
- розширені налаштування мережевого екрану і конфігурації маршрутизатора;
- QoS для таких додатків, як VoIP, онлайн-ігри, а також мультимедійні потоки;
- режими роботи маршрутизатора - повторювач бездротового зв'язку, бездротова точка доступу, бездротовий міст, або навіть поєднання всіх трьох режимів;
- використання динамічних DNS послуг для підтримки фіксованого доменного імені для постачальників Інтернет-послуг, які не надають статичну IP-адресу;
- доступ через SSH або Telnet;
- в маршрутизаторах з портами USB підтримується спільне використання пристроїв друку, обмін файлами у Windows-мережах (через SAMBA), USB-аудіо і практично будь-який інший пристрій, який можна під'єднати до USB-порта;
- моніторинг мережі в режимі реального часу;
- розширений Веб-інтерфейс з підтримкою AJAX;

На теперішній час існує три версії прошивок OpenWRT:

- OpenWRT 0.9 WhiteRussian;
- OpenWRT 8.09 Kamikaze;
- OpenWrt 10.03 BackFire.

Прошивка OpenWrt 10.03 BackFire записана на маршрутизатор D-LINK DIR 320, яка була попередньо налаштована для використання з цим маршрутизатором, за допомогою протоколу tftp. Маршрутизатор налаштований для використання в домашніх умовах, з метою забезпечення доступу до мережі Інтернет.

sK1 Project: Презавантаження (перспективи розвитку sK1 і UniConverter)

Новіков І.Є.

sK1 Project (<http://sk1project.org>), igor.e.novikov@gmail.com

Проект sK1 включає в себе векторний редактор sK1, орієнтований на професійне використання в поліграфії, і консольний конвертер векторних графічних форматів UniConverter. Редактор sK1 починався як форк проекту Skencil в 2003 р., оскільки автори Skencil зупинили розробку. На даний момент джерельному коду проекту вже більше 12 років. За 7 років незалежної розробки sK1 в проекті було реалізовано багато унікальних особливостей, завдяки яким проект став відомий широкому колу користувачів і отримав ряд міжнародних нагород. Але з огляду на те, що архітектура вихідного проекту Skencil розроблялася для набагато більш скромних цілей і досить давно, назріла необхідність кардинальної перебудови проекту.

За роки розвитку проекту кардинально змінилися його цілі: спочатку Skencil розроблявся, як простий векторний редактор зі спрощеною MVC (Model-View-Controller) архітектурою, в якій не передбачалася ні модульність, ні мультиплатформність. MVC компоненти були жорстко зав'язані один з іншим, що робило портування проекту вкрай важким.

У процесі розробки UniConverter'a на кодової базі sK1 нам вдалося частково розділити компоненти, оскільки до проекту були жорсткі вимоги в плані мультиплатформності. Але в результаті ми отримали два проекти (sK1 і UniConverter) з ідентичною кодовою базою, яку доводилося регулярно синхронізувати, що гальмує розробку. Подальший рефакторинг проектів дозволив виділити в загальний пакет найбільш важливі елементи - фільтри імпорту/експорту, рушій шрифтів і управління кольором. Це вирішило частково питання синхронізації, але загалом сильно розробку не спростила.

У процесі рефакторингу з'ясувалося, що в моделі документа і фільтрах імпорту/експорту спочатку закладені архітектурні недоліки. Тривалий аналіз коду, поряд з оглядом поточних змін у використовуваних інструментах і бібліотеках дав невтішну картину - проекту потрібно вкрай глибока переробка, щоб відповідати нинішньому стану речей.

Паралельно виникла необхідність зміни віджетсета. Skencil розроблявся в ті роки, коли Qt і Gtk + бібліотеки ще тільки починали розвиватися і тому інтерфейс програми створювався з використанням Tk віджетів, які на той момент вважалися вже стабільними і традиційними для додатків, написаних на Python. Але на жаль, Tk віджети досить сильно ізольовані від сучасних десктопних оточень і за довгі роки їх функціонал серйозно не розширився. Нам довелося дописувати безліч додаткових речей, щоб хоч якось забезпечити інтеграцію в Gnome і KDE: ці зміни вилилися в цілий додатковий пакет sk1sdk. І при цьому глибокої інтеграції так і не було досягнуто: програма мала інородні зовнішні вигляд і поведінку у всіх популярних графічних оболонках.

Спочатку було прийнято рішення перенести додаток на Gtk, оскільки на базі цих віджетів побудовані найбільш популярні графічні опенсорсні програми та Gnome є стандартним середовищем для найпопулярніших десктопних дистрибутивів: Ubuntu і Mint. Початкові розробки вже показали економічність додатків в плані витрат ресурсів і практично моментальний старт.

Подальший аналіз моделі виявив, що можна серйозно поліпшити показники споживання пам'яті. На даний момент sK1 в споживанні ресурсів RAM подібний до пропрієтарних додатків типу CorelDRAW і набагато більш економічний, ніж вільний редактор Inkscape. Але користувачі висувають більш серйозні запити, наприклад можливість роботи з об'єктами типу гільйошів (векторні сітки), які містять десятки тисяч компонентів. Розроблена нами нова модель документа дозволяє вкрай економічно витрачати пам'ять і вона набагато краще відповідає задуманій архітектурі програми: замість стандартного MVC патерну ми планує задіяти MVP (Model-View-Presenter), який дозволяє уникнути дуплікації коду в процесі виконання і спрощує організацію багатодокументного додатки.

У підсумку, на даному етапі проект фактично переписується з нуля. Але цей крок досить виважений і серйозно продуманий: нарощування функціоналу на старій кодової базі призведе до того, що зробити зміни в базових компонентах буде вкрай складно, а повноцінний автоматичний рефакторинг, як в мовах з суворою типізацією, в Python, на жаль недоступний.

І остання, найголовніша мета, яка буде реалізована після глибокої перебудови проекту - детальна переробка всіх фільтрів імпорту/експорту. Користувачів UniConvertor'a не цікавлять деталі імплементації моделі та інших модулів. Перш за все їх хвилює як можна більш повна підтримка різних форматів векторної графіки. І власне щоб спростити розробку і підтримку цих фільтрів, ми і затіяли весь цей процес. Спочатку фільтри імпорту/експорту створювалися під нескладні потокові формати. Але більшість сучасних форматів векторної графіки мають досить складну внутрішню структуру. Тому поточковий розбір таких форматів вкрай утруднений. На це також накладається той факт, що по суті в попередньому редакторі Skencil архітектура фільтрів розроблялася на основі процедурного патерну, що характерно для поточкових форматів. Ми плануємо перевести розбір форматів на об'єктні моделі типу DOM, що серйозно полегшує розробку і підтримку фільтрів імпорту/експорту, оскільки дозволяє фрагментувати імплементацію, уникаючи процедурного стилю «спагеті» (в якому через якийсь час, не те що сторонньому розробнику, але і навіть автору з власною працею важко розібратись).

Создание общедоступной образовательной среды школ РФ на основе свободного программного обеспечения, 2007-2010. Взгляд участника проекта.

Алексей Новодворский

«Альт Линукс», Москва, РФ.

В 2007-2010 годах в России был реализован проект разработки и внедрения комплекта свободного программного обеспечения для общеобразовательных учебных заведений. В то же время были проведены закупки лицензий проприетарного ПО на три года. 1 января 2011 оба проекта завершились и наступило время подведения первых итогов. «Альт Линукс» был одним из активных участников проекта, в докладе представлен взгляд на достигнутое и оценка перспектив.

Задачи и финансирование.

Работы по обоим проектам, получившим названия СБПО (Стандартный базовый пакет программного обеспечения) и ПСПО (Пакет свободного программного обеспечения) велись на основании распоряжения Правительства РФ №1447-р. На проект СБПО было потрачено 2 700 млн. руб. (около \$90 млн.), на проект ПСПО около 250 млн. руб. (около \$8,3 млн.) , из них 115 млн. руб. на обучение. Кроме того, в рамках создаваемого с 2005 г. хранилища электронных образовательных ресурсов (ЭОР) нового поколения (Федеральный центр информационных образовательных ресурсов, <http://fcior.edu.ru>) была предусмотрена разработка многоплатформенных ЭОР и средств их исполнения, на что было потрачено около 600 млн. руб (около \$20 млн.).

Реализация проекта ПСПО.

В начале реализации проекта ПСПО список его задач был существенно расширен. Комплект ПСПО, созданный в 2008 году включал:

- Три дистрибутива Linux для разных поколений компьютеров (128Mb , 256Mb, 1Gb памяти), а также дистрибутив «Нау Линукс» на основе Scientific Linux
- Терминал-сервер с низкими требованиями к терминалам (от 16Mb памяти)
- Репозиторий ПСПО для разработки и обновления пакета
- Сервер школы с веб-интерфейсом управления, общими сетевыми функциями, виртуализацией, средствами создания локального зеркала репозитория, системой дистанционного обучения Moodle и MediWiki. В 2009 и 2010 годах в инициативных разработка «Альт Линукс» функциональность сервера была расширена, в частности был включен электронных дневник и журнал РУЖЭЛЬ.
- Лекции и методические материалы по всем основным приложениям и системе в целом.

В 2010 году за счет финансирования государственно-частного фонда фирмой Etersoft была разработана бесплатная для школ система

Wine@Etersoft, позволяющая запускать бесплатную версию системы управления школой «1С:Хронограф». В конце того же года за счет бюджета был выпущен бесплатный плеер ЭОР для «Альт Линукс», проигрывающий сейчас более 5 тысяч ЭОР нового поколения.

За счет негосударственного финансирования и энтузиастами были разработаны:

- Дистрибутив Edumandriva с комплектом документации (Александр Казанцев, Глазов).
- Учебники информатики для 8-11 классов, ориентированный на использование многоплатформенных приложений и разных ОС (Угринович). Разработка еще двух учебников ведется.
- Многочисленные (сотни) свободных видеоуроков.
- Свободные книги по основным приложениям и ПСПО в целом («Альт Линукс»)
- Сотни методических разработок (методисты и учителя)
- Комплекты свободных программ для Microsoft® Windows™ (раные авторы).

Результаты.

По оценкам самых разных специалистов, разработанные свободные и бесплатные ПО и учебные материалы полностью удовлетворяют потребности преподавания информатики и по своему разнообразию, возможностям комплексного использования в учебном процессе и пресловутой стоимости владения превосходят СБППО. Удалось во многом решить проблему с документацией и методматериалами, постепенно решается проблема учебников. Без Windows™ в классе информатики можно обойтись, что признано как руководителями образования, так и многими учителями.

При помощи «Школьного сервера» решена проблема организации сетевой среды учебного заведения, а также его образовательной инфраструктуры. По сути, ранее такой типовой общедоступной интеграции сетевых ресурсов в школах не было.

Проблема использования компьютеров на уроках по разным предметам называлась многими экспертами главным основанием необходимости Windows™. Использование ЭОР во многом ее решает, однако появившиеся в самом конце 2010 года возможности еще недостаточно известны. Мешает также отсутствие дистрибьюции ЭОР на физических носителях, — не все школы имеют достаточную пропускную возможность сети для их простого использования.

С появлением доступных средств для запуска системы управления школой «1С:Хронограф», а также свободного дневника и журнала РУЖЭЛЬ, стало возможным решение организационно-административных задач учебного заведения.

На основе разработанных свободных курсов региональные институты повышения квалификации включают в свои программы изучения СПО и внедрение его в учебный процесс.

Работает централизованная служба технической поддержки, а также службы техподдержки в некоторых регионах.

Таким образом, все основные компоненты общедоступной образовательной среды школ РФ на основе свободного программного обеспечения созданы и прошли апробацию.

Проблема 01.01.2011

Первого января 2011 года истек срок лицензий СБППО. Федеральные власти еще в 2008 году объявили об отказе от дальнейшего финансирования закупки лицензий, возложив решение на субъекты федерации, а многие из них -- на муниципалитеты и даже школы. Предложения Microsoft были заманчивы по цене – \$8 за Windows и Office в год на компьютер, но при условии покупки лицензий на все компьютеры школ региона.

К настоящему времени менее 10 регионов (из 83) приняли решение о закупках ПО Microsoft, другое проприетарное ПО, входившее в СБППО, почти не закупается в расчете на использование свободных аналогов, в том числе для Windows.

Текущее положение дел будет подробнее освещено в докладе.

Перспективы.

- Поощрение учителей и школ, внедряющих СПО, в частности за экономию бюджетных средств.
- Создание информационных ресурсов, обеспечивающих размещение, контроль качества и удобный доступ ко всем элементам образовательной среды на основе СПО.
- Развитие региональных центров СПО на основе университетов и профильных фирм.
- Развитие разработки СПО, в частности в рамках проекта «Национальной программной платформы»

Заключение

Проект ПСПО сопровождался многочисленными публичными скандалами, расследованиями и отставками. Мы не освещаем их в докладе, так как даже несмотря на весь негатив, проект можно признать успешным и даже для очень многих – неожиданно успешным.

Результаты были бы еще лучше, если бы в ходе работ 2009-2010 годов не была во многом утеряна его социальная составляющая, замененная на административное давление.

Програмний комплекс моделювання процесу функціонування космічних систем

Петрожалко В. В., Фриз С. П.

Житомирський військовий інститут ім. С. П. Корольова Національного авіаційного університету, petrozhalco@mail.ru

У доповіді висвітлюються проблемні питання, які виникають при вивченні складних космічних дисциплін. Наводяться приклади програмних продуктів, що використовуються у навчанні, висвітлюються їх недоліки та запропоновано напрями їх вдосконалення.

Відповідно до «Загальнодержавної цільової науково-технічної програми України на 2008 – 2012 роки», космічна діяльність в Україні є однією з пріоритетних галузей народного господарства. Основу такої діяльності складають теоретичні дослідження, розробка, виготовлення та застосування за призначенням (експлуатація) космічних засобів. Розуміння фізичних процесів, що відбуваються на усіх цих етапах є досить важливим як для інженерів, які експлуатують так і для конструкторів, які розробляють космічні системи (КС). Особливе місце в цьому належить аналізу функціонуванню космічного апарата (КА) на усіх етапах життєвого циклу. Певну складність має засвоєння закономірностей руху КА. Науковою основою для його вивчення є теорія польоту КА, яка має ще й інші назви, наприклад, динаміка польоту, механіка польоту, космодинаміка.

У Житомирському військовому інституті ім. С.П. Корольова Національного авіаційного університету (ЖВІ НАУ) вже понад 30 років готують фахівців для експлуатації КС і окреме місце в цій підготовці займає вивчення дисципліни «Теоретичні основи польоту КА». Крім того, в інституті поряд із підготовкою вітчизняних спеціалістів уже декілька останніх років йде підготовка і перепідготовка іноземних слухачів з питань супутникових систем фотографічного, оптико-електронного і радіолокаційного спостереження. На методіку навчання цих слухачів впливає той факт, що вони зазвичай володіють українською (російською) мовами лише на початковому рівні, мають недостатню технічну підготовку. До того ж, щоб описати всі процеси, які відбуваються під час орбітального польоту КА потрібно застосовувати досить складний понятійний та математичний апарат. Тому розробка такого програмного комплексу (ПК), за допомогою якого можливо моделювати різноманітні процеси, що відбуваються в КС, а також наглядно демонструвати зв'язок математичних виразів із реальними фізичними величинами змодельованих за допомогою ЕОМ, є особливо важливим для підготовки як іноземних так і вітчизняних спеціалістів в космічній галузі.

На теперішній час існують вільні програмні засоби, які в певній мірі здатні відображати трасу та параметри орбітального руху КА. Із тих ПК, які вільно розповсюджуються, доцільно відмітити Orbiron, WX-track та ін. Але їхні можливості дуже обмежені в напрямку моделювання процесу

функціонування усіх елементів КС, відображення їх взаємовпливу та взаємозалежності. Математичний апарат, що використовується в даних програмах, зазвичай, не показується, і тому його дослідження або заміна з метою експерименту або вдосконалення неможливо.

Отже, ПК моделювання процесу функціонування космічних систем, який забезпечить краще розуміння та засвоєння курсантами, слухачами та студентами навчальної програми з космічних дисциплін, а також дасть можливість проводити дослідження процесів, що відбуваються в елементах КС є необхідним інструментом у навчанні спеціалістів за різними спеціалізаціями. Пропозиції розробників програмних продуктів у даному напрямі не залишаться поза увагою у ЖВІ НАУ.

Використання вільного програмного забезпечення у професійній підготовці майбутніх інженерів

Покришень Д. А., Дрозд О. П., Сподаренко І.Й.

Чернігівський державний технологічний університет, кафедра вищої та прикладної математики секція прикладної інформатики pokryshen@ukr.net

Розглядаються передумови впровадження вільних програмних засобів у навчально-виховний процес майбутніх інженерів. Зроблено огляд деяких вільних програмних продуктів, які можуть бути використані у професійній підготовці інженерів.

Проблемою у використанні комп'ютерних програм є вирішення правових та фінансових проблем, пов'язаних з придбанням і легальним використанням програмного забезпечення, в тому числі і операційної системи (ОС) для якої відбувається надбудова прикладного програмного забезпечення (ПЗ).

Сучасне масове програмне забезпечення не задовольняє педагогів та користувачів за великою кількістю причин, до яких можна віднести: відсутність україномовного (різномовного) інтерфейсу, невідповідність апаратної частини вимогам програм. Перепоною у використанні того чи іншого ПЗ стає і його платність.

Немає потреби говорити, що правильний вибір операційної системи потребує ретельного зважування всіх "за" і "проти". Адже помилка в цьому питанні призводить до невиправданих витрат на розробку і впровадження прикладних програм, експлуатацію самої обчислювальної системи і її зв'язок з іншими обчислювальними системами і мережами. Для школи, вузу чи іншого навчального закладу вибір операційної системи є дуже важливим фактором ще й тому, що від цього залежить, які вміння і навички отримають учні і як вони зможуть їх застосувати в подальшому.

При виборі ОС слід враховувати наступні фактори: підтримка великої кількості апаратних засобів; наявність необхідного прикладного ПЗ, можливість використання вже існуючого ПЗ; відповідність стандартам; антивірусний захист; захист системних файлів та файлів користувачів; технічна підтримка; ліцензійна чистота ОС та ПЗ.

З вищесказаного випливає необхідність використання досить стабільної, невибагливої та бажано безкоштовної системи.

Після вибору ОС для використання у навчальному процесі, виникає необхідність визначення критеріїв, на які слід опиратись при виборі прикладного ПЗ: методична доцільність, інтуїтивно-зрозумілий інтерфейс, україномовний інтерфейс, апаратна сумісність, програмна сумісність, ліцензійна чистота.

Сучасні інформаційні технології розвиваються у кількох напрямках. З метою встановлення найбільш перспективних та очікуваних у майбутньому напрямків поширення ІТ проводиться аналіз використовуваного програмного забезпечення, публікацій в періодичних виданнях та Інтернет,

нових програмних продуктів, оголошених напрямків досліджень провідних фірм інформаційної індустрії. Проведений аналіз дозволяє зробити висновки, що сьогоденні студенти будуть досить широко використовувати: мережеві технології; Інтернет-технології: електронну пошту, електронне справочництво, проведення конференцій, соціальні мережі; технології баз даних і доступу до них; машинезалежні технології доступу та опрацювання даних — програмування на Java, мультимедіа та гіпермедіа технології, HTML5, WebGL; системи символного перетворення та середовища автоматизованого проектування.

Проведений аналіз сучасних ОС показав доцільність використання дистрибутивів Linux, що відповідає всім вищезгаданам критеріям та задовольнить потреби сучасного студента.

Для підтримки професійної підготовки майбутніх інженерів на основі ВПЗ можна використовувати у навчальному процесі програмне забезпечення різного призначення: Free Pascal, Maxima, QCAD.

Free Pascal – це вільно розповсюджуваний компілятор мови програмування Pascal. Крім повної сумісності з Borland Pascal 7 та Object Pascal – Delphi у ньому є і додаткові функції. Є можливість його використання у різних ОС, наприклад, дистрибутивах Linux, MacOS X, Windows.

До програм символного перетворення можна віднести ПЗ Maxima. Даний програмний продукт цікавий тим, що його можна використовувати як альтернативу комерційним системам символного перетворення Maple і Mathematica. Крім числових розрахунків також можна проводити аналітичні перетворення та будувати графіки функцій на площині або об'ємі, диференціювання, інтегрування, розрахунки з матрицями та багато іншого. ПЗ Maxima може використовуватись у різних ОС: дистрибутивах Linux, MacOS X, Windows.

Символьні перетворення можна реалізувати і за допомогою програмного засобу SMath Studio. Даний програмний продукт може використовуватись у різних ОС для звичайних комп'ютерів та ноутбуків, а також для кишенькових ПК та смартфонів. Для використання SMath Studio необхідно встановити для Windows – .NET Framework, для Linux – Mono.

Збереження створених документів у форматі HTML дозволяє досягти їх універсальності. Створені на КПК документи можна відкривати та редагувати на звичайних комп'ютерах і навпаки.

Для початку роботи з даним програмним продуктом не потрібно мати спеціальних знань, тому що він має інтуїтивно зрозумілий інтерфейс та реалізований різними мовами (більше 20 мов).

