molpit
International open laboratory of advanced science and technology

Make IT

Разработка компонентного программного обеспечения для научного моделирования и инженерии

Денисов Иван Андреевич

аспирант СФУ

План

1. Компонетно-ориентированное программирование и языки семейства Оберон
2. Основы работы в среде BlackBox Component Builder
3. Примеры использования среды для математического моделирования и инженерии

Наследование реализации — это GOTO девяностых

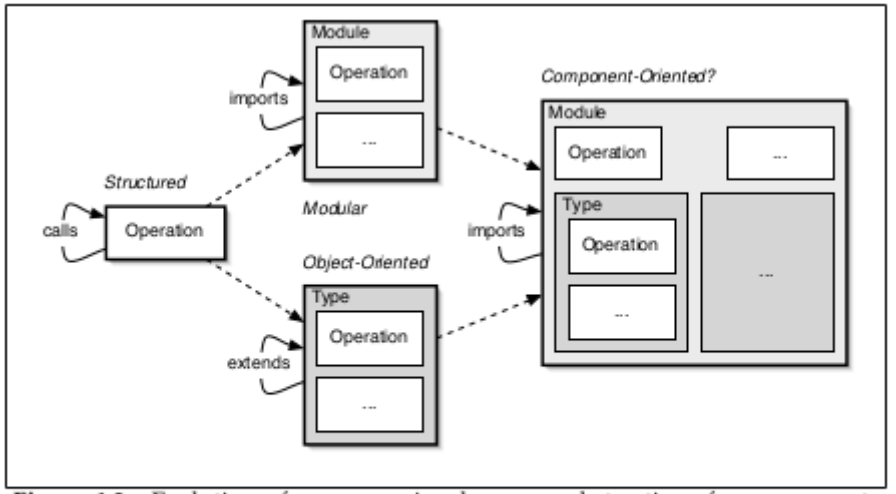


Figure 1.2: Evolution of programming language abstractions for components through various software development paradigms. (Dashed arrows indicate evolution, repeated arrows for recursive relationships omitted for clarity.)

Десятилетие	Технология программирования	Ключевое нововведение
1940-е	машинные коды	программируемые машины
1950-е	ассемблерные языки	символы
1960-е	языки высокого уровня	выражения и независимость от машины
1970-е	структурное программирование	структурные типы и управляющие конструкции
1980-е	модульное прогаммирование	отделение интерфейса от реализации
1990-е	объектно-орентированное пр-е	полиморфизм
2000-е	компонентно-ориентированное пр-е	динамическая и безопасная композиция

Таблица 4-1. Эволюция императивного программирования

Component-Oriented Programming Languages: Why, What, and How (2003) Peter Froehlich [University of California, Irvine, USA]

