

32

2014

U

Q

U

BLENDER: материалы

KRITA 2.8

Первые впечатления

PhotoFiltre 7

От ASCII до GPGPU

История компьютерной графики

Культовые игры:

QUAKE

TellTale Games

Сказки для взрослых

+ многое
другое!

Независимый электронно-познавательный журнал.
Издается с 2008 г. Доступен по CC-BY-NC-SA





FPS

№32

FPS - бесплатный, свободно распространяемый электронный журнал, посвященный различным видам цифрового творчества. Тематика FPS охватывает разработку игр и игровых движков, обработку изображений и звука, уроки по трехмерному моделированию и ретуши фотографий, а также свежие новости из мира цифровых технологий.

FPS - это журнал для программистов, художников, моделлеров, линуксоидов, энтузиастов движения СПО, хакеров и просто творческих людей, ищущих и умеющих находить в этом мире качественную «пищу для ума».

Мы ратуем за свободу слова и свободный обмен информацией - журнал распространяется на условиях лицензии Creative Commons (CC-BY-NC-SA). Мы рады любому сотрудничеству, приветствуем любые новые идеи и приглашаем в наш авторский коллектив всех желающих!

Журнал издается в Казани с января 2008 г. и в данный момент выходит раз в два-три месяца.

© 2008-2014 Редакция журнала «FPS». Некоторые права защищены. Все названия и логотипы являются интеллектуальной собственностью их законных владельцев и не используются в качестве рекламы продуктов или услуг. Редакция не несет ответственности за достоверность информации в материалах издания и надежность всех упоминаемых URL-адресов. Мнение редакции может не совпадать с мнением авторов. Материалы издания распространяются по лицензии Creative Commons Attribution Noncommercial Share Alike (CC-BY-NC-SA), если явно не указаны иные условия.

Главный редактор: **Тимур Гафаров**
Дизайн и верстка: **Наталья Чумакова**
Обложка: **Тимур Гафаров**
Корректор: **Наталья Чумакова**

По вопросам сотрудничества обращайтесь по адресу:
gecko0307@gmail.com

Читайте в номере:

- **Blender**
 - Новости
 - Материалы в Blender. Часть 1
 - Обзор дополнений. Выпуск 11
- **2D-графика**
 - GIMP: новости
 - Krita. Первые впечатления
 - PhotoFiltre 7. Обзор новшеств
- **Кодинг**
 - Язык D: новости
 - Стадии программирования
 - C++14
- **История компьютерной графики**
 - От ASCII до GPGPU
- **Linux-гейминг**
 - Игровые новости из мира Linux
- **Культовые игры**
 - Quake
- **TellTale Games**
 - Сказки для взрослых
- **RetroGame**
 - Коллекция олдскульного геймера
- **Linux**
 - Полезные команды
- **#GamerGate**
 - Истинное лицо информационной войны



Blender

Новости

Разработчики Blender выпустили новую версию пакета – 2.72. Этот релиз обеспечивает поддержку рендеринга объемов и подповерхностного рассеивания в Cycles на GPU, также в Cycles был интегрирован NPR-рендер Freestyle (правда, без поддержки предпросмотра). Интерфейс обогатился новым типом контекстного меню – радиальным, элементы которого располагаются по кругу.

Кроме того, в **Blender 2.72** появился узел постобработки для рендеринга «световых лучей» – это эффект, при котором лучи света становятся видимыми, проходя сквозь не полностью прозрачную воздушную среду (например, облако пыли). Этот феномен может быть воссоздан «честно» – при помощи объемного рендеринга, однако это не самый быстрый способ, поэтому было разработано специальное средство для отрисовки световых лучей в двумерном пространстве, на этапе постобработки.



Blender 2.72

Release Candidate



Открылся новый полезный ресурс от отечественных авторов – онлайн-библиотека материалов **Blendermada**. Главная цель проекта – создать коллекцию качественных материалов для Blender, которые можно было бы загружать и использовать в один клик. Для этого был создан специальный аддон для Blender 2.70 и выше, позволяющий соединиться с сервером Blendermada и импортировать в проект материалы. В данный момент сервис работает в режиме альфа и работает только с Cycles, но в будущем не исключена поддержка других рендер-движков.

Авторы принимают заявки на включение сторонних материалов в свою библиотеку – все они распространяются по лицензии CC-0 (эквивалент Public Domain). Клиент и сервер Blendermada являются свободным ПО и доступны на GitHub.



Evermotion, разработчики рендер-движка NOX, перевели свой продукт в OpenSource! NOX – это физически корректный движок рендеринга, полностью интегрированный с Blender и 3ds Max (поддерживается также Cinema 4D). Код представлен под лицензией Apache и свободен для коммерческого использования и модификации. Основные возможности NOX включают поддержку честного DoF и его имитации, подповерхностного рассеивания, инстанцирования, постобработки и многое другое.

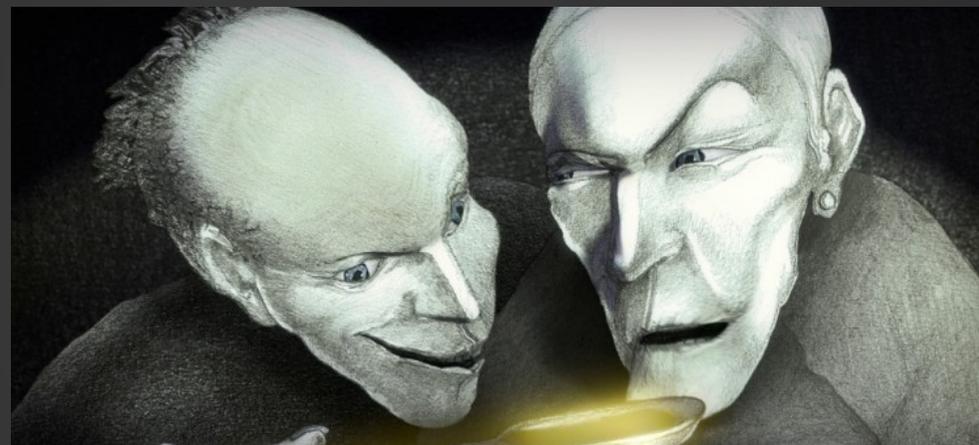
Подробности – на официальном сайте: <http://www.evermotion.org/nor>



Арно Кутюрье, автор **Suicidator City Generator**, анонсировал выход в течение ближайших месяцев новой версии своего знаменитого генератора городов. В Suicidator 0.7 дебютирует поддержка Cycles (генерация материалов Cycles), при этом можно будет свободно переключаться между Cycles и V-Ray. В будущем не исключена поддержка LuxRender. Проект получит новое название.



Напомним, текущая версия SCG – 0.5.7. Бесплатная версия имеет ограничения на площадь генерируемых городов, на размер текстур и детализацию домов (однако разрешается использование аддона в любых целях, в том числе коммерческих). Полная версия стоит €14.95, причем единожды купив лицензию, вы получаете бесплатные обновления в будущем. Один евро от каждого заказа идет в помощь фонду Blender Foundation.



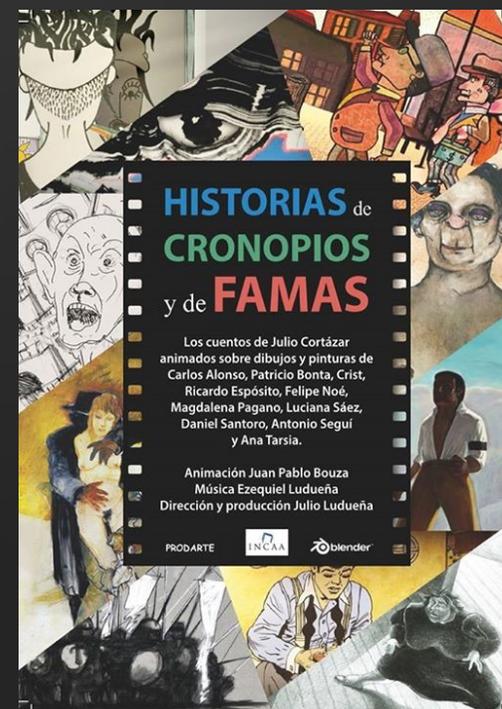
Одновременно с этим, состоялось обновление **Blend4Web** – платформы для создания браузерных трехмерных приложений. Пакет тесно интегрирован с Blender, воспроизведение контента осуществляется средствами WebGL, Web Audio и других нативных браузерных технологий.

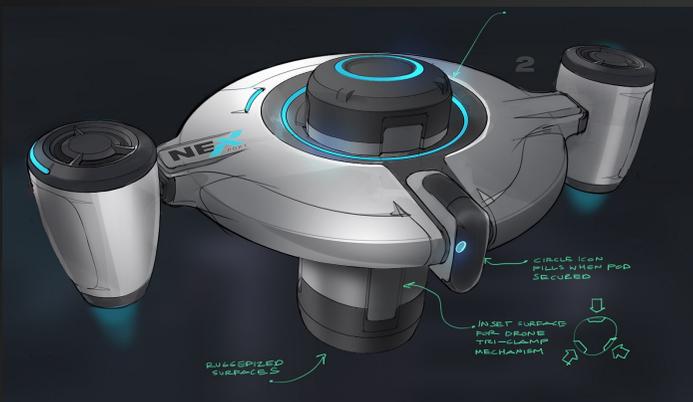
В новой версии Blend4Web 14.09 дебютировал новый редактор для визуального программирования логики приложений – NLA Script, основанный на встроенном в Blender редакторе нелинейной анимации (NLA). Также было существенно улучшено качество отрисовки теней, добавлена возможность смешивания различных анимаций на одном объекте, расширена функциональность редактора узлов.

Интересно, что разработчики Blend4Web начали принимать участие в улучшении самого Blender – так, ими был предложен патч, исправляющий некорректное поведение бликов на прозрачных участках поверхности.

Недавно прошла премьера полнометражного 90-минутного фильма «Historias de Cronopios y de Famas», созданного аргентинской студией с использованием Blender, GIMP, Ardour и других свободных программ. Картина отличается необычной сюрреалистической графикой в духе творчества Луиса Фелипе Ноз, Антонио Сеги и Карлоса Алонсо, а сюжет основан на одноименном романе Хулио Кортасара – «Жизнь хронопов и фамов».

Трейлер можно посмотреть здесь: <https://www.youtube.com/watch?v=8qF9yZ01fjY>





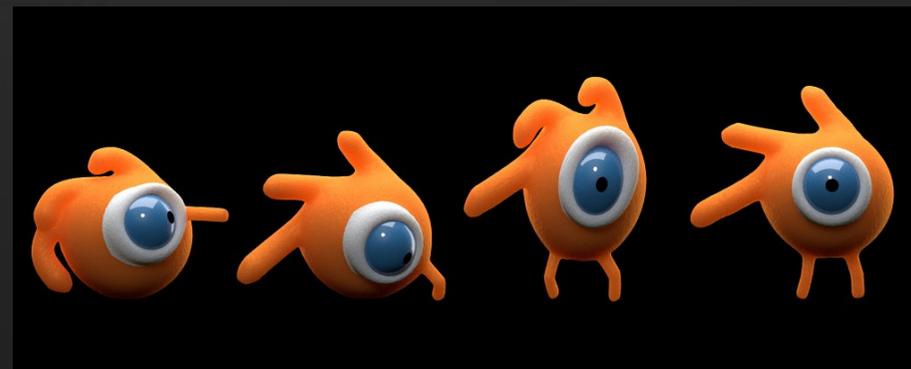
Еще один интересный проект – новая короткометражка от двух корифеев блендеровского сообщества: Колина Леви, режиссера «Синтел», и Эндрю Прайса, создателя портала BlenderGuru. В планах – снять научно-фантастический фильм в сеттинге футуристического Лос-Анджелеса о мире, где всю работу за людей выполняют дроны. Планируется задействовать реальные съемки и множество спецэффектов, которые будут создаваться, естественно, в Blender.