До недоліків можна віднести: відсутність побудови графіка функції за заданими точками, відсутність роботи з масивами, обчислення тільки визначених інтегралів. Також до недоліків можна віднести відсутність методичної та навчальної літератури.

Крім мультиплатформенності та наявності української мови у інтерфейсі програми до переваг використання саме SMath Studio можна віднести: невибагливість до якості екрану (240x240 пікселів), побудова графіка функції на площині та у просторі, числові та символні перетворе-

ння, можливість побудови циклічних та розгалужених конструкцій, використання комплексних чисел, редагування документів Mathcad, наявність довідника з математики.

Серед вільного програмного забезпечення, як альтернатива Mathcad можуть бути використані і інші програмні продукти, такі як FreeCAD, PythonCAD, QCAD, Varkon, Linuxcad, Varicad, Cucas, Tomcad, Thancad, Fandango, Lignumcad.

Розглянемо ще декілька програмних продуктів, призначених для інженерів і розроблених для мобільних платформ, а саме для кишенькових комп'ютерів та смартфонів.

PocketProfile Lite та Pocket Zhelezobeton працюють на Pocket PC, WM, Windows Mobile 2003-6.0. Для використання їх на Windows Mobile 2003 SE необхідно встановити Microsoft .Net Compact Framework. Дані програмні продукти розповсюджуються безкоштовно.

Програмний продукт PocketProfile Lite використовується для розрахунку геометричних характеристик поперечних перерізів сортаментів металопрокату на КПК (момент інерції, опір та інше), а також їх перегляд.

Програма Pocket Zhelezobeton призначена для перегляду типорозмірів та інших характеристик залізобетонних конструкцій, що використовуються у цивільному та промисловому будівництві.

Для вирішення проблем, пов'язаних з роботою у операційній системі та вибором апаратного засобу (домашній комп'ютер, ноутбук, нетбук, планшет, кишеньковий комп'ютер, смартфони на різних ОС), на нашу думку більш доцільно розробляти програмні засоби з веб-інтерфейсом (наприклад, таких як Google Docs). Це надасть можливість використовувати програмний продукт незалежно від вибору програмної та апаратної частини.

Використання вільного програмного забезпечення у професійній підготовці майбутніх інженерів дозволить збільшити зацікавленість у навчанні, збільшити рівень інформаційної культури, дозволить самостійно обирати програмний засіб для подальшої трудової діяльності, виховати законослухняного громадянина.

Інтелектуальна власність, ввз та світова економіка ноосферне та індустріальне виробництво: конкурентні підходи

Радченко Ю.А.

*пошукач Інституту економіки та прогнозування НАНУ, координатор УАВВІЗ,
Київ. yuriradchenko@gmail.com*

На прикладі моделей розповсюдження "пропріетарної" та "вільної/відкритої" інтелектуальної власності в доповіді проаналізовано структурні відмінності між індустріальним та постіндустріальним виробництвом, процес сучасного становлення технічного та технологічного рівня ноосфери. Аналізуються відмінності між індустріальним та мережевим (ноосферним) способами виробництва та економічні зміни, що відбуваються у зв'язку з "вивантаженням" корпораціями до мережі частини виробничих ланок. Запропоновано підходи до необхідної зміни математичного апарату індустріальної економіки, який стає все менш релевантним у постіндустріальних умовах, а також ряд нових термінів у сфері економіки інтелектуальної власності.

Термін "Нова економіка" щільно увійшов до наукового та аналітичного вжитку, репрезентуючи економіку інформаційну, інноваційну, цифрову – тобто практично економіку інтелектуальної власності (далі також "ІВ") в цілому. Феномени "нової економіки" розглядаються зазвичай лише як нові види специфічних товарів ("об'єктів інтелектуальної власності") – в той час, як належної уваги (принаймні, у вітчизняних працях) не приділяється тому важливому фактові, що для самого виникнення цих нових товарів необхідною як правило є і абсолютно нова структура, топологія виробництва.

У цій невідповідності якраз і полягає марність зусиль здобути "інноваційну" (читай – "постіндустріальну") економіку, нічого не змінюючи в економічному і соціальному базисі країни. Виробити інтелектуально-насичений, тобто постіндустріальний за типом, товар у рамках суто-індустріальної виробничої структури – важко або й неможливо через ряд системних протиріч у підходах на всіх етапах розробки, створення і серійного випуску продукту.

По-перше, це протиріччя ітераційності (постійного, збагаченого новими даними повернення до початкової точки процесу або окремого блоку розробки продукту), що є основною умовою виробництва нових продуктів ІВ – і конвеєрності (тобто жорстко заданого технологічно і незворотного руху виготовлення) для індустріальних продуктів. По-друге, товар індустріального виробництва є товаром, як правило не пристосованим до модифікації – в той час як товар постіндустріальний часто має набагато більш складне життя, і за певних умов може змінювати навіть базові свої характеристики. По-третє, індустріальний товар є предметним, потребує для розповсюдження спеціально організованої логістики. Натомість функція логістики, розповсюдження – є апіорно "вбудованою" у саму суть товару постіндустріального.

Ці протиріччя є найбільш наочними. Високотехнологічна промисловість Радянського Союзу, зокрема в ІТ сфері, програла Заходу не так тому, що не мала відповідних технологій – як тому, що не змогла перейти від індустріально-конвеєрного до ітераційного, або хоча б до змішаного процесу виробництва.

Кожному з типів виробництва – індустріальному і постіндустріальному – явно (хоча й не виключно!) відповідають і преференційні форми власності: жорстко-"пропрієтарна" в індустріальному світі і вільна/відкрита у світі постіндустріальному.

Аналізуючи реалії "Нової економіки", що народжується на наших очах – ми повинні сконцентрувати увагу на особливостях не лише самого товару з високим вмістом інтелектуальної власності, а і на специфічних рисах структури і топології його виробництва; і на специфіці технологічної бази, яка це виробництво уможливує; і на тих соціальних аспектах, які "зв'язують" всі інші фактори до єдиної працюючої системи. З огляду на вищесказане, нам видається доволі важливим чітко акцентувати як відмінності у виробничих системах індустріальної та постіндустріальної доби, так і визначити, як саме залежать від обраної виробничої системи форми власності на кінцевий продукт.

Питання "боротьби" між альтернативними способами виробництва, індустріальним та постіндустріальним, неправомірно зведене до "боротьби" між "пропрієтарною" та "вільною" моделлю інтелектуальної власності. Тим не менше суперечка щодо "первородства" того чи іншого виду інтелектуальної власності не є конструктивною. Набагато більш важливим є питання, хто і за який кошт "поставив на ноги" високотехнологічну промисловість, заклавши технологічний фундамент постіндустріального суспільства. В цьому сенсі величезною є роль державного сектору – який в Україні, на відміну від розвинутих зарубіжних країн [2], нажаль, вже довгі роки залишається або абсолютно пасивним, або навіть відкрито-ворожим до питань структурної модернізації виробництва.

Проте звична, "вертикальна" схема виробництва інтелектуально-насичених продуктів є на даний момент, по-перше, надзвичайно витратною (що в першу чергу залежить від необхідної для масштабних проектів кількості і вартості робочої сили) [2], і по-друге – не враховує нову (постіндустріальну) топологію. Альтернативний спосіб виробництва (який, насправді, далеко перевищив задані його першим офіційним "апологетом" Еріком Реймондом [3] рамки сфери програмного забезпечення, і який чудово вписується у класичні розробки В.І.Вернадського [4], його послідовників [5] і П.Т.де Шардена [6] – надає в цьому сенсі суттєвих переваг, що демонструють і приклади відповідних міжнародних проектів.

На відміну від "вертикального" (корпоративного) способу виробництва, який визначається чіткою ієрархією елементів та відповідним "вертикальним" плануванням – "горизонтальний" спосіб виробництва має іншу структуру, детерміновану структурою транспортного рівня ноосфери (мережі) як площини рівноправних та з'єднаних за однаковими технічними

правилами елементів. Оскільки рейтинг та авторитетність кожного "вузлу мережі" формується не згідно офіційного статусу або посади в національних рамках, а безпосередньо як результат визнання іншими фахівцями – виробничу структуру мережі в ідеалі можна представити як сітку рівноправних елементів, кожний з яких має можливість прямого контакту з кожним іншим. Таким чином в проектах, які формуються в умовах мережі, топологічно відсутній єдиний центр прийняття рішень. Це:

а) дозволяє виконувати проект в режимі "переключення компетенцій";

б) практично знімає обмеження на ресурс технічних потужностей та людський ресурс (що, однак, не означає, що проект не повинен мати відповідного бюджету).

Незважаючи на те, що реальна активність учасників мережі, згідно як практики, так і відомого дослідження Мішеля Бергмана [7], є незначною (і, вочевидь може бути обчислена за формулою $a_n = k(100 \times 0,25^n - 1)$, де a_n – рівень "піраміди" мережевої активності за Бергманом, а множник "k" (0,75 для галузі програмування) може до певної міри залежати від сфери виробництва) – цього відсотку достатньо, щоб "зняти" питання про критичний брак виробничих ресурсів. З іншого боку, можна очікувати поступове збільшення активного відсотку користувачів мережі з підвищенням загальної технічної і технологічної освіченості й підготовки.

Відповідно, "суто-вертикальної" виробничої структури на сьогодні на світовому ринку притримуються лише компанії, жорстко пов'язані із "речовим" та конвергентним (з превалюванням речової складової) сектором (включаючи, звісно, і сировинну галузь); інші йдуть шляхом диверсифікації.

Ця ситуація змушує нас зробити важливий висновок про конвергентність сучасної корпоративної структури, і про лише часткову придатність старих математичних і статистичних методів обчислення важливих для корпорацій економічних параметрів на сучасному етапі. На зміну їм можуть бути запропоновані новий погляд на структуру сучасної економіки (з точки зору застосованого способу виробництва), а також нові економетричні підходи до розрахунку що цілісних бізнес-процесів, що окремих економічних параметрів – як-от собівартості та ціни інтелектуально-насичених товарів, впливу "паразитарних" ринків, які з'являються від час продажу "юридичного повітря" (ліцензій) та стимуляції монополістичними компаніями "чорного" ринку в рекламних цілях, тощо.

Література:

1. Радченко Ю. Ноосферна модель виробництва проти індустріальної: на початку філогенетичного розгалуження / Ю.А.Радченко / Економіст, № 6 (284), с.4-9 // Київ, НДЦ інновацій та конкурентоспроможності, 2010.
2. Best Companies to Work For 2007. /CNN Money; Fortune // [Електрон. ресурс]. – режим доступу: <http://money.cnn.com/magazines/fortune/bestcompanies/2007/index.html>
3. Raymond Eric. The Cathedral and the Bazaar. / Eric Steven Raymond // [Електрон. ресурс]. – режим доступу: <http://catb.org/~esr/writings/cathe->

- dral-bazaar/cathedral-bazaar/ [Російський переклад: Електрон. ресурс] – <http://www.osp.ru/os/1999/09-10/177856>
4. Вернадский В. Несколько слов о ноосфере. / В.И. Вернадский // [Електрон. ресурс]. – режим доступу: <http://www.trypillya.kiev.ua/vernadskiy/noosf.htm>
5. Каракo П. Философия и методология науки: В.И. Вернадский. Учение о биосфере./ П.С.Каракo. // - Мн.:Экоперспектива, 2007. - 208 с ISBN 985-469-188-8.
6. Шарден Т. Феномен человека. / Тейяр де Шарден П. // М.: Наука, 1987.
7. Bergman M. The 'Lottery Syndrome' and Recent Open Source Statistics / Michael K. Bergman. // [Електрон. ресурс]. – режим доступу: <http://www.mkbergman.com/148/the-lottery-syndrome-and-recent-open-source-statistics/>
8. Радченко Ю. Реальна економіка об'єктів інтелектуальної власності: приведення базових розрахунків до постіндустріальних вимог. / Радченко Ю.А. // Зб. "Теоретичні і практичні аспекти економіки інтелектуальної власності", с. 77-82 / Маріуполь, видання ПДТУ, 2010.

Про досвід використання ОС LINUX у навчальному процесі Львівського національного медичного університету імені Данила Галицького

Риковський П.А.

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

В доповіді висвітлюється досвід і досліди з впровадження вільного програмного забезпечення у навчальному процесі Львівського національного медичного університету імені Данила Галицького, накопиченого за 10 років викладацьким та інженерним колективом кафедри медичної інформатики. Метою доповіді є заклик до обговорення проблем і завдань у становленні предмету інформатики для непрофільних вищих навчальних закладів.

Починаючи з 2001 року вивчення предмету “Медична інформатика” на кафедрі медичної інформатики Львівського національного медичного університету імені Данила Галицького проводиться в середовищі ОС Linux. Навчальна база – це 3 лабораторії загальною кількістю 30 р.м.с. Для формування робочих місць використовуються інсталяційні збірки SlackWare і Ubuntu. Протягом навчального року на базі цих лабораторій проходять курс медичної інформатики 17 груп медичного факультету (по 14 студентів у кожній), 11 груп стоматологічного (теж по 14 студентів), а також 5 груп студентів іноземного відділу (загалом 54 особи).

Для оволодіння основними методиками використання комп'ютерних технологій у медицині, структурування медичної інформації, основами роботи з базами даних, експертними системами, знайомство та вміння використовувати медичні інформаційні ресурси всесвітньої мережі Інтернет окрім засобів самої операційної системи (WEB–браузери (Mozilla, Konqueror, Links), поштові клієнти (Kmail, Pine), Gimp) використовується таке програмне забезпечення: StarOffice, Octave, GNUplot, GNUmed, 3DSlicer, IDE Kuzya, CodeBlocks.

Викладачі кафедри при хорошій підготовці з використання комп'ютерних систем самостійно, без проходження курсів перепідготовки, освоювали ОС Linux та супровідні аплікації протягом 1 – 3 тижнів. Зокрема, це стосувалось монтування зовнішніх пристроїв, відмінності у використанні офісних пакетів (кодування, конвертація та портування документів), друку, графічного та WEB–дизайну.

Адмініструванню ОС Linux характерні різноманітні засоби інсталяції та клонування, в тому числі мережні (Acronis, Clonezilla), що полегшує супровід та підтримку працездатності лабораторій. Щодо противірусного та спам-захисту – його повністю перекладено на вхідні сервери університету.

Використання вільного програмного забезпечення в навчанні і наукових дослідженнях у Львівському національному університеті імені Івана Франка

Апуневич С. Є., Злобін Г. Г., Рикалюк Р. Є., Шувар Р.

Львівський національний університет імені Івана Франка, rer@franko.lviv.ua

В доповіді аналізується стан використання вільного програмного забезпечення в навчанні та наукових дослідженнях у Львівському національному університеті імені Івана Франка

Використання вільного програмного забезпечення в ЛНУ імені Івана Франка можна розділити за такими напрямками:

- серверні застосування — Linux (Debian, Open SuSE), Unix FBSD;
- навчання — операційна система (Debian, Open SuSE), офісний пакет (OpenOffice), засоби програмування (gcc, Kuzya IDE, Qt Creator), математичні пакети (Octave, Labplot), технологія термінал-сервер;
- студентська наукова робота — операційна система (Debian, Open SuSE), офісний пакет (OpenOffice), засоби програмування (gcc, Kuzya IDE, Qt Creator), математичні пакети (Octave, Labplot), системи керування базами даних, емулятори апаратних засобів і операційних систем;
- наукові дослідження — операційна система (Debian, Open SuSE), офісний пакет (OpenOffice), засоби програмування (gcc, Kuzya IDE, Qt Creator), математичні пакети (Octave, Labplot), організація обчислювальних кластерів (Open SuSE, Pelican HPC).

Вільне програмне забезпечення для організації серверів застосовується практично у всіх підрозділах ЛНУ імені Івана Франка.

Термінал-серверну технологію використовують в двох корпусах факультету електроніки:

- головний корпус факультету — операційна система сервера і тонких клієнтів Linux OpenSuSE 11.2, термінал-сервер LTSP 5 (реалізація Kiwi-LTSP). Технічні характеристики обладнання — термінал-сервер ПЕОМ Phenom II X4 з 8 ГБ ОЗП, тонкі клієнти на бездисккових ПЕОМ класу Pentium II / Pentium III (500–700 МГц) з 256 МБ ОЗП;
- корпус факультету електроніки на вул. Ген. Тарнавського 107 — операційна система сервера і тонких клієнтів AspLinux 9.1, термінал-сервер LTSP 5 (реалізація Kiwi-LTSP). Технічні характеристики обладнання — термінал-сервер ПЕОМ AMD 800 МГц з 1.5 ГБ ОЗП, тонкі клієнти на бездисккових ПЕОМ класу Pentium II / Pentium III (500–700 МГц) з 128 МБ ОЗП.

В навчальній роботі вільне програмне забезпечення використовують на факультетах:

- електроніки (курси “Обчислювальна техніка і програмування”, “Системне програмування і операційні системи”, “Практична робота в ОС Linux”, “Програмування та алгоритмічні мови”, “Чисельні методи”,

“Математичні методи інформаційних технологій”, “Комп’ютерні мережі”, “Архітектура комп’ютерів”);

- прикладної математики (“Операційні системи”, “Комп’ютерні мережі”, “Багатокористувацькі комп’ютерні системи”);
- фізичному (спецкурси кафедри астрофізики);
- природничому коледжі (курси “Операційні системи”, “Мережі”).

Спектр вільних програмних засобів, які використовуються в студентській науковій роботі надзвичайно широкий — від офісного пакету OpenOffice.org.ukr до емулятора ОС Android. Окремо хочеться звернути увагу на роботу студентської групи користувачів Linux, яка працює на факультеті електроніки з лютого 2008 року. Ця група створена за ініціативою самих студентів, засідання проводяться щотижня (www.pllug.totalh.com). Тематика засідань визначається членами групи без втручання викладачів. Основним здобутком членів групи наразі є оболонка Kuzya IDE (www.kuzya.sf.net).

Вільне програмне забезпечення в наукових дослідженнях використовується на факультеті електроніки (паралельні обчислення — кафедри радіофізики, радіоелектронного матеріалознавства, фізики напівпровідників), фізичному факультеті (кафедра астрофізики [gcc]), в астрономічній обсерваторії (паралельні обчислення на кластерних системах, програмування мовами Fortran 77 та Fortran 90/95, C та C++ за допомогою компіляторів GCC та Intel Compiler і засобів розробки, наукова візуалізація за допомогою Gnuplot, обчислення в системах Octave та GNU R, підготовка тексту в TeXLive, системи керування телескопами у реальному часі, рестрація даних спостережень та їх обробка).

Помилки адресації в електронних мережах та методи їх коректування

Северин М.В.

Одеська національна академія зв'язку ім. О.С. Попова, *n_severin@ukr.net*

Розглянуто питання обробки помилок адресації в е-мережах. Запропоновано методи автоматичної корекції помилок в електронних адресах.

У мережі Інтернет досить поширені випадки помилкового набору електронної адреси. Причинами таких помилок можуть бути як неуважність користувача, так і труднощі запам'ятовування електронної адреси. Адже електронна адреса, сформована з букв латиниці та службових символів, може містити в собі набори символів, які є транслітераціями «іноземних» слів, скорочення слів, або взагалі неосмислені послідовності символів. Помилки такого роду у записах URL призводять до втрати потенційних клієнтів сайту, а при пересиланні листів (e-mail) - до недоотримання листа адресатові.

Дослідження проблеми «помилки набору» у мережі може, у певних випадках, містити і «комерційну складову». Так, у [1] розглядаються питання, пов'язані з залученням відвідувачів на сайт (розкручування сайту) за рахунок використання друкарських помилок при ручному наборі URL-адреси у браузері користувача. Для залучення «чужих» клієнтів на свій сайт здійснюється реєстрація «хибного» сайту. Тобто реєструється домен, близький за назвою до популярного домену (у якості імені домену вибирається запис, який містить найбільш ймовірну помилку при наборі згаданої назви). Такий метод отримав назву «туре-in-traffic», і його застосування найбільш характерне для відомих і легко запам'ятовуваних сайтів, введення адреси яких не викликає великих труднощів і тому часто виконується безпосередньо у рядку браузера.

Не менш розповсюджені і помилки запитів в пошукових системах. Вони також досліджуються, і за аналогією з помилками адресації використовуються для залучення клієнтів на свій сайт [2]. І хоча у запитах, на відміну від електронних адрес, використовується осмислена інформація, статистика пошукових систем свідчить про велику кількість орфографічних помилок або описок на зразок: А) спотворення символу (кількох символів), В) пропуску символу (ів), С) вставки «зайвого» символу (ів), D) переставлення кількох (часто - сусідніх) символів. Сьогодні у пошукових системах застосовуються заходи, спрямовані на усунення помилок подібного роду. Так, при введенні «спотвореного» тексту, система пропонує (підказує) користувачеві можливі варіанти «правильних» слів зі словника. Використання сервісу пошуку і виправлення помилок дає позитивні результати як у плані зниження навантаження на пошуковий сервер, так і в плані підвищення якості надавання ним послуг.