http://oberoncore.ru/library/pfister_component_software4

Эволюция парадигм программирования

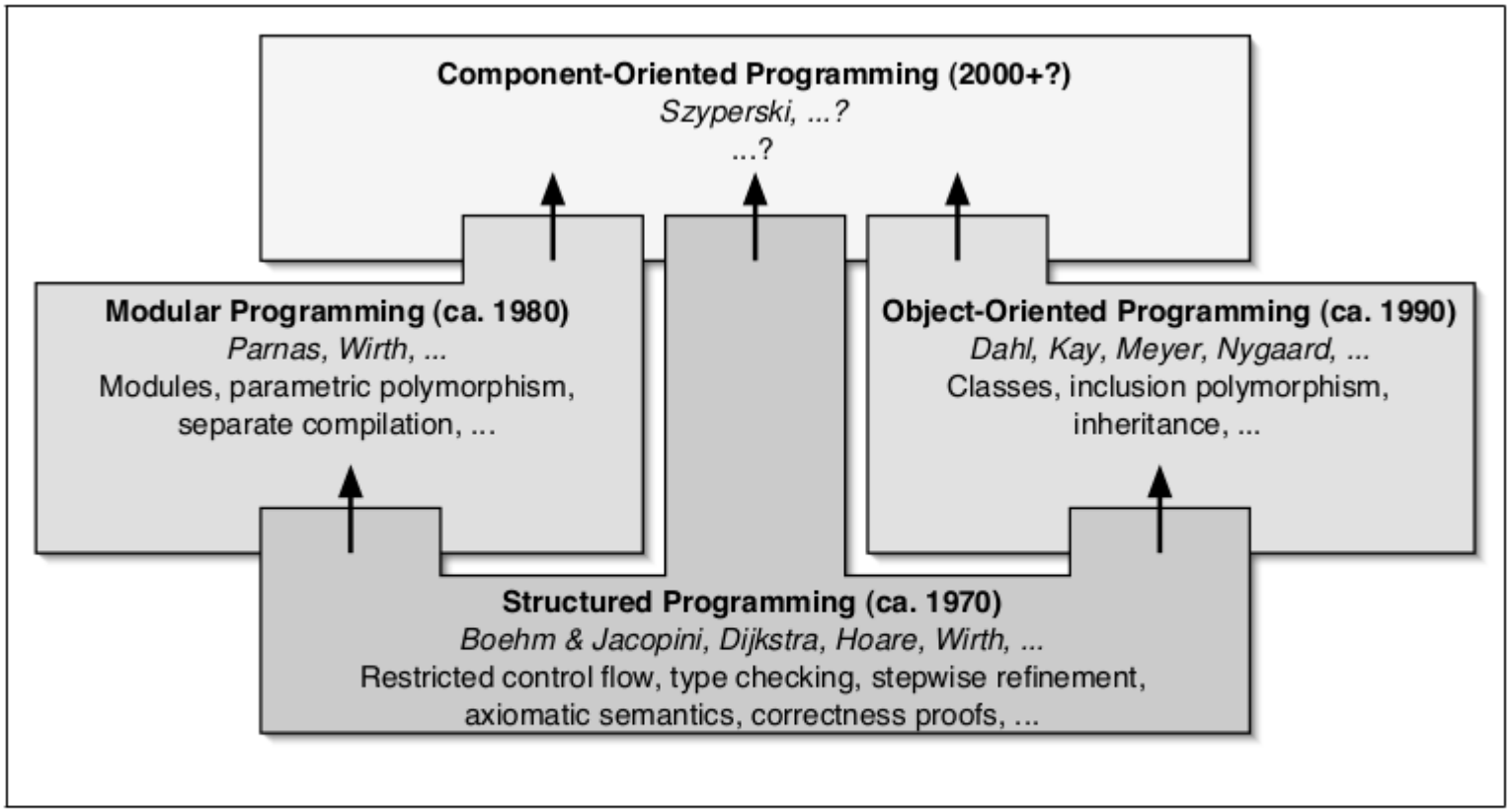


Figure 2.4: A biased view on the evolution of software development paradigms. Arrows indicate the flow of (certain) concepts, dates are approximate and (highly) debatable.

ООП

Полиморфизм, позднее связывание и сокрытие информации работают совместно, чтобы сделать возможным четкое разделение интерфейса и реализации, и следовательно — обеспечить поддержку для компонентного ПО.

http://oberoncore.ru/library/pfister_component_software4
Pfister C. Component Software — 1997, Cuno Pfister (Oberon Microsystems, Inc.)

Компонентное ПО

Компонентное ПО — программная система, собираемая из независимых (друг от друга) блоков по их спецификации кем угодно. Эти блоки (компоненты) — продукт производства разработчиков ПО. «Традиционный» подход разработки отличает как раз единица продукции — готовая, *монолитная программа*, сборку которой осуществляет (контролирует) только её разработчик.

Компонент — элемент конструкции ПО, сравниваемый с **«чёрным ящиком»**:

- зафиксирован *интерфейс экспорта*, через который предоставляются «услуги» другим компонентам;
- зафиксирован *интерфейс импорта*, через который затребуются "услуги" других компонентов;
- детали реализации интерфейса скрыты.

Компонент является единицей упаковки, развертывания и загрузки.

Компонентно-ориентированное программирование

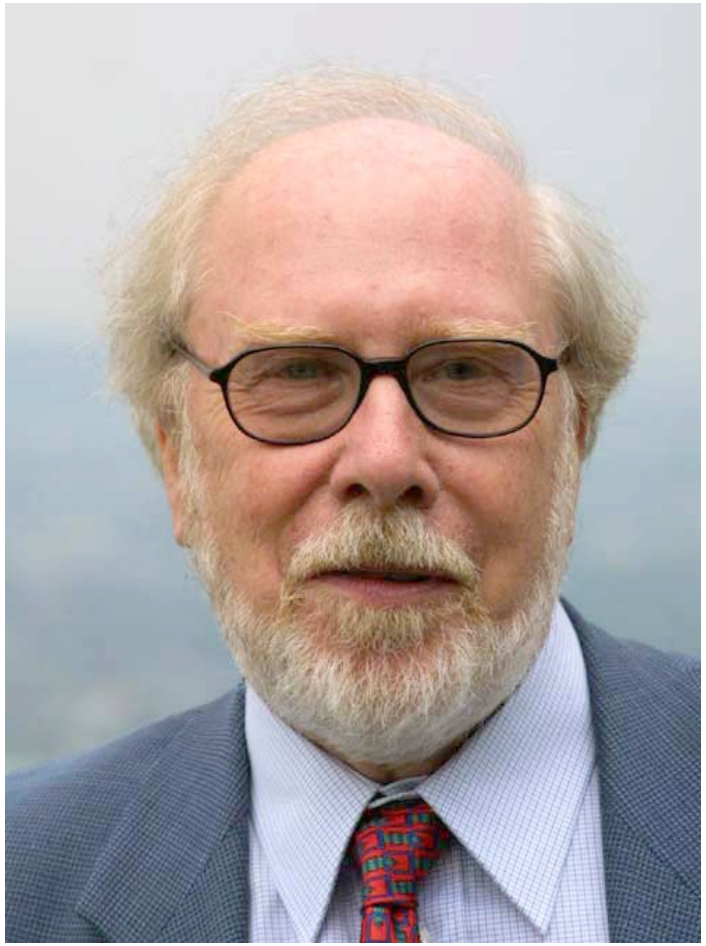
- Современный язык программирования — это нечто большее, чем только нотация для реализации мелких алгоритмов.
- Хорошо спроектированный язык программирования также поддерживает программирование в большом. Чтобы оставаться управляемыми, большие программы должны разбиваться на компоненты, которые взаимодействуют только через определенные интерфейсы. Хороший язык программирования может использоваться не только как язык реализации, но также и как язык описания и спецификации интерфейса.

http://oberoncore.ru/library/pfister_component_software4
Pfister C. Component Software — 1997 (Oberon Microsystems, Inc.)

Отличия от ООП

- Компонент — «независимый модуль программного кода, предназначенный для повторного использования и развертывания».
- Может содержать «множественные классы».
- Как правило, независим от конкретного языка.

Никлаус Вирт (Niklaus Wirth)



род. 15.02.1934 (Швейцария)

Автор языков:

- (1963) Euler
- (1966) Algol-W
- (1968) PL360
- (1970) Паскаль
- (1976) Modula
- (1979) Modula-2
- (1988) Оберон

http://www.oberon2005.oberoncore.ru/

Сложность синтаксиса языков программирования

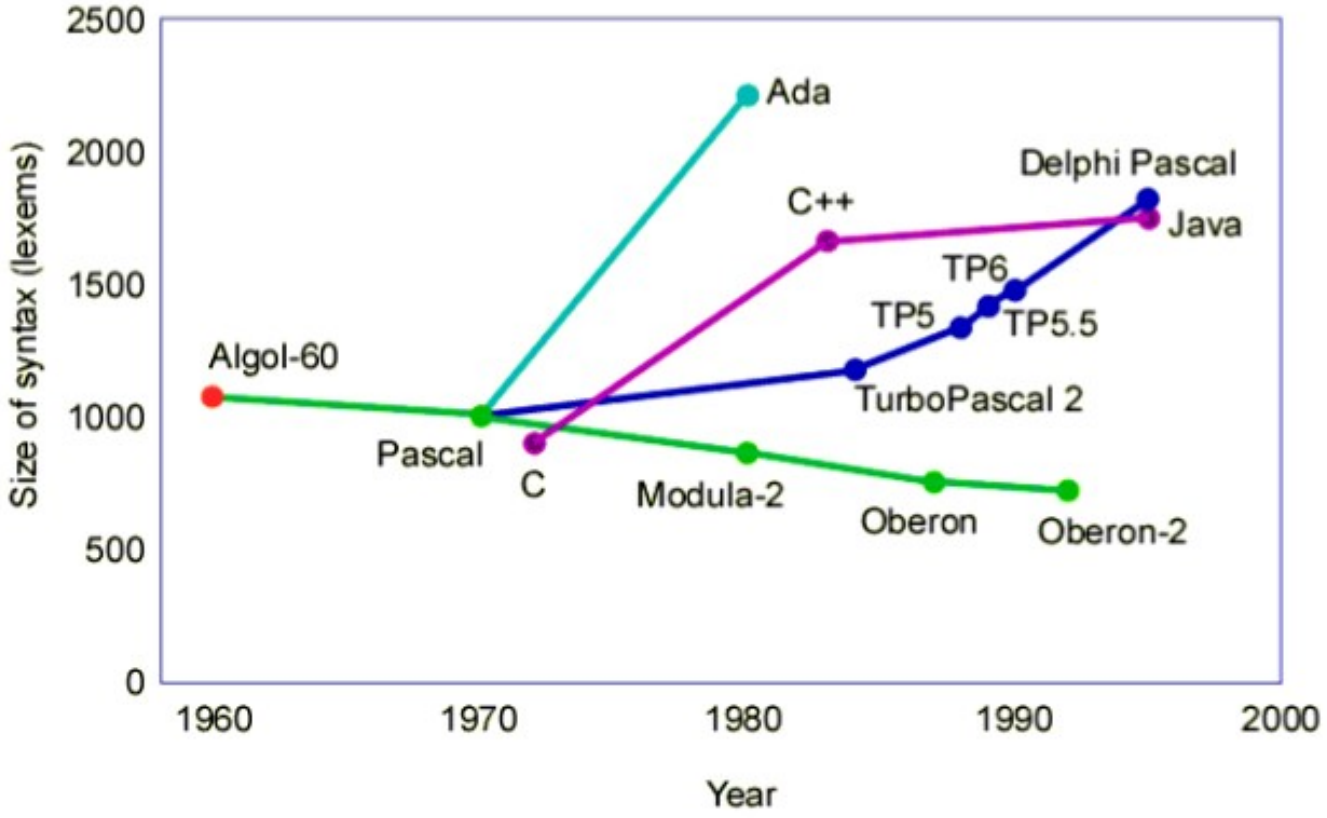


Figure 1. Complexity of syntax of programming languages

S.Z.Sverdlov (University of Vologda, Russia)

История КОП

- Основу компонентой парадигмы как образа рынка компонентов заложил Douglas McIlroy в 1969.
- В 1987 году Никлаус Вирт, унифицировав, предложил для языка Оберон паттерн написания блоков. Блок, удовлетворяющий требованиям этого паттерна, называется компонентом. Данный паттерн сформировался при изучении проблемы хрупких базовых классов, возникающей при построении объемной иерархии классов. Паттерн заключался в том, что компонент компилируется отдельно от других, а на стадии выполнения необходимые компоненты подключаются динамически.
- В 1989 году Бертран Мейер предложил идею единого взаимодействия между вызываемым и вызывающим компонентами. Эта идея воплотилась в виде готовых решений: CORBA, COM, SOAP и протоколу JAVA. Впоследствии поддержка со стороны языка осуществилась в компонентном Паскале.

McIlroy, Malcolm Douglas (January 1969). "Mass produced software components". Software Engineering: Report of a conference sponsored by the NATO Science Committee, Garmisch, Germany, 7-11 Oct. 1968. Scientific Affairs Division, NATO. p. 79.

Oberon

Подлинной жемчужиной творчества Вирта стал проект Oberon (1988). Созданная почти два десятилетия назад система Oberon (Oberon System, http://www.oberon.ethz.ch) играет в наши дни приблизительно ту же роль, что в начале 1980-х годов играли проекты Alto и Xerox Star знаменитого центра Xerox PARC, откуда взяли начало современные персональные компьютеры и текстовые редакторы. Для таких корпораций, как Microsoft, IBM и Sun Microsystems, проект Oberon стал источником плодотворных идей, среди которых можно выделить документо-ориентированный интерфейс, браузеры, промышленные языки разработки ПО (Java и C#), машинно-независимый мобильный код (JVM и .NET CLR), аплеты, компонентное ПО, динамическую компиляцию (JIT, AOC, DAC), смарт-теги, веб-службы и др.

ETH Oberon

Богатырев Р. Никлаус Вирт — патриарх надежного программирования // МИР ПК. ДИСК. 2004. С. 1–7.

Oberon-2

Оберон-2 был разработан в 1991 году в Швейцарской высшей технической школе (Цюрих) Никлаусом Виртом и Ханспетером Мёссенбёком, которые теперь работают в Институте системного программного обеспечения (SSW) Университета им. Иоганна Кеплера в Линце (Австрия). Оберон-2 — расширенный набор Оберона и полностью с ним совместим. Оберон-2 стал переработкой конструкции Объектного Оберона.

Кратко дополнения, внесённые в Оберон-2, заключаются в следующем:

- добавлены процедуры, связанные с типом, допускающие переопределение для порождённых типов (приблизительный аналог виртуальных методов в других объектно-ориентированных языках);
- в язык возвращен оператор цикла с шагом FOR;
- добавлена возможность экспорта описаний в режиме «только для чтения».

Компилятор **XDS** Новосибирской фирмы **Excelsior** для языков Modula-2 и Oberon-2 позволяет создавать приложения быстрее, чем написанные на C++!!!

http://www.excelsior-usa.com/xds.html

Компонентный Паскаль

«Ренессанс Оберона»

Oberon-2 был задавлен Java, которая вошла на рынок для которого был предназначен Oberon-2.

Новый виток в их жизни начался в том момент, когда в ETH созрела идея создания небольшой компании (Oberon microsystems), ориентированной на внутренний швейцарский рынок (в основном, это были промышленные роботы).

Главным мотором стал Куно Пфистер, а архитектором Component Pascal — Клеменс Шиперски.

Богатырев Р. Судьба Оберона // МИР ПК. ДИСК. 2005. С. 1–8.

Java vs. Component Pascal

Заключение

В качестве профессора американского университета я должен приветствовать Java уже по той причине, что этот язык позволяет безболезненно отказаться в преподавании от дидактически несостоятельных Си и Си++. Java представляет собой приемлемый компромисс: современный язык программирования, разработанный в традиции, близкой к Паскалю, Модуле-2 и Оберону, но с сознательной попыткой замаскировать это синтаксисом, сходным с синтаксисом Си. В качестве практика я, таким образом, приветствую язык Java везде, куда он планомерно входит как альтернатива Си или Си++. Однако те, кто уже обрел столь же мощный, но эстетически более совершенный язык программирования, такой как Оберон или Ада-95, спокойно могут держаться за него, не боясь оказаться в невыгодной "языковой ситуации".

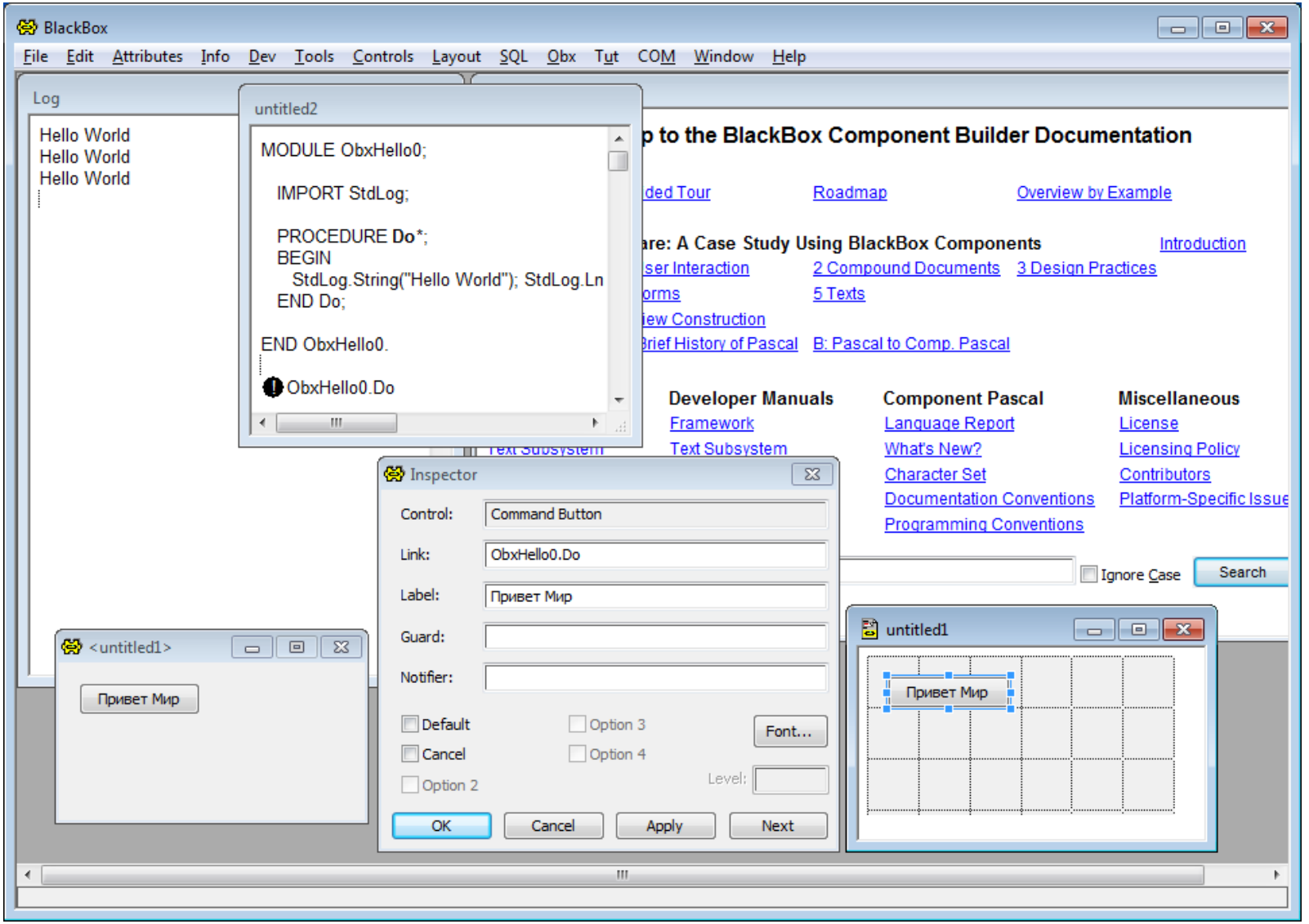
1. Пфистер К. Component Pascal и Java: что лучше // МИР ПК. ДИСК. 2005. С. 1–2.

2. Франц М. Java: критическая оценка // МИР ПК. ДИСК. 2005. С. 1–5.

Книги на русском

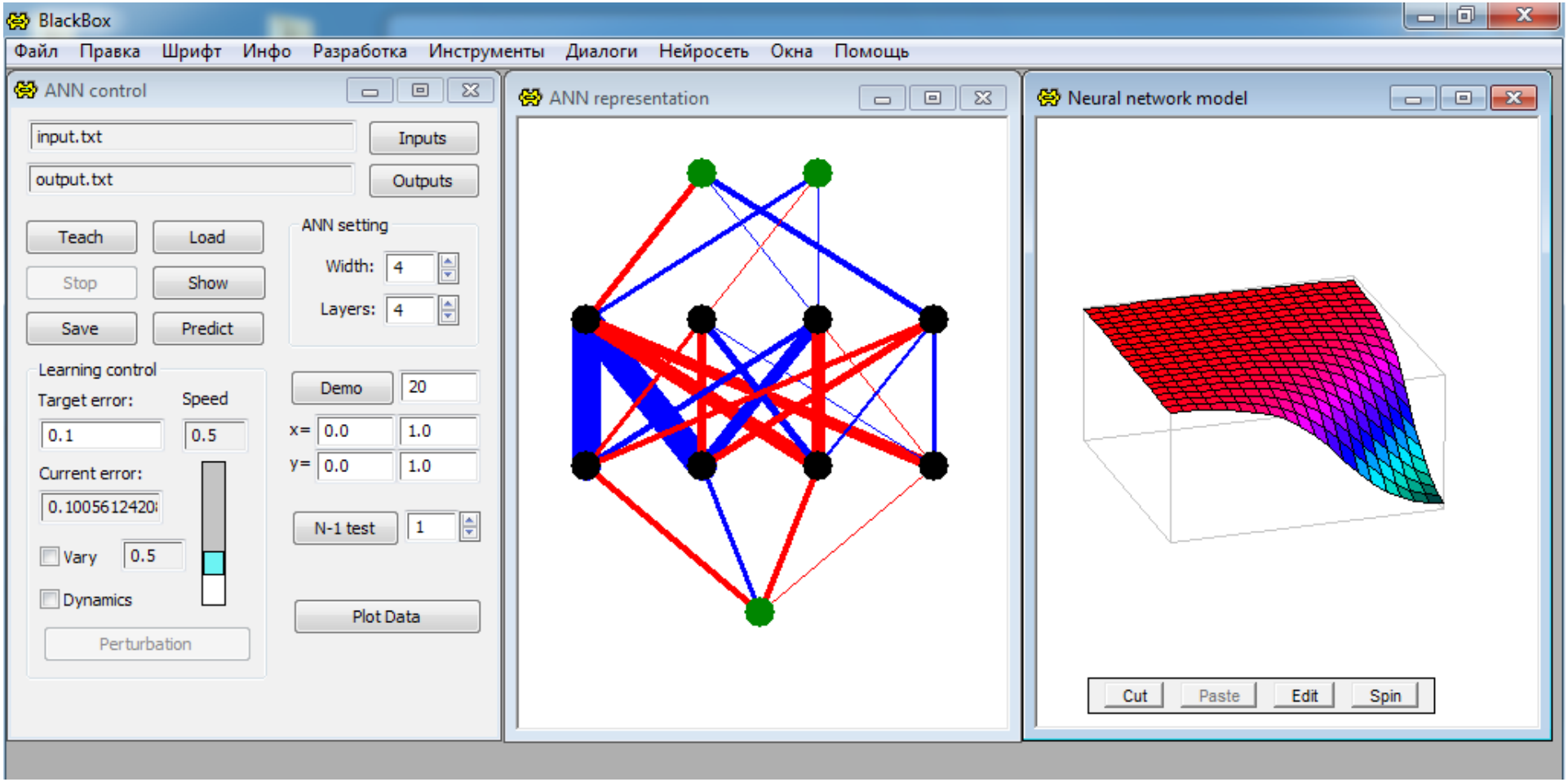
Информатика-21 международный общественный научно-образовательный проект

http://www.inr.ac.ru/~info21/literatura.htm

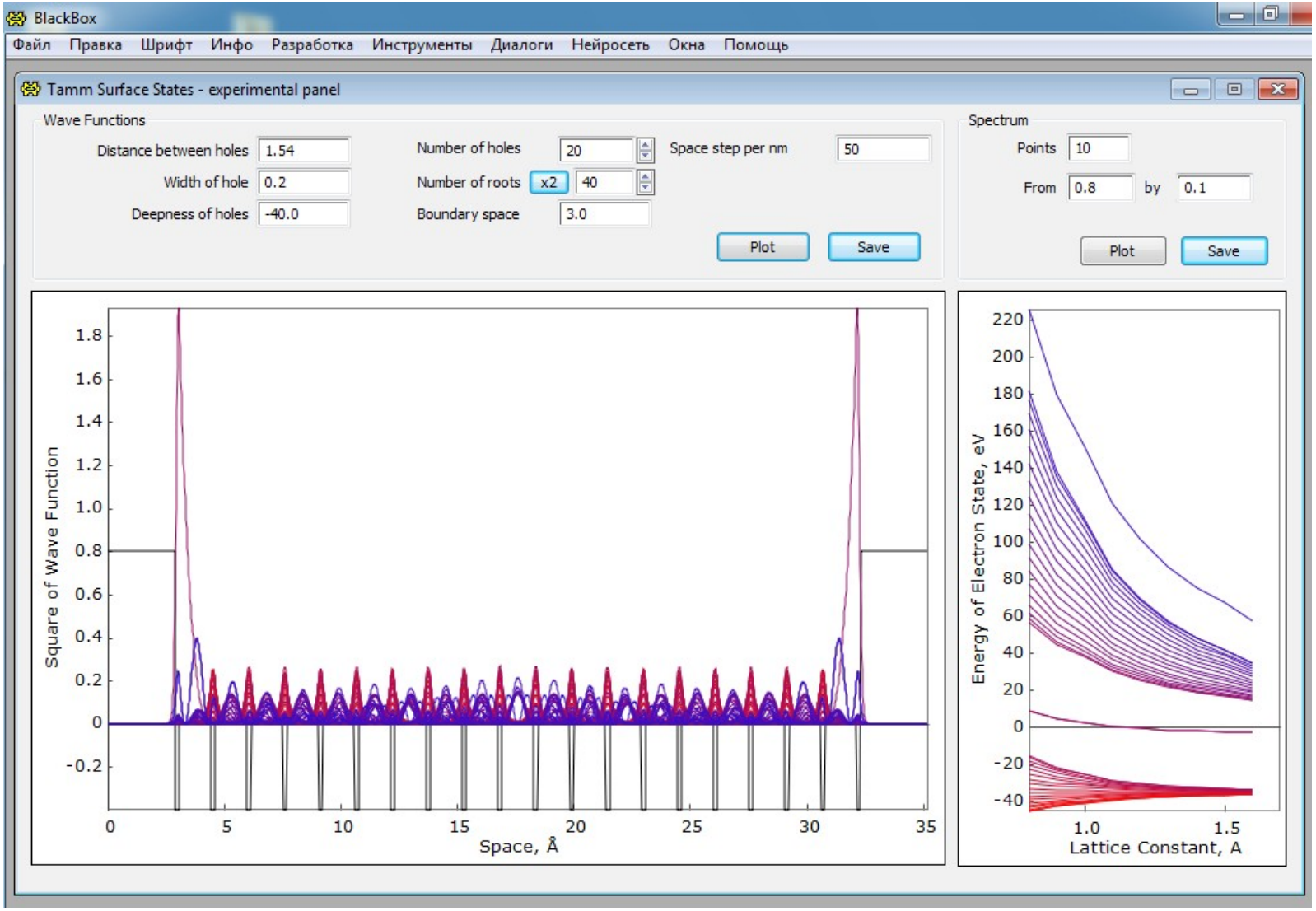

BlackBox
File Edit Attributes Info Dev Tools Controls Layout SQL Obx Tut COM Window Help
Log
Hello World
Hello World
Hello World
untitled2
MODULE ObxHello0;
IMPORT StdLog;
PROCEDURE Do*;
BEGIN
StdLog.String("Hello World"); StdLog.Ln
END Do;
END ObxHello0.
ObxHello0.Do
p to the BlackBox Component Builder Documentation
ded Tour
Roadmap
Overview by Example
are: A Case Study Using BlackBox Components
Introduction
ser Interaction
2 Compound Documents
3 Design Practices
orms
5 Texts
iew Construction
Brief History of Pascal
B: Pascal to Comp. Pascal
Developer Manuals
Framework
Text Subsystem
Component Pascal
Language Report
What's New?
Character Set
Documentation Conventions
Programming Conventions
Miscellaneous
License
Licensing Policy
Contributors
Platform-Specific Issue
Ignore Case
Search
Inspector
Control: Command Button
Link: ObxHello0.Do
Label: Привет Мир
Guard:
Notifier:
Default
Cancel
Option 2
Option 3
Option 4
Level:
Font...
OK
Cancel
Apply
Next
<untitled1>
Привет Мир
untitled1
Привет Мир

Примеры применения **BlackBox Component Builder** для моделирования и инженерии

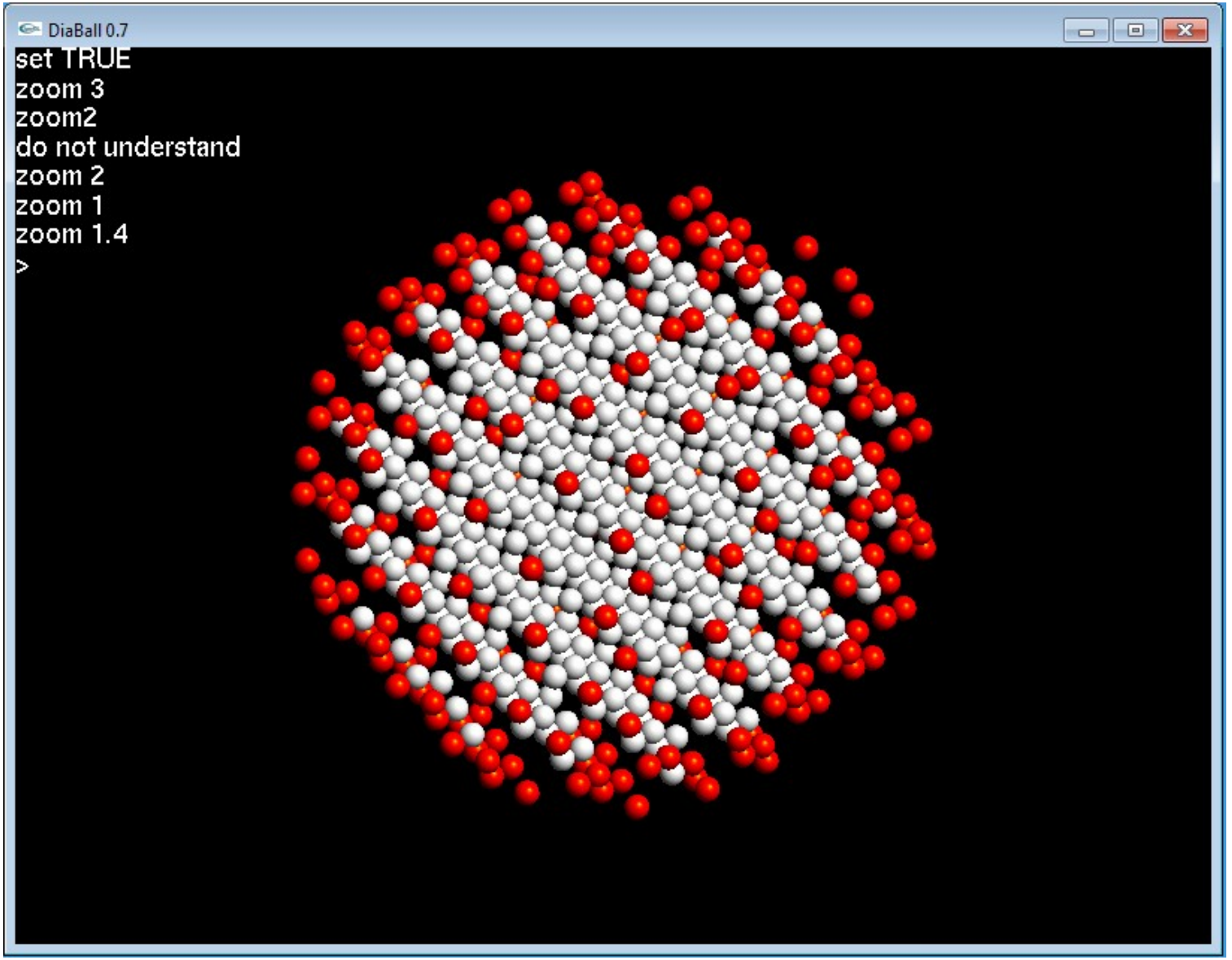
Нейросети


BlackBox
Файл Правка Шрифт Инфо Разработка Инструменты Диалоги Нейросеть Окна Помощь
ANN control
input.txt
Inputs
output.txt
Outputs
Teach
Load
Stop
Show
Save
Predict
ANN setting
Width: 4
Layers: 4
Learning control
Target error:
Speed
0.1
0.5
Current error:
0.1005612420
Vary
0.5
Dynamics
Perturbation
Demo
20
x= 0.0 1.0
y= 0.0 1.0
N-1 test
1
Plot Data
ANN representation
Neural network model
Cut
Paste
Edit
Spin

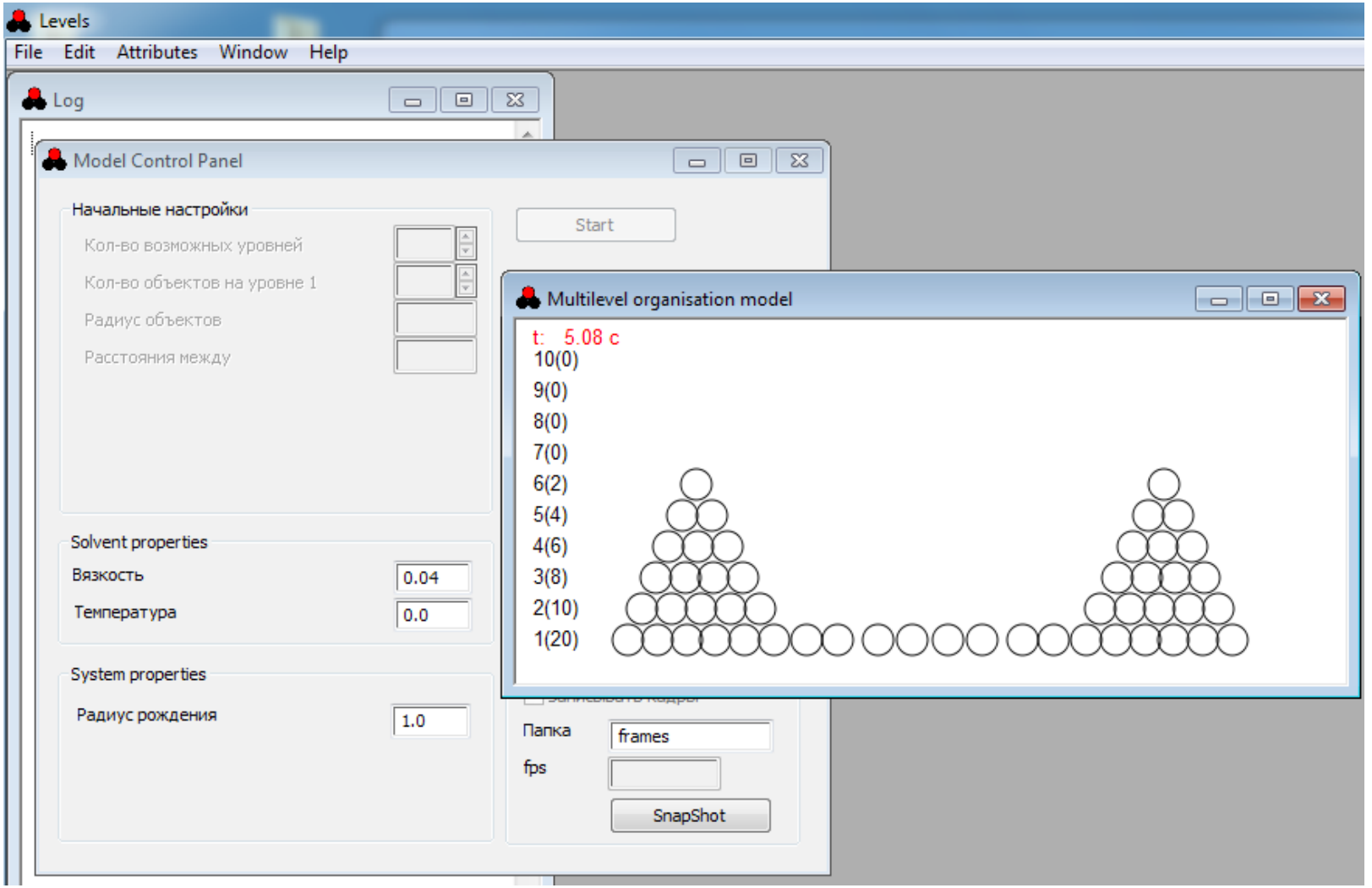
Квантовая механика



Физика твердого тела + OpenGL



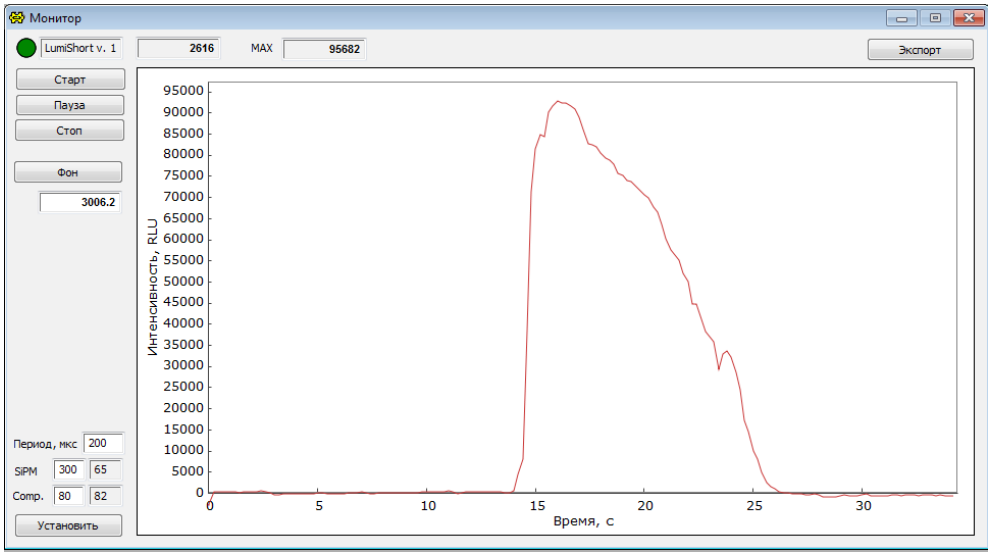
Динамика самосборки



Биолюминометр

Программа для МК LPC2103 (ARM7) написана на языке Oberon-07.

Приложения для сбора, визуализации и анализа данных написано на Компонентном паскале в среде BlackBox Component Builder



Astrobe для Oberon-07