

# Лаборатория беспроводных и мобильных технологий ПетрГУ-Nokia-NSN: организация и результаты

А. В. Воронин, С. И. Баландин, Ю. А. Богоявленский,  
Д. Ж. Корзун, К. А. Кулаков, А. И. Шабаев

Петрозаводский государственный университет



Конференция “Преподавание ИТ в России”, 17—18 Мая, 2010

# Содержание

## 1 Предпосылки создания лаборатории

- Фундаментальность образования - “Обратный” подход и открытое ПО
- Роль языка C
- Дисциплина “Разработка ПО”
- Аппаратно–программная инфраструктура поддержки открытого ПО — openSUSE — 1994—2000—2009
- Командные студенческие проекты

## 2 Лаборатория: основные моменты

- Направления работы
- Модификация учебного плана
- Обучение и развитие инфраструктуры
- Волны проектов разработки

## 3 Заключение. Полученные результаты



# Фундаментальность образования - “Обратный” подход

Анализ Ядер совокупностей базовых знаний определенных в Computing Curricula 2005 дисциплин Информатика, Информационные Системы, Разработка ПО и Информационные Технологии:

<http://www.cs.karelia.ru/news/2006/files/sorucm-ybgv-ru.pdf>

- Ядро ИН входит в остальные Ядра
- Все Ядра содержат много математических курсов
- Стандарт бакалавриата 010500 «Прикладная математика и информатика»:
  - ▶ формирует математическую культуру
  - ▶ позволяет включить в учебный план ядра ИН, ИС, РПО, ИТ

**“Обратный” подход - включение необходимых инженерных дисциплин в стандарты, обеспечивающие высокую математическую квалификацию**



# Фундаментальность образования - открытые программные платформы. 1

Система лицензий open source, free

- ОПП — открытый код + бесплатный доступ
- Архитектуры ЭВМ — рост сложности, разнообразия
- Важнейшее свойство ОС — переносимость
- Linux — грид, кластеры, мейнфреймы, ПК, смартфоны, Интернет планшеты

**Access Linux Platform, Google Android, LiMo Platform, Maemo, Moblinux, Moblin, MotoMagx, Openmoko, Palm webOS, Qt Extended, Snow Leopard (iPhone), Ubuntu Mobile**



# Фундаментальность образования - открытые программные платформы. 2

Инструменты ОПП формируют:

- архитектурную культуру
- культуру прямого использования ЯВУ (C/C++)-системное программирование
- потенциальную способность к разработке ОС
- навыки владения классическими инструментами разработки
- способность к работе и в патентованных средах



# Роль языка С

Язык С с I семестра — далее практические занятия на открытых платформах

ВВЕДЕНИЕ В КУРС  
Язык программирования Си:  
краткая характеристика

- Небольшой, быстрый в освоении язык (функциональность вынесена в библиотеки)
- Один из первых в истории ЯВУ, сохранивший широкое распространение и в наши дни
  - Компоненты ядра и оболочек современных ОС
  - Реализация сетевых протоколов
  - Платформы и приложения для встроенных систем
- Минимум проверок на этапах трансляции и выполнения, свобода и производительность

© Петрозаводский Государственный Университет, 2009  
© Александр Бородин, 2009

<http://cs.karelia.ru/~aborod/inf/2010/slides/cslides-01.pdf>

Старший преподаватель А. В. Бородин: [aborod@cs.karelia.ru](mailto:aborod@cs.karelia.ru)



# Роль языка C

Язык C с I семестра — современные системы разработки ПО — опираются на идеи и конструкции C

## ВВЕДЕНИЕ В КУРС

Почему каждый программист должен владеть языком Си?

- Большинство «промышленных» технологий разработки наследуют синтаксис и библиотеки функций языка Си
  - C++, D
  - Perl, PHP, JavaScript
  - Java, C# (платформа .NET)
  - ...
- Язык стандартизован на международном уровне (ISO/IEC 9899:TC2)

© Петрозаводский Государственный Университет, 2009  
© Александр Бородин, 2009



# Роль языка С

Язык С с I семестра — С шире всех применяется на практике

## ВВЕДЕНИЕ В КУРС

Почему каждый программист должен владеть языком Си?

- Уверенно лидирует в списке самых востребованных языков (данные 2009 г.) для проектов с открытым кодом
  - С 40.34% (-0.6%)
  - C++ 13.43% (-0.6%),
  - Java 10.29% (-0.7%),
  - Javascript 7.6% (+2.1%),
  - Shell 7.05% (-1.9%),
  - PHP 5.19% (+0.3%),
  - Python 2.63% (-0.1%)
  - SQL 2.65% (+1.1%),
  - Perl 2.43% (-0.8%),
  - C# 1.32% (0.1%),
  - Ruby 1.01% (+0.2%),
  - Assembler 0.83% (-0.4%),
  - Pascal 0.73% (-0.2%),
  - TCL 0.28% (-0.1%),

© Петрозаводский Государственный Университет, 2009  
© Александр Бородин, 2009





# Дисциплина “Разработка ПО” в ПетрГУ. Общие положения

## Направления подготовки на математическом факультете

- Прикладная математика и информатика (1993)
- Информационные системы (1999)

## Учебные планы

- Специалист (5 лет), Бакалавр + Магистр (4+2 года)
- ACM Computing Curricula (1991, 2001, 2005)
- Хельсинкский университет: Общее ядро учебного плана (с 2001 г.)
- Программа FRUCT :  
Программирование для MID (с 2008 г.)

## Командные студенческие проекты: Иницирующие (2003–2004 гг.)

- Научно–ориентированный *Web-SynDic*,  
<http://websyndic.cs.karelia.ru/>
- Совместный проект *DaCoPan*  
(с Хельсинкским университетом),

**Web-SynDic system**

Architecture diagram showing a Client connected to an Internet cloud, which is connected to a Web-SynDic server. The server components include: Ask Server, Algorithm Server, Data Ware, External Server, External Index, External Converter, and External Generator.

**Input:**  
 $x1 + x2 = 2x3 + x4$   
 $x5 + x6 + x7 + x8 = x9$

**Output:**  
 $x1 = 2x3 - x4$   
 $x5 = x9 - x6 - x7 - x8$

**Start-up/progress:**  
 Requirement analysis: Jul – Aug 2003  
 Design: Sep – Nov 2003  
 Implementation: Nov – Dec 2003  
 Testing: December – Jan 2004  
 (a working version: Mar 2004)  
 Release: Aug – Nov 2004

Department of Computer Science 27

**DaCoPan Project**

Start: 09/2003  
 Finish: 01/05/2004

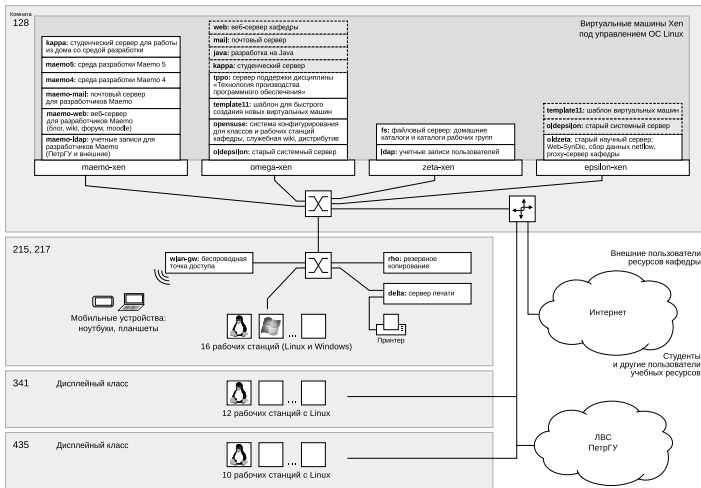
Implementation: 09/2003 - 01/2004  
 Documentation: 01/2004 - 01/2004  
 Test: 01/2004 - 01/2004

Forum: 01/2004 - 01/2004  
 Test plan: 01/2004 - 01/2004  
 Integration testing: 01/2004 - 01/2004

International team



# Аппаратно-программная инфраструктура поддержки открытого ПО — openSUSE — 1994—2000—2009



# Командные студенческие проекты (с 2003/04 гг.), весна

## Команды 3–6 разработчиков (студентов)

- Организованный и контролируемый процесс
- Менеджер - студент
- 15 часов на человека в неделю, 15 недель

## Заказчики

- Сотрудники факультета, IT парк, компании
- Требования к ПО
- Аттестация

## Инструктор

- Баланс между обучением и получением продукта
- Мониторинг прогресса, консультации, контроль
- Оценивание



# Лаборатория беспроводных и мобильных технологий ПетрГУ-Nokia-NSN

## Схема работы



## Направления работы лаборатории

- Разработка учебных курсов по OS Маемо (среда Moodle) и OS Symbian
- Подготовка разработчиков и экспертов для промышленных проектов (например в IT парке)
- Разработка близких к промышленным проектам в OS Маемо
- Акцент на исследовательскую составляющую
- Интеллектуальные сетевые пространства - Smart Spaces
- Разработка ПО в OS Symbian
- Кооперация в рамках Программы FRUCT — Finnish Russian Universities Cooperation in Telecommunications



## Модификация учебного плана 2009 — 2010

Семестр	Дисциплина
Первый курс	
1	Введение в программирование на языке C Введение Shell
2	C++ и структуры данных Архитектура IA-32 с элементами ассемблера gas Разработка интерфейсов в средах GTK/Qt
Второй курс	
3	Компьютерные сети UNIX Programming Программирование в среде Java
4	Операционные системы Программирование в OS Maemo
Третий курс	
5	Технология разработки ПО Программирование в OS Symbian



# Обучение и развитие инфраструктуры

Интенсивные курсы (1...6 дней)  
семестровый курс:

- Летняя школа, Авг. 2008 (ПетрГУ)
- Зимняя школа, Фев. 2009 (ПетрГУ)
- Тренинг Апр. 2009 (FRUCT5, ГУАП)
- Сент. 2009 (Нижний Новгород)
- Фев. 2010 (ПетрГУ, Qt)

Развитие инфраструктуры программных инструментов:

- Терминальный сервер с SDK
- Web сервер для доступа к учебным материалам (Moodle)
- Репозитории проектов SVN, Mercurial, Git
- Wiki студентов разработчиков
- Bugzilla для управления ошибками



# Командные студенческие проекты лаборатории

## Организация

- Инструктор выполняет многие функции менеджера
- Смешанные команды: студенты 1 – 6 курсов
- 18 часов на человека в неделю (15 для обычных проектов)
- Индивидуальные учебные планы
- Регулярные межпроектные семинары

## Фазы профессионального роста студентов

- Осенний семестр (3 курс): минипроект
- Весенний семестр (3 курс): подключение к проектам
- Лето (Июль–Август): Практика в IT парке, демо–прототип
- Осень/Весна: студенты 4 курса тестирование и публикация





## Сетевая поддержка разработки проектов

Ресурс: Открытые платформы для мобильных устройств

<http://oss.fruct.org>

### Синергия : Обучение+проекты ПО

- Новости
- Форум для Российских разработчиков
- Идеи, опыт и ПО от участников
- Учебные материалы и техническая документация
- wiki процесса разработки проектов

### Технологии реализации

- Wordpress
- PhpBB
- MediaWiki
- Moodle
- Bugzilla: <http://oss.fruct.org/bugzilla/>



## Выполняющиеся проекты: Первая волна, февраль 2009

- **MySocials**, C, 12168 LOC Клиент социальной сети VKontakte  
wiki: <http://oss.fruct.org/wiki/Маемо-VKontakte>  
исходный код: <http://gitorious.org/mysocials>
- **Kimi**: C, 10788 LOC Органайзер в пространстве событий Интернет  
wiki: <http://oss.fruct.org/wiki/Маемо-Google>  
исходный код: <http://gitorious.org/kimi>
- **Scribo**: Python, 7014 LOC Клиент Маемо для блога LiveJournal  
wiki: <http://oss.fruct.org/wiki/Маемо-Blogs>  
исходный код: <http://gitorious.org/scribo-client>
- **MobileTrade**: C++, 8964 LOC Клиент Маемо для web сервисов  
торговой или аукционной системы  
wiki: <http://oss.fruct.org/wiki/Маемо-Business>  
исходный код: <http://gitorious.org/mobile-trade>

Пакеты проектов загружаются на устройство N900 из  
репозитория: <http://маемо.org/packages/>



## Выполняющиеся проекты: Вторая волна, февраль 2010

- **MySocials-Testing:** Комплексное тестирование проекта MySocials  
wiki: <http://oss.fruct.org/wiki/MySocials-Testing>
- **Scribo-Testing:** Комплексное тестирование проекта Scribo  
wiki: <http://oss.fruct.org/wiki/Maemo-Blogs-Testing>
- **M3-Weather:** Клиент информации о погоде на платформе Smart-M3  
wiki: <http://oss.fruct.org/wiki/M3-Weather>
- **Sensors:** Приложение, использующее сенсоры для рисования  
wiki: <http://oss.fruct.org/wiki/Sensors>



# Исследовательский проект “Развитие платформы Smart-M3”

SmartSlog: ПетрГУ генератор библиотек ANSI C для онтологий Smart-M3

<http://sourceforge.net/projects/smartslog/>

- Отображение OWL описания онтологии в код ANSI C (библиотека онтологии)
- API для программирования агентов Smart-M3
  - ▶ процессор знаний КР
  - ▶ операции в высокоуровневых элементах сущностей онтологии вместо низкоуровневых элементов на основе триплетов
- Платформа Smart-M3 разрабатывается:
  - ▶ В Программе Artemis JU в Софии (“разумные” объекты для интеллектуальных приложений)
  - ▶ В национальном Финляндском исследовательском проекте DIEM (Device interoperability ecosystem)



# Характеристики процесса разработки проектов

- Унификация схожих частей проектов
- Еженедельные межпроектные семинары
- Проектирование на основе прецедентов
- Локальное хранение данных (SQLite или RDF)
- Стиль кодирования
  - ▶ вариант стиля K&R для C/C++
  - ▶ инструмент Doxygen для самодокументирования кода
- Использование разных инструментов (C, C++, Qt, GTK, Python, SQLite, RDF)
- Автоматизированное системное тестирование
  - ▶ основано на прецедентах
  - ▶ действия с интерфейсом заменяются тестирующим кодом (предопределенные сценарии и проверки)
  - ▶ каждый тест содержит несколько проверок
- Соответствие нормам разработки интерфейса
- Анализ потребления ресурсов с помощью инструмента Valgrind



## Взаимодействие с международным сообществом разработчиков

- В проектах активно используются свежие технологии
- Члены наших команд сообщают авторам о недочетах в продуктах через bugzilla, форумы, листы рассылки

Некоторые недавно обнаруженные недочеты

- Утечки памяти в PySide (версия языка Python для Qt)
- Некоторые классы Qt не поддерживают работу с серверами proxy
- Некоторые классы Qt не полностью поддерживают протокол SOAP
- Наши студенты перенесли в Maemo 5 fremantle библиотеку работы с форматом json и поддерживают ее



# Заключение. Полученные результаты

- Маето 5, Маето 6 → MeeGo ...
- Современные инструменты внедрены в учебный процесс
- Создан коллектив, разработаны сетевые проекты
- Разработан и реализован процесс воспитания разработчиков
- Интеллектуальные пространства (Smart Spaces), платформа Smart-M3

<http://www.cs.karela.ru>

<http://oss.fruct.org>

<http://wiki.fruct.org>



Спасибо за внимание!

Зав. кафедрой Информатики и математического  
обеспечения Ю. А. Богоявленский: [ybgv@cs.karelia.ru](mailto:ybgv@cs.karelia.ru)