У представленій доповіді пропонуються методи корекції помилок для вирішення проблем з помилковим введенням електронних адрес. Ці

методи ґрунтуються на так званій «теорії подібності скінченних послідовностей» (ТПСкП) [3], у якій розроблені чисельні міри оцінки «подібності» символічних рядків, що характеризуються, зокрема, спотвореннями видів A) - D). Застосовуючи методи ТПСкП, веб-сервер, у випадках отримання помилкової URL-адреси, міг би виконати пошук «близької» (можливої) адреси і перенаправити запит на неї, або ж разом з повідомленням про відсутність шуканого сайту запропонувати список сайтів, адреса яких «близька» за написанням.

Такий підхід прийнятний і для електронної пошти. Наразі тут застосовують наступні прийоми обробки «неправильних» адрес. Агент передачі повідомлень МТА (Mail Transfer Agent) у випадках відсутності зазначеного відправником адресата формує повернений лист (bounce message). Це повідомлення відсилається назад відправникові, коли поштова скринька одержувача не існує або недоступна. (Існують різні шаблони таких листів, і вміст повернутого листа може відрізнятись у залежності від налаштувань поштового сервера). У тексті повернутого листа вказується текст помилки, адреса поштової скриньки, список кодів помилок, і причина, за якою лист не було доставлено. Також до bounce message може бути додано оригінал листа або деяка його частина.

Таким чином, відправлена користувачем термінова і важлива кореспонденція може не досягти адресата через помилку (помилки) у записі електронної адреси. А у разі незбереження ним копії листа - до його втрати. Якщо ж повторне відправлення листа можливе, його відправнику необхідно буде з'ясувати правильну адресу, і у разі складності такого з'ясування він може намагатися підібрати її методом перебору. Це призводить як до збільшення навантаження на мережу, так і до зниження якості надання послуг.

У традиційному поштовому зв'язку існує вимога прийняття всіх заходів щодо доставлення листа адресатові, у тому числі і щодо з'ясування нового місця перебування адресата. Електронна пошта є альтернативою традиційній, і якість надання нею сервісу має бути принаймні «не гірше». Враховуючи досвід, отриманий у традиційній пошті, перш ніж відправити e-mail-повідомлення про неможливість доставлення листа адресату, слід вжити всіх заходів із його «відшукання».

Широке поширення в мережі серверів, побудованих на Linux / Unix-платформах з використанням вільного програмного забезпечення [4], дає можливість розробникам не тільки вивчати вихідні коди, але й виконувати їх модифікацію. Найбільшу популярність серед МТА, що використовуються на Linux / Unix-платформах, набули: Sendmail, Postfix, Exim та Qmail. З урахуванням цього, можна запропонувати - як один з методів боротьби з помилками адресатів - впровадження у МТА модуля, що дозволяє виконувати корекцію помилкової або ж пошук «близької» адреси. У разі можливості однозначного визначення адресата, модуль дозволить виконати коригування адреси і доставлення листа. При неможливості однозначно визначити адресата у повернений лист може бути вміщено перелік «близьких» адрес. Автор розробляє модуль корекції

«помилкових» адрес для MTA Postfix та Qmail.

Розглянемо приклади оцінок близькості електронних адрес, отриманих при використанні методів ТПСкП. У якості «шаблону» візьмемо електронну адресу автора: n_severin@ukr.net. Оцінки близькості будуть виконані тільки для першої частини адреси (до символу @), оскільки ми тут вважаємо, що повідомлення знаходиться на сервері адресованого домену, і на поштового агента покладено тільки завдання пошуку адресата у своєму домені.

У таблиці справа від помилкової адреси показано значення ТПСкП-міри його «близькості» до правильної адреси.

nseverin	0.889	n_sveerin	0.889
m_severin	0.889	n_siverin	0.889
n-severin	0.889	n_sivirin	0.778
nv_severin	0.900	n_civerin	0.778

З прикладу видно, що незначні спотворення (помилки в одному з символів, перестановки символів) «мало» спотворюють електронну адресу. Встановивши поріг міри близькості, при якому адреси, «близькі» до правильної, припустимо вважати результатами «друкарських помилок», рівним 0.88, шість з восьми наведених вище адрес можна автоматично «виправити». Звичайно, на практиці всі адреси у наведеному списку могли б належати іншим користувачам мережі. Якщо така ситуація вважається високо-ймовірною, згадані шість адрес можна включити в bounce message. Отже, навіть у випадках неможливості або неприпустимості автоматичної корекції адреси, запропоновані методи роблять bounce message більш інформативним. Таким чином, застосування методів ТПСкП дозволить як підвищити якість роботи сервісу, так і знизити «помилковий» трафік мережі.

Література

1. С. Жарков [Кривые пальцы, приносящие трафик](http://www.linkz.ru/promotion/krivuee_palmztcue_prinosyascie_trafik.html) // URL: http://www.linkz.ru/promotion/krivuee_palmztcue_prinosyascie_trafik.html
2. С. Кокшаров Роль опечаток в SEO // URL: <http://devaka.ru/articles/seo-misprints>
3. Леоненко Л.Л., Поддубный Г.В. Теория подобия конечных последовательностей и ее приложения к распознаванию образов // Автоматика и телемеханика. - 1996. - №8. - с.119–131.
4. Security Lab Исследование: «Идеальный почтовый сервер» // URL: <http://www.securitylab.ru/analytics/309889.php>

Використання Скретч на уроках інформатики

Семенюк В.М.

НВК "Школа-садок "Софія", escuela@ukr.net

Розглянуто використання програмного середовища Скретч на уроках інформатики в середній школі. Представлено навчальний посібник, який допомагає освоїти принципи роботи з цим середовищем та базові поняття алгоритмів.

Інформатизація системи освіти як одна із ланок загального процесу розвитку суспільства, з одного боку, має на меті підвищення ефективності навчання завдяки розширенню обсягів інформації та вдосконалення методів її застосування, а з другого – спрямована на те, щоб користувачі могли застосувати інформаційні технології в особистій професійній діяльності та навчально-виховному процесі.

Однією із складових завдань курсу інформатики середньої школи є розвиток в учнів умінь самостійно опановувати та раціонально використовувати програмні засоби різного призначення, цілеспрямовано шукати й систематизувати інформацію, використовувати електронні засоби обміну даними. Розділ програми “Базові поняття програмування” [1] ставить завдання ознайомити учнів з такими фундаментальними поняттями інформатики, як алгоритм, програмний об’єкт, атрибути об’єкта тощо. Вивчаючи цю тему, учні не повинні власне програмувати, вони мають лише зрозуміти основні принципи, за якими функціонують комп’ютерні програми. Підкреслю, що автори дотримуються об’єктно-орієнтованої парадигми програмування, згідно з якою прототипом програми є не алгоритм, а об’єктне середовище, кероване подіями. Цей підхід визначив послідовність подання матеріалу в темі: лише після понять програми, об’єкту та події вводиться поняття алгоритму, як основи програмної логіки, що є однією зі складових програмного забезпечення.

Для реалізації цієї ідеї цілком добре підходить середовище програмування Скретч [3], яке дає змогу дітям створювати власні анімовані та інтерактивні програми, ігри, історії тощо. Цими творчими доробками можна обмінюватися дсередині міжнародного середовища, яке поступово формується в мережі Інтернет. Скретч базується на традиціях мови Лого і Лего. Можливо не всі в Україні знайомі з програмним середовищем АЛГО і його виконавцями Роботом і Черепашкою [2], але вже конструктори Лего знають всі.

У середовищі Скретч використовується метафора цеглинки Лего, з якої навіть найменші діти можуть зібрати прості конструкції. Але, почавши з малого, можна далі розвивати і розширювати своє уміння будувати і програмувати. Скретч створювався спеціально для того, щоб підлітки 10 - 16 років використовували його самостійно в мережі позашкільного навчання на базі комп’ютерних клубів, в містах і районах з низьким рівнем доходів. Користувачі можуть збирати свої програми-процедури з блоків так само,

як вони збирали конструкції з цеглинок Лего. З конструкцій і структур, що управляються, можуть бути зібрані різні агенти, що виконують прості інструкції і, отже, володіють певною поведінкою. Ці агенти можуть взаємодіяти між собою і відтворювати співтовариство в середовищі Скретч.

Для повноцінної роботи з цим середовищем та використання його функціональних можливостей був створений посібник «Керівництво для користувача середовища Скретч». Він містить розділи, що стосуються інтерфейсу програми, написання скриптів, створенню об'єктів та підключення аудіо чи відео. Розглянуто також технологію створення програм з використанням понять базових алгоритмів. Наведені приклади містять покрокові детальні інструкції, що стосуються роботи з графічним редактором, який вмонтований в дане середовище і дозволяє створювати користувачу власні об'єкти. В результаті виконання простих команд над цими об'єктами може складатися модель, в якій взаємодітимуть безліч об'єктів, наділених різними властивостями.

Література.

1. І.О. Завадський, Ю.О. Дорошенко, Ж.В. Потапова. Інформатика. Навчальна програма для учнів 10-11 класів загальноосвітніх навчальних закладів. Рівень стандарту. Лист МОН. 2010.
2. В.Петрів. Програмне забезпечення та методика вивчення програмування. Львів, 2003
3. <http://www.scratch.mit.edu> – сайт міжнародної спільноти Скретч.

Використання вільного програмного забезпечення для створення програми керування інформаційним автоматом

Злобін Г., Скляр В., Чмихало О., Шевчик В.

Львівський національний університет імені Івана Франка,
buoyantgrumbler@gmail.com

Система відображення мультимедійного контенту складається з апаратної та програмної частин і призначена для використання в операційній системі Linux.

Апаратна частина системи відображення мультимедійного контенту складається із системної плати з процесором, оперативною пам'яттю, жорстким магнітним диском, графічним контролером і дисплеєм та блоком живлення. Для керування системою призначені:

спецклавіатура, під'єднана до інтерфейсу Centronics;

підсистема дистанційного керування, яка складається із інфрачервоного приймача, під'єданого до інтерфейсу RS-232, та пульта дистанційного керування теле-радіоапаратурою. Блок-схема апаратної частини зображена на рис.1.



Рис.1. Апаратна частина системи відображення мультимедійного контенту

На рис. 2-3 зображені схеми спецклавіатури і ІЧ-приймача.

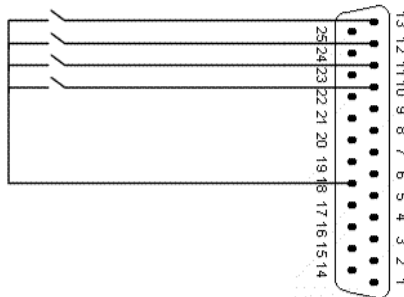


Рис. 2. Під'єднання спецклавіатури до інтерфейсу Centronics

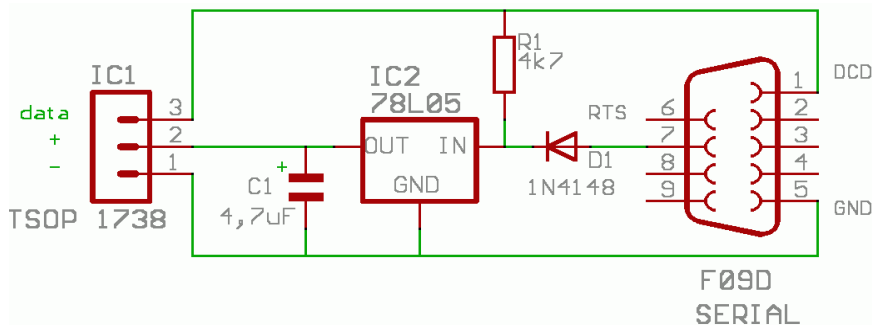


Рис. 3. Під'єднання ІЧ-приймача до інтерфейсу RS-232

Програмна частина системи відображення мультимедійного контенту складається із:

- операційної системи Linux;
- програвача відеофайлів Mplayer;
- переглядача pdf-файлів Poppler
- програми із відкритим кодом LIRC;
- програми керування відображенням мультимедійного контенту WonderBox , яка була розроблена учасниками проекту.

Блок-схема програмної частини системи відображення мультимедійного контенту зображена на рис.4.



Рис. 4. Блок-схема програмної частини системи відображення

Програма керування відображенням мультимедійного контенту — це універсальна програма, яка здійснює програвання відеороликів, відтворення графічних зображень та pdf-файлів. Програма відображення мультимедійного контенту написана мовою C++ з використанням бібліотеки Qt4. Інтерфейс програми складається з двох частин: вікна відтворення та вікна вибору файлів і каталогів (рис. 5). Вікно відтворення призначене для відображення вибраних користувачем відеороликів, графічних зображень, документів, та фонових відеокліпів. У випадку вибору користувачем документу чи відео, воно розкривається на всю площину екрану для зручного

їх перегляду Вікно вибору файлів і каталогів містить список, що дає змогу переглядати вміст теки з мультимедійними файлами, переміщуватись між вкладеними теками та вибрати файл для перегляду.



Рис. 5. Інтерфейс програми WonderBox



Рис. 6. Вікно вибору файлів і каталогів програми WonderBox

Керування програмою відбувається за допомогою чотирьох спеціалізованих клавіш: “Вперед”, “Назад”, “Відтворити/Пауза”, “Зупинити”. Також для переміщення по дереву файлів і каталогів, керування рівнем гучності та запуску/зупинки відтворення вибраних файлів можна використовувати пульт дистанційного керування. Відповідно до дій користувача, програма може перебувати у трьох режимах:

- режим вибору файлів і каталогів;
- режим перегляду;
- режим відтворення фонових відеороликів.

Розглянемо функціонування у кожному з них.

Після завантаження програми вона переходить у режим вибору файлів і каталогів. Користувач може вибирати мультимедійний файл, переміщаючи

курсор вибору по списку за допомогою клавіш “Вперед” та “Назад”. Для перегляду вибраного файлу необхідно натиснути “Відтворити/Пауза”. Після вибору мультимедійного файлу, який буде переглядатися, програма переходить у режим перегляду, в якому клавіші “Вперед” та “Назад” слугують для гортання документу або для перемотування відео вперед або назад. За допомогою клавіші “Відтворити/Пауза” можна призупинити відтворення відеоролика та знов відновити відтворення, натиснувши її повторно. Після натискання клавіші “Зупинити” або після закінчення відеоролика, програма повертається у режим вибору файлу.

Якщо протягом заданого проміжку часу не відбувається ніяких дій, спрямованих на вибір файлу зі списку, то програма автоматично переходить у режим відтворення у фоні. Тоді випадковим чином вибирається один з відеороликів, призначених для фонового відтворення, і запускається його перегляд. Після закінчення відтворення програма переходить до наступного випадкового ролика. Процес триває доти, доки користувач своєю дією не виведе програму з цього режиму. Для цього достатньо натиснути одну з клавіш переміщення по списку (“Вперед” або “Назад”). Натискання клавіші призведе до переходу у режим вибору файлів і каталогів.

Програма передбачає ведення журналу роботи системи. В цей журнал записується інформація про час запуску та завершення роботи, а також про час початку та кінця перегляду зображення, відеоролика або документа. Це може слугувати корисною інформацією для аналізу і побудови рейтингу файлів, котрі викликають найбільше зацікавлення користувачів даної системи.

Реалізація керування системою і перегляду мультимедійних файлів була спрощена, оскільки є доступними сторонні бібліотеки, компоненти і програмні засоби для більшості типових цілей.

До них належить Poppler, бібліотека для відображення Portable Document Format (PDF), яка лежить в основі цілого ряду широко вживаних програм перегляду в форматі PDF (таких як "Evince", "Okular", "ePDFView", "Vindaloo" та інші). Poppler - це відгалуження від Xpdf переглядача PDF-файлів, який поширюється під ліцензією GNU General Public License.

Для відтворення відеофайлів у даному проекті використовується MPlayer — програвач мультимедіа з відкритим вихідним кодом. Програма працює у більшості сучасних операційних систем. Вона не вимагає значних системних ресурсів, використовує власний набір бібліотек кодування медіа-форматів. MPlayer підтримує великий спектр медіа-форматів. Іншою важливою властивістю цього програвача є широка підтримка різних пристроїв виведення зображення таких як X11, Quartz Compositor, DirectX, VESA, SDL, VIDIX, DGA, Framebuffer, GGI, VDP AU включно з такими своєрідними пристроями виведення як ASCII-Art та Color ASCII-Art (тобто є можливість переглядати відео в текстовому режимі). MPlayer є консольною програмою, тому для здійснення керування програвачем було написано додатковий клас, який переводить Mplayer у режим Slave і керує ним. Виведення відеопотоків відбувається у вікні відображення системи,

що й забезпечується можливістю MPlayer'a використання різних пристроїв виведення.

Для реалізації дистанційного керування використана програма із відкритим кодом LIRC (Linux Infrared Remote Control). LIRC складається із модуля ядра і двох фонових системних процесів, один з яких декодує сигнали від пульта керування, які були прийняті інфрачервоним приймачем, і надає отриману інформацію через сокет про натиснені кнопки пульта дистанційного керування іншому. Другий фоновий процес, отримавши через сокет цю інформацію, виконує операції керування програмою відображення мультимедійного контенту (регулювання звуку, навігація по дереву файлів і каталогів тощо).

Тестування в Moodle як елемент менеджменту якості освіти: перший досвід

Сергієнко В.П., Сліпухіна І.А.

*Національний педагогічний університет ім.М.П.Драгоманова,
vpsergienko@npu.edu.ua*

На підставі практичного впровадження в навчальний процес коротко проведено огляд можливостей, переваг, недоліків і перспектив використання платформи Moodle у вищих навчальних закладах

Он-лайн освіта сьогодні — це система управління навчанням (СУН), яка потребує використання специфічного програмного забезпечення. Часто можна зустріти використання альтернативних термінів, таких як «віртуальне навчальне (або освітнє) середовище» або «система дистанційної освіти». Можна впевнено сказати, що всі студенти сьогодні широко використовують Інтернет, включаючи такі його можливості, як дискусійні форуми, чати, журнали, інструменти автоматизованого тестування, різноманітний он-лайн контроль і консультацій тощо.

Домінуючою і популярною серед академічних установ сьогодні є платформа з відкритим вихідним кодом Moodle, на базі якої сьогодні впроваджується СУН, зокрема, в НПУ ім.М.П.Драгоманова.

До очевидних переваг використання Moodle можна віднести такі:

- вартість володіння ліцензією постійно знижується;
- економія коштів може бути використана для розробки контенту;
- свобода від прив'язки до продажу конкретних виробів;
- Moodle - комунікації є надзвичайно ефективним ресурсом;
- можливе налаштування під свої потреби.

Як недоліки Moodle можна відзначити:

- отримати підтримку не так легко: форум містить великий обсяг інформації, яку іноді досить важко знайти
- сьогодні проблемою залишається також відсутність належної технічної підтримки з боку спеціально підготованих фахівців.

На платформі Moodle можна використати всі переваги веб-тестування:

- тестування виключає витрати і зусилля щодо друку завдань, їх розповсюдження, збору та перевірки контрольних робіт;
- значно полегшується відстеження успішності студентів і може бути створена її бази даних;
- є можливість створення звітів та отримання даних для дослідження ефективності методів навчання;
- висока надійність;
- задоволення потреб студентів в об'єктивності та ефективності тестування;

Оцінка ефективності навчального середовища є одним з найбільш ефективних заходів в освіті. Ґрунтовно розроблений тест може дати достатню

інформацію про роботу студентів. За наявності швидкого зворотного зв'язку тестування може значно поліпшити якість навчання, самотійну роботу студентів.

Тест модуль у Moodle має велику кількість опцій і утиліт, роблячи його максимально гнучким. Є можливість створення тестів з різними типами питань, випадково згенерувати тести зі списку питань, дозволити студентам повторити спробу або пройти тест кілька разів, і отримати відмітку щодо всіх спроб.

Серед усього спектру можливих тестових завдань у Moodle виділяються такі типи:

- питання множинного вибору: один або кілька варіантів відповіді;
- короткі відповіді на питання: слова або фрази;
- питання «Так/Ні»;
- питання на відповідність;
- випадкові питання;
- чисельні питання (з допустимим діапазоном значень);
- вбудовані питання-відповіді (закритий стиль);
- вбудовані фрагменти текстів і графіки.

Отже, чи є перспективним Moodle в якості СУН? Хоча на форумах можна зустріти достатньо критики програмного забезпечення з відкритим вихідним кодом, зокрема, думок, що ця платформа цікава тільки для ІТ-фахівців і є занадто важкою для основних (пересічних) користувачів в установці та використанні. Тим не менше, більше 66% Moodle - користувачів ідентифікували себе при реєстрації як вчителі й викладачі, дослідники он-лайн навчання чи адміністратори мереж навчальних закладів.

Одна з сильних сторін Moodle — спільнота, яка виросла навколо цього проекту. І розробники, і користувачі беруть участь у дискусійних форумах, обміні порадами, розміщенні фрагментів коду, допомагаючи новим користувачам. Низька вартість Moodle, гнучкість і простота використання допомагає тримати інформаційні технології в межах досяжності людей з обмеженими технічними і фінансовими ресурсами. І ще. Moodle не може існувати сам по собі. Якщо викладачі не відповідально, творчо ставляться до виробництва і дизайну контенту, створення он-лайн матеріалів, то Moodle, або будь-який інший подібний навчальний засіб, буде залишатися порожньою оболонкою, як гарний автомобіль, але без водія.

LINUX та VIRTUALBOX у навчанні абстрактних понять теорії операційних систем

Спірін О.М., Сверчевська О.С.

Житомирський державний університет імені Івана Франка, oleg.spirin@gmail.com, sverchevska@gmail.com

Визначено окремі методичні підходи до навчання абстрактних понять теорії операційних систем. Детально розглянуто методичний прийом, спрямований на підвищення ефективності засвоєння студентами таких абстрактних понять, як процес, потік, задача, адресний простір. Наведено приклади реалізації прийому під час проведення лекцій навчальних занять на основі використання вільно поширюваного програмного забезпечення Linux та VirtualBox.

Вивчення курсу "Операційні системи" ("Операційні системи та системне програмування") є важливою частиною підготовки майбутнього фахівця з інформатики. Вагоме значення мають не лише практичні навички, які студенти одержують у процесі проходження курсу, а й теоретичні знання, оскільки саме останні забезпечують глибину розуміння сенсу практичних дій та механізмів, що стоять за цими діями.

Досвід показує, що теоретичну частину курсу студенти нерідко засвоюють із певними труднощами. Це передусім пов'язано з потребою засвоєння великої кількості абстракцій (процес, потік, задача, контекст, адресний простір тощо). Ускладнює ситуацію ще й те, що вищезгадані абстракції належать до базових понять теорії операційних систем, тож відсутність у студента розуміння цих абстракцій чи недостатнє володіння ними спричиняє проблеми у подальшому вивченні курсу.

Варто враховувати також і особливості студентської аудиторії. Студентам, котрі обрали своїм фахом інформатику, часто властиве радше практичне, ніж теоретичне мислення. Під час навчання це нерідко виявляється в неналежному ставленні до необхідності опанування теоретичних знань та переоцінюванні значення лише практичних умінь і навичок. Наявність у студента таких установок може призводити до суперечливої ситуації: з одного боку, він зараховує себе до самоуків, практиків, прагматиків, а з іншого боку – далеко не завжди має достатньо наполегливості, терпіння і навичок самостійної роботи, аби відповідати цьому образу. Крім того, мотивувати такого студента до збалансованого навчання теорії та практики операційних систем надзвичайно важко.

Це породжує актуальну проблему удосконалення та розробки науково-обґрунтованих методик у навчанні майбутніх фахівців інформатики теорії операційних систем. Частковою проблемою є розробка окремих методичних прийомів, спрямованих на підвищення ефективності засвоєння студентами низки абстрактних понять теорії операційних систем з використанням вільно поширюваного програмного забезпечення. Мета дослідження – розробити методичний прийом для опанування студентами абстрактних понять процесу, потоку, задачі, адресного простору та описати використа-

ння цього прийому під час розгляду поняття "процес" на лекційному занятті.

Насамперед зазначимо, що для таких понять, як процес, неможливо забезпечити повну наочність. Проте повна наочність тут і не важлива, достатньо запропонувати студенту доступне зору підтвердження того, що речі, про які йдеться, насправді існують і використовуються, маючи сенс поза означеннями і схемами.

Завдання прийому – показати процеси наживо у реальній операційній системі, підкреслюючи тісний зв'язок матеріалу демонстрації з матеріалом, що розглядається на лекції.

Для реалізації прийому необхідно, аби на комп'ютері, з якого здійснюватиметься демонстрація, було встановлено програмний засіб для роботи з віртуальними машинами (наприклад, VirtualBox). За допомогою VirtualBox запусимо заздалегідь створену віртуальну машину з попередньо встановленою операційною системою (пропонуємо Linux, наприклад, Ubuntu Linux чи інший популярний безкоштовний дистрибутив [1], [2]). У межах віртуальної машини продемонструємо, де і яким чином у Linux можна спостерігати за роботою процесів. Для цього підходять як засоби графічної оболонки (системний монітор), так і засоби командного рядка (команди `top`, `ps` тощо). Те, що відбувається на комп'ютері викладача, транслюється для всієї аудиторії – проектором чи по локальній мережі.

Розглянемо особливості реалізації прийому на прикладі системного монітору. Засобами останнього доцільно показати студентам:

- появу нового процесу при запуску програми;
- інформацію про процес (ім'я, PID, стан, користувач тощо);
- ієрархію процесів;
- зупинку процесу;
- знищення процесу як наслідок закриття вікна програми;
- знищення процесу вручну тощо.

Зазначимо, що описана демонстрація не є поясненням до лабораторної роботи. Використання її покликане знайомити студентів з процесами як такими, а не із засобами маніпулювання ними. Тому коментарі на зразок "натискаємо праву кнопку", "вибираємо пункт меню", "переходимо на вкладку" тут краще звести до мінімуму.

Оскільки завантаження операційної системи потребує певного часу, варто запустити віртуальну машину заздалегідь і переконатися, що система успішно завантажилася і що успішно пройдено процедуру авторизацію. Крім того, реалізація запропонованого прийому потребує технічної бази достатньої потужності. Це, передусім, стосується комп'ютера, з якого здійснюється трансляція. Здатність наявного обладнання впоратися з поставленою задачею також краще перевірити заздалегідь.

Прокоментуємо вибір використаних програмних засобів.

Вибір ОС Linux не єдиний можливий. Проте використання за цих обставин ОС Windows вважаємо менш доцільним, оскільки тоді матимемо певні дидактичні обмеження. Зокрема, у Windows відсутня ієрархія про-

цесів, а тому показати зв'язки між батьківськими і дочірніми процесами не вдасться. Проте цілком виправданим був би вибір іншої unix-подібної ОС, наприклад, FreeBSD.

Вибір VirtualBox (не лише для реалізації даного прийому, а й для проведення лабораторних занять) пояснюється наступними факторами:

- належність VirtualBox до вільно поширюваного ПЗ (що, поряд із іншими свободами обставинами, означає його безкоштовність);
- популярність VirtualBox [2], [3];
- наявність у VirtualBox україномовного інтерфейсу.

Обираючи між засобами графічного інтерфейсу та командним рядком, варто також орієнтуватися на склад студентів: яка кількість їх вже працювали з Linux чи UNIX, чи багато з них мають досвід роботи з командним рядком. Якщо так, то є сенс замість системного монітору використати командний рядок. Якщо ні, то краще зупинитися на засобах графічного інтерфейсу. Обрати командний рядок можна і тоді, коли немає технічної можливості встановити на комп'ютер, з котрого вестиметься трансляція, Linux з графічним інтерфейсом.

Запропонований прийом може бути застосований і під час навчання інших абстракцій, а також для пропедевтики навчання інших операційних систем у межах лекційного курсу.

Література

1. Linux Distributions – Facts and Figures. – Режим доступу: <http://distrowatch.com/stats.php?section=popularity>.
2. Readers' Choice Awards 2010. – Режим доступу: <http://www.linuxjournal.com/content/readers-choice-awards-2010>.
3. Five Best Virtual Machine Applications. – Режим доступу: <http://lifehacker.com/5715803/best-virtual-machine-application-virtualbox>.

Стан українського FOSS-сегменту на кінець 2010 року

Сподарець Дмитро

Одеський національний університет імені І.І. Мечникова,
m31@root-ua.com www.m31.org.ua

В статті аналізується стан українського FOSS-сегменту на кінець 2010 року. Розглянуто такі напрямки: використання FOSS в освітньому та державному секторах; FOSS та бізнес; основні FOSS заходи в Україні; провідні українські web-ресурси FOSS тематики; спеціалізовані тематичні видання в Україні.

FOSS в державному секторі

23 грудня 2009 було підписано розпорядження Кабміну № 1588-Р «Про схвалення Концепції державної цільової науково-технічної програми використання в органах державної влади програмного забезпечення з відкритим кодом». Державний комітет інформатизації України (ДКІ) розгорнув активну діяльність за цією програмою і провів ряд круглих столів за участю провідних компаній і фахівців FOSS напрямків України. Але чергові вибори в країні і реорганізація ДКІ трохи загальмували, навіть заморозили даний процес.

5 липня 2010 був ліквідований ДКІ, а на його базі створено Державний комітет з питань науки, інновацій та інформатизації. 9 грудня 2010 року цей Комітет був реорганізований в Державне агентство з питань науки, інновацій та інформатизації України, діяльність якого координується Міністерством освіти та науки, молоді та спорту України. Будемо сподіватися, що в 2011 році нове агентство повернеться до втілення в життя програм, розпочатих ДКІ.

FOSS в освітньому секторі

Поки управлінські структури в державі зайняті реорганізаційними питаннями і налагодженням роботи, вчителі шкіл, викладачі ВНЗ та всі, хто пов'язаний з освітнім процесом, у навчальній роботі самостійно розпочали застосовувати вільне/відкрите програмне забезпечення. На підставі досвіду ентузіастів, після проведення низки конференцій FOSS Sea 2008 і OSDN Conference 2008, з'явився проект edu.root.ua - всеукраїнська ініціатива з використання вільного програмного забезпечення в освіті та науці. На базі Веб-сторінки, створеної ініціаторами, збираються різні методичні матеріали, здійснюється предметне спілкування, координація та працює розсилка - <https://groups.google.com/group/floss-in-ua>. На жаль, до кінця 2010 року ініціатива була фактично заморожена і тепер потребує нових сил. Відомо, що в 2010 році багато працівників наукової та освітньої галузі України самостійно та локально почали використовувати ВПЗ. Координатори ініціативи закликають їх «вийти з підпілля» і долучитися до розробки спільних стратегічних проектів. Ми всі зацікавлені в налагодженні нових зв'язків, обміні досвідом, проведенні спільних заходів та акцій...

FOSS і бізнес

В останні роки український бізнес поступово освоював вільне програмне забезпечення. І вже сьогодні багато компаній активно використовують його у своїй роботі: на робочих станціях і серверах, в телекомунікації і на своїх веб-ресурсах, для розробки власного внутрішнього ПО

В Україні вже працюють представники провідних дистрибутивів світу, серед яких можна виділити такі: ALT Linux - NetStyle, Media Magic; Ubuntu - UALinux; Novell і RedHat - Ustar, Softline, MUK, БАКОТЕК та інші. Більш розгорнуту інформацію щодо підтримки Linux в Україні можна знайти на Веб-сторінці - <http://linux.kiev.ua/support/>.

Основні FOSS-заходи України

Серед найбільш масштабних українських FOSS заходів можна виділити:

- OSDN Conference - проводиться щорічно восени у Києві conference.osdn.org.ua;
- FOSS Sea - проводиться щорічно влітку в Одесі foss-sea.org.ua;
- FOSS Lviv - проводиться щорічно взимку у Львові conference.linux.lviv.ua
- FOSS Fest - проводиться двічі на рік: влітку - на узбережжі Чорного моря, взимку - на одному з гірськолижних курортів Карпат foss-fest.com
Проводиться також ряд спеціалізованих, але більш локальних за масштабом заходів: конференція OpenKyiv (openkyiv.org.ua), конференція KyivBSD (kyivbsd.org.ua), конференція «Відкрите Програмне забезпечення в освіті, науці, бізнесі» (ulcc.org.ua/projects/conf2010/wiki), конференція WebCamp (webcamp.in.ua), Free and Open Source Software Conference (fossua.org), зустрічі одеської спільноти Odessa FOSS Meeting (foss-fest.com).

Провідні українські Веб-ресурси FOSS-тематики

Хоча в українському FOSS сегменті існує досить велика кількість дрібних веб-ресурсів, слід виділити основні: портали OSDN.org.ua і Root.UA, форум Linux.org.ua, соціальну мережу блогового виду OpenLife.org і сайт асоціація UAFOSS.org.ua.

Спеціалізовані тематичні видання в Україні

Офіційних видань FOSS-тематики в Україні два - журнал RootUA (root-ua.com) і газета FOSS News (foss-news.com). Нещодавно з'явився новий електронний журнал UserAndLINUX (ualinux.com). Слід назвати ще два електронні журнали, які сприяють розвитку російськомовної FOSS-спільноти: Full Circle (ubuntu.ru/fullcircle) і Open Source (osa.samag.ru).

Розробка системи спектральної діагностики димової плазми

Сподарець Д.В., Драган Г.С.

Одеський національний університет імені І.І. Мечнікова, m31@root-ua.com

Для розробки системи спектральної діагностики плазми використовувався комплекс TERM, який представляє собою набір модулів для обробки аналогових сигналів, отриманих із світлочутливої матриці. Комплекс TERM дозволяє реєструвати в реальному часі тепло- і електрофізичні параметри конденсованої та газової фази димової плазми, такі як концентрацію електронів і атомів, які легко іонізуються, в газовій фазі, температуру частинок і газу[1]. При необхідності розширення функціонального призначення системи, комплекс TERM дозволяє проводити діагностику як контактними, так і безконтактними методами.

Використання контактних методів для дослідження димової плазми призводить до значного впливу на її параметри. Це пов'язано з різного роду реакціями, які відбуваються на поверхні плазми, що контактує з вимірювальним датчиком. Тому актуальним завданням є розробка системи діагностики димової плазми з мінімальним її збуренням.

Базою для діагностичного комплексу був обраний атомно-абсорбційний спектрофотометр С-115, який працює в двох режимах: емісійному і абсорбційному. Перший режим дозволяє вимірювати інтенсивність випромінювання плазмового факела як функцію довжини хвилі, тобто зареєструвати власний спектр плазмового факела. Аналізуючи даний спектр, можна отримати температурні параметри газової та конденсованої фази плазмового факела. Другий - сумарну інтенсивність випромінювання лампи і плазмового факела як функцію довжини хвилі. У цьому випадку випромінювання лампи, проходячи через плазмовий факел, буде поглинатися, а за рахунок того, що лампа має тільки лінії в своєму спектрі, буде відбуватися абсорбція на певних довжинах хвиль. Наприклад, якщо взята калієва лампа, то буде відбуватися поглинання спектральних ліній калію, що дозволить визначити кількість атомів калію в плазмовому факелі.

В якості світлочутливої матриці в нашій роботі використовується Веб-камера. Після отримання спектру, Веб-камера передає відео-потік на обробку в комплекс TERM, де відбувається покадровий аналіз. Спочатку в кадрі виділяється робоча область, яка попіксельно аналізується. Результатом даного аналізу є матриці значення інтенсивностей за трьома кольорами: червоний (R), зелений (G), синій (B). Далі залежно від того, який використовувався в спектрофотометрі світлового фільтру, буде застосовуватися відповідна матриця значень інтенсивностей і відповідні алгоритми для розрахунку необхідних параметрів плазми.

Програмний комплекс базується на вільному / відкритому програмному забезпеченні, що дає можливість досить просто адаптувати, модернізувати і доповнювати його необхідним функціоналом під кожну конкретну задачу. Як базова платформа використовується операційна

система ALT Linux [2]. Основною мовою є C++, яка використовується у зв'язці з QT4, що дозволяє збирати необхідні модулі під різні платформи (як мобільні, так і настільні). Основною бібліотекою для роботи з відео потоком і первісним його аналізом є бібліотека комп'ютерного зору OpenCV [3].

Таким чином, використання відкритих технологій суттєво спрощує процес розробки і модернізації різного дослідницького обладнання і дозволяє вченим приділяти більше часу розробці фізичних теорій і їх експериментальній перевірці.

Література

1. Сподарец Д.В. Расширение возможностей аппаратно-программного комплекса TERM для исследования низкотемпературной плазмы // Сборник тезисов 7-й конференции разработчиков свободных программ, г. Переславль. М.: 2010. С. 81-85.
2. <http://www.altlinux.org>
3. <http://opencv.willowgarage.com>

Особливості програмного забезпечення в електронному навчанні

Жарких Ю.С., Лисоченко С.В., Сусь Б.Б., Третьак О.В.

Київський національний університет імені Тараса Шевченка, binsuse@gmail.com

Розглянуті переваги методів і технологій електронного навчання при застосуванні відкритого програмного забезпечення. Основну увагу приділено організації навчального процесу, створенню електронних курсів з природничих наук і дистанційного лабораторного практикуму. Використання відкритих кодів в програмному забезпеченні дає можливість впровадження електронних технологій в навчальний процес в стислі терміни, зміни і адаптації його контенту відповідно до вимог учбового закладу.

Актуальність проблеми.

Забезпечення належного рівня освіти потребує розвитку дидактичних систем, методів, методик і технологій навчання. Тому актуальними стають навчальні системи, побудовані на новітніх технологіях, що ґрунтуються на комп'ютерній техніці і досягненнях інформатики. Електронне навчання є важливим доповненням до традиційних способів навчання і поширюється на ті області освіти, де її функціонування неможливе або неефективне. Електронне навчання може бути корисним також для студентів стаціонарного навчання, оскільки в зв'язку з реформуванням освіти із врахуванням позицій Болонського процесу значна частина навчальної діяльності студентів зорієнтована на самостійну роботу. Однак самостійна навчальна діяльність студентів потребує відповідної організації і дидактичного забезпечення.

Для ефективної організації навчального процесу при дистанційній формі навчання існує велика кількість електронних систем для організації і управління навчальними курсами (LMS) як комерційних, так і на основі відкритого програмного коду.

Комерційні програмні продукти часто мають закриту структуру і в такому вигляді пропонуються користувачеві. Комерційна програма виконує функції, закладені власником програмного коду, а не користувача, тому якщо такий програмний продукт чимось не задовольняє потреб користувача, то він не в змозі що-небудь змінити і вдосконалити програму, хоча така потреба існує у зв'язку з розвитком навчальних планів і методик навчання. Таким чином, комерційний програмний продукт залишається ніби законсервованим до виходу наступної версії, коли можливі необхідні виправлення чи доповнення. Однак це не завжди можливо, тому з часом деякі комерційні продукти навіть втрачають підтримку (наприклад, система Learning Space). До того ж комерційні програмні продукти коштують досить дорого.

Програмні продукти на основі відкритого коду (open source) мають такі ж функціональні можливості, як і комерційні, однак можуть мати ряд переваг перед комерційними, оскільки вони постійно розвиваються зусиллями великої кількості програмістів з усього світу. Завдяки відкритості та доступності програмного коду їх легко аналізувати, змінювати,

вдосконалювати, модифікувати і таким чином адаптувати до потреб конкретного навчального закладу. Деякі з них стали навіть більш якісними, ніж комерційні. На основі відкритого програмного коду розроблена велика кількість електронних систем для організації і управління навчальним процесом. Важливо, що більшість програм з відкритим кодом є безплатними.

Слід зазначити, що використання відкритого програмного коду має й іншу позитивну сторону, оскільки заохочує студентів технічних факультетів до вивчення програмування. Створювати власний проект студентам важко, оскільки для цього необхідно навчитись добре складати програми, вивчити багато вихідних кодів та написати багато власних програм. Але вони можуть з цікавістю допрацьовувати чи вдосконалювати готові блоки системи.

Характеристика системи дистанційного навчання на основі відкритого коду.

Система повинна містити засоби організації навчання, наявність навчального матеріалу і можливість його подачі студентові, засоби контролю (тестування), надійної і зручної комунікації.

Універсальної технології створення електронних навчальних курсів ще немає, тому кожен розробник використовує свою технологію.

Як зазначається в [1,2] електронні навчальні курси можна поділити на 2 типи: з підтримкою LMS (Learning Management System – система керування навчанням) і без такої взаємодії. LMS виконують ряд функцій – функцію планування навчального процесу, поширення навчального матеріалу до студентів, функцію контролю (тестування) і комунікаційну функцію, яка здійснює зв'язок між викладачем та студентами. Якщо розробляється електронний навчальний з підтримкою LMS, то необхідно передбачити, які параметри будуть передаватися в базу даних системи LMS для подальшого аналізу викладачем (оцінки, затрачений час на виконання читання тексту, виконання завдання тощо).

Особливості створення електронного навчального курсу з фізики.

Опис фізичних явищ потребує, з одного боку, формалізації, моделювання, застосування математичного апарату, а з іншого – розвитку просторової уяви, відчуття динаміки фізичних процесів, чуттєвого сприйняття. У підручниках з фізики дуже часто використовуються графічні зображення, вони логічно вибудовуються, ускладнюються і врешті рисунки набувають досить складного вигляду, важкого до сприйняття у традиційному представленні. Комп'ютер дає можливість забезпечити поступовість викладу через послідовність окремих кадрів таким чином, що попередні кадри зберігаються, а наступні поступово ускладнюються, що важко зробити у друкованому варіанті через значне зростання обсягу підручника. Наприклад, при розгляді питання “Геометричне додавання коливань” необхідно згадати правила додавання векторів. На рис. 1 показана послідовність розвитку побудови рисунка через послідовне ускладнення наступних кадрів.

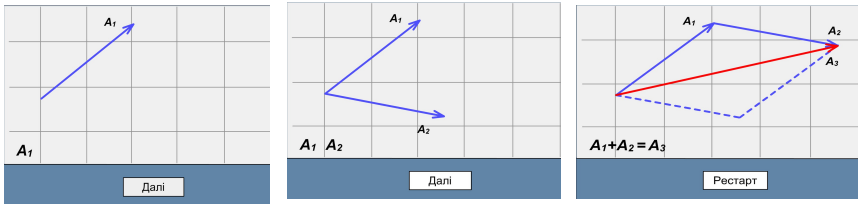


Рис.1

Для пояснення явищ, процесів в електронному варіанті курсу дуже легко можна використати різні кольори зображень ліній, елементів рисунка, продемонструвати динаміку процесу за допомогою кліпу.