Сейчас авторы ищут добровольцев для команды – талантливых художников и аниматоров. Сотрудничать можно на удаленной основе – всем желающим дается уникальная возможность поработать в крупном проекте, бок о бок с профессионалами киноиндустрии!

Отправить свою заявку на участие можно по [следующей ссылке](#).



Авторы небезызвестного проекта «Агата» – казанская анимационная студия **Propellers** – планирует выпустить собственный русскоязычный видеокурс по Blender – «LevelUp». Основной задачей была поставлена помощь начинающим пользователям освоить Blender на уровне, достаточном для создания серьезных проектов. Тематика курса будет охватывать лицезвую анимацию, систему частиц, симуляцию жидкости, динамику волос, шейдеры на основе узлов, секреты работы с Cycles, симуляцию дыма и огня, композитинг, работу с Free-style и многие другие продвинутые возможности пакета. Видеоматериалы курса будут публиковаться на YouTube по мере готовности.





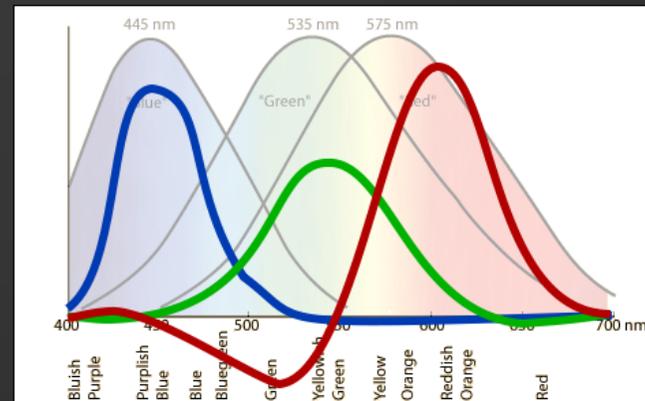
Материалы в Blender

Часть 1

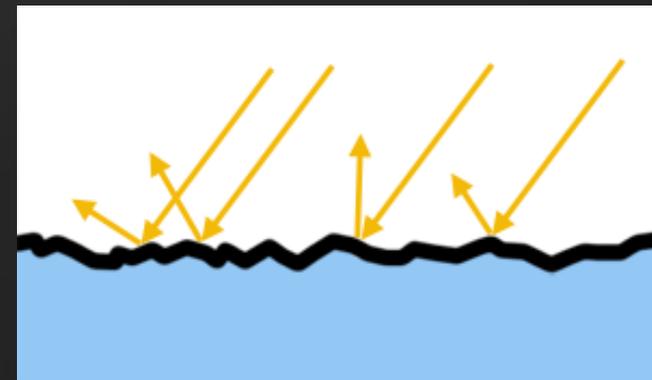
Материал - это набор параметров, определяющих характер поверхности объекта (а в некоторых случаях - и его объема). Они включают цвет (если быть точным, несколько компонентов цвета для разных составляющих освещенности), текстуру, параметры прозрачности, отражения и преломления и многие другие. В общем случае, эти параметры определяют закон, по которому свет должен отражаться от поверхности объекта.

Чтобы создавать на компьютере действительно качественные и реалистичные изображения, необходимо понять, как моделируется освещенность виртуальных объектов. В реальном мире свет состоит из мельчайших частиц, называемых фотонами. Фотон имеет свойства, присущие как волнам, так и элементарным частицам. Фотонов настолько много, что обычно можно пренебречь тем, что световое излучение состоит из отдельных частиц (это важно только в квантовой механике), и рассматривать его как непрерывный поток энергии. В этом случае к свету можно применить статистические законы и смоделировать его на компьютере.

Поток энергии отрывается от источника света и распространяется в пространстве, пока не столкнется с каким-либо объектом. При этом одна часть энергии поглощается веществом объекта, а другая - отражается (поэтому мы видим объекты как темные или светлые). Отраженный поток фотонов меняет свою длину волны в зависимости от свойств вещества, в результате чего мы воспринимаем у разных объектов различные цвета. Также некоторая часть фотонов проходит сквозь материал, и объект выглядит прозрачным. Проходя через вещество, световой поток может преломляться (менять направление) и рассеиваться.

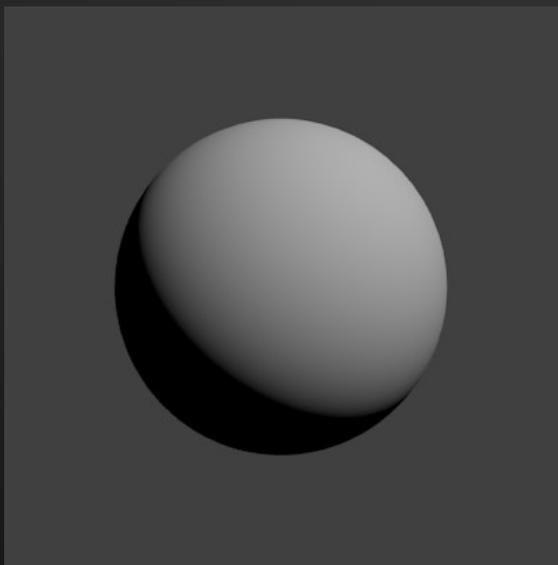


Но если объекты отражают свет, почему же далеко не любая поверхность может служить зеркалом? Все дело в том, что идеальным зеркалом является только идеально гладкая поверхность, в то время как обычные поверхности в той или иной степени шероховаты - то есть, состоят из множества микроскопических граней-отражателей. Свет, попадая на поверхность объекта, многократно отражается от этих микрограней и рассеивается в пространстве, в результате чего мы не видим точных отражений, а только сплошной цвет.

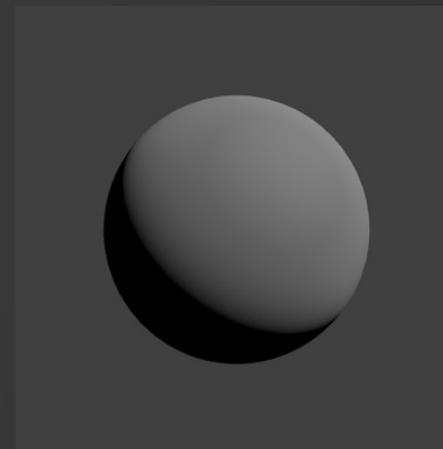


В компьютерных моделях, описывающих материал объекта, этот феномен сведен к простым математическим формулам, по которым можно вычислить степень рассеянной (или, как обычно говорят, диффузной) освещенности в любой точке заданной поверхности. Самая простая такая формула - закон Ламберта (Lambert), который определяет интенсивность диффузной освещенности в точке как косинус угла между направлением света и нормалью к поверхности в этой точке.

Модель Ламберта хорошо подходит только для сравнительно гладких поверхностей. Для моделирования шероховатой, бархатистой или запыленной поверхности часто используют диффузную модель Орена-Найара (Oren-Nayar), которая основана на предположении, что поверхность состоит из множества бесконечно малых микрограней, освещение каждой из которых описывается моделью Ламберта. Модель Орена-Найара имеет параметр для контроля шероховатости поверхности (Roughness). Этот параметр определяет, сколько света отразится назад в направлении источника света.



Lambert



Oren-Nayar

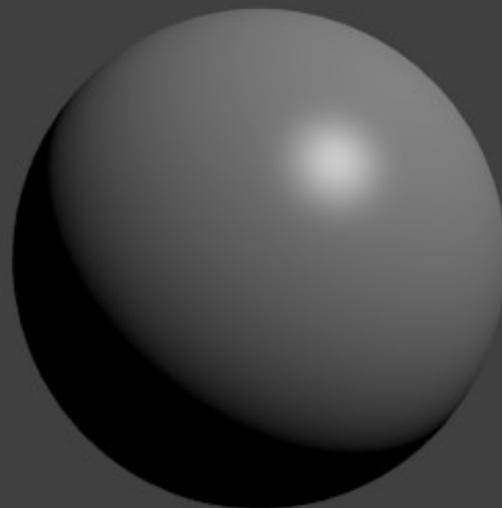
В Blender чаще всего используются именно эти две диффузные модели. Помимо диффузной, используется также бликовая составляющая освещенности. Бликовая составляющая (specular term) - это количество света, зеркально отраженного поверхностью. Блик - это прямое отражение источника света на поверхности объекта. Если учесть, что в компьютерной графике используются идеализированные объекты, возникает закономерный вопрос: почему точечный источник света, не имеющий объема и невидимый сам по себе, отражается как относительно крупный размытый световой блик? Этот феномен также объясняется наличием микрограней: они имеют собственные вектора нормалей, отклонение которых от основной нормали поверхности меняют интенсивность зеркально отраженного света.

Цвет блика может не совпадать с цветом материала. Это справедливо для некоторых многослойных материалов - например, пластик представляет собой «слоеный пирог» из пигмента и прозрачного полимера: блик дают прозрачные слои, а диффузное рассеивание - цветные. Однородные материалы такого эффекта не производят, и блики на них имеют тот же цвет, что и сами материалы. Яркий пример - металл.

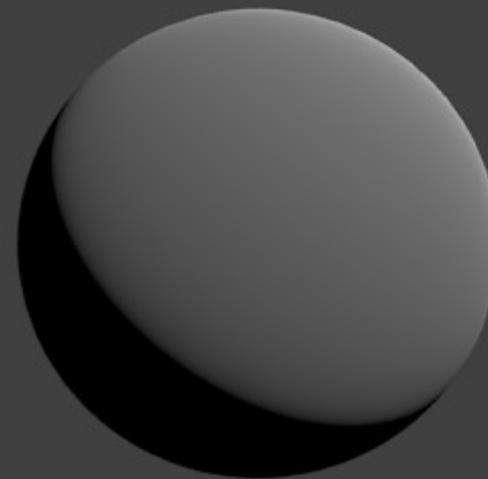
Для сравнительно гладких материалов (таких, как пластик или металл) обычно пользуются эмпирической моделью Фонга (Phong). Она не соответствует точному физическому описанию отражения света, но в большинстве случаев позволяет достичь приемлемых реалистичных результатов. Формула Фонга основана на простом наблюдении: блестящие поверхности дают маленькие и резкие блики, в то время как матовые - большие и размытые.

Более согласованная с физикой модель, которая поддерживается в Blender - модель Кука-Торренса (Cook-Torrance). Она основана на допущении, что поверхность состоит из микрограней, каждая из которых является идеальным зеркалом.

Зная, где и как правильно применять эти модели, можно моделировать объекты, по внешнему виду максимально приближенные к реальным. Так, для матовых поверхностей вроде камня, бетона или бумаги лучше всего подходит модель Ламберта. Блики на пластике, фарфоре, металле, матовом стекле имитируются моделью Фонга. Бархат, вельвет, ковры и некоторые другие виды тканей лучше всего воссоздаются моделями Орена-Найара и Кука-Торренса.



Phong



Cook-Torrance

Продолжение следует...



Обзор дополнений Blender

Выпуск 11

Благодаря удобному и мощному API для языка Python, Blender поддается практически неограниченному расширению. Наш журнал отслеживает выход новых полезных дополнений для Blender, которые могут заинтересовать пользователей, использующих программу в качестве инструмента для разработки игр или создания игрового контента.

Если вы разрабатываете собственное дополнение или просто нашли в Интернете чей-то интересный проект, будем очень рады, если вы напишете нам об этом и поделитесь ссылкой. Пишите на gecko0307@gmail.com, либо в наше сообщество: http://gplus.to/fps_community

Zero Brush

Автор: Blender Sensei

Дополнение, облегчающее рисование по текстурам. Zero Brush берет на себя всю рутинную работу по подготовке UV-развертки, настройке материалов и текстур – вам остается только начать творить!

Ссылка:

<http://blendersensei.com/zero-brush/>

BakeTool

Автор: Cogumelo Softworks

BakeTool – это набор инструментов, облегчающих работу с технологией запекания (baking), не так давно дебютировавшей в Cycles. Они включают многоканальное запекание и быстрый доступ к некоторым настройкам.

[Ссылка.](#)



BAKE TOOL

Cogumelo
SOFTWARE

- Cycles and Blender Internal Render Support
- Multi Pass With Configs Per Pass
- Easy Bake for Single Objects or Full Scenes
- Individual and Atlas Mode
- Bake to Target
- Original Pack UV Space Atlas Function!
- Auto Save Externally and Pack in .Blend
- Auto Config Image Format for Best Quality/ File Size
- More Coming!

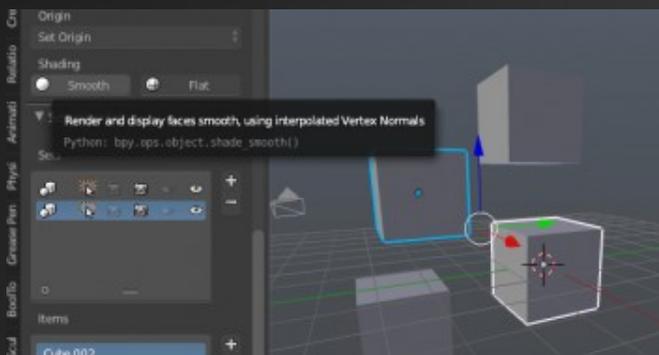


Selection Sets

Автор: Джон Мэттьюз

Менеджер выделений для Blender, позволяющий сохранять списки выделенных объектов и восстанавливать их в любое время. По сути, альтернатива группам объектов (Groups).

[Ссылка.](#)



Render+

Автор: Januz

Коммерческое дополнение, дополняющее систему рендеринга Blender новыми функциями. Так, с ним вы можете запускать рендеринг в отдельном фоновом процессе – при закрытии окна Blender рендеринг продолжится. Кроме того, под Linux и Mac OS X можно включить системные уведомления, оповещающие о завершении рендеринга. Из других полезных возможностей: автовыключение компьютера, выполнение кода на Python по завершении задачи, автосохранение отрендеренного изображения и т.д.
Цена: \$12

[Ссылка.](#)



Muscle System

Автор: Coloremblem

Система мускульной анимации, наподобие встроенной в Maya, 3ds Max или Cinema 4D.

[Ссылка.](#)



Create Tab Hider

Автор: Leon

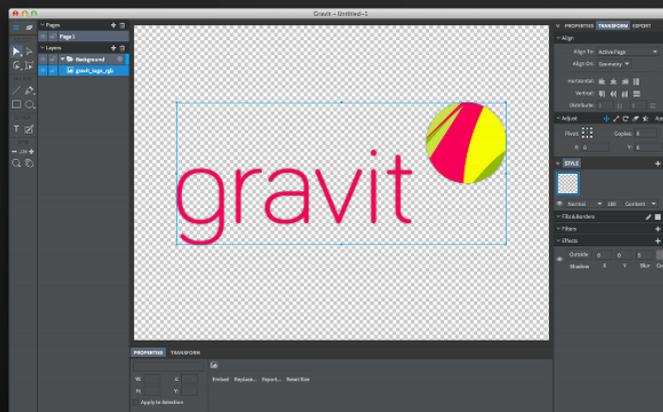
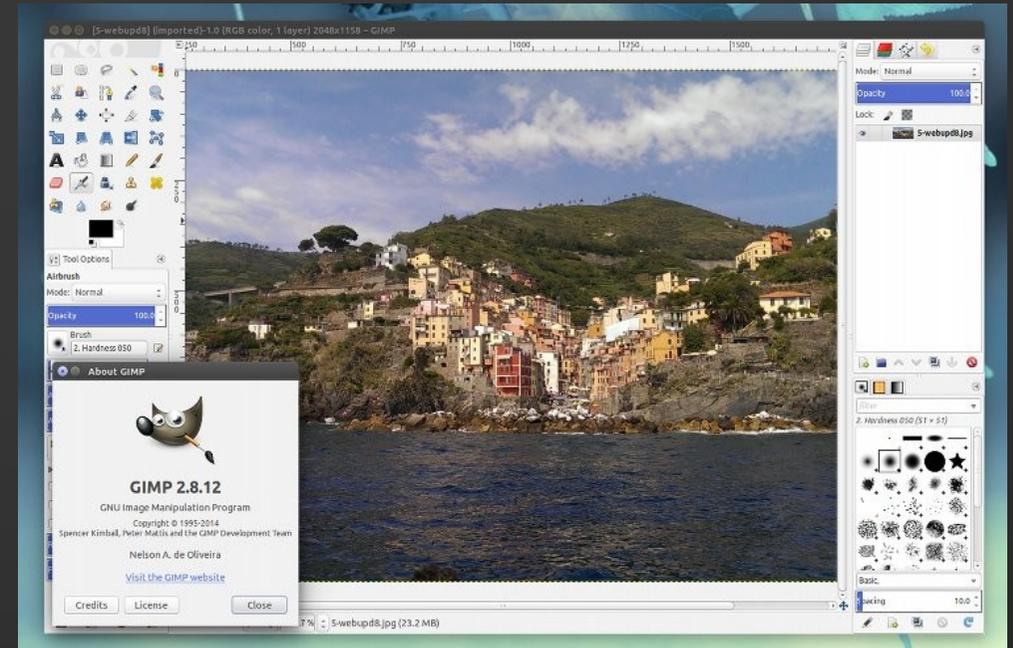
Не всем нравятся вкладки на панели инструментов, появившиеся в Blender 2.70. Это небольшое дополнение поможет скрыть вкладку Create, если вы ей не пользуетесь – большинство опытных пользователей для создания новых объектов просто нажимают Shift+A.

[Ссылка.](#)

GIMP: Новости

Недавно состоялся корректирующий релиз **GIMP 2.8.12**. В новой версии внесено 16 изменений, связанных с устранением ошибок. Из исправлений можно отметить повышение надежности загрузки поврежденных файлов XCF, удаление опции для отмены вывода уведомления о закрытии измененного изображения, обеспечение drag & drop между изображениями в однооконном режиме, запуск сервера скриптов по умолчанию только на 127.0.0.1.

Компания Quasado анонсировала скорое открытие кода проекта **Gravit**, в рамках которого развивается графический редактор для веб-дизайнеров. Gravit позиционируется как продолжение идей проприетарных продуктов Fireworks и Freehand, разработка которых была свернута компанией Adobe. Редактор написан целиком с использованием веб-технологий (HTML5, CSS3 и JavaScript) и поставляется в форме браузерного приложения. Начиная с первого стабильного выпуска, проект планируется развивать в качестве открытого продукта.



Состоялся выход новой ветки графического редактора **Krita 2.8**. Начиная с версии 2.8.0, Krita теперь доступна и пользователям Windows. Примечательно, что к портированию были причастны в том числе сотрудники Intel. Кроме того, были проведены многочисленные улучшения пользовательского интерфейса с целью оптимизации производительности пакета. В состав программы вошел популярный плагин G'MIC, также реализована функциональность псевдобезразмерного холста. Подробный обзор windows-порта Krita читайте ниже.

После успешного релиза Krita 2.8, команда разработчиков приступила к работе над следующим релизом – 2.9, средства для которого планируется собрать через Kickstarter. В частности, планируется обеспечить поддержку оплачиваемой работы в режиме полного рабочего дня сразу двух разработчиков - это первая цель кампании по сбору средств. Если начальная сумма будет значительно превышена, то разработчики наметили дополнительную цель – портирование Krita на Mac OS X.

Совместно с сообществом художников составлен список функций, которые пользователи хотят увидеть в версии 2.9 – он включает в себя такие интересные задачи, как улучшенную совместимость с Photoshop и дальнейшее улучшение инструмента трансформации и деформации изображения.

Напомним, Krita – это свободный растровый графический редактор, развиваемый в рамках проекта KDE и входящий в состав офисного пакета Calligra. Проект курируют фонд Krita Foundation и организация KO GmbH (Krita Studio).



Уважаемые читатели!

Наш журнал регулярно выходит на протяжении почти 6 лет – с января 2008 года. Все эти годы он оставался бесплатным изданием, предлагая публике эксклюзивный контент с минимумом рекламы. Мы всегда работали на совесть – не ради денег, а на благо наших читателей. «FPS» был и остается проектом энтузиастов и полностью независимым изданием – мы не защищаем интересы корпораций или политиков, мы пишем о том, что считаем нужным и важным. Мы стоим за свободу слова и творчества, за обмен информацией и знаниями: все материалы журнала можно беспрепятственно копировать, распространять и использовать в любых производных работах.

И мы надеемся, что так будет продолжаться и дальше. Но на создание новых номеров у авторов уходит достаточно много сил и времени, которые никак материально не компенсируются. Поэтому, если вам нравится журнал, и вы хотели бы, чтобы он жил, развивался, становился больше и качественнее, просим **поддержать его электронной валютой** – при помощи [Bitcoin](#), [WebMoney](#), [PayPal](#) или [Яндекс.Деньг](#), любой суммой на ваше усмотрение. Для нас важен любой, даже маленький вклад!

Наш WMR-кошелек: **R120156543694**

Адреса Bitcoin:

16PSGbj5foeqMN8isdoyiKvWYGM9V5idFk
16XaSt1U5eXWG7EAKuEMpFE6M6fPia5o4F
1PdNHTL5nJsZJGXyNj4c5xPW3eApEoB9pQ
1MFauzBqewUN8MWPpy7DqGeLiJayxCqGBCL
1MSPN7TXuTPbuGjT1AQxex3ECjRgPhpFEA
1NB1xXoyJ71beEPpmmKioWMaon2uCigCeQ

Адрес PayPal:

gecko0307@gmail.com

Яндекс.Деньги:

410012052560079

*Заранее благодарны!
Редакция*



Мобильный FPS



Теперь любимый журнал всегда с вами!

Читайте FPS на мобильных устройствах:
скачайте приложение для Android или iOS!

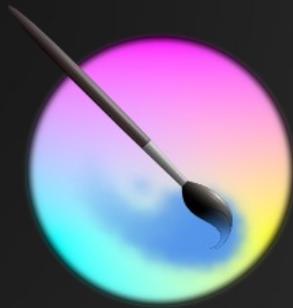


Available on the
App Store



ANDROID APP ON
Google play

Разработчик приложения: цифровое издательство St.Appler <http://www.stappler.org/>



KRITA

Первые впечатления

Начало 2014 года ознаменовалось важным событием: стабилизацией графического редактора Krita на ОС Windows. До сих пор отсутствие полноценной поддержки Windows мешало этому свободному редактору, уже довольно известному в рядах линуксоидов-KDE'шников, обзавестись армией пользователей, незнакомых с Linux, как это уже стало с GIMP. Теперь настало время исправить ситуацию...

Хоть я и являюсь идейным пользователем Linux, я решил начать знакомство с Krita именно с windows-порта. Установка прошла без нареканий, хотя немного смутил вес дистрибутива Krita 2.8.3 – более 100 Мб.

Учитывая, что GIMP со включенным в пакет рантаймом GTK и множеством плагинов редко занимает больше 80 Мб, возникает закономерный вопрос – откуда же такой объем? Впрочем, вопросы о размерах приложений в наше время звучат больше как риторические – назад, к килобайтам, уже дороги нет...

Интерфейс порадовал компактностью и отсутствием лишнего. Классическое расположение панелей в духе Photoshop порадует опытного дизайнера, так и не привыкшего к многооконному GIMP (хотя в последних версиях там есть точно такой же однооконный режим – тут у Krita преимуществ мало). По умолчанию стоит модная темно-серая тема оформления.

Почти сразу оценил приятную мелочь: во время рисования нажатие Shift позволяет на лету изменить размер кисти – очень удобно, не нужно залезать в настройки инструмента. Также порадовало то, что размер кисти – не экранный, а физический: то есть, при уменьшении масштаба просмотра, кисть также пропорционально уменьшается.

В программе очень много готовых кистей, имитирующих реальные художественные инструменты: разные виды графических материалов, красок, распылителей и т.д. Можно устанавливать новые кисти. Во время рисования кисти могут, в зависимости от их сложности, иногда слегка подтормаживать, но это не очень критично.



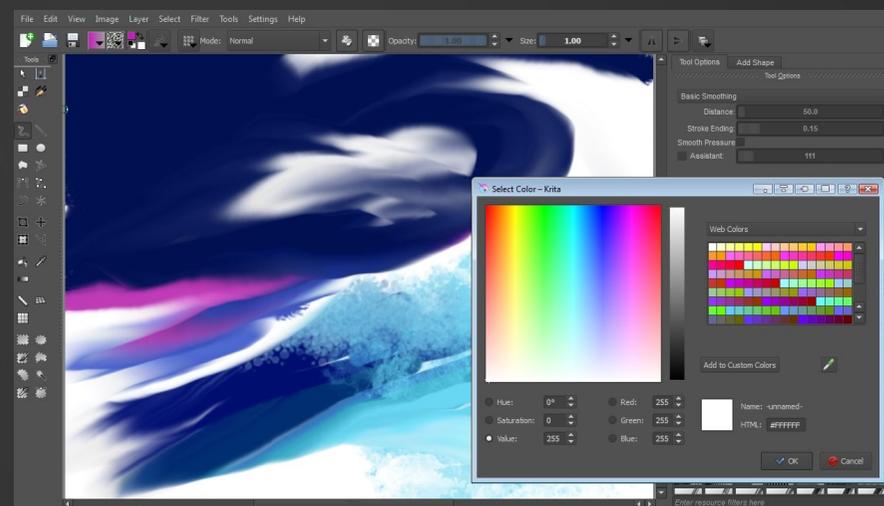
Version: 2.8.3 (git c49997b) (x86)

Кроме обычной кисти, есть интересный инструмент «мультикисть». Он позволяет рисовать несколько штрихов за раз с использованием различных настраиваемых алгоритмов – так, порадовала поддержка режима зеркального рисования. Работает этот инструмент слегка тормознuto, но терпимо.



Другие инструменты тоже выполнены качественно. Например, мне очень понравилась реализация выделений: есть режим выделения кистью с поддержкой логических операций – каждый штрих может добавлять в выделение новую область.

Также есть быстрое выделение при помощи кривых – а в GIMP необходимо сначала создать кривую, затем конвертировать ее в выделение, что требует нескольких дополнительных действий. Кроме того, есть удобный инструмент быстрой обрезки холста – в GIMP есть только обрезка по контуру выделения.



Рисование геометрических фигур на порядок более простое и интуитивное, нежели в GIMP, где для этого используется отдельный плагин со своим интерфейсом. Здесь фигуры рисуются напрямую на холсте – есть прямоугольник, эллипс, полигон. Контур фигуры определяется текущей кистью. Впрочем, в GIMP есть возможность рисовать любые фигуры путем оконтуривания кривой, и технически тут у Krita особых преимуществ нет – разве что только в плане эргономики.

Очень порадовал инструмент трансформации слоев и выделений – он работает здесь гораздо быстрее, чем в GIMP. Есть различные способы трансформации, включая 3D-поворот по всем трем осям и деформацию по сетке. Большой плюс заключается в том, что все виды трансформации объединены в один неразрушающий контейнер: пока вы не смените инструмент, любые операции могут быть отменены без потери качества изображения. Но есть и определенное неудобство: ручной поворот (с зажатым Shift) работает по умолчанию с осью X, а не Z. Как это изменить, я так не нашел.

В GIMP меня всегда раздражала необходимость перезапуска программы для изменения размеров сетки – у Krita такого недостатка нет. Есть несколько готовых пресетов размеров сетки, но, к сожалению, нельзя добавлять свои.



Подводя итог: по инструментам Krita однозначно получает твердую пятерку. От GIMP ее в этом отношении заметно отличает только отсутствие инструментов рисования текста, а также «умных ножниц», которые, впрочем, я использую довольно редко.

А что касается текста, то он и в GIMP реализован не настолько хорошо, чтобы представлять собой серьезное преимущество.

Перейдя к фильтрам, я сразу убедился, что Krita больше ориентирована на рисование кистями и мало подходит для ретуши фотографий. Фильтры очень медленные, скорость предпросмотра по сравнению с GEGL оставляет желать лучшего.

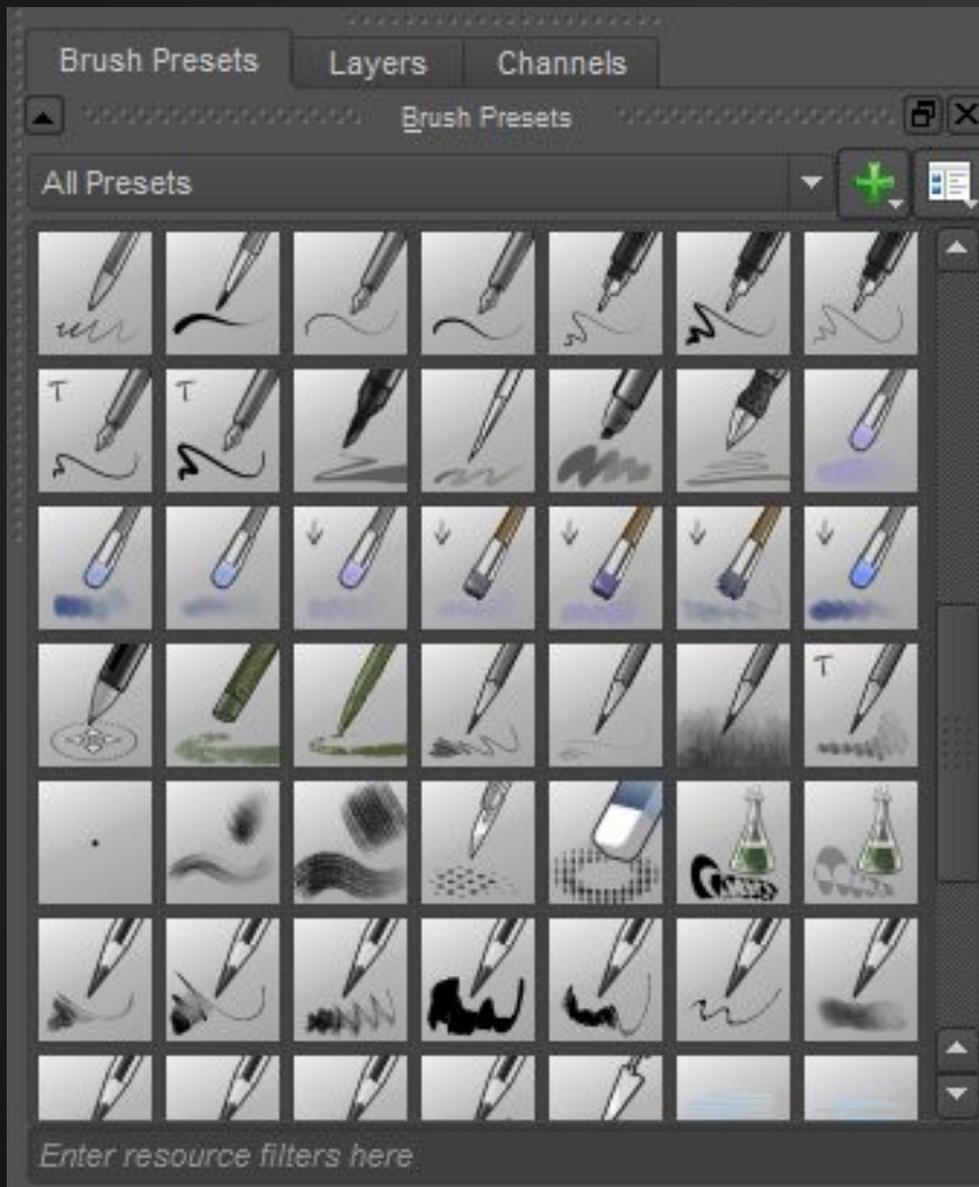
Нет шкалы прогресса фильтрации, равно как и возможности ее отмены. Как только вы начинаете активно пользоваться фильтрами, редактор начинает бесконтрольно потреблять память, и, в результате, часто падает или виснет. И это на системе с гигабайтом памяти – в то время как GIMP работает вполне сносно даже при 512 мегабайтах!

Самых фильтров не так много, но среди них есть достаточно полезные – например, Lens Blur. Я часто использую фильтр корректировки цветных кривых, но он тут не самый удобный: нет кнопки сброса, и вообще цветные кривые и кривая яркости почему-то вынесены в два отдельных модуля, хотя им логичнее быть в одном.

Единственное, что по-настоящему порадовало: в комплекте с программой идет незабвенный плагин G'MIC, который давно полюбился пользователям GIMP. Он представляет собой огромную коллекцию качественно реализованных фильтров и эффектов, а также фреймворк для создания своих собственных.

Серьезный минус Krita – нет экранной палитры с часто используемыми цветами, как в GIMP. Ее частично заменяет постоянно присутствующий на экране цветовой круг, но его трудно назвать замечательной палитрой.

Диалог с палитрой вызывается по нажатию кнопки, но, к сожалению, пока он активен, рисовать нельзя.



Но минусы на этом, пожалуй, заканчиваются и начинаются серьезные преимущества. Так, Krita «из коробки» поддерживает формат OpenRaster (ORA), что делает ее совместимой с MyPaint. Само собой, поддерживается полный набор основных графических форматов (JPG, PNG, TIFF и т.д.), включая PSD, а также экспорт в PDF. В программе много различных цветовых режимов: RGB, XYZ, CMYK, LAB, YCbCr. Есть возможность выбора глубины цвета: 8, 16 и 32 бит на канал (32 бита – с плавающей запятой). Также есть поддержка цветовых профилей, что просто незаменимо в полиграфии.

Итог. По общим впечатлениям, программа напоминает некий гибрид GIMP и MyPaint: есть мощный кистевой движок, но не забыты и стандартные функции графического редактора общего назначения. Конечно, в плане удобства рисования кистями до MyPaint ей далеко, а в плане обработки фотографий она значительно уступает GIMP, но как «среднее арифметическое» между этими двумя комбинациями – например, для создания коллажей и рисования поверх фотографий – вполне может пригодиться. Windows-порт юзабелен, но стабильность еще далека от совершенства, поэтому как основной инструмент художника под Windows я бы его пока не рекомендовал.

Тимур Гафаров

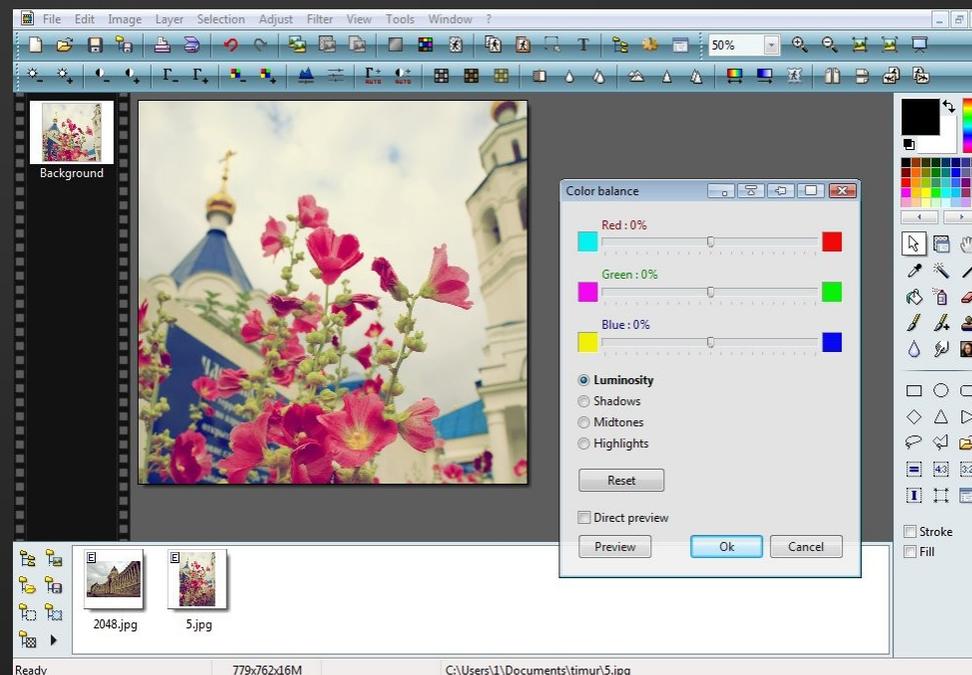
PhotoFiltre 7: Обзор новшеств

PhotoFiltre – это минималистичный бесплатный графический редактор, который в свое время был чуть ли не панацеей для тех, кто не мог позволить себе мощные коммерческие решения.

В те времена, когда Windows-версия GIMP была далека от стабильности, а возможности бюджетных персональных компьютеров практически не позволяли комфортно работать с Photoshop, легковесный и быстрый PhotoFiltre спасал положение – его можно было использовать практически для любых нужд, от обработки фотографий высокого разрешения до пиксель-арта, художественного текста и веб-графики. Так, вся графика в ранних номерах нашего журнала – обложки, растровые заголовки, обработка иллюстраций – делалась полностью в PhotoFiltre.

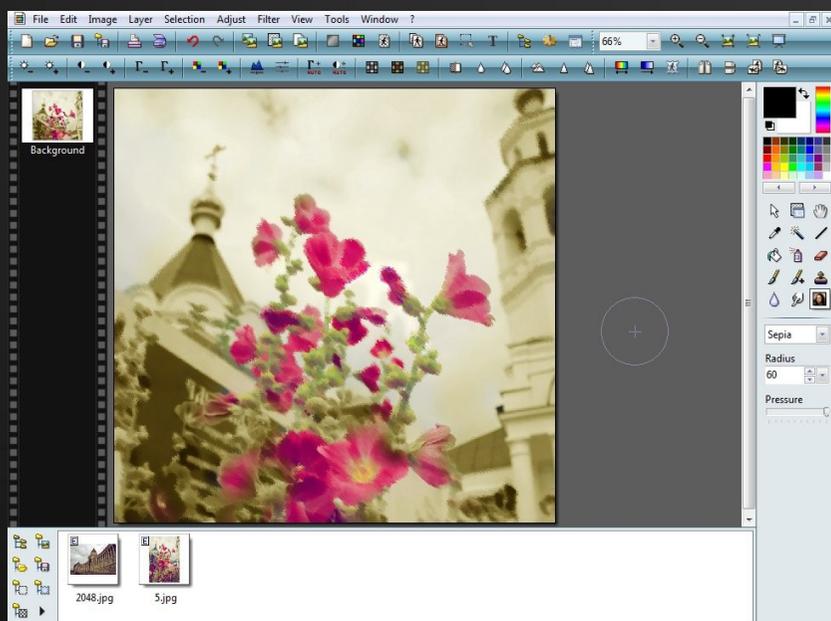
Кроме рисования при помощи базовых инструментов и кистей, редактор позволяет регулировать яркость, контрастность, насыщенность, исправлять гамму, накладывать всевозможные фильтры (акварель, пастель, чернила и т.д.). Кроме того, можно работать с декоративным текстом, оптимизировать графику, создавать поздравительные конверты и открытки из готовых шаблонов. Есть возможность добавлять свои кисти, маски, текстуры для заливки, а также подключать дополнительные локализации и плагины. На официальном сайте имеется SDK для создания своих плагинов.

На протяжении целых восьми лет актуальной была шестая версия программы – минорные релизы, в основном, исправляли баги. Разработчик Антонио да Круз делал основной упор на коммерческой shareware-версии пакета – PhotoFiltre Studio, а бесплатный вариант, казалось, совсем забросил. Основными особенностями «Студии» были поддержка слоев и альфа-каналов, инструмента «художественная кисть», анимации при сохранении в формат GIF, множественного выделения, формата RAW и множество других более мелких улучшений. В бесплатной PhotoFiltre поддержки слоев не было, но даже в таком виде программа шла на ура.



Впрочем, когда пользователи начали массово переходить на GIMP и Paint.Net (я в это время и вовсе перешел на Linux), популярность PhotoFiltre стала понемногу падать. В этой связи совершенно неожиданным был выход в 2013 году новой, седьмой ветки программы!

PhotoFiltre 7 – это, по сути, слегка урезанная бесплатная версия PhotoFiltre Studio. Она полноценно поддерживает слои, в том числе прозрачные. Слои можно перемещать и масштабировать. Что самое интересное, есть зачаточная поддержка неразрушающих эффектов слоев а ля Photoshop: в качестве такового доступен эффект тени (Drop Shadow). Следует отметить, что в GIMP поддержка такого рода функциональности еще не реализована и ожидает полной интеграции GEGL.



Также в PhotoFiltre 7 слои безразмерные – то есть, можно рисовать за пределами слоя, и его размер будет адаптирован автоматически. В GIMP такого, опять-таки, нет – безразмерные слои я до сих пор видел только в MyPaint, но это редактор несколько иной направленности.

Еще одна новинка «родом» из PF Studio – «художественная кисть», прямого аналога которой также нет в GIMP. Она позволяет применять к изображению фильтр из набора «Artistic» частично – путем ручного рисования штрихов. Набор доступных фильтров включает «пуантилизм», «аэрографию», «растушевку», «масляную живопись», «мастихин», «акварель», «пастель», «стекло», «оттенок», «обесцвечивание», «сепию» и «монохром». Можно указать радиус и нажим закрашивания.

Конечно, в целях сохранения коммерческой целесообразности, разработчик оставил некоторые существенные различия между PhotoFiltre 7 и PhotoFiltre Studio. Так, за деньги вы покупаете поддержку RAW и анимированного GIF, фильтров от Photoshop (в формате 8BF), инструментов Nozzles, «обратный ластик» и «деформация», а также эффектов границ для слоев. Стоит все это 29 евро. С другой стороны, отсутствующая функциональность может быть восполнена плагинами – если, конечно, у вас есть время и возможность написать их.

В остальном же PF7 – отличное дополнение к GIMP и MyPaint. PhotoFiltre запускается практически со скоростью просмотрщика – он может пригодиться, например, когда нужно быстро отредактировать изображение, а тяжеловесный пакет запускать нет времени. Ну и не стоит забывать про уникальные новые функции, которые мы еще нескоро увидим в GIMP и его аналогах...

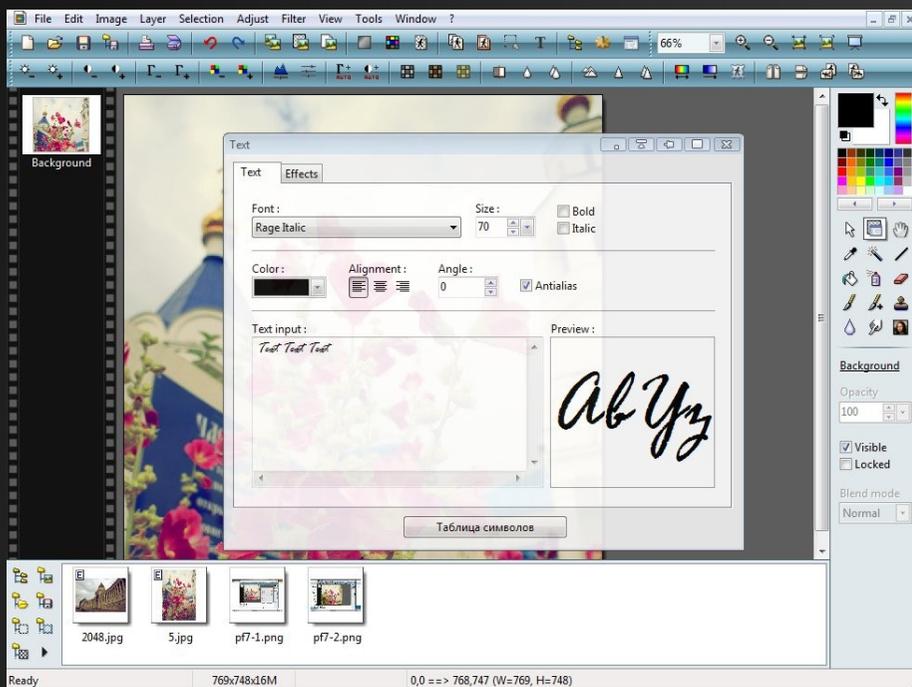
Blender

настольная книга

«Blender. Настольная книга» – это проект от журнала «FPS» по созданию полноценного русскоязычного электронного руководства по основам работы в Blender 2.6. Целевая аудитория – начинающие пользователи программы (как перешедшие со старых версий, так и начинающие знакомство с Blender «с нуля»). Книга будет представлять собой сборник статей, охватывающих различные аспекты использования Blender, скомпонованных по принципу «от простого к сложному».

Издание будет распространяться бесплатно, по лицензии Creative Commons BY SA. На данный момент активно ведется подготовка текста книги.

К работе над книгой приглашаются все желающие! На почтовый ящик редакции (gecko0307@gmail.com) принимаются статьи и уроки, а также общие советы и предложения. Кроме того, книге нужны графические материалы: авторские художественные работы, интересные скриншоты, демонстрационные рендеры, схемы, диаграммы и т.д. Весь Ваш вклад в книгу обязательно будет учтен, и Ваше имя будет указано в списке авторов.



Скачать программу можно на <http://www.photofiltre-studio.com/pf7-en.htm>. PhotoFiltre 7 бесплатна для персонального и некоммерческого использования – для использования в коммерческих целях нужно покупать лицензию.

Кстати, пользователей Linux тоже можно оснастить: во-первых, последние версии PhotoFiltre неплохо работают под Wine, а во-вторых, существует проект энтузиастов по созданию полноценного linux-клона программы – PhotoFiltre-LX: <http://www.photofiltre-lx.org>.

Сейчас доступны исходники альфа-релиза (по лицензии GPL), но авторы принимают предзаказы для готовящейся к выходу коммерческой версии PhotoFiltre-LX Studio. Заявлено, что доходы от продажи покроют стоимость разработки, после чего проект станет полностью бесплатным и свободным!

Тимур Гафаров



Язык D

Новости «с Марса» свежие релизы и обновления

DMD 2.066.0

Вышла новая версия референсного компилятора DMD 2.066.0. Крупных нововведений в спецификации языка нет, улучшения, в основном, касаются только вспомогательных инструментов самого компилятора. Так, добавлена опция `-vgs`, которая выводит места, где программа неявно выделяет память (это могут быть изменения размеров динамических массивов, объединения строк и т.п.). Опция использует новый атрибут `@nogc` для анализа.

Кроме того, добавлена поддержка пространств имен C++ - языковая конструкция `extern(C++, namespace)`. Также появилась поддержка конструкторов для встроенных типов: `auto foo = long(1)`. Кстати, в DMD недавно был принят патч с долгожданной поддержкой формата объектных файлов COFF под Win32: вывод объектников в этом формате включается опцией `-m32mscoff`. Это позволит использовать с DMD на 32-битных системах альтернативные линкеры - например, GNU ld или MS Link. Напомним, ранее DMD поддерживал только формат OMF и, соответственно, под Windows мог работать только с OMF-линкерами, которых сейчас катастрофически мало. Затем поддержка COFF была реализована в 64-битной версии компилятора.

<http://dlang.org/download.html>

LDC 0.14.0-alpha1

Состоялся выход альфа-версии LDC 0.14.0 - компилятора D, использующего LLVM для генерации машинного кода. Релиз основан на фронтенде D2 2.065.0 и бэкенде LLVM 3.1-3.4.2.

<https://github.com/ldc-developers/ldc>
<http://forum.dlang.org/post/ixayhuvchsxuwmosebro@forum.dlang.org>

Обновление сайта GDC

Проект GDC по разработке компилятора D с бэкендом GCC обновил свой сайт. Теперь он использует `vibe.d` в качестве движка, страницы написаны на Markdown.

<http://gdcproject.org>

Профайлер Tharsis.prof

Хорошая новость для разработчиков игр: анонсирован Tharsis.prof - специализированный игровой профайлер для проектов на D. Он позволяет собирать информацию о вычислительной нагрузке пок кадрово, посредством RAII-подобного API. Характерно также отсутствие в профайлере обращений к сборщику мусора и вообще скрытых аллокаций - всю память пользователь должен предоставить самостоятельно.

<https://github.com/kiith-sa/tharsis.prof>

D:GameVFS 0.2

Виртуальная файловая система, предназначенная для разработчиков игр - D:GameVFS - обновилась до версии 0.2. Библиотека пока находится на ранней стадии разработки: API нестабилен, нет поддержки архивов, равно как и возможности удалять файлы и директории.

<https://github.com/kiith-sa/D-GameVFS>

dfuse

Разработчики Facebook открыли свой биндинг к библиотеке fuse. Поддерживает libfuse 2.8 и выше, работает на Linux и Mac OS X.

<https://github.com/facebook/dfuse>

SonarQube

Поддержка D недавно появилась для SonarQube - открытой платформы для управления качеством кода.

<http://www.sonarqube.org>

<https://github.com/economicmodeling/sonar-d-plugin>

Cerealed 0.5.0

Библиотека Cerealed обновилась до версии 0.4.1. Главное нововведение релиза - переход с классов на структуры со статическим полиморфизмом. Напомним, Cerealed - это библиотека бинарной сериализации, альтернатива таким решениям, как Orange и std.serialization.

<https://github.com/atilaneves/cerealed>

D:YAML 0.5

Вышла новая версия D:YAML - парсера языка разметки YAML. Проект ставит целью создание полноценной YAML-библиотеки для D2/Phobos. В этом релизе улучшена совместимость с DMD 2.066, сокращено количество обращений к сборщику мусора, а также проведены другие виды оптимизации. Документация по проекту перенесена на отдельный сайт.

<https://github.com/kiith-sa/D-YAML>

Поддержка D в Xamarin Studio

Разработчики Mono-D объявили о выходе плагина поддержки D для Xamarin Studio, которая является коммерческой версией IDE MonoDevelop.

<http://addins.monodevelop.com/Project/Index/27>

DCD 0.3.1

Доступна бета-версия сервера автозаполнения кода DCD 0.3.1. Это, в основном, исправляющий релиз, устраняющий некоторые баги.

Напомним, DCD - это демон автодополнения, совместимый практически со всеми IDE и текстовыми редакторами, где есть поддержка плагинов или скриптов. На данный момент DCD работает с Textadept, Kate/KDevelop, Vim, Emacs и Zeus.

<https://github.com/Hackerpilot/DCD>

DSnippets

Набор сниппетов D в формате UltiSnips для редактора Vim. Содержит полезные шаблоны кода - такие, например, как шаблоны перегрузки операторов opBinary/opUnary/opOpAssign.

<https://github.com/kiith-sa/DSnips>

Стадии программирования

Все программисты проходят несколько этапов профессионального роста. Недавно на сайте <http://gamedev.net> была представлена остроумная модель, описывающая эмоциональную эволюцию личности программиста - она основана на знаменитых «5 стадиях принятия неизбежного» из психологии.

Отрицание

Изначально программист всегда переоценивает свои возможности, поэтому берется за непосильные для него задачи. Существует много вещей, устройство которых он не может себе представить - это приводит к тому, что начинающий создает себе искусственную действительность и строит собственные теории устройства сложных программ. Этот этап называется отрицанием, так как программист находится в состоянии отрицания реальности в пользу собственных воображаемых конструкций.

Гнев

По мере роста навыков, программист сталкивается со все более сложными проблемами. На этом этапе он уже осознает границы своих возможностей и учится признавать их. Но это его сильно расстраивает: осознание своего бессилия приводит его в ярость.

Торг

Этот этап - следствие внезапного прозрения. Программист понимает, что компьютер - это безумная и непредсказуемая дьявольская машина, которая делает все, чтобы разрушить ваш мир. И для того, чтобы добиться желаемого, приходится идти на постоянные компромиссы - «сделки с дьяволом».

Депрессия

Приходит второе прозрение: на самом деле, компьютер не безумная дьявольская машина - безумие свойственно самому программисту. Осознав полную бессмысленность своих попыток понять реальность, программист ломается и впадает в длительную апатию.

Принятие

Не все добираются до этого этапа. Все ужасы остаются далеко позади, и программист достигает глубокого внутреннего спокойствия. Когда он понимает, что в жизни есть вещи важнее, чем подчинять компьютер своей воле, он преодолевает свою одержимость и начинает принимать все как есть...

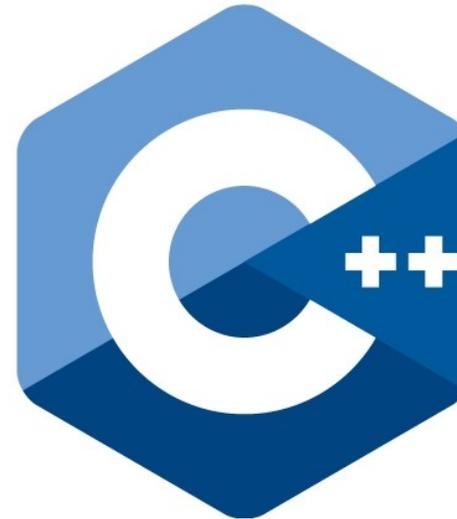


C++14: новая версия стандарта

Недавно было объявлено о готовности нового международного стандарта для языка программирования C++ – ISO/IEC 14882:2014(E), известного также под именем C++14. В настоящее время стандарт прошел утверждение и перешел на стадию подготовки документа к публикации, на которой будет проведена работа по редакторской правке орфографических ошибок и опечаток.

C++14 примечателен самым коротким сроком подготовки спецификации – так, C++11 развивался более 10 лет. Новый стандарт не содержит серьезных изменений и рассматривается в качестве корректирующего расширения стандарта C++11 с рядом улучшений и дополнений:

- Возможность указания чисел в двоичном представлении, например «a = 0b101010;»
- Поддержка автоматического определения типа, возвращаемого любой функцией, в которой используется несколько выражений return, содержащих значение одного типа;
- Дополнительный синтаксис определения типа – decltype (auto), который можно использовать для определения результирующего типа возвращаемого функцией значения;
- Поддержка захвата переменных лямбда-функций;
- Поддержка задания шаблонов для переменных;
- Расширение числа возможностей, которые можно использовать внутри функций constexpr. В частности, C++14 позволяет объявлять локальные переменные и использовать выражения if, switch, for, while, do-while;



- Добавлен атрибут «[[deprecated]]», при помощи которого можно пометить элемент устаревшим, после чего останется возможность использования этого элемента, но будут выводиться предупреждения, что в будущем его поддержка может быть прекращена.

Полноценная поддержка C++14 уже реализована в LLVM/Clang 3.4, а частичная поддержка C++14 доступна для пользователей GCC, IBM C++ и Microsoft C++.

Напомним, что в языке D все вышеперечисленные возможности существуют уже достаточно давно и, что характерно, в большинстве случаев они реализованы более основательно и элегантно, поскольку D не оглядывается на обратную совместимость с C.

История компьютерной графики

Применение графики (в особенности, трехмерной) на персональных компьютерах практически безгранично. Одной из самых популярных сфер являются, несомненно, игры - трудно назвать современную игру, которая бы не требовала видеокарты с 3D-ускорением. Кроме того, 3D-графика всегда была востребована в области научной визуализации и в инженерных приложениях.

Она используется при моделировании производственных процессов, в архитектуре и дизайне, в медицине и авиации. 3D-графика не реального времени широко применяется художниками и кинематографами при создании анимации и спецэффектов в кино.

С учетом вышесказанного, интересно было бы окунуться в прошлое и отследить основные этапы развития технологий компьютерной графики - эта сравнительно молодая отрасль за полвека существования успела пережить не одну революцию...

60-е. Первые компьютеры

Ранние компьютеры представляли собой большие панели с рядами переключателей и ламп. Инженеры работали днями и даже неделями, чтобы запрограммировать эти машины и прочесть результаты расчетов.

Полезную информацию программистам предоставляли светодиоды: можно сказать, что первой формой компьютерной графики была панель мигающих лампочек. Первые программисты рассказывают, что писали программы для создания узоров из мигающих огней!

Компьютерная графика в более-менее современной ее форме появилась в 60-х, когда компьютеры научились выводить информацию на экран электронно-лучевой трубки (ЭЛТ-монитора). Программисты развлекались созданием узоров и изображений, составленных на экране символами таблицы ASCII. Позднее это вылилось в самостоятельную форму компьютерного искусства - ASCII-арт.

Вскоре в дополнение к буквам стало возможным отображение простейшей графики: были разработаны первые алгоритмы рисования прямых и кривых линий. В 1961 году программист Стив Рассел возглавил проект по созданию первой компьютерной игры с графикой - «Spacewar!» для машины PDP-1.

В 1963 году американский ученый Айвен Сазерленд создал программно-аппаратный комплекс Sketchpad, который позволял рисовать точки, линии и окружности на ЭЛТ при помощи цифрового пера. Поддерживались базовые действия с примитивами: перемещение, копирование и т.д. По сути, это был первый в истории векторный графический редактор.



В середине 60-х гг. появились разработки в промышленных приложениях компьютерной графики. Так, под руководством Т. Мофетта и Н. Тейлора фирма Itek разработала цифровую электронную чертежную машину. В 1964 году General Motors представила систему автоматизированного проектирования DAC-1, разработанную совместно с IBM.

70-е. Начало эры ПК

Наиболее знаменательным событием в области компьютерной графики было создание в конце 70-х персонального компьютера - до этого компьютеры можно было найти лишь в больших корпорациях, университетах и исследовательских центрах. В 1976 году появился компьютер Apple I, годом позже - его более массовый преемник Apple II, который стал предвестником бума всемирной компьютеризации.

Обладая собственным дисплеем, Apple II предлагал уникальные для того времени графические возможности: 24 строки в текстовом режиме и разрешение 280x192 при 6 цветах - в графическом. По сравнению с более ранними машинами, эти возможности были хорошо документированы и просты в изучении. Тем самым, Apple II обозначил начало революции в области персональных компьютеров: это была машина для масс, а не только для ученых или инженеров.

Кстати, с тем чтобы отразить уникальную на тот момент особенность - цветную графику - логотип на корпусе компьютера содержал цвета радуги. Этот логотип Apple использовала вплоть до 2000 года.

Первоначально областью применения ПК от Apple были не графические приложения, а работа с текстами и электронными таблицами, но впоследствии их графические возможности очень быстро завоевали популярность среди фотографов, видеомонтажеров и специалистов по печати.



ПК в его современной форме появился в начале 80-х, когда компания IBM выпустила свой знаменитый PC (IBM 5150). В то время рынок персональных компьютеров был разделен между Apple и 8-битными компьютерами Commodore, Atari и Tandy, поэтому IBM старалась не упустить момент.

Архитектура IBM PC оказалась столь удачной, что используется до сих пор: так, его процессор Intel 8088 содержал набор команд, который применяется и в современных процессорах - именно от него берет свое начало архитектура x86. Популярности PC способствовала его открытая спецификация: любой сторонний производитель мог создать периферийное оборудование и ПО для этого компьютера без покупки какой-либо лицензии. Совсем скоро мир увидел первые IBM PC-совместимые компьютеры, которыми мы, по сути, пользуемся и сегодня.

80-е. Ранние GPU

Одной из ключевых особенностей PC была системная шина ISA со стандартными слотами, что позволяло подключать к компьютеру разнообразные платы расширения - видео-, звуковые, сетевые и прочие адаптеры. Со временем она была вытеснена стандартом PCI.

В те годы одним из самых мощных видеоадаптеров для PC был Hercules. Данная карта могла создавать растровые изображения с разрешением 720x348. Недостатком было то, что каждый пиксель имел только два состояния - «включено» или «выключено». Впрочем, на то время даже такая монохромная растровая графика на ПК была великим достижением!

До карт Hercules были карты CGA (Color Graphics Adapter). Появившиеся вместе с первыми IBM PC, эти карты поддерживали разрешение 320x200 пикселей и отображали одновременно 4 из 16 цветов. При использовании двух цветов возможно было еще большее разрешение - 640x200, но оно было не таким эффективным, как разрешение Hercules: цветные мониторы тогда стоили немало.



По сегодняшним стандартам возможности CGA были ничтожны, но в то время они превосходили графические возможности любительских домашних компьютеров Commodore или Atari. Не имея достаточного разрешения для бизнес-графики, CGA использовались, в основном, в играх и текстовых процессорах.

Следующим большим прорывом в графике персональных компьютеров стали представленные IBM карты EGA (Enhanced Graphics Adapter). Они поддерживали более 25 строк цветного текста в текстовом режиме и 16-цветные растровые изображения размером 640x350 - в графическом. Другие технические улучшения устранили некоторые проблемы мерцания CGA, обеспечили более гладкую анимацию. Теперь реализация сложных игр, бизнес-графики и даже трехмерной графики на ПК стала не только возможной, но и целесообразной. Это было огромным шагом со времен CGA, но в целом компьютерная графика все еще находилась в младенчестве.

Последним серийным стандартом для ПК, установленным IBM, был VGA (от Vector Graphics Array - «Матрица векторной графики», а не от Video Graphics Adapter - «адаптер видеографики», как часто ошибочно расшифровывают эту аббревиатуру). Карты с поддержкой VGA обеспечивали 16 цветов при разрешении 640x480 и 256 цветов при разрешении 320x240. Эти 256 цветов выбирались из палитры, вмещавшей более 16 миллионов возможных цветов. Именно тогда начался расцвет графики на ПК: начали появляться красочные игры, графические редакторы и другие прикладные программы с полноценным GUI.

Помимо того, для своих рабочих станций компания разработала мощную графическую карту IBM 8514. Она позволяла работать с небывалым по тем временам разрешением 1024x768 при 256 цветах! В компании предполагали, что такие мощности будут востребованы только для инженерных и научных программ, но оказалось, что и простые потребители хотят большего. Недалеко видность IBM стоила ей места законодателя рынка графики для ПК.

Вскоре начали подниматься другие производители, поставившие карты стандарта Super VGA, которые могли работать со все большим разрешением, используя все больше цветов. Вначале это было 800x600, затем - 1024x768 при 256, 32000 и 65000 цветах. В настоящее время карты с 24-битным цветом могут отображать 16 миллионов цветов при разрешениях, значительно превышающих 1024x768.

90-е. 3D-графика

Ранние потребительские видеокарты не ускоряли графические вычисления, выводя на экран непосредственно кадр безо всяких предварительных операций. Поэтому первые трехмерные игры были достаточно простыми - все вычисления по растеризации полигонов, составляющих трехмерные модели, приходилось производить на CPU. Вычисления эти были достаточно ресурсоемки, и сложная 3D-графика в реальном времени оставалась за гранью возможностей тогдашних персональных компьютеров.

Прародителем современных 3D-игр считается I, Robot от компании Atari (1983 г). В середине 80-х появилась первая игра линейки Microsoft Flight Simulator. Чисто полигональные движки тогда практически не использовались, наряду с полигональными моделями широко применялись спрайты, так что «полного 3D» еще не существовало. Наиболее ярким примером сочетания полигональных и спрайтовых объектов является игра Alone in the Dark (1992 г). Эта технология применяется и сегодня, хотя уже достаточно редко.



Некоторые привычные 3D-технологии, широко используемые сейчас, были изобретены именно в ту эпоху - например, текстурное окрашивание и использование Z-буфера. Пионером в области полноценной трехмерной графики в играх для ПК стала компания id Software с ее культовыми Wolfenstein 3D, Doom и Quake. На консолях первыми полноценную игровую 3D-графику предложили PlayStation и Nintendo 64.



Незадолго до появления Quake вышла первая версия Game SDK - прародителя DirectX, разработанного для Windows 95. В это же время получил распространение среди программистов графический стандарт OpenGL от Silicon Graphics, изначально разработанный для CAD-систем еще в 1992 году.

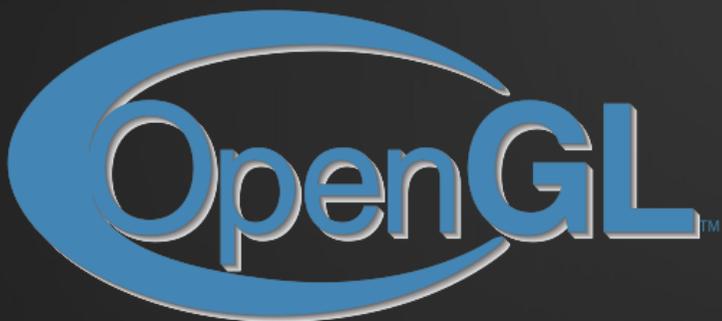
На рынке игровых видеоускорителей на тот момент бал правила компания 3DFX, создавшая в 1994 году первый полноценный графический API для игровых нужд - Glide. Именно ее производству принадлежали первые видеокарты с 3D-ускорением - Voodoo Graphics. Надо ли говорить, какой популярностью среди игроков и разработчиков игр пользовался 3D-ускоритель: адаптер мог обрабатывать до 1 миллиона треугольников и 45 миллионов пикселей в секунду, полностью брал на себя вычисления, относящиеся к 3D-сцене.

Достаточно интересным было подключение Voodoo к компьютеру: так как 2D-часть на акселераторе полностью отсутствовала, для работы была необходима еще и обычная видеокарта, которая подключалась к ускорителю. Совместное использование 3D- и 2D-карт было существенным недостатком, поэтому сразу после Voodoo Graphics вышла Voodoo Rush, сочетавшая в себе оба модуля. Впрочем, качество этой карты оставляло желать лучшего, поэтому она не смогла завоевать признание программистов.

Контроль над аппаратной частью и API позволил 3DFX на несколько лет стать практически монополистом на рынке графики на ПК. Но закрытость Glide сыграла с компанией злую шутку - когда на рынке появились карты с поддержкой OpenGL и DirectX, их встретили с распростертыми объятьями...

90-е. OpenGL и DirectX

История OpenGL берет свое начало в далеком 1982 году: именно тогда в Стэнфордском университете была разработана концепция графической машины, на основе которой фирма Silicon Graphics Inc. в своей рабочей станции IRIS реализовала конвейер рендеринга.



SGI специализировалась на производстве высокопроизводительных рабочих станций для инженерных расчетов и визуализации данных, а эта область отчаянно нуждалась в универсальном интерфейсе для вывода графики - для создания программ, способных работать на любом графическом оборудовании. Поэтому было принято решение стандартизировать метод доступа к графической аппаратуре на уровне прикладного API.

Так была разработана графическая библиотека IRIS GL, ставшая предтечей OpenGL. На ее основе в 1992 году был разработан и утвержден графический стандарт OpenGL.

Изначально созданный для профессионального сектора, OpenGL очень прочно обосновался в нем. Он использовался на инженерских рабочих станциях, в научной сфере и в образовании. На рынке компьютерных игр OpenGL изначально пользовался куда меньшей популярностью, нежели главный его конкурент - Direct3D: корпорация Microsoft взялась за разработку собственного графического API специально для ОС Windows.

Впрочем, порт OpenGL под Windows тоже существовал, но работал он куда хуже и медленнее чем «родной» Direct3D. Получилось так вовсе не из-за преимущества API от Microsoft, а по причинам скорее политическим. Порт, реализованный программистами MS, работал через медленный программный порт MCD, а не через быстрый ICD (Installable Client Driver). Сделано это было, разумеется, с прицелом на то, что разработчики игр будут работать с Direct3D, а не с OpenGL, уступающим ему в быстродействии.

Поначалу так и было, но вскоре появилась ICD-реализация OpenGL, разработанная компанией Silicon Graphics, а потом и многие производители акселераторов стали включать в комплект драйверов собственные реализации OpenGL, позволяющие наиболее эффективно использовать все возможности ускорения, которые предоставляет та или иная видеокарта.

Благодаря механизму расширений, производители могли свободно добавлять в видеодрайвер новые, не предусмотренные в OpenGL функции, что позволяло программистам использовать самые новые возможности видеокарт, не дожидаясь, пока их внесут в ядро стандарта. Это, в перспективе, выгодно отличало OpenGL от Direct3D, полностью зависимого от Microsoft.

Почувствовав, что теряет контроль над событиями, Microsoft решила прекратить войну стандартов и предложила SGI разработать единый API. Проект получил название Fahrenheit, и работа началась. MS потребовала убрать из Интернета все ссылки на SGI-реализацию OpenGL, и прекратить его дальнейшую поддержку, что и было сделано.

Увы, но API Fahrenheit не суждено было появиться на свет. Microsoft, преследуя свои выгоды, решила реализовать новый API только через Direct3D, и сотрудничества не вышло. За OpenGL-часть проекта отвечала SGI, но компания переживала не лучшие времена, специалисты уходили в nVidia, так что энтузиазма у представителей SGI не было.

Так бы и помер OpenGL тихо и незаметно, не получи он мощную поддержку со стороны id Software. Джон Кармак буквально за неделю переписал графический движок Quake под OpenGL - и это заставило зашевелиться весь мир.

Кармак продемонстрировал, что десяток строк кода OpenGL соответствует двум-трем страницам кода Direct3D, выполняющего ту же работу!

Игра имела большой успех, вскоре стали появляться продолжения и модификации - графический движок Quake стал популярен, начали выходить игры использующие его, а, следовательно, и OpenGL. id Software проводила политику выпуска игр для нескольких различных платформ, и OpenGL позволял делать это гораздо проще. Стандарт выжил, и большая гонка продолжилась.

К тому времени Microsoft вложила слишком много ресурсов в свой интерфейс Direct3D, чтобы отступить от начатого. Она начала формировать мнение, будто OpenGL предназначен для научной визуализации, а Direct3D - для визуализации в реальном времени.

В официальной литературе от MS OpenGL описывался как что-то наподобие трассировщика лучей, а не как интерфейс визуализации в реальном времени - а, между тем, он изначально разрабатывался именно в таком качестве! Это и привело к тому, что под Windows до сих пор Direct3D остается более популярным, нежели OpenGL...

Время шло, Microsoft стала монополистом на рынке программного обеспечения для ПК. Direct3D развивался, в него включались новые возможности, в играх стали использоваться все более изощренные технологии. Рынок видеокарт с 3D-ускорением резко расширился: на арену вышли NVIDIA, Maxtor, S3, а через некоторое время и ATI. Еще в конце 1997-го NVIDIA выпустила первый видеоадаптер, сочетающий в себе 2D- и 3D-ускорители, и при этом превосходящий по производительности Voodoo - NVIDIA Riva128.

В конце 1998 года вышел DirectX 6, позволявший разработчикам использовать новые технологии: Stencil, W-буфер, мультитекстурирование и многое другое. 3D-технологии развиваются с невообразимой скоростью, революция следует за революцией.

В конце 1999 года выходит DirectX 7 - очередной прорыв. На этот раз появляется аппаратный T&L (Transforms and Lighting) - модуль, отвечающий за преобразование геометрии и аппаратный расчет освещения. Постепенно оттесняя 3DFX, лидером в производстве 3D-ускорителей становится NVIDIA. С появлением RivaTNT2 в 1999 году NVIDIA отказывается от поддержки API Glide.



2000-е. Программируемый шейдер

Начало нового тысячелетия оказалось очень символическим: на рубеже веков были практически переломлены основные принципы построения трехмерной графики в видеокарте. На смену фиксированному конвейеру пришел программируемый – широкое распространение приобрели фрагментные (или, иначе, пиксельные) шейдеры, позволяющие программистам полностью контролировать процесс закрасивания полигонов.

Именно версия поддерживаемых шейдеров становится основным отличием между видеокартами следующих поколений: шейдеры появляются, например, в NVIDIA GeForce 4 и ATI Radeon 8500, поддерживаются в DirectX 8.

Знаковым также стал выход спецификации OpenGL 2.0 в 2004 году. Стандарт был существенно дополнен: в ядро OpenGL был внесен высокоуровневый шейдерный язык GLSL (OpenGL Shading Language), который позволил программистам заменить фиксированный конвейер OpenGL небольшими программами на C-подобном языке для создания различных сложных и трудно-достижимых ранее эффектов: Bump Mapping, Parallax Mapping, HDR и т.д. Аналогичный язык HLSL от Microsoft стал частью DirectX 9.



Впрочем, еще до введения GLSL, существовала возможность разрабатывать спецэффекты на платформозависимых ассемблерах (их поддержку в форме расширений OpenGL включали в драйверы некоторые производители видеокарт) и высокоуровневом кросс-платформенном языке Cg от NVIDIA, который компилировал шейдеры в специфичное для той или иной видеокарты представление.

В середине «нулевых» Microsoft допустила большой просчет, недооценив рынок мобильных платформ. Огромной популярностью стали пользоваться мобильные телефоны, оснащенные большими экранами и мощными процессорами. Компания Apple, в конце 80-х упустившая рынок персональных компьютеров, в мобильную эпоху стала производителем №1. Все больше игр и приложений стало создаваться для смартфонов и планшетов, где основным графическим интерфейсом был и остается все тот же кросс-платформенный OpenGL – а, вернее, его подмножество, названное OpenGL ES. В таких условиях использование OpenGL для разработки игр стало как никогда более оправданным с точки зрения бизнеса.



В 2008 году концерн Khronos Group выпустил спецификацию OpenGL 3.0, включавшую поддержку GLSL 1.3, массивы текстур, более гибкие объекты буфера кадра (FBO) и ряд других новых функций. Чего не было в OpenGL 3.0, так это серьезного пересмотра API, ожидаемого многими разработчиками, и ставшего доступным лишь много месяцев спустя.

В 2009 году Khronos Group анонсировал OpenGL 3.1. В новой версии произведена чистка компонентов, которые были объявлены устаревшими, но оставались в OpenGL 3.0 для сглаживания перехода на новую версию API.

Одновременно продолжалась эволюция DirectX: одним из наиболее важных прорывов стал DirectX 10, в котором дебютировали геометрические шейдеры - промежуточные между вершинными и пиксельными. Они дали возможность производить манипуляции над уже определенным массивом треугольников после окончания работы вершинного шейдера - и, самое главное, допускали произвольное изменение их геометрии. Следующая версия - DirectX 11 - привнесла поддержку аппаратной тесселяции, вычислительных шейдеров и многого другого.



2010-е. GPGPU и трассировка лучей

Прогресс никогда не стоит на месте - графические ускорители продолжают развиваться. Уже в «нулевых» стало ясно, что графические процессоры можно использовать не только для графики реального времени, но и для неграфических вычислений - стало реальностью ускорение на GPU физических расчетов, симуляции поведения частиц и многого другого. Появились специальные программные интерфейсы для адаптации вычислений под GPU - это CUDA от NVIDIA и OpenCL от AMD. Возник даже специальный термин - General-purpose GPU (графический процессор общего назначения).

Для более глубокой оптимизации приложений, использующих GPU для расчетов, программистам необходимо было более тесное взаимодействие с аппаратной частью видеокарт, и новые версии OpenGL предусматривали, в том числе, и эти потребности.

В 2010 году ARB утвердил спецификацию OpenGL 4.0. Среди нововведений можно отметить расширенную интеграцию с OpenCL (Open Computing Language), многопоточный рендеринг, обновление GLSL до версии 4.0, поддержку 64-х битных операции с плавающей точкой, а также многочисленные улучшения производительности.

Еще одна удивительная технология, ставшая реальностью в 2010-х годах - это трассировка лучей в реальном времени. Сама по себе трассировка не является чем-то новым, она применяется со времен первых попыток построить трехмерное изображение на компьютере. Она позволяет строить точные тени, отражения и преломления, симулировать поведение реальных световых лучей. Но основным ее недостатком всегда была высокая вычислительная сложность, долгое время не позволявшая использовать ее в приложениях реального времени.

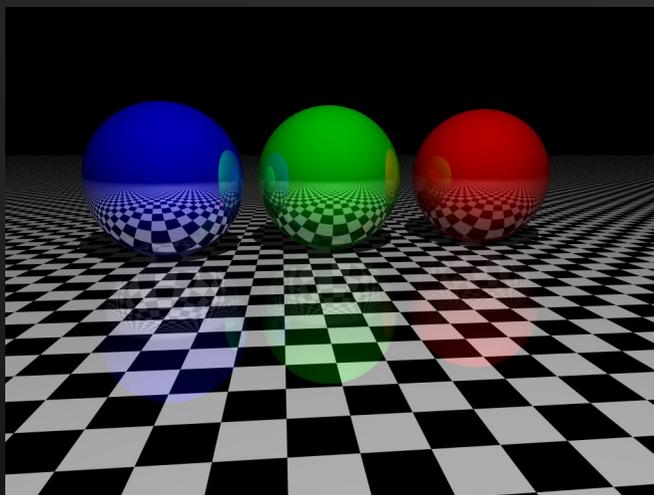


Еще одна удивительная технология, ставшая реальностью в 2010-х годах - это трассировка лучей в реальном времени.

Сама по себе трассировка не является чем-то новым, она применяется со времен первых попыток построить трехмерное изображение на компьютере. Она позволяет строить точные тени, отражения и преломления, симулировать поведение реальных световых лучей. Но основным ее недостатком всегда была высокая вычислительная сложность, долгое время не позволявшая использовать ее в приложениях реального времени.

Это стало реальностью только сейчас, когда уровень развития GPU позволил использовать их для оптимизации различных параллельных алгоритмов. Все дело в особенностях архитектуры: если современные CPU содержат всего несколько ядер (1, 2, 4, реже 8), то в графическом процессоре их количество может достигать нескольких сотен!

Разница в архитектуре обуславливает и разницу в принципах работы: CPU исторически предполагает последовательную обработку информации, а GPU рассчитан на массивно параллельные вычисления.



Трассировка лучей является примером превосходно распараллеливаемой задачи. Если вам нужно отрендерить изображение размером 640x480, то необходимо отследить ход порядка 300 000 лучей, при этом каждый из них может быть обработан независимо от остальных, что и делает трассировку столь привлекательной для реализации на GPU.



Эта технология, разумеется, еще молодая - трассировка реального времени только начинает получать свое применение в играх и программах трехмерного моделирования. Но, если утверждать, что будущее компьютеров - за параллельными вычислениями, то трассировка лучей претендует на роль ведущей технологии в компьютерной графике будущего!

Тимур Гафаров



Игровые новости из мира Linux

Недавно обновился клиент Steam для Linux: новая версия примечательна обновлением оформления и упрощением интерфейса, значительным обновлением встроенного веб-браузера и другими изменениями. Общее число игр для Linux, доступных в каталоге Steam, доведено до 630.



Игры для Linux теперь предлагает и популярный сервис GOG.com (Good Old Games). В настоящее время для загрузки доступно более 50 наименований, портированных под Linux – в их числе Floutout, Darklands, Blake Stone, Realms of the Haunting, Pirates и Colonization. Порты распространяются в пакетах для LTS-выпусков Ubuntu и Mint, для пользователей других дистрибутивов также подготовлены универсальные сборки *.tar.gz, которые можно установить вручную.

Кстати, о классических играх. Любите The Neverhood? Тогда вас наверняка заинтересует проект ScummVM – полностью свободный кроссплатформенный интерпретатор классических квестов, позволяющий запускать многие игры прошлого на платформах, для которых они изначально не предназначены.



В ScummVM 1.7.0 добавлена поддержка вывода с использованием OpenGL, представлены многочисленные улучшения в GUI, расширены возможности движка AGOS, добавлена экспериментальная поддержка консоли OUYA, заметно улучшен порт PS2, включены порты для платформ Bada и Tizen.

Из игр, поддержка которых появилась в ScummVM 1.7.0, можно отметить The Neverhood, Return to Ringworld, Chivalry is Not Dead, Mortville Manor и Voyeur.



А компания id Software недавно опубликовала под лицензией GPLv2 исходники классического платформера Commander Keen 3.5, известного также как Keen Dreams – это одна из первых игр, созданных Джоном Кармаком. Изначально игра была выпущена в 1992 году для MS-DOS, но в прошлом году была портирована на Android.

Для компиляции оригинальной игры требуется Borland C++ 3.1, но энтузиасты уже начали подготовку Linux-порта, переведенного на библиотеку SDL и адаптированного для сборки при помощи GCC.



Стартовала кампания по сбору средств на создание свободных ресурсов (изображений и звуков) для игры по мотивам классической экономической стратегии Caesar III. Свободная реализация движка игры уже готова – частично она была получена путем реверс-инжиниринга оригинала, что позволило максимально точно перенести игровую механику.

Движок распространяется по лицензии GPLv3 и поддерживает Linux, Android, Haiku, Mac OS X и Windows. Основной проблемой является то, что для полноценной игры требуются оригинальные ресурсы, которые теперь необходимо воссоздать с нуля.

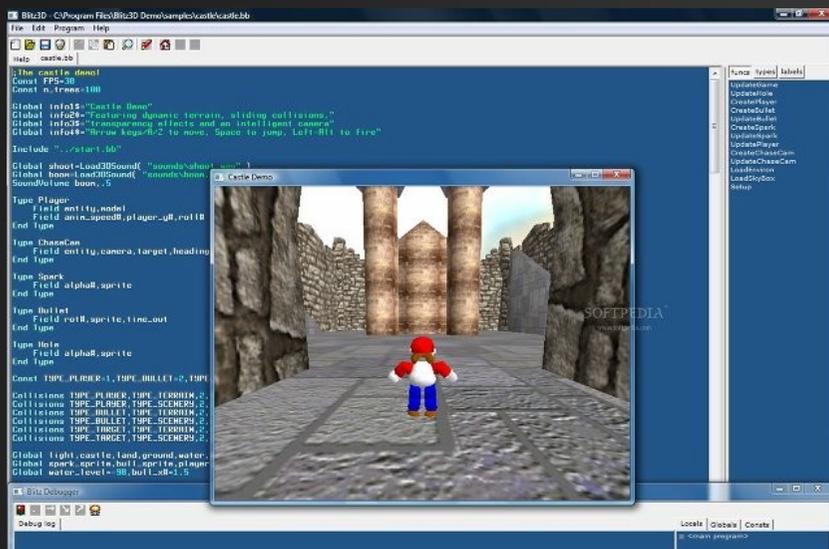
Еще одна интересная стратегия, которая недавно обзавелась Linux-версией – это транспортный симулятор Train Fever. Интересно, что игра была разработана с привлечением 300 тысяч евро через систему совместного финансирования. Проект развивается как продолжение таких игр, как Transport Tycoon, Railroad Tycoon, Cities in Motion и Open TTD. Игра охватывает развитие транспорта с 1850 по 2050 год.



Особенностью игры является процедурная генерация городов. В задачи игрока входит строительство инфраструктуры – железнодорожных путей и станций, создание автобусных и трамвайных маршрутов, а также организация грузоперевозок. Главная цель игрока – развить свою транспортную компанию до огромной корпорации.

Компания Blitz Research Ltd открыла исходные коды двух своих продуктов подряд: Blitz3D и Blitz Plus. Blitz3D – это среда разработки трехмерных игр на языке Blitz Basic с использованием DirectX 7, включающая компилятор, отладчик, IDE, графический движок, полный комплект документации и множество примеров кода. Она пользовалась большой популярностью в начале и середине «нулевых» за счет своей простоты и дружелюбности к начинающим игроделам – на основе этого движка было выпущено множество бесплатных и коммерческих игр. Blitz Plus – аналогичная среда для создания 2D-игр и GUI-приложений.

В последние годы Blitz3D практически не развивался, будучи вытесненным Blitz Max – более свежим коммерческим решением от той же компании. Сейчас, за счет поддержки сообщества, вполне можно ожидать много нововведений в этих движках – поддержку современных графических API, порт на Linux, уход от привязки к Basic и т.д. Исходники Blitz3D и Blitz Plus доступны на [GitHub](#) по лицензии zlib/libpng.



Кстати, BlitzResearch сейