

Санкт-Петербургский государственный университет  
Математико-механический факультет

Кафедра системного программирования

## Разработка средств для создания обучающих демонстраций в среде QReal:Robots

Курсовая работа студента 371 группы  
Чернова Дмитрия Владимировича

Научный руководитель

аспирант кафедры системного  
программирования Мордвинов Д.А.

Санкт-Петербург  
2014

[Разработка средств для создания обучающих демонстраций в среде QReal:Robots](#)

[Введение](#)

[Постановка задачи](#)

[Обзор существующих решений](#)

[Обучающая инфраструктура в различных программных продуктах](#)

[Актуальность задачи](#)

[Реализация](#)

[Обучающие демонстрации](#)

[Интерпретация скриптов](#)

[API пользовательского интерфейса](#)

[Информационные сообщения и указательные стрелки](#)

[Фасад к пользовательскому интерфейсу](#)

[Заключение](#)

[Результаты](#)

[Дальнейшее развитие](#)

[Список литературы](#)

## **Введение**

В недалеком прошлом возможностью использовать ЭВМ могли позволить себе лишь университеты и крупные компании. Поэтому чаще всего только сам программист мог пользоваться тем, что он создал. Через несколько десятилетий персональные компьютеры появились почти у каждого человека, поэтому появилась проблема обучения возможностям программного продукта рядового пользователя. И в настоящее время эта проблема актуальна для всех программных систем, предназначенных для людей, не связанных с программированием или обучающихся ему.

Большое количество современных программных продуктов нуждаются в технической поддержке и подробной справке. Причем этот этап занимает больше всего времени и затрат в жизненном цикле программного обеспечения. Это доказывает факт того, что невозможно ограничиться только разработкой программного продукта. Даже хорошо продуманный пользовательский интерфейс не гарантирует для пользователя возможности решить его задачу. Столь же важно научить конечных пользователей взаимодействовать с ним.

Во многих заказных проектах должны быть люди, отвечающие за освоение программного продукта на стороне заказчика. Продукты для более широкой аудитории зачастую включают в себя обучающую инфраструктуру в том или ином виде. Наиболее интересно решение внедренной в программную среду инфраструктуры, так как оно зачастую наиболее наглядно и информативно.

## **Постановка задачи**

Целью данной курсовой работы стало исследование концепций обучения программным средам и применение полученных знаний в реализации обучающей инфраструктуры в среде визуального программирования роботов QReal:Robots — среды обучения основам программирования и кибернетики.

В связи с особенностями QReal:Robots, наилучшим решением является обучающая инфраструктура, встроенная в программный продукт. Это позволит проводить обучение непосредственно в процессе взаимодействия со средой.

Реализуемая технология должна взаимодействовать с пользовательским интерфейсом среды и дать возможность без труда разработать комплекс обучающих программ для обучения пользователя. В доказательство этого факта полученную инфраструктуру необходимо апробировать.

# Обзор существующих решений

## Обучающая инфраструктура в различных программных продуктах

Рассмотрим различные способы организации обучения программным средам.

На мой взгляд можно выделить три основных типа получения знаний о программном продукте:

- **Текстовый вариант (справка).** Является наиболее популярным способом организации обучения. Представляет собой хорошо систематизированный набор информации о программном продукте в текстовом формате.  
*Плюсы:* наименьшие временные затраты, наиболее полный вариант организации обучения.  
*Минусы:* затрудняет понимание, подробное и зачастую избыточное преподнесение информации, практическая часть обучения формируется самим обучающимся.
- **Видео-демонстрации.** Зачастую реализуется пользователями, чтобы показать решение какой-либо задачи с помощью обозреваемого программного продукта.  
*Плюсы:* наглядно показывает возможности, не требует наличия программного продукта на пользовательской машине.  
*Минусы:* в процессе видео демонстрации решается только одна конкретная задача, поэтому требует полного просмотра даже для разрешения простых проблем. Сильно привязана к интерфейсу версии, на которой производилась запись.
- **Обучающая инфраструктура, встроенная в программный продукт.** Этот способ используется, например, в компьютерных играх. Представляет собой обучающую программу, позволяющую приобретать знания о программном продукте, работая непосредственно с ним и только с ним.  
*Плюсы:* обучение при работе со средой позволяет более эффективно усваивать информацию.  
*Минусы:* затраты на разработку модели и встраивание в среду, при реализации необходимо затронуть все основные возможности программного продукта.

## Актуальность задачи

В данный момент в среде QReal:Robots присутствует лишь громоздкая справка о возможностях среды. Это вступает в конфликт с тем фактом, что целевой аудиторией данного программного продукта являются школьные учителя, а также их ученики, причем начиная с начальной школы. Данный способ организации обучающей инфраструктуры задает высокий порог вхождения, поэтому дети этого возраста скорее всего не смогут быстро усваивать информацию о среде, пользуясь этой справкой, а многие вообще сразу же ее закроют. Для учителей это также актуально, так как это поможет им составлять программу и проводить уроки с использованием среды QReal:Robots.

После сравнения всех типов организации обучения в программном продукте было принято решение разработать модель обучающей инфраструктуры с учетом возможностей и целей среды QReal:Robots.

В первую очередь необходимо выработать модель, в соответствии с которой будет происходить обучение возможностям среды. Она должна разрабатываться в соответствии с целевой аудиторией и обладать возможностью наглядно демонстрировать возможности среды. Ввиду целевой аудитории QReal:Robots: школьных учителей и учеников, были рассмотрены программные продукты низкого порога вхождения. Хорошим примером стали игры. Они имеют наиболее низкий порог вхождения и при этом используются почти каждым. Для наглядности рассмотрим процесс обучения в компьютерной игре Warcraft 3.

В этой компьютерной игре доступно обучение. В нем коротко и информативно описываются возможности при самостоятельном общении с программной средой. За несколько минут усваиваются основы и дальнейшее изучение проходит гладко. Пользователи данной компьютерной игры за короткое время усваивают свои возможности в ней и при этом не занимаются поиском нужной информации в справке и не смотрят обучающие видеоролики. Обучение слабо обособлено от самой игры, поэтому проходит почти с таким же интересом, как и полноценная игра.

# Реализация

## Обучающие демонстрации

Для среды QReal:Robots как для среды программирования, которой требуется максимально низкий порог вхождения, демонстрации внутри среды станут очень эффективным способом обучения возможностям программного продукта. Предполагается создание мощного API для QReal:Robots на скриптовом языке. Возможности, доступные в API должны позволить создать обучающую инфраструктуру в среде.

API добавит гибкости системе, обучающую инфраструктуру, встроенную в среду, для пользователей, а также новые возможности для разработчиков и тестировщиков.

Для создания демонстраций было решено использовать скрипты на языке QtScript (расширенный специально для библиотеки Qt ECMAScript). Этот язык был наиболее подходящим для задачи, так как он позволяет получать доступ к объектам в Qt, инструментария к языку программирования C++, с помощью которого и написано приложение QReal:Robots.

Также скриптовый язык обеспечит необходимую гибкость для создания множества обучающих демонстраций.

Последним шагом станет подготовка комплекса обучающих демонстраций, использующих реализованные средства. После реализации API необходимо проиллюстрировать его возможности, создав полноценную демонстрацию.

## Интерпретация скриптов

Библиотека Qt имеет в комплектации возможность встраивания интерпретатора скриптов на языке QtScript, причем с возможностью доступа к необходимым объектам.

В связи с этим в QReal:Robots был встроен интерпретатор QtScript с возможностью доступа к API пользовательского интерфейса.

В результате с помощью коротких и понятных скриптов реализуются демонстрации, взаимодействующие с пользовательским интерфейсом среды, плюс несколько возможностей, превращающих демонстрации в полноценное обучение.

## API пользовательского интерфейса

Действия, задаваемые скриптом, выполняются с помощью класса ScriptAPI (интерфейс приложения) к пользовательскому интерфейсу QReal:Robots. API предоставляет необходимые возможности, для создания обучающих демонстраций, используя малую долю информации о внутренней структуре кода. Демонстрации реализуются короткими скриптами. В реализуемые возможности входит:

- Управление виртуальным курсором;
- Отправка информационных сообщений;

- Отрисовка указательных стрелок, направленных на элементы пользовательского интерфейса;
- Доступ к пользовательскому интерфейсу многих частей системы, в том числе к компонентам интерфейса инструментального плагина Robots (двумерная имитационная модель);
- Отрисовка связей между блоками на сцене;
- Взаимодействие с палитрой;
- Редактирование свойств блоков;
- Взаимодействие с выпадающими списками (например, выбор сенсора).

Подсказка : Добро пожаловать в среду QReal:Robots!

Рис. 1.

Подсказка : Добро пожаловать в среду QReal:Robots!

Рис. 2.

На рис.1 изображено сообщение в начале его появления, на рис.2 его уничтожение с использованием анимации свойства прозрачности.

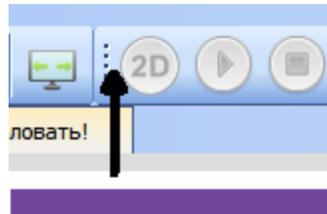


Рис. 3

На рис.3 изображена стрелка, указывающая на кнопку окна 2d модели робота.

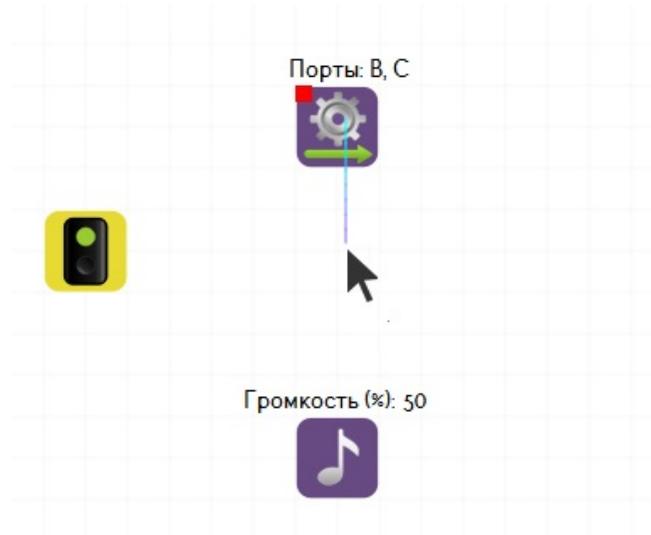


Рис. 4.

На рис. 4 изображен процесс соединения связью двух элементов на сцене.

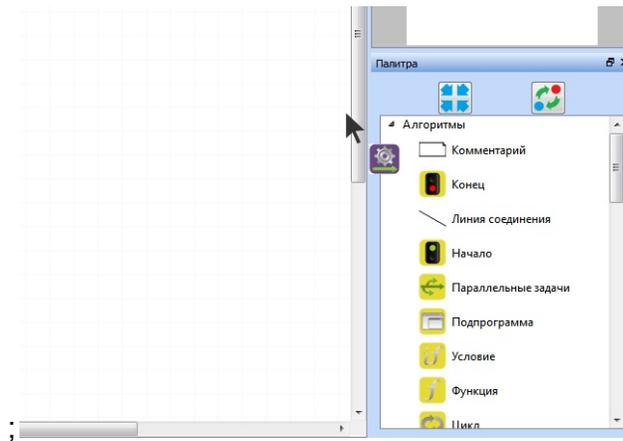


Рис. 5.

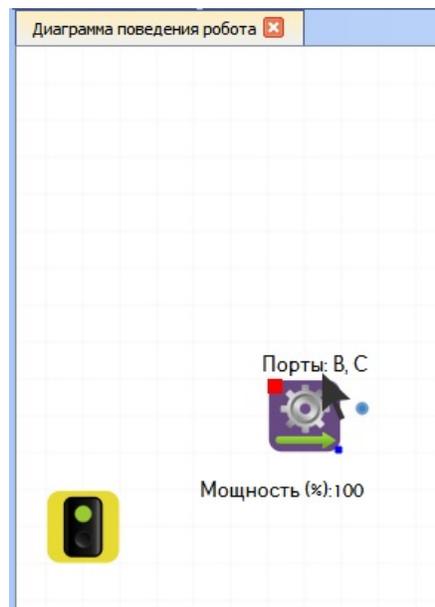


рис. 6.

На рис.5-6 элемент палитры переносится на сцену.

## Виртуальный курсор

Основным инструментом пользователя в среде графического программирования является компьютерная мышь, поэтому необходима реализация виртуального курсора, который будет эмулировать работу компьютерной мыши. В связи с этим был реализован виртуальный курсор, который представляет из себя статический элемент графического интерфейса.

Среди возможностей виртуального курсора:

- moveTo: виртуальный курсор передвигается к указанному элементу пользовательского интерфейса за указанное время;
- sceneMoveTo: аналог предыдущего пункта для сцены;
- (left/right)Button(Press/Release): виртуальный курсор посылает виртуальные события мыши указанному виджету;
- scene(left/right)Button(Press/Release): аналог предыдущего пункта для графической сцены;
- dragPaletteElement: виртуальный курсор берет элемент палитры по его идентификатору и создает его в нужном месте на сцене. Результатом возвращает Id элемента на сцене. Ввиду того, что графический вид палитры не показывал все элементы одновременно, было принято решение на первое время ввести пробную эмуляцию переноса блока, которая понятна для пользователя, но более проста в исполнении;
- drawLink: виртуальный курсор рисует соединительную линию между двумя элементами жестом мыши, реализацией которых занимался автор статьи [6], ориентируясь по Id элементов на сцене.

## Информационные сообщения и указательные стрелки

Важным фактором обучения является не только показ возможностей программной среды, но и объяснение происходящих действий. Для этого были введены информационные сообщения. Для более удобного восприятия они реализованы как всплывающие подсказки. Подсказка задается сообщением и продолжительностью жизни.

Также были реализованы стрелки-указатели, задаваемые углом поворота и целевым элементом пользовательского интерфейса. Их наличие позволяет помогать пользователю ориентироваться в графическом интерфейсе среды.

## Фасад к пользовательскому интерфейсу

Так как в процессе разработки QReal:Robots графический интерфейс претерпевает некоторые изменения, то ориентирование функций ScriptAPI (скриптовый интерфейс приложения к графическому интерфейсу среды) по координатам неприемлемо. Следовательно, необходим класс, который будет выдавать нужные ссылки на элементы

пользовательского интерфейса по запросам, после чего мы можем взаимодействовать с ними с помощью виртуального курсора.

По этой причине были реализованы классы `guiFacade` и `robotsGuiFacade` (включен в плагин `Robots`), которые являются реализацией шаблона проектирования “фасад”. Они предоставляют информацию об элементах пользовательского интерфейса главного окна и окна плагина `Robots` соответственно.

Среди их возможностей:

- получить элемент пользовательского интерфейса по его типу и имени;
- получить сцену (с текущей вкладки, или сцену 2d модели плагина `Robots`);
- получить элемент палитры по его идентификатору.

# Заключение

## Результаты

На текущий момент в среду QReal:Robots был встроен интерпретатор скриптового языка QtScript. В сценариях доступны возможности доступа к пользовательскому интерфейсу среды QReal и плагина Robots, управление виртуальным курсором, а также отправка информационных сообщений и отрисовка указательных стрелок.

Все возможности реализуются через API к пользовательскому интерфейсу среды и ориентируются по его элементам.

Технология была апробирована созданием нескольких обучающих демонстраций, показывающих возможности полученной технологии, а также перспективность ее развития.

## Дальнейшее развитие

В дальнейшем планируется продолжить разработку данной технологии в следующих направлениях:

- Обеспечение функциональной поддержки всей функциональности среды в обучающих демонстрациях;
- Создание наиболее информативного набора демонстраций, который покрывает большую часть основных возможностей платформы;
- Генерация обучающих демонстраций в процессе взаимодействия пользователя со средой.

Ссылка на GitHub репозиторий:

<https://github.com/DmitriyChernov/qreal/tree/tutorial>

## Список литературы

[1] Design Patterns: Elements of Reusable Object-Oriented Software. – 1994 г.

[2] ECMAScript Programming Language

<http://www.ecmascript.org/>

[3] Qt - cross-platform application and UI framework for developers using C++ or QML, a CSS & JavaScript like language.

<http://qt-project.org/>

[4] Warcraft 3 official page

<http://us.blizzard.com/en-us/games/war3/>

[5] Распознавание нарисованных диаграмм в проекте QReal, курсовая работа, СПбГУ, математико-механический факультет, кафедра системного программирования. – 2012 г.

[http://se.math.spbu.ru/SE/diploma/2012/s/Osechkina\\_diploma.pdf](http://se.math.spbu.ru/SE/diploma/2012/s/Osechkina_diploma.pdf)

[6] Стив Круг «Не заставляйте меня думать!». – 2002 г.