Подібним чином можна виконувати також складні доведення за допомогою формул. При виведенні формули важливо дотримуватися принципу поступовості розгляду. Є можливість наочно демонструвати заміни величин у формулі шляхом перенесення, зміною кольору тощо. З врахуванням цих особливостей нами створений електронний курс загальної фізики (Механіка) який функціонує в LMS з відкритим кодом Moodle [3]

Одним з важливих елементів навчального процесу, особливо в природничих науках, є лабораторні роботи, без яких неможливо уявити формування фахівця підготовленого до участі в процесі виробництва. Незамінними складовими процесу дистанційного навчання можуть стати дистанційно виконувані лабораторні роботи і віртуальні симулятори що включаються в LMS. Такі віртуальні лабораторні роботи замінюють реальні в умовах заочного здобуття освіти і можуть бути одним з навчальних етапів для студентів стаціонарної форми навчання. Використання електронних лабораторних робіт має особливу актуальність в умовах недостатнього рівня матеріально-технічної бази при викладанні природничих дисциплін у периферійних навчальних закладах. В цьому випадку дистанційне виконання лабораторних робіт може бути чи не єдиним шляхом швидкого підняття рівня випускників і підготовки їх до незалежного тестування та подальшого навчання в умовах Болонського процесу, який характеризується підвищеними вимогами до самостійної роботи студента. Однією з причин, що стоїть на заваді найширшому використанню електронних лабораторних робіт є використання закритого коду. Перехід до спільних стандартів і відкритого коду в системах дистанційного навчання поєднаний з об'єднанням локальних мереж вищих навчальних закладів дає можливість впровадження дистанційних лабораторних робіт в навчальний процес в стислі терміни, їх зміни і адаптації відповідно до вимог навчальної програми.

Література

1. [Електронний ресурс]. Електронні дані. 2007. Режим доступу: <http://moodle.org>, вільний.
2. Jason Cole. Using Moodle. Teaching with the popular open source.
3. www.edu.uninet.kiev.ua

Обеспечение учебного процесса по курсу «Информатике и ИКТ» в образовательных учреждениях в условиях внедрения СПО

Недосеков А.В., Торский Р.Б.

Негосударственное образовательное учреждение «Открытый молодёжный университет», г. Томск, Россия, Roman.Torskiy@itdrom.com

В 2008 году Томская область стала участником федерального пилотного проекта по внедрению СПО. Кроме Томской области в эксперименте участвовали Пермский край и Татарстан.

В образовательных учреждениях сложилась сложная ситуация: пришлось удалить весь (или почти весь) нелегализованный софт. У педагогов возник вопрос - как преподавать информационные технологии без него? В 2007 году участники Рабочей группы при Региональном центре развития образования (РЦРО Томской области) исследовали большое количество дистрибутивов. Пять отобранных дистрибутивов установили в своих школах. Ещё через полгода компания Интрайс предложила учителям школ Томской области дистрибутив на базе Кубунту, адаптированный к нуждам учебного процесса (это происходило в рамках областного проекта). Затем в школы одновременно начали поступать пакеты СБППО и ПСПО (это уже федеральный проект), первый для установки, второй — для апробации. Параллельно проводились курсы повышения квалификации, обучающие учителей и школьных системных администраторов работе с новыми системами.

Открытый молодёжный университет сотрудничает с Образовательными учреждениями России и ближнего зарубежья в области преподавания Информатики и ИКТ с 2001 года и к решению проблемы СПО подошёл ещё в 2006 году. Отчасти из-за того, что представители некоторых школ начали задавать вопрос: «А когда у вас будут практикумы под Linux?», отчасти из-за того, что это интересно. Участие Томской области в федеральном проекте по внедрению СПО значительно ускорили процесс перехода к разработке курсов под СПО.

На сегодня методистами «Открытого молодёжного университета» разработаны 16 курсов, направленных на изучение СПО школьниками с 5 по 11 класс. Это курсы по направлениям: Информационные процессы и технологии, офисные технологии, программирование, компьютерная графика, мультимедиа и дизайн:

№ п/п	Наименование учебного курса	Необходимое свободное программное обеспечение, устанавливаемое как под ОС Windows и так под ОС Linux
1.	Волшебный компьютер	Пакет OpenOffice.org (Writer, Calc). Браузеры Mozilla FireFox, Opera.

2.	Занимательное ЛОГО-знание	Приложение MSWLogo
3.	Азбука офиса	Пакет OpenOffice.org (Writer, Impress)
4.	Юный дизайнер	Приложение Gimp версия 2.6. Приложения Paint (для ОС Windows) и KolourPaint (для ОС Linux).
5.	Мир информационных технологий	Пакет OpenOffice.org (Writer, Calc). Браузер Mozilla FireFox.
6.	Математическое и компьютерное моделирование	Пакет OpenOffice.org (Writer, Calc, Base)
7.	Практическое моделирование. Компьютерный эксперимент	Среда программирования Lazarus. Пакет OpenOffice.org (Writer, Calc)
8.	Увлекательные уроки программирования. Pascal	Среда разработки Borland Pascal под DosBox
9.	Объектно-ориентированное программирование в среде Delphi	Среда программирования Lazarus
10.	Офисные технологии: текстовые документы и мультимедийные презентации	Пакет OpenOffice.org (Writer, Impress)
11.	Офисные технологии: электронные таблицы и основы баз данных	Пакет OpenOffice.org (Calc, Base)
12.	Компьютерная графика. CorelDRAW	Приложение Inkscape версия 0.47
13.	Секреты компьютерной графики	Приложение Gimp версия 2.6
14.	Основы компьютерного дизайна	Приложение Gimp версия 2.2.17. Приложение Inkscape версии 0.45.
15.	Основы издательского дела	Программа SCRIBUS версия 1.3.5

16	Подготовка к ЕГЭ— 2011 по информатике	Пакет OpenOffice.org (Writer, Impress, Calc). Среда разработки Borland Pascal под DosBox
----	--	--

Каждый из 16 курсов обеспечен Интерактивным электронным учебником, учебным пособием и базой контрольно-измерительных материалов: контрольные работы, материалы для проведения итогового контроля и материалы и задания по проектной деятельности.

В 2010-2011 году 800 Образовательных учреждений и 13 130 учащихся используют в образовательном процессе курсы под СПО, разработанные НОУ «Открытый молодёжный университет».

Для помощи педагогам образовательных учреждений в переходе на СПО были разработаны очные 48 часовые обучающие семинары по темам:

- Основы алгоритмизации и программирование в 8-11 классах;
- Математическое и компьютерное моделирование в 8-11 классах, которые регулярно проводятся преподавателями «Открытого молодёжного университета», выезжающими в регионы.

Очный мастер-класс для учащихся и педагогов «Фракталы и рекурсия в среде программирования MSWLogo»

Очный 4-х часовой семинар «Переход школ на СПО: проблемы и перспективы»

Хочется верить, что учебно-методические комплекты «Открытого молодёжного университета» значительно облегчат учителю переход к свободному программному обеспечению. Во всяком случае, мы делаем для этого всё возможное.

Інформаційно-аналітична система управління навчальним процесом ВНЗ на базі MOODLE

Триус Ю.В.

Черкаський державний технологічний університет, tryusyv@gmail.com

У тезах анансовано проект створення та впровадження у діяльність ВНЗ 3-4 рівнів акредитації інформаційно-аналітичної системи управління навчальним процесом (ІАС УНП) в умовах кредитно-модульної системи навчання та європейської інтеграції вищої освіти України.

Проведений аналіз існуючих вітчизняних і зарубіжних інформаційних систем управління ВНЗ показав, що:

- існуючі системи не забезпечують на потрібному рівні зворотній зв'язок в системі управління навчальним процесом ВНЗ, що є необхідною умовою покращення навчального процесу на рівні як окремої дисципліни, так і навчальних підрозділів ВНЗ;
- у цих системах практично відсутні засоби моделювання навчального процесу, які б надавали можливість на основі даних про результати навчальної діяльності студентів, професійної діяльності викладачів, функціонування навчальних підрозділів ВНЗ оптимізувати параметри організації і контролю навчального процесу, прогнозувати показники успішності і якості навчання, а також рівень професійної підготовки майбутніх фахівців у різних галузях людської діяльності;
- найбільш поширені ІАС управління ВНЗ є комерційними продуктами, з англійським і російськомовним інтерфейсом, вимагають наявності ліцензованого програмного та апаратного забезпечення високої вартості і, як правило, не враховують специфіки українських ВНЗ.

Метою проекту, що пропонується авторським колективом ЧДТУ, є створення та впровадження у діяльність ВНЗ 3-4 рівнів акредитації інформаційно-аналітичної системи управління навчальним процесом (ІАС УНП) в умовах кредитно-модульної системи навчання та європейської інтеграції вищої освіти України.

У проєкті враховані зазначені вище недоліки, використовуються сучасні методи прийняття рішень та імітаційного моделювання, web-технології, при цьому основними критеріями вибору засобів створення системи є: відкритість, безкоштовність, простота застосування та незалежність від системного програмного та апаратного забезпечення.

ІАС УНП ВНЗ є складовою цифрового університету і повинна забезпечити кінцевим користувачам, які приймають рішення (в межах своїх повноважень) зручний доступ до відповідних даних, моделей та інших інформаційних ресурсів з метою прийняття рішень у напівструктурованих і неструктурованих ситуаціях, пов'язаних з навчанням студентів і управлінням діяльністю навчальних підрозділів ВНЗ. Даний проєкт є наступним етапом після створення інформаційно-аналітичної системи контролю та оцінювання навчальної діяльності студентів ВНЗ (ІАСКОНДС ВНЗ), що

розроблялася в ЧДТУ у 2009-2010 р.р. за Держбюджетною темою №ІТ/535-2009 (реєстраційний №0109U006094) за Державною програмою “Інформаційні та комунікаційні технології в освіті і науці” на 2006-2010 р.р.

Інформаційно-аналітична система управління навчальним процесом ВНЗ повинна:

- забезпечити єдине місце, де всі члени університетської спільноти мають виконувати всі ділові транзакції (операції);
- забезпечити структурну єдність університету (на академічному, адміністративному та особистісному рівнях), а також інтеграцію всіх процедур автоматизації та супроводу ділових процесів;
- забезпечувати доступ зацікавленим особам до інформаційних ресурсів та служб ВНЗ через єдиний графічний інтерфейс;
- підтримувати єдину систему ідентифікації та перевірки прав доступу користувачів до корпоративних інформаційних ресурсів та служб;
- забезпечити інформаційні засоби управління навчальним процесом ВНЗ;
- підтримку розробки нових освітніх технологій управління та їх моделювання;
- забезпечити швидке впровадження нових освітніх технологій та технологій управління в навчальний процес ВНЗ.

Основні завдання ІАС УНП ВНЗ:

- оптимізувати складові навчального процесу ВНЗ за рахунок регламентації зв'язків між структурними підрозділами;
- забезпечити контроль доступу до інформації;
- збереження конфіденційності і неможливості спотворення даних;
- усунути дублювання даних;
- автоматизувати рутинні операції працівників навчальних підрозділів ВНЗ;
- забезпечити єдине інформаційне робоче середовище для керівного складу, професорсько-викладацького складу і студентів ВНЗ.

В основу створення ІАС УНП ВНЗ покладено такі принципи:

- орієнтація на національні і міжнародні освітні стандарти вищої освіти;
- дотримання сервіс-орієнтованої архітектури;
- дотримання модульності компонентів системи;
- використання компонентів ПЗ ІАС лише з відкритими ліцензіями та вихідним кодом;
- дотримання поступового та ітеративного процесу розвитку ІАС;
- забезпечення якості продукту як результат дотримання дисципліни процесу розробки;
- інформаційне забезпечення продукту та супутні послуги з підтримки життєвого циклу продукту в конкретному місці впровадження.

Однією з ключових складових ІАС буде підсистема моделювання навчального процесу, яка на основі методів об'єктно-орієнтованого

моделювання складних систем та технології імітаційного моделювання систем з використанням мереж Петрі буде враховувати складні взаємозв'язки між усіма суб'єктами навчального процесу і стане ефективним інструментом для розробки та дослідження нових технологій управління навчальним процесом ВНЗ. В межах проекту також планується дослідити та розробити системну архітектуру ІАС з використанням технології корпоративної шини служб (ESB) на технологічному стеку JEE 5/6 для забезпечення можливості інтеграції ІАС УНП ВНЗ з такими складовими цифрового університету, як ІАС управління людськими ресурсами ВНЗ, ІАС „Електронний диспетчер”, ІАС «Електронний деканат», ІАС «Кафедра», ІАС управління навчальним навантаженням, ІАС контролю та оцінювання навчальної діяльності студентів ВНЗ.

Обрана технологія і сама система будуть задовольняти таким вимогам:

- орієнтація на національні й міжнародні освітні стандарти у галузі вищої освіти;
- дотримання сервіс-орієнтованої архітектури;
- забезпечення стабільної роботи із значною кількістю користувачів;
- підтримка чіткого розподілу прав користувачів на одержання та зміни інформаційних ресурсів;
- забезпечення модульності кінцевого продукту та його здатності до інтеграції в корпоративну інформаційну систему ВНЗ.

Технологічна архітектура ІАС УНП ВНЗ передбачає використання наступних компонентів з відкритим вихідним кодом:

для серверної частини:

- операційні системи (FreeBSD, GNU/Linux);
- системи управління базами даних (PostgreSQL, MySQL, Firebird);
- проміжного програмного забезпечення (Apache Tomcat, Apache HTTPd, Oracle Glassfish, RedHat JBoss, Apache ServiceMix);
- засобів створення і підтримки (PHP, CMS Joomla!);
- системи електронного навчання LCMS Moodle;
- для клієнтської частини: web-браузер (Mozilla Firefox, Google Chrome).

В результаті реалізації проекту планується створити університетський портал як інформаційно-аналітичне середовище для підтримки освітньо-наукових процесів у межах ВНЗ на основі використання сучасних інформаційних технологій і телекомунікаційних засобів. ІАС управління навчальним процесом ВНЗ може бути запроваджена у всіх ВНЗ 3-4 рівнів акредитації України, а при належній локалізації системи її можна пропонувати на зарубіжних ринках програмних продуктів у галузі освіти.

Використання CMS JOOMLA! та LCMS MOODLE у ВНЗ

Франчук В.М.

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

Багато навчальних закладів і інших організацій, що планують розгорнути свої освітні послуги в глобальній мережі Інтернет, часто мають справу з деякими існуючими технологічними рішеннями, що є недостатньо адаптованими до існуючих умов і не відповідають реальним потребам. При цьому часто кошти спрямовуються на придбання зовні привабливих технологічних рішень всесвітньо відомих торгових марок. Однак впровадження дорогих комерційних систем в умовах місцевого ринку далеко не завжди приносить позитивний результат, а наступне доопрацювання або адаптація до потреб, що постійно змінюються, в багатьох випадках стає неможливим.

Joomla! – це система управління вмістом сайту (від англ. Content Management System (CMS)). CMS Joomla! (далі система Joomla!) – це вільнопоширюване програмне забезпечення для динамічного управління веб-сайтом безпосередньо на сервері (офіційний сайт <http://joomla.org>).

Стандартний пакет системи Joomla! може бути легко і швидко встановлений користувачами без спеціальної підготовки. Після встановлення та запуску системи 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 Розділів, Категорій і Об'єктів на сайті. Користувач, який вмiє набирати текст за допомогою стандартних комп'ютерних засобів, може швидко оволодіти правилами роботи з системою Joomla!.

Крім цього, для системи Joomla! тисячі розробників зі всього світу створюють нові та удосконалюють вже існуючі шаблони і розширення (Компоненти, Модулі і Плагіни). Наприклад: системи опрацювання документів, мультимедіа галереї, системи для електронної комерції, форуми, чати, блоги, календарі, засоби для опрацювання даних і побудови звітів, системи банерної реклами, засоби розсилання електронних повідомлень і т. д.

На сьогодні для системи Joomla! написано близько 2000 розширень (Компоненти, Модулі, Плагіни), різні інтеграції та мости із форумами, галереями та іншими системами.

Знайти приклади розширень (Компоненти, Модулі і Плагіни) для системи Joomla! можна на сайті <http://extensions.joomla.org>.

Дізнатися більше про розробку прикладних програмних засобів для системи Joomla! можна на сайті для розробників <http://developer.joomla.org>.

Основні характеристики системи:

- система Joomla! написана з використанням мови PHP та баз даних MySQL;
- багаторівневий доступ зареєстрованих користувачів, як до Адміністра-

тивної частини так і до Фронтальної частину сайту;

- зрозуміла структура Розділів та Категорій для зручного структурування вмісту;
- легке створення та редагування матеріалів (статей);
- вміст може додаватись як окремо Адміністраторами та Менеджерами із Адміністративної частини, так і Публікаторами, Редакторами, Авторами та звичайними зареєстрованими користувачами із Фронтальної частини сайту;
- редагування матеріалів за допомогою візуального редактора (аналог текстових редакторів, наприклад *OOO Writer*);
- використання програмних продуктів сторонніх розробників;
- робота системи в ОС Linux, FreeBSD, Windows, Solaris та ін.

За допомогою системи Joomla! можна створити як сайт візитку, персональний сайт так і великі потужні портали, корпоративні сайти, інформаційні проекти.

MOODLE (Modular Object Oriented Distance Learning Environment) – це система управління навчальним контентом (LCMS – Learning Content Management Systems). 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.

Система MOODLE (www.moodle.org) розповсюджується безкоштовно за принципами ліцензії Open Source. Кожний користувач має доступ до джерела-коду системи і може його змінювати залежно від своїх цілей та бажань.

Автор концепції системи MOODLE австралієць Martin Dougiamas. Головною його метою було створення системи, відмінної від доступних на ринку, а саме такої, в якій враховувалися б педагогічні аспекти, що базуються на основах пізнавальної психології, а особливо однієї з її течій, яка іменується конструктивізмом.

В цій теорії припускається, що студент (учень) – активний суб'єкт, який самостійно створює свою власну систему знань, користуючись при цьому доступними йому джерелами знань. Роль вчителя (тьютора) в мотивуванні і підтримці своїх підопічних полягає головним чином в підготовці завдань для самостійного опрацювання, оцінюванні результатів їх виконання, коригуванні знань студентів (учнів) і т.п. Використання цих завдань сприяє формуванню в студента (учня) нових знань. Відповідно до основ суспільного конструктивізму, конструйоване знання найбільш ефективно, коли учні навчаються в співпраці. Це можливо тоді, коли учень працює в групі, ділячись досвідом і думками, і будучи відкритим для досвіду і думок інших.

Перевагою LCMS MOODLE є той факт, що почавши від її появи, тобто з 1999 року, вона неодноразово була модифікована і доповнена новими

рішеннями і інструментами. Програмне забезпечення платформи описано мовою PHP та з використанням безкоштовних загальнодоступних баз даних (MySQL, PostgreSQL і ін.). Систему MOODLE можна встановити на будь-яку операційну систему (Unix, Linux, MS Windows,).

Система MOODLE включає набір модулів, використання яких дає можливість співпрацювати на рівнях учень-учень і учень-вчитель. До цих модулів належать: голосування (опитування), анкети, чати, форуми, уроки, журнали, тести, словники, семінари, wiki, завдання і т.д. (всього біля 35 модулів).

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

Моделювання задач електротехніки у XCOS

Філь І.М.

Донецький національний технічний університет, iron.ivan1991 @ gmail.com

Розглянуто вільно поширюваний додаток Xcos, як середовище для моделювання електричних ланцюгів і енергосистем. Розглянуто структуру цього додатка. Показаний приклад моделювання перехідних процесів в електричних колах з елементами різної структури.

Розрахунок електричних ланцюгів - це одне з основних завдань енергетики. Моделювання електричних ланцюгів у програмних пакетах полегшує і прискорює вирішення таких завдань. У даній доповіді додаток Xcos розглядається як середовище для моделювання електричних схем.

При запуску Xcos з'являються два вікна. Перше з них - Palette browser, браузер компонентів (рис. 1), служить для вибору елементів моделювання. Потім елементи «перетягуються» в друге вікно (рис. 2), яке призначене для створення і редагування моделей.

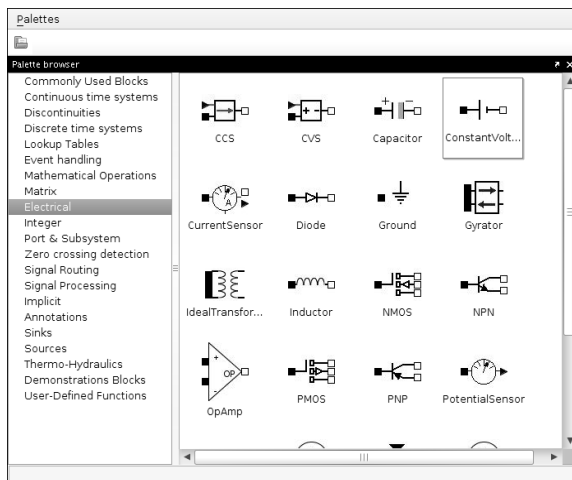


Рис. 1: Браузер компонентів

У вікні для створення моделей (рис. 2) можна також налаштувати час моделювання, точність та інші параметри, для чого використовується команда Simulation-Setup. При подвійному клацанні миші по елементу відкривається діалогове вікно для зміни параметрів даного елемента.

На мал. 1 видно деякі елементи бібліотеки Electrical, в якій є моделі вимірювальних приладів (датчиків) для напруги, струму, індуктивності, конденсатора, активного опору, діода, транзисторів, трансформатора, вимикача, різних джерел напруги, струму.

В Xcos можна змоделювати перехідні і встановлені режими в електричних ланцюгах. В якості прикладу розглянемо задачу моделювання перехідного процесу в електричному ланцюзі, представленою на рис. 2.

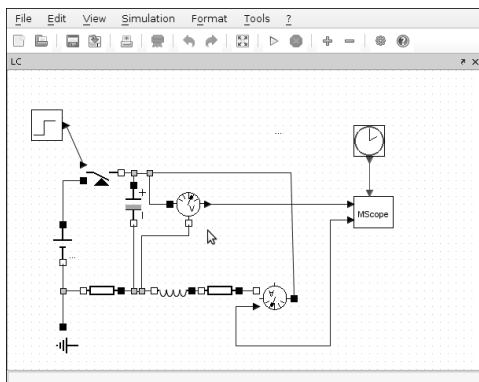


Рис. 2: Вікно для редагування моделей. RLC-ланцюг.

Через деякий проміжок часу вимикач відключає ланцюг від джерела напруги і відбувається перехідний процес. На рис. 3 зображені графіки зміни струму на індуктивності (графік 2) і напруги на ємності (графік 1).

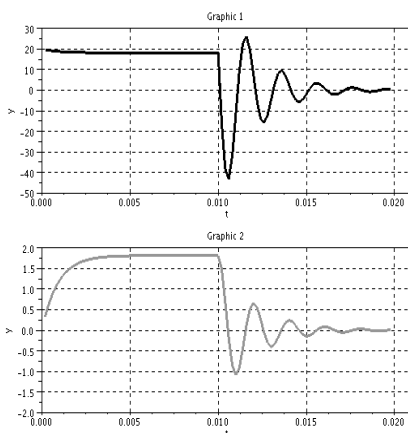


Рис. 3: Зміна струму на індуктивності, напруги на ємності в RLC-ланцюзі

Проведений автором аналіз дозволяє стверджувати, що система моделювання Xcos може використовуватися поряд з пропріетарної програмою Simulink для моделювання перехідних і усталених режимів в електричних ланцюгах будь-якої складності і конфігурації.

Автор рекомендує використовувати останню версію Xcos для розв'язання завдань електротехніки та впровадження програми в навчальний процес, але це дещо ускладнюється недостатньо повною документацією і відсутністю будь-якої літератури російською і українською мовами.

Локалізація системи MOODLE

Франчук Н.П.

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

Деякі навчальні заклади використовують системи управління навчальним контентом, зокрема LCMS MOODLE, для надання освітніх послуг в глобальній мережі Інтернет. Ці системи мають різні локалізації і є недостатньо перекладеними (локалізованими) українською мовою.

MOODLE (Modular Object Oriented Distance Learning Environment) – це система управління навчальним контентом (LCMS – Learning Content Management Systems). За допомогою даної системи можна створювати навчальні курси і проводити як аудиторне навчання, так і навчання на відстані. Система MOODLE розповсюджується безкоштовно. Користувачі мають доступ до джерела-коду системи і можуть його змінювати залежно від своїх цілей та бажань, зокрема здійснювати переклад (локалізацію) даної системи.

Локалізацією програмного забезпечення (перекладом програм) називається адаптація ПЗ під національні вимоги, включаючи забезпечення роботи програм із регіонально-залежними форматами даних і форматами друку, зміну символів, які використовуються, рисунків, кольорових комбінацій, музичних фрагментів тощо, згідно з культурою цільової аудиторії тощо. Часто, говорячи про локалізацію, мають на увазі всього тільки переклад інтерфейсу на цільову мову.

Оновити мовний пакет (мовний пакет містить файли із змінними і їх значеннями) можна з офіційного сайту (<http://www.moodle.org>) системи MOODLE (перекладено 20,4%) або з сайту (<http://moodle.co.ua>) «Українська спільнота користувачів Moodle» (перекладено 96,4%, але з помилками), виконавши наступні кроки:

- 1)Завантажити мовний пакет (як правило zip-архів);
- 2)Розархівувати його в каталог <шлях до moodldata>/lang;
- 3)Вибрати мову сайту за замовчуванням українську (якщо це не було зроблено при встановленні системи).

Після перезавантаження (оновлення) головної сторінки сайту отримаємо локалізовану версію системи MOODLE.

Також переклад мовного пакету можна редагувати засобами системи MOODLE. Для цього потрібно в головному меню адміністратора (див. Рис. 1) вибрати послідовність посилань: Мова інтерфейсу > Редагування > Редагування змінних.

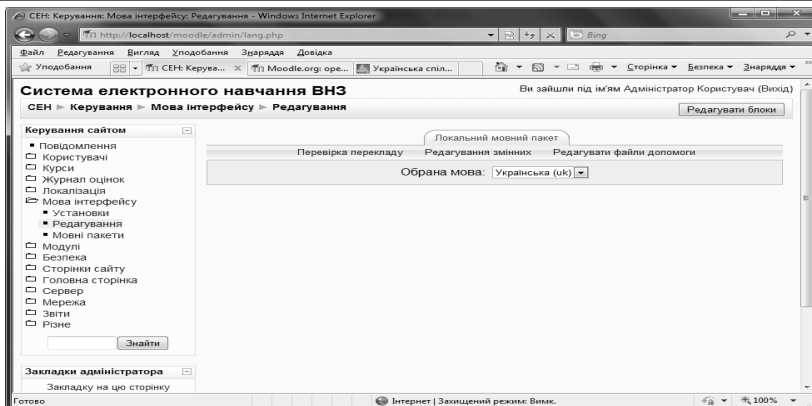


Рис. 1.

В доповіді планується більш детально розкрити досвід здійснення перекладу LCMS MOODLE.

Література

1. Moodle.org: open-source community-based tools for learning. [Електронний ресурс]. – Режим доступу: <http://www.moodle.org>
2. POLYGLOT – Переклад в сфері інформаційних технологій – Latvia – Riga. [Електронний ресурс]. – Режим доступу: <http://www.polyglot.lv/uk/it-pereklad.html>
3. Українська спільнота користувачів Moodle. [Електронний ресурс]. – Режим доступу: <http://moodle.co.ua>

З досвіду використання вільного програмного забезпечення при вивченні інформатики у виші

Харченко В.М.

Ніжинський державний університет ім. Миколи Гоголя, volmkhar@gmail.com

У тезах проаналізовано можливість використання вільного програмного забезпечення у вишах, описано досвід його використання при підготовці майбутніх вчителів інформатики, вказано на переваги й недоліки використання такого забезпечення.

В умовах побудови багатьма країнами світу інформаційного суспільства все більш актуальною стає проблема переходу навчальних закладів на вільне програмне забезпечення. Згідно з [1], це дасть змогу не тільки зекономити матеріальні затрати закладів, а й заохотить учнів до навчання та виховає з них високоморальних членів суспільства.

Ніжинський державний університет ім. Миколи Гоголя здійснив перехід на використання вільного програмного забезпечення в 2007 р. Тоді студенти всіх спеціальностей перейшли на використання Linux, OpenOffice.org, GIMP, Inkscape, Free Pascal та Lazarus. Значною проблемою при такому переході стала майже повна відсутність україномовної навчально-методичної літератури, яку можна було б рекомендувати студентам. Щоб вийти з такої ситуації, викладачами кафедри прикладної математики та інформатики були створені електронні курси з необхідних дисциплін на основі системи управління навчанням Moodle.

Як показав досвід використання вільного програмного забезпечення, студенти отримують достатні уявлення про використання інформаційних технологій в повсякденному житті. Набувши вміння працювати із вказаним раніше вільним програмним забезпеченням, студенти швидко переходять на інші його різновиди, зокрема й на такі комерційні продукти як Microsoft Windows, Microsoft Office тощо. Це важливо тому, що програмне забезпечення змінюється набагато швидше, ніж людина закінчує виш.

Оскільки шкільні підручники з інформатики орієнтовані на використання пропріетарного програмного забезпечення, то при навчанні майбутніх вчителів інформатики довелося повернутися до застосування програмних продуктів фірми Microsoft, закупивши Microsoft Office та використовуючи програму MSDN Academic Alliance. Проте і цим студентам даються розрахункові роботи, які вони повинні виконати з використанням вільного програмного забезпечення. А при вивченні курсів “Використання інформаційних технологій в освіті і науці”, “Шкільний курс інформатики та методика його викладання” показується можливість використання замість Microsoft Office – OpenOffice.org та Scribus, Adobe Photoshop – GIMP, CorelDraw – Inkscape тощо. При цьому зауважуємо, що достатньо повного аналога Microsoft Publisher немає, тому для його заміни слід використовувати Scribus і Quanta Plus. Доцільність такого переходу визнають вже

не тільки в інших країнах, а й на Україні. Згідно з [2], при оснащенні комп'ютерних класів вільним програмним забезпеченням 11 загальноосвітніх шкіл м. Прилуки економиться близько 500 тис.грн.

Враховуючи постійні затримки з друкуванням підручників та швидке їх старіння, активно пропагуємо майбутнім вчителям використовувати вільні системи управління навчанням. Оскільки система Moodle [3] досить розвинена і надійна, то для створення навчальних курсів рекомендуємо саме її. Причому, дану систему можна використовувати не тільки для наповнення навчальним матеріалом та перевірки знань, а й для виконання учнями практичної роботи з інформатики по спілкуванню у форумах та чатах.

Пропагуючи вільне програмне забезпечення наголошуємо й на деяких труднощах його використання: відсутність підтримки розробників, менша сфера функціональності [4], недостатня кількість навчально-методичної літератури. Усунення останнього недоліку можливе завдяки створенню порталів [5, 6], проте їх наповнення бажає бути кращим.

Отже, використання вільного програмного забезпечення має свої переваги і недоліки, проте досвід такого використання показує, що переваг значно більше за недоліків.

Література

1. Stallman R. Why Schools Should Exclusively Use Free Software. [//www.gnu.org/philosophy/schools.html](http://www.gnu.org/philosophy/schools.html)
2. Школи намагаються переходити на безкоштовне програмне забезпечення. [// osvita.ua/school/news/7173](http://osvita.ua/school/news/7173)
3. Довідка про Moodle. [// moodle.co.ua/mod/resource/view.php?id=11](http://moodle.co.ua/mod/resource/view.php?id=11)
4. Перспективи та можливості впровадження вільного програмного забезпечення в навчальних закладах та державних установах України. [// old.niss.gov.ua/Monitor/june2009/15.htm](http://old.niss.gov.ua/Monitor/june2009/15.htm)
5. Збірника документації з Unix/Linux українською мовою. [//docs.linux.org.ua](http://docs.linux.org.ua)
6. Вільне та відкрите програмне забезпечення [//edu.root.ua/index.php](http://edu.root.ua/index.php)

Программный комплекс для расчета процессов переноса водорода в металлах

Чихрай Е.В.¹, Тажыбаева И.Л.2, Кульсартов Т.В.²

¹⁾ НИИЭТФ КазНУ им.аль-Фараби, Алматы, Казахстан; chikhray@physics.kz

²⁾ ИАЭ НЯЦ РК, Алматы, Казахстан

В работе приводится описание и математическая формулировка задачи переноса водорода и его изотопов (дейтерия и трития) в металлах, постановка задачи численного решения системы дифференциальных уравнений диффузии в среде с ловушками и равномерно распределенным или точечным источником диффузанта. Описывается реализация решения полученной аппроксимацией конечными разностями системы линейных уравнений методом прогонки (алгоритм Томаса). Описаны примеры реализации основных модулей программы расчета, результаты тестирования ее на экспериментальных кривых.

Одним из важнейших методологических аспектов современного материаловедения, в особенности связанного с созданием новых материалов для работы в критических условиях высоких температур, давлений и радиационной нагрузки, является изучение процессов массопереноса водорода. Исследование одной только роли водорода в процессах коррозии, охрупчивания и старения металлов в настоящее время заставляет исследователей искать новые подходы и методы, как экспериментов, так и анализа. Одним из наиболее мощных методов такого анализа экспериментальных данных традиционно является компьютерное моделирование внутренних процессов в металлах и сплавах в присутствии водорода и под различными рода внешними воздействиями: радиационном, температурном, химическом, механическом и т.п.

В НИИЭТФ КазНУ уже более десяти лет, как используется для научных и образовательных целей и постоянно совершенствуется собственный комплекс программ для расчета параметров взаимодействия водородотвердое тело. Необходимость разрабатывать собственный набор программ обусловлена целым рядом причин – и недоступностью дорогих лицензионных продуктов, и их ограниченностью в некоторых аспектах. Скажем такой известный продукт как ANSYS, базирующийся на аппроксимации конечными элементами, не очень подходит для случая многослойных систем типа метал-оксид-покрытие, когда толщина каждого из слоев отличается на три-четыре порядка от предыдущего. Кроме того иногда необходимо иметь более развитое описание ловушек для диффузанта или процессов на поверхности или границах раздела.

Представленный набор программ ограничивается рассмотрением только одномерных процессов массопереноса и только таких, для которых возможно составление описывающей процесс системы дифференциальных уравнений. Такое ограничение, на самом деле не является очень жестким, так как охватывает большинство исследовательских (экспериментальных) и значительную часть практических задач диффузии и массопереноса

водорода.

В наиболее типичном случае процесс переноса водорода в металле описывается системой дифференциальных уравнений диффузии в однородной среде с ловушками ограниченной емкости (по модели Мак-Набба-Фостера). Задача для одномерного случая состоит в численном решении системы дифференциальных уравнений вида

$$\begin{cases} \frac{\partial C}{\partial t} = D(t) \cdot \frac{\partial^2 C}{\partial x^2} - N \frac{\partial m}{\partial t} + G(x, t) \\ \frac{\partial m}{\partial t} = k_1(t) \cdot (1 - m)C - k_2(t) \cdot m \end{cases}$$

где C – концентрация диффузанта (в нашем случае – водорода) в металле; $D(t)$ – коэффициент диффузии; N – число (концентрация) ловушек ограниченной емкости в металле; m – степень заполнения ловушек ($0 \leq m \leq 1$); $\partial m / \partial t$ – уравнение динамики заполнения-опустошения ловушек (только в свободные ловушки происходит захват водорода, и только из занятой ловушки водород может выйти в металл); k_1 – коэффициент (скорость) захвата водорода в ловушки, k_2 – коэффициент (скорость) высвобождения водорода из ловушки; $G(x, t)$ – функция описывающая некий источник водорода в металле, t – время.

В работе вкратце описывается физика рассматриваемого процесса, этапы его численного моделирования: выбор схемы аппроксимации дифференциальных уравнений конечными разностями; составление системы линейных алгебраических уравнений; выбор и обоснование метода их численного решения; программирование. Большая часть работы посвящена описанию и использованию созданных в лаборатории конструкционных материалов ядерно-энергетических установок НИИЭТФ КазНУ набора программ для численных расчетов таких диффузионных процессов. Программы рассчитывают динамику распределения диффузанта по толщине образца, динамику заполнения-высвобождения ловушек и временную зависимость потока десорбции диффузанта из металла при его проницаемости при постоянной температуре или при нагреве в заданном диапазоне температур с указанной скоростью.

Разработанный программный комплекс может использоваться студентами и докторантами, специализирующихся по физике твердого тела, радиационной физике, экспериментальной ядерной физике и изучающих методы численного моделирования переноса вещества в рамках спецкурсов "Взаимодействие излучений с веществом", "Диффузионные процессы в твердых телах", "Радиационно-стимулированные процессы в кристаллах", "Ядерная и термоядерная энергетика" и соответствующих лабораторных практикумов.

Розробка і використання web-інтерфейсів для роботи з системами комп'ютерної математики

Чичкар'юв Є.А.

*м.Маріуполь, Приазовський державний технічний університет,
influence@meta.ua*

Представлений аналіз існуючих рішень з розробки web-інтерфейсу до обчислювальних пакетів - Sage, Octave, R, Maxima і ін, проаналізовано переваги і недоліки аналогічних пропріетарних рішень для Maple, Matlab, MathCad. Представлені результати розробки і випробування обчислювального сервера для проведення лабораторного практикуму з різних навчальних дисциплін, заснованого на відкритому програмному забезпеченні (Octave, Scilab, R, Maxima)

Сьогодні для роботи з обчислювальними пакетами широко використовують різні варіанти клієнт-серверних технологій з доступом користувачів до обчислювального ядра того чи іншого пакету через web-інтерфейс. У цьому випадку на стороні клієнта необхідний лише Веб-переглядач, а вся обчислювальна робота виконується на стороні сервера.

Досить характерний приклад - Matlab Web server (MWS), що організовує віддалену роботу з MatLab. Скрипти MatLab, призначені для віддаленого запуску з використанням MWS, вимагають деякого доопрацювання, що полягає у створенні web-інтерфейсу (у найпростішому варіанті зчитуються дані з полів html-форми). Інший характерний приклад - Mathcad Calculation Server, що дозволяє працювати з MathCad-документами в мережі. Аналогічні розширення існують і для Maple (MapleNet) або Mathematica (webMathematica).

Технологічно засоби організації віддаленого доступу не однакові, але характерним прикладом може бути webMathematica, заснована на технологіях сервлетів та JSP. Що стосується обчислювальних пакетів з відкритим кодом, то найбільш розвинені засоби для організації віддаленої роботи існують для R. Найбільш розвиненим є R-пакет, що дозволяє запустити TCP / IP-сервер для доступу інших програм до можливостей R - це Rserve (див. <http://www.rforge.net/Rserve/>). Цей пакет володіє досить широкими можливостями як для організації віддаленої роботи, так і для взаємодії з R через localhost (зокрема, з OpenOffice). Поряд з Rserve, можливо і низькорівнева взаємодія java-R-java за допомогою пакетів JRI і rJava (<http://www.rforge.net/JRI> і <http://www.rforge.net/rJava>). Поряд з досить складним Rserve, існує й простіший у використанні Rphp (<http://dssm.unipa.it/R-php/>), що дозволяє організувати віддалену роботу з R за допомогою web-сервера Apache і набору скриптів на php. Існують й інші варіанти web-інтерфейсу R (див. FAQ на <http://cran.r-project.org>).

Пакети, призначені для віддаленої роботи з Octave і Maxima (<http://hara.mimuw.edu.pl/web octave/web/> і <http://maximaphp.sourceforge.net/>), з організації та можливостям мало відрізняються від Rphp.

Під час запуску обчислювального ядра того чи іншого пакету з php (за

допомогою функцій `passthru` або `shell_exec`) важливою проблемою є фільтрація команд користувача для блокування потенційно небезпечних функцій або команд конкретного пакету, що вимагають доступу до файлової системи сервера.

Пакети, написані на `php`, зазвичай мають і функції системи управління контентом - реєстрацію користувачів, зберігання, редагування та видалення скриптів користувачів тощо. Можливості віддаленої роботи з обчислювальними пакетами забезпечує і пакет Sage (<http://sagemath.org/>). За допомогою Sage-notebook можна віддалено працювати з цілою низкою математичних пакетів, що входять до складу Sage, а також публікувати в Інтернет інтерактивні документи («склеюючи» скрипти для різних пакетів і результати їх роботи за допомогою коду на `python`).

Як показало випробування різних варіантів організації віддаленого доступу до обчислювальних пакетів, для організації студентського практикуму та дипломного проектування найбільш простим і зручним рішенням (з перспективою використання спільно з системою дистанційного навчання Moodle) виявилася розробка скриптів доступу на `php` і `javascript` і розробкою `html`-форм для різних завдань.

Веб-сервер Apache був розгорнутий під управлінням ОС Linux Ubuntu. Передбачена робота з пакетами Octave, R, Maxima (з можливістю розширення кола використовуваних обчислювальних додатків). Для кожного пакета розроблений (на базі Rphp) комплекс скриптів для інтерактивного виконання команд, а також виконання файлів конкретного користувача (після реєстрації та входу в систему під своїм логіном, при цьому створюється каталог даного користувача) або звернення до стандартних моделей і сценаріями, забезпеченим формами і протоколами для введення вихідних даних та інтерпретації результатів розрахунку. Для зберігання бази даних аутентифікації використана MySQL.

Перевага цієї розробки полягає в широких можливостях налаштування на конкретне завдання. Зокрема, з її використанням були розроблені система статистичного аналізу результатів поплавно контролю в чорній металургії (R), система статистичного аналізу та прогнозу тимчасових рядів (Maxima), пакет моделювання динамічних систем з розподіленими і зосередженими параметрами (Maxima, Octave) і ін.

Застосування вільного програмного забезпечення для дистанційного навчання у вищих навчальних закладах

Захарченко В.М., д.т.н., проф., Шапо В. Ф., к.т.н., доцент

Одеська національна морська академія, stani@te.net.ua

Проаналізовано організаційні та технічні проблеми, що виникають під час експлуатації систем дистанційного навчання вищих навчальних закладів. Запропоновано структуру, яка дає змогу об'єднати територіально розосереджені підрозділи навчального закладу.

В останні роки українські ВНЗ активно впроваджують програмні комплекси, що дозволяють експлуатувати системи дистанційного навчання (СДН). Найбільш популярні з них розглянуто в роботах [1, 2, 3]. Використання СДН дозволяє отримати додаткові позитивні результати у навчанні зацікавлених студентів [4, 5]. У зв'язку з частою неможливістю вкладати в придбання дорогого програмного забезпечення (ПЗ) власні кошти, недостатнім рівнем володіння новими інформаційними технологіями рядом співробітників ВНЗ та іншими причинами в більшості випадків вибір зупиняється на безкоштовно поширюваній СДН Moodle. Вона регулярно оновлюється, має нескладний інтерфейс і широкий спектр можливостей для роботи студентів, викладачів, розробників навчально-методичних матеріалів та адміністраторів [1, 2]. Поступово з'являються друковані або електронні видання, в яких з різним ступенем деталізації описані процедури створення дистанційних курсів, навчально-методичних матеріалів і адміністрування СДН [6, 7].

Суттєва відмінність Одеської національної морської академії (ОНМА) від ряду українських ВНЗ полягає в наявності великих відокремлених структурних підрозділів в Ізмаїлі (Ізмаїльський факультет, ІФ) та Маріуполі (Азовський морський інститут, АМІ). Крім того, сотні курсантів, студентів, слухачів курсів підвищення кваліфікації щорічно проходять багатомісячну плавальну практику на суднах дальнього плавання та потребують отримання доступу до навчально-методичних матеріалів і консультацій викладацького персоналу, перебуваючи далеко від дому та ВНЗ. Такі можливості можуть бути надані наступними способами:

1. Передавання друкованих або електронних навчально-методичних матеріалів на судна при зміні екіпажів суден (непередбачувано і ненадійно).
2. Пересилання електронних навчально-методичних матеріалів електронною поштою або завантаження електронних навчально-методичних матеріалів з бази даних ОНМА по мережі Інтернет (успішно реалізується протягом 13 років).
3. Використання вільно поширюваної СДН Moodle дає змогу реалізувати п.2 і поступово підготувати викладачів та студентів для регулярного використання широкого спектру її можливостей.

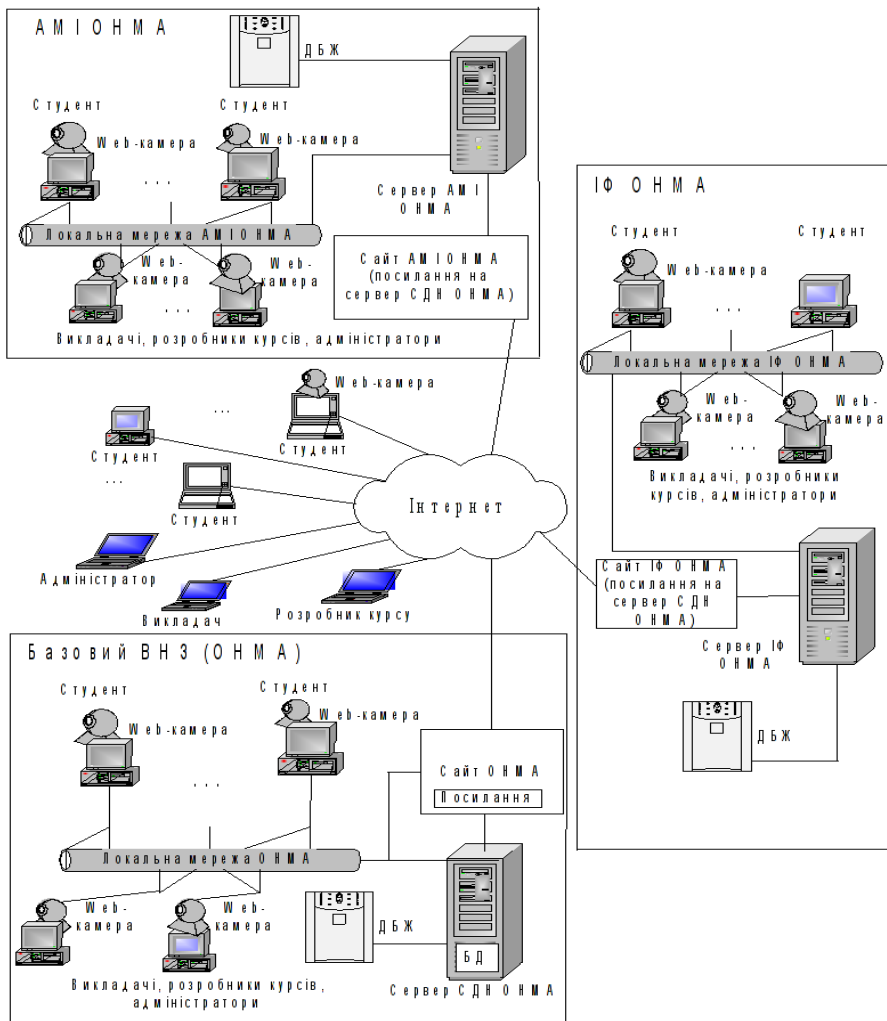


Рис. 1. Структура інформаційної взаємодії підрозділів ОНМА

Реалізована в ОНМА схема дистанційної інформаційної взаємодії між учасниками навчального процесу великих відокремлених підрозділів ОНМА зображена на рис. 1.

Відомо, що в багатьох випадках студенти починають вивчення матеріалу дисциплін з навчально-методичних посібників, підготовлених викладачами ВНЗ, в якому вони навчаються. Саме ці посібники простіше (у порівнянні з літературою інших авторів і правовласників) розміщувати в СДН або Інтернет, оскільки можна звести до мінімуму проблему порушення авторських прав та інші юридичні аспекти.

В теперішній час СДН ОНМА побудована на програмному комплексі Moodle 1.9.6 та забезпечує роботу більше ніж 8400 користувачів (у т. ч. 210 викладачів). Для швидкої реєстрації великої кількості користувачів створено спеціальне ПЗ та розроблено відповідний регламент. Активність і зацікавленість використання СДН курсантами денної форми навчання та студентами заочної форми навчання постійно зростає.

Література

1. Шапо В.Ф. Анализ программных средств систем дистанционного обучения // Информационные технологии в учебном процессе: труды 4-го научно-методического семинара. – Одесса: ЮГПУ. – 2003. – С. 77 – 78.
2. Кухар А., Мишко С., Бушуев Е. Электронное обучение: начало // Компьютерное обозрение. – №30. – 2010. – С. 17 – 24.
3. Шапо В.Ф. Применение систем дистанционного обучения для повышения уровня конкурентоспособности учебных заведений и специалистов // Мережева розробка курсів тренінгу з менеджменту у сфері транспорту і логістики: монографія / Ун-т м. Падерборн (Німеччина) та ін. – К.: НТУ, 2008. – С. 417 – 431.
4. Винников В.В., Шапо В.Ф. Подготовка квалифицированных кадров в области логистики с использованием систем дистанционного обучения // INCEL 08: E-learning в высшей школе – проблемы и перспективы: труды международной научно-практической конференции. – Одесса: ОНПУ – НТУ «ХПИ». – 2008. – [Электронный ресурс]. – <http://cde.kpi.kharkov.ua/tempus/incel/>.
5. Винников В.В., Шапо В.Ф. Подготовка специалистов в области логистики с применением дистанционного обучения // Стратегия качества в промышленности и образовании: труды 4-й международной конференции. – Т.2. – Варна: технический университет. – 2008. – С. 536 – 539.
6. Захарченко, В.Н., Шапо, В.Ф. Руководство по созданию учебно-методических материалов в системе дистанционного обучения (e-learning) на базе программного комплекса Moodle: методические рекомендации / В.Н.Захарченко, В.Ф. Шапо. – Одесса: ОНМА, 2009. – 28 с.
7. Мясникова Т.С., Мясников С.А. Система дистанционного обучения MOODLE. – Харьков: Издательство Шейниной Е.В., 2008. – 232 с.

Реалізація розподілених обчислень на основі грід-платформи з відкритим кодом BOINC

Шийка Ю.Я., Шувар Р.Я.

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

Розглянуто особливості реалізації розподілених обчислень на основі грід-інфраструктури. Представлено особливості вибраної грід-платформи BOINC. Описано проект розподіленої обробки растрових зображень.

Однією з галузей використання вільного програмного забезпечення є реалізація нових комп'ютерних технологій на основі програмного забезпечення з відкритим кодом. Зокрема останнім часом стрімко розвиваються технології розподілених обчислень (грід). Грід або грід-інфраструктура - це розподілена програмно-апаратна комп'ютерна мережа з принципово новою організацією обчислень і управління потоками завдань і даних.

Така комп'ютерна інфраструктура призначена для об'єднання обчислювальних потужностей окремих робочих станцій чи організацій. На основі грід-технології здійснюється інтегрування регіональних і навіть національних обчислювальних комп'ютерних інфраструктур для створення об'єднаних інтернаціональних ресурсів, для розв'язування науково-технічних задач, що потребують дуже великих обчислювальних ресурсів.

Одна з основних частин програмного забезпечення грід-інфраструктури – так зване “Middleware” (проміжне програмне забезпечення). Його завданням є організація та об'єднання ресурсів у мережі, забезпечення автоматичної взаємодії “машина-машина” (M2M) для об'єднання у єдину обчислювальну грід-мережу. Очевидно, що від вибору проміжного програмного забезпечення значною мірою залежить структура грід-мережі та вимоги до адаптації програм, що будуть виконуватися. В зв'язку з цим постає задача вибору проміжного програмного забезпечення, враховуючи його можливості та поставлені до нього вимоги і задачі.

Серед відомого Middleware програмного забезпечення є як платні так і безкоштовні платформи. Серед них найбільш поширені: NorduGrid, Unicore, MOBIDICK, NeuroGrid, CrossGrid, GridLab, SAGA API, BOINC та ін.

Однак найбільший інтерес викликають платформи з відкритим кодом, оскільки вони забезпечують найбільшу гнучкість і можливість відносно легкого впровадження власних проектів розподілених обчислень.

Проаналізувавши можливості різних грід-платформ, для реалізації розподілених обчислень у локальній мережі факультету електроніки було вибрано платформу з відкритим кодом BOINC (Berkeley Open Infrastructure for Network Computing). Серед особливостей цієї платформи варто відмітити:

1. Автономність проектів – різні BOINC проекти є незалежними, кожен з них керує власними серверами та базами даних. Немає центрального

каталогу або процесу затвердження проектів.

2. Гнучкість для учасників – учасники можуть приймати участь в декількох проектах, вони контролюють, в яких проектах приймають участь і як їх ресурси розподілені між цими проектами. Коли проект не працює або не має завдань, ресурси розподіляються між іншими проектами.
3. Гнучка структура програм – існуючі програми написані різними мовами (C, C + +, Fortran) можуть працювати як додатки VOINC практично без змін. Додаток може складатися з декількох файлів (наприклад, декілька програм і об'єднуючий скрипт). Нові версії програм можуть бути впроваджені без необхідності будь-яких дій учасників проекту.
4. Безпека – VOINC захищений проти атак. Наприклад, використання цифрового підпису при розповсюдженні виконавчих файлів забезпечує захист від розповсюдження зловмисниками вірусів під виглядом задач VOINC.
5. Продуктивність сервера і масштабованість – один сервер середнього класу може відправляти задачі і обробляти результати мільйонів робочих місць на день. Серверна архітектура також масштабована, що дозволяє легко збільшити потужність сервера за рахунок додавання нових машин.
6. Доступність вихідного коду – платформа VOINC розповсюджується вільно з відкритим кодом на умовах Lesser General Public License. Проте, VOINC додатки не обов'язково повинні бути з відкритим вихідним кодом.
7. Незалежність від платформи учасника – клієнт VOINC доступний для більшості поширених платформ (Mac OS X, Windows, Linux та інших Unix-подібних систем). Клієнт може використовувати кілька процесорів, в тому числі і GPU.
8. Спільнота учасників – VOINC надає веб-інструменти, такі як дошки оголошень, профілі користувачів, особисті повідомлення, тощо. Це дозволяє об'єднувати учасників проектів у формі інтернет-спільнот.

На основі платформи VOINC у міжкафедральній лабораторії комп'ютерних інформаційних технологій факультету електроніки реалізовано проект віртуального суперкомп'ютерного центру розподілених обчислень для виконання складних обчислювальних науково-практичних і навчальних задач. Здійснено встановлення та налаштування серверної частини грид-платформи VOINC під управлінням Debian Linux на сервері міжкафедральної лабораторії комп'ютерних інформаційних технологій факультету електроніки. Участь в розподілених обчисленнях приймають комп'ютери локальної мережі факультету, надаючи свої незадіяні ресурси.

В рамках цього проекту реалізовано задачу розподіленої обробки растрових зображень великого розміру, а саме їх сегментацію за кольорами. Для цього використано програму сегментації растрових зображень, яку було адаптовано для розподіленої обробки, за принципом поділу великого завдання (обробка великого зображення, чи багатьох зображень) на окремі незалежні підзадачі, які порівняно легко виконуються на окремих комп'ютерах учасників розподілених обчислень.

Таким чином реалізовано серверну та клієнтську частини розподіленої сегментації растрових зображень. Завданням серверної частини є формування підзадач, перевірка результатів виконання підзадач та формування остаточного результату. А завданням клієнтської частини є виконання переданих їй окремих підзадач і передача результатів серверній частині.

Отримана система розподілених обчислень використана для обробки даних дистанційного зондування Землі.

Реалізація високопродуктивної обчислювальної системи на базі ОС LINUX

Шувар Р. Я., Бойко Я. В.

Факультет електроніки Львівського національного університету імені Івана Франка, boyko@electronics.wups.lviv.ua

Описано систему паралельних обчислень на базі графічних процесорів загального призначення (GPGPU). Наведено результати її тестування і висновки щодо можливостей застосування та вдосконалення.

У процесі розв'язання сучасних науково-технічних завдань все частіше виникає потреба здійснення масштабних обчислень. Серед апаратно-програмних рішень для реалізації високопродуктивних обчислень (суперкомп'ютери, кластери) в останній час набуває прискореного розвитку використання спеціалізованих обчислювальних пристроїв, у першу чергу — графічних процесорів (GPU). Особливості архітектури сучасних GPU дають змогу досягати високих показників продуктивності та створювати ефективні з економічної точки зору обчислювальні системи. Труднощі при створенні програмного забезпечення для таких пристроїв в значній мірі вирішуються використанням спеціалізованих високорівневих мов програмування та бібліотек, які реалізують відповідні стандарти GPGPU. Найбільш відомими є відкритий стандарт OpenCL і пропрієтарний CUDA, які реалізовані у системах розробки основних виробників графічних процесорів. З метою запровадження такого роду обчислень у Львівському національному університеті імені Івана Франка на факультеті електроніки створено прототип високопродуктивної обчислювальної системи [1]. Її базова архітектура: SMP x86_64, графічний адаптер GeForce GTX 465 (11 мультипроцесорів, 352 ядра). Зважаючи на явні переваги ОС Linux для проведення високопродуктивних обчислень [2], використовувалась ОС openSUSE Linux 10.3 x86_64. Виходячи з наявного апаратного забезпечення, застосовувалась технологія NVIDIA CUDA. Пікова продуктивність на стандартних тестах становила 470 Gflops для чисел з одинарною точністю і 95 Gflops — з подвійною точністю. Проведено також обчислення зонної структури Si в рамках теорії функціонала густини (метод псевдопотенціалу), у якому на GPU виконувалися лише операції лінійної алгебри та швидкого перетворення Фур'є. У порівнянні з обчисленнями на хост-системі отримано приблизно двократний приріст продуктивності. Описана система може бути вдосконалена як шляхом нарощування потужності CPU та GPU, так і об'єднання вузлів у Beowulf-кластер і комбінації технологій CUDA/OpenCL і MPI. Планується широке використання системи у науковому та навчальному процесах факультету, а в перспективі — включення її у національну програму GRID.

1. <http://www.electronics.wups.lviv.ua/index.php?p=1&o=1&subf=0&idn=99>

2. Top500 Supercomputer Sites. <http://www.top500.org/>

Використання MACROMEDIA SHOCKWAVE PLAYER під час вивчення атомної та ядерної фізики студентами технічних спеціальностей

Чуйко Г.П., Яремчук О.М., Добровольська А.С.

*м. Миколаїв, Чорноморський державний університет ім. Петра Могили,
helga_30@mksat.net*

В даній статті розглядається застосування програмних продуктів під час вивчення атомної та ядерної фізики у вищих навчальних закладах технічного спрямування.

У зв'язку з важливістю поняття інформації останнім часом значна увага приділяється проблемам методики введення цього поняття в курсі інформатики як у вищих навчальних закладах [3, 4, 5]. Зокрема в роботі М.І. Жалдака [3] зроблено аналіз навчальних посібників і підручників для середніх навчальних закладів з метою виявлення некоректностей при використанні таких понять як інформація, оцінка і вимірювання інформації, її передавання, зберігання, опрацювання і т.д. В результаті цього аналізу, спираючись на роботи Ф.Л. Бауера, Г. Гооза [1], Д. І. Блюменау [2], Н. М. Моїсеева [4], А. П. Суханова [7], в яких інформація розглядається як результат відображення оточуючого світу у свідомості людини, зроблено такі висновки:

- “говорити в шкільних навчальних посібниках і підручниках про кількість інформації, об’єм інформації і т.п. принаймні некоректно”;

- “в шкільних навчальних посібниках і підручниках слід уникати слова інформація, використовуючи в разі потреби слова повідомлення, дані, матеріал і т. ін.”;

- “на будь-яких “носіях інформації” – в книгах, листах, на гнучких і жорстких дисках, компакт дисках, в мережі Internet і т. д., ніякої інформації немає, а є лише всеможливі повідомлення, програми і дані (які є теж повідомленнями про порядок виконання операцій чи про характеристики різноманітних об’єктів), аудіо і відеозаписи (які теж є повідомленнями)”;

- “некоректно говорити, що інформація зберігається в базах даних (сама назва говорить, що в базах даних зберігаються саме дані), в електронних таблицях, в графічних зображеннях тощо”;

- “спілкуючись, люди обмінюються повідомленнями, передають саме повідомлення, але не інформацію, знання, думки, емоції, оскільки людина, яка прийняла (чи віднайшла на довготривалих носіях) деяке повідомлення, може зрозуміти і витлумачити таке повідомлення не зовсім так або навіть зовсім не так, як людина, яка надіслала (чи залишила) таке повідомлення”.

Для того щоб технічні вищі навчальні заклади України відповідали вимогам сучасного суспільства, світовим тенденціям розвитку вищої освіти і могли забезпечувати високу якість підготовки і конкурентноспроможність своїх випускників на міжнародному ринку праці, зокрема з точки зору вміння застосувати інформаційно-комунікаційні технології в

навчанні, їм необхідно більш активно застосовувати:

- різноманітні програмні продукти, такі як Macromedia Shockwave Player, Macromedia Flash Player 8 (рис. 1);
- телекомунікаційні технології (корпоративну мережу ВНЗ, глобальну мережу Internet) для організації самостійної, індивідуальної, пошукової, науково-дослідної роботи викладачів і студентів.

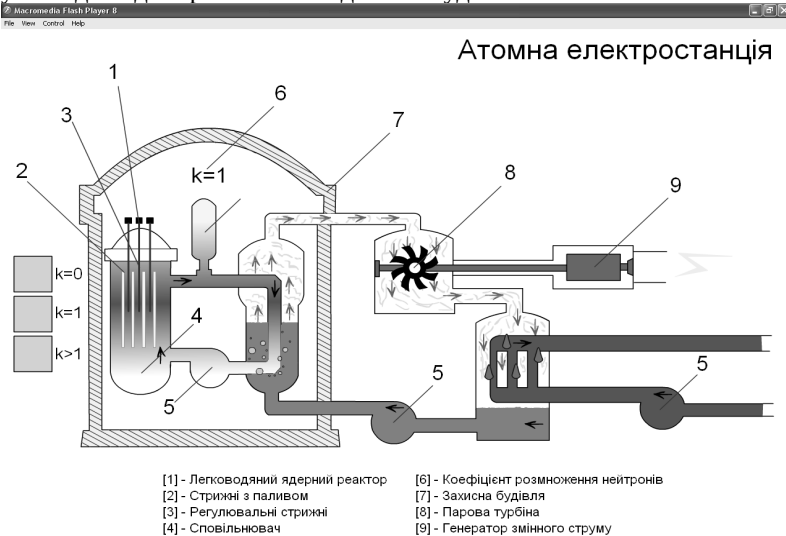


Рис. 1. Приклад анімації роботи атомної електростанції

Під час вивчення атомної та ядерної фізики студентами технічних спеціальностей, зокрема студентами ЧДУ імені Петра Могили спеціальностей «Приладобудування» та «Комп'ютерна інженерія», застосовують Shockwave - це технологія, яка дозволяє використати векторну 3D-графіку, Flash-ігри та інші інтерактивні елементи в Інтернеті. Саме для відображення цих даних був створений Shockwave Player. Він працює у вигляді плагіна с браузерами Internet Explorer, Opera, Mozilla Firefox, Netscape и Netscape сумісними. Без встановлення цієї програми інтерактивні сайти, створені за технологією Macromedia Shockwave, не будуть правильно відображатися у вищезазначених браузерах.

Flash Player – це віртуальна машина, на якій виконується завантажений з Інтернету код flash-програми. В основі анімації Flash лежить векторний морфінг, тобто плавне «перетікання» одного ключового кадра в інший. Це дозволяє робити складні мультиплікаційні сцени, задаючи лише декілька ключових кадрів. Продуктивність Flash Player при показі анімації в декілька разів перевищує продуктивність віртуальної машини Javascript в браузерах, що підтримують попередній стандарт HTML5.

Література

1. Бауэр Ф. Л., Гооз Г. Информатика: В 2 ч. – М.: Просвещение, 1990. – Ч. 1. – 336 с.

2. Блюменау Д. И. Информация и информационный сервис. – Ленинград: Наука. Ленинградское отделение. – 1989. – 192 с.
3. Жалдак М. І. Про деякі методичні аспекти навчання інформатики в школі і педагогічному університеті // Науковий часопис НПУ імені М. П. Драгоманова. Серія №2. Комп'ютерно-орієнтовані системи навчання: Зб. наук. праць / Редкол. – К.: НПУ імені М. П. Драгоманова. – №2 (9). – 2005. – С. 3–14.
4. Моисеев Н. Н. Алгоритмы развития. – М.: Наука, 1987. – 304 с.
5. Морзе Н. В. Методика навчання інформатики: Навч. посіб.: У 4 ч. / За ред. акад. М. І. Жалдака – К.: Навчальна книга, 2003. – Ч.ІІ: Методика навчання інформаційних технологій. – 287 с.
6. Морзе Н.В. Основи методичної підготовки вчителя інформатики: Монографія.– К.: Курс, 2003. – 372 с.
7. Суханов А.П. Информация и прогресс. – Новосибирск: Наука. Сибирское отделение. – 1988. – 192 с.

Про комп'ютерний супровід викладання геометрії

Яхненко І.В., Лутфуллін М.В.

Полтавський національний педагогічний університет ім. В.Г. Короленка
yaxnenko@yandex.ru; M.Lutfullin@i.ua

Обґрунтовано необхідність використання наочності у викладанні геометрії. Розглянуто можливості систем комп'ютерної математики, зокрема Maxima, для побудови унаочнень під час вивчення студентами аналітичної та диференціальної геометрії.

Такий навчальний предмет як геометрія важливий не тільки для підготовки спеціалістів, чия професія пов'язана з математикою, але і для розвитку будь-якої освіченої людини. Як в часи виникнення математики як науки геометричні фігури були містком, що пов'язували її із повсякденним життям, так і сьогодні застосування математичних досягнень до розв'язання різноманітних прикладних завдань вимагає розвиненого образного мислення, сформованих просторових уявлень і уяви.

З основними просторовими фігурами і їх властивостями людина знайомиться в дошкільному віці. Серед шкільних навчальних дисциплін найбільш важливе значення для розвитку просторового мислення мають геометрія та креслення, яке зараз викладають далеко не в усіх школах. Педагогічні дослідження показують, що більшість учнів і студентів мають наочно-образний тип мислення. Для людей з таким видом мислення наочність є необхідною для ефективного розв'язання задач і важливою ланкою при встановленні зв'язку нового поняття з уже відомими поняттями. В процесі вивчення аналітичної геометрії уявлення про просторові фігури повинні стати достатньо глибокими і служити для засвоєння властивостей пов'язаних з ними понять алгебри та аналізу.

Основним об'єктом геометрії як науки є просторова фігура, тому висновки, які роблять студенти при вивченні аналітичної геометрії, базуються на зображеннях фігур і спираються на наочні уявлення. Структура просторового образу істотно залежить від характеру наочної основи, оскільки вона допомагає створити студентам образ досліджуваного об'єкта, щоб в подальшому вони могли сформулювати правильні висновки.

Розуміння задач властивостей просторових фігур найкраще можна досягти дослідженням моделей цих фігур та виконанням їх якісних креслень. Проте виконання "твердої" моделі до кожної задачі неможливе, а креслення і рисунки студенти перших курсів виконують часто на незадовільному рівні, так що вони скоріше затумують розуміння задачі ні прояснюють його.

В умовах обмеженого навчального часу вельми корисним може бути використання систем комп'ютерної математики (СКМ), що дозволить швидко будувати моделі фігур та їх взаємного розташування і виконувати громіздкі обчислення. Звичайно, зображення на екрані на відміну від "твердої" моделі ми сприймаємо лише зором, але сучасні СКМ дозволяють

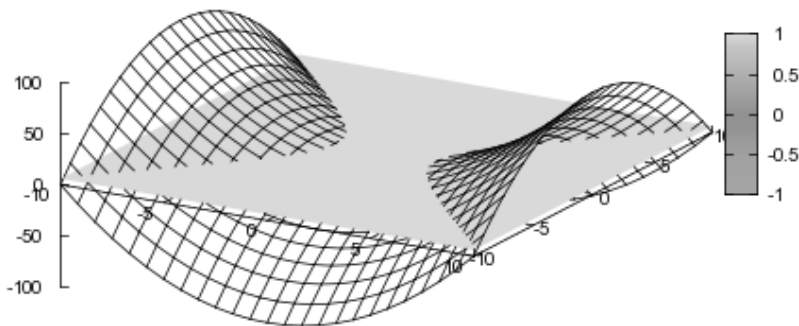
його розглядати з різних точок зору, крім того використання анімації дозволяє розглядати властивості, що залежать від параметрів (наприклад вигляд кривої від ексцентриситета, кількість точок перетину від коефіцієнтів рівнянь фігур) і розуміти динамічні процеси і перетворення (поверхні обертання, циклоїда, репер Френе).

При використанні СКМ збагачується досвід студентів, з'являється можливість експериментувати: порівнювати різні частини і елементи фігури; при виділенні і засвоєнні істотних ознак понять, що формуються, забезпечувати варіацію ознак фігури; доповнювати зображення необхідними елементами; виключати зайвий елемент; змінювати зображення окремих елементів об'єкта; коригувати місце окремих елементів у зображенні.

Серед СКМ досить популярними є Maple, Derive, GRAN 3D та інші [2]. У цій роботі ми розглядаємо можливості СКМ Махіма, що є вільним ПЗ і розвивається під ліцензією GPL [3,4]. Побудову графіків Махіма здійснює з допомогою пакета Gnuplot, який і самостійно може бути використаний для побудови геометричних зображень з навчальною метою.

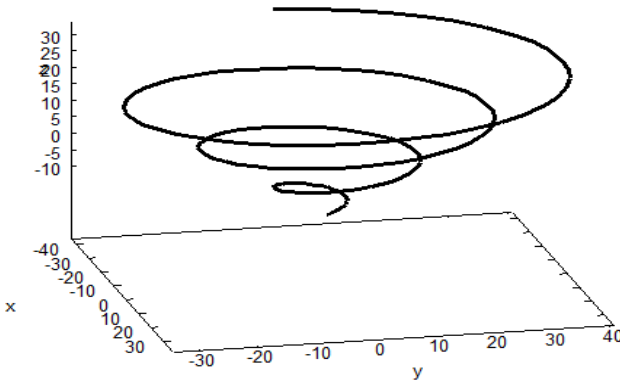
Далі наводимо декілька зображень, отриманих з допомогою цих ПЗ.

Перетин гіперболічного параболоїда і площини

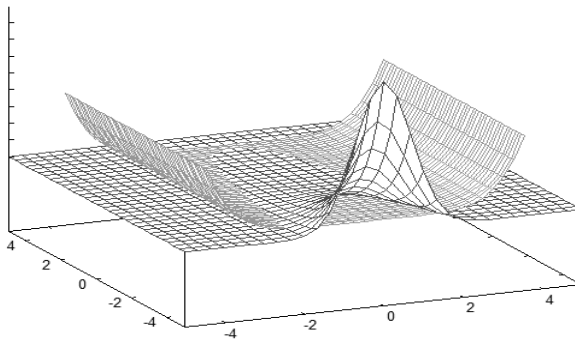


Наочність, зокрема комп'ютерна, допомагає студентам розвивати свою просторову уяву і формувати правильні і різносторонні уявлення про властивості геометричних об'єктів. Таким чином, вона протистоїть вербалізму, чисто словесному навчанню, проведеному у формі абстрактних міркувань, зміст яких не завжди зрозумілий студентам чи учням.

Конічна гвинтова лінія



Перетин поверхонь $z=300*\exp(-x^2-y^2)-1$ та $z=5*x^6+2*y$



Література

1. Ленчук И.Г., Лысенко Н.П., Лось Л.В., Фонарюк Е.В. Упорядочение профессиональных качеств студентов обогащением вычислительных предложений стереометрии конструктивным содержанием // Вісник Черкаського університету, № 150, 2009. (див. також http://www.nbu.gov.ua/portal/Soc_Gum/Vchu/N150/N150p152-164.pdf)
2. Сінько Ю.І. Системи комп'ютерної математики та їх роль у математичній освіті // [Електронний ресурс] <http://ite.ksu.ks.ua/?q=en/node/532>
3. Пронкевич С. Maxima — Минимализм в математике // [Електронний ресурс] <http://www.ky.by/index2009361105.htm>
4. Тарнавский Т. Maxima — максимум свободы символьных вычислений // [Електронний ресурс] <http://maxima.sourceforge.net/ru/maxima-tarnavsky-1.html>

The Open Data, Open Society report

Marco Fioretti

M. Fioretti, freelance writer/teacher/researcher (<http://mfioretti.com>) Email mfioretti@nexaima.net, <http://mfioretti.com>

This report is the first output of a research project about openness of public data in EU local administrations. The full report was finished in October 2010, is released under a Creative Commons cc-by license and can be freely downloaded from the web site of the DIME project ² or from Sant'Anna school. The report is also integrally republished on <http://stop.zona-m.net>³, split in separate pages with comments open to anonymous readers, in order to facilitate as much as possible feedback and discussion on each single part (but please do check the notes to readers first!). This report only contains the introduction about this.

Goals and structure of this report

The report discusses the current and potential role, in a truly open society, of raw Public Sector Information (PSI) that is really open, that is fully accessible and reusable by everybody. The general characteristics of PSI and the conclusions are based on previous studies and on the analysis of current examples both from the European Union and the rest of the world.

Generation, management and usage of data constituting what is normally called PSI is a very large topic. This report only focuses on some parts of it. First of all, we only look here at really “public” PSI, that is information (from maps to aggregate health data) that is not tied to any single individual and whose publication, therefore, raises no privacy issues.

It is also important to distinguish between actual raw data (basic elements of information like numbers, names, dates, single geographical features like the shape of a lake, addresses...), their results (more or less complex documents, policies, laws...) and the procedures and chains of command followed to generate and use such results, that is to vote or, inside Public Administrations, to take or implement decisions.

So far, discussion and research on Open Data at national level has had relatively more coverage, even if much of the PSI that has the most direct impact on the life of most citizens is the one that is generated, managed and used by local, not central, administrations and end users (citizens, businesses or other organizations). Creation of wealth and jobs can be easier, faster and cheaper to stimulate, especially in times of economic crisis, at the local level. Finally, open access to public data is much more necessary for small businesses than for big corporations, since the latter can afford to pay for access to data anyway (and high prices of data may also protect them from competition from

2 <http://www.dime-eu.org/editions>

3 <http://stop.zona-m.net/2010/01/the-open-data-open-society-report-2/>

smaller companies).

For all these reasons, the main focus of this report will be on the raw data that constitute “public” PSI as defined above. This is the reason why in this report the terms “raw data” and “PSI” are practically interchangeable. We will also focus on the local dimension of Open PSI, that is raw data directly produced by, or directly relevant for, local communities (City and Regions), and on their direct impact on local government and local economy.

Chapters 2 and 3 summarize the importance of data in the modern society and some recent developments on the Open Data front in Europe. Chapter 4 explains why raw PSI should be open, while Chapter 5 shows the potential of such data with a few real world examples from several (mostly EU) countries. Chapter 6 looks at some dangers that should not be ignored when promoting Open Data and Chapter 7 proposes some general practices to follow for getting the most out of them.

Why and how FOSS scripting languages are important at school and at home

Marco Fioretti

M. Fioretti, freelance writer/teacher/researcher (<http://mfioretti.com>) Email mfioretti@nexaima.net

FOSS scripting languages like Unix shells, Perl and Python are extremely productive, flexible and powerful programming tools for everybody. They are much easier to learn and deploy on any operating system than programs in compiled languages like C or C++, and yet they can automate or greatly speed up many tedious computer activities in any area of daily computing.

The topic of this paper is at the intersection of two tracks of the FOSS Lviv 2011 Conference: FOSS in Education and FOSS at home.

The central point of the paper is that scripting languages like Perl or the shell are not relevant only for system administrators and other software professionals and much easier than they look. These languages have no license costs and can be very efficient and useful tools even at school and at home, in at least two ways: on one level, they can save lots of time when studying or in normal home computing tasks, even on older/limited computers.

On another level, they are a very efficient, easy to set up and extremely portable way to teach programming, from primary school to adult, professional training classes. Besides, unlike other languages used to teach programming, these are tools directly usable and valuable when looking for a job (not just in the software industry, but in any business sector).

This paper discusses in depth all these theses and the fact that scripting languages are much easier to learn than normally thought. The paper also provides practical, immediately useful, real world examples of how non programmers (especially, but not exclusively teachers, students and small

business owners!) can quickly learn these tools and can benefit from them for their daily, non programming computer-related needs.

Promoting scripting languages in this way can also be an indirect, but very effective way to promote larger adoption of FOSS in all branches of society. The paper also discusses this point and suggest some strategies for effective promotion and teaching of these tools in schools and small businesses. In spite of its topic or, we should say, just because of it, the paper is non-technical and its intended audience is the general public (but especially teachers).

References:

OpenDocument Format scripting examples, M. Fioretti, 2010:
<http://freesoftware.zona-m.net/tag/odf-scripting/>

How to survive the GNU AUTOTOOLS

Rob Savoye (RS)

Open Media Now <rob@openmedianow.org>

Surviving the GNU Autotools is a technical talk for anyone that has to use the Autotools, namely Autoconf, Automake, and Libtool.

After a brief discussion of the history behind Autoconf and configuration tools in general, the basics of using the Autotools are discussed. This is oriented towards anyone who has to configure and build a free software package using the Autotools. Then I dive into the gory details of how to debug configure scripts and Makefiles. Some coverage of basic Unix shell utilities and bourne shell programming will also be covered.

Otwarta infrastruktura przetwarzania rozproszonego berkeley – wspomaganie międzynarodowych obliczeń naukowych

Swierczewski L.

College of Computer Science and Business Administration in Łomża,

University of Warsaw, lswierczewski@pwsip.edu.pl

Wraz z ciągłym rozwojem nauki i technologii informatycznych środowiska naukowe coraz bardziej odczuwają potrzebę wykorzystywania olbrzymich zasobów obliczeniowych. Nie wszyscy jednak mogą uzyskać dostęp do wysokowydajnych komputerów dużej mocy, których zakup i eksploatacja jest bardzo droga. Alternatywnym rozwiązaniem może być platforma BOINC, która umożliwia przetwarzanie rozproszone danych na komputerach połączonych siecią Internet i udostępnionych przez wolontariuszy.

W latach 40. XX wieku pojawiły się pierwsze urządzenia elektroniczne służące do dekryptażu szyfrogramów słynnej maszyny Enigma. To właśnie te lata najczęściej uznaje się za początek dziedziny nauki jaką jest Informatyka. Pomimo tego, że pierwsze komputery zbudowano około 70 lat temu to rozważań na temat możliwości konstrukcji różnego typu maszyn obliczeniowych i algorytmów, które mogłyby być przez nie realizowane dokonywano nawet setki lat temu. Informatyka – z łac. informatio co w tłumaczeniu oznacza "wyobrażenie" tak naprawdę wywodzi się z matematyki, w której kiedyś była tylko wyodrębnionym działem. Dział ten pod wpływem rozwoju nauki i techniki z czasem bardzo szybko się rozrósł i wyewoluował do oddzielnej dyscypliny. Ojcowie Technologii Informacyjnej najprawdopodobniej nawet nie wyobrażali sobie jak duży wpływ na ludzkość będzie miał ich twór i że komputery kiedyś będzie można odnaleźć praktycznie wszędzie. Dzisiaj bez wykonywania skomplikowanych obliczeń nie obyłyby się ani fizyka, ani chemia, ani biologia, z której ostatnio wydzielił się aktualnie bardzo dynamicznie rozwijany kierunek: bioinformatyka. Bez szybkich komputerów nie poradziłby sobie także przemysł, ani współczesne wojsko, które szczególnie nacisk kładzie na przepływ informacji i ultraszybkie rozwiązywanie problemów decyzyjnych i obliczeniowych.

Dzisiaj, w XXI wieku wszystkie liczące się ośrodki badawcze posiadają własne centra superkomputerowe. We wszystkich z nich pracują znakomici naukowcy i programiści. Od 1993 roku prowadzony jest prestiżowy ranking TOP500, w którym można odnaleźć najszybsze jawne komputery na Ziemi. Koszt budowy i utrzymania w pełnej sprawności tak wydajnych systemów komputerowych jest bardzo wysoki i wielokrotnie przekracza dziesiątki milionów euro. W pewnym stopniu alternatywą dla systemów scentralizowanych jest przetwarzanie rozproszone danych. Jedno z najciekawszych tego typu rozwiązań zaprezentował Uniwersytet Kalifornijski w Berkeley. BOINC1 - Berkeley Open Infrastructure for Network Computing jest znane szerokim gronom internautów z projektu SETI@Home2, w którym każdy kto posiada komputer i dostęp do Internetu może wziąć udział w analizie spektrum fal elektromagnetycznych i poszukiwaniu charakterystyk, które nie mogły powstać w naturalny sposób, a co za tym idzie - mogą być dowodem na istnienie cywilizacji pozaziemskich. Sama zasada działania BOINC jest bardzo prosta: naukowcy uruchamiają serwer, który generuje zadania i rozsyła je do komputerów internautów. Po wykonaniu przez maszyny wszystkich obliczeń wyniki są z powrotem odsyłane do serwera gdzie podlegają analizie przez zespół naukowy. Można mieć bardzo duże wątpliwości co do udostępniania zasobów i zachowania własnego bezpieczeństwa. Powinniśmy jednak wziąć pod uwagę fakt, że prace badawcze zazwyczaj prowadzone są przez bardzo renomowane instytucje takie jak Harvard University, Oxford University lub Massachusetts Institute of Technology. Ryzyko ataków i uzyskania nieautoryzowanego dostępu do systemu klienta istnieje zawsze – każdy komputer podłączony do globalnej sieci bez względu na zainstalowane oprogramowanie może być zaatakowany. Platforma UC: Berkeley jest

wykorzystywana w szerokiej gamie obliczeń dotyczących takich dziedzin jak: matematyka (PrimeGrid3, Collatz Conjecture4), fizyka (Leiden Classical5, LHC@home6), astronomia (Cosmology@home7, Einstein@home8) lub też biochemia (Rosetta@home9). Należy także dodać, że BOINC jest rozwiązaniem całkowicie darmowym z otwartym kodem, który jest rozwijany przez ludzi z całego świata. Oczywiście nie wszystkie problemy można rozwiązać z wykorzystaniem tego typu rozwiązania, jednak jak widać cieszy się ono na arenie międzynarodowej dość dużą popularnością głównie ze względu na stosunkowo niski koszt przeprowadzanych obliczeń, dzięki rozbiciu ich na wielu pasjonatów. Ogólne dostępne zasoby siatkowe gromadzą nie tylko znaczne moce obliczeniowe, które często nie ustępują tym „czysto” superkomputerowym, ale także budują społeczności ludzi zainteresowanych danymi zagadnieniami i angażujących się w rozwój badań, gdzie bez względu na wiek, wykształcenie, zawód lub narodowość każdy może mieć wpływ postępowania nauki.

Bibliografia:

1. <http://boinc.berkeley.edu/>
http://en.wikipedia.org/wiki/Berkeley_Open_Infrastructure_for_Network_Computing
2. <http://setiathome.berkeley.edu/>
3. <http://www.primegrid.com/>
4. <http://boinc.thesonntags.com/collatz/>
5. <http://boinc.gorlaeus.net/>
6. <http://lhcatome.cern.ch/lhcatome/>
7. <http://cosmologyathome.org/>
8. <http://einstein.phys.uwm.edu/>
9. <http://boinc.bakerlab.org/rosetta/>

Збірник наукових праць

ТЕЗИ МІЖНАРОДНОЇ НАУКОВО-ПРАКТИЧНОЇ КОНФЕРЕНЦІЇ
FOSS LVIV-2011

Під редакцією:
Злобін Г.Г., Апунович С.Є., Машков В., Апунович С.В.

Формат 60*84/16. Умовн. друк. арк. 10,4
Тираж 100. Зам.

Видавець та виготовлювач: Львівський національний університет імені
Івана Франка, вул. Університетська, 1, м. Львів, 79000

Свідоцтво про внесення суб'єкта видавничої справи до Державного реє-
стру видавців, виготівників і розповсюджувачів видавничої продукції:
Серія ДК№3059 від 13.12.2007 р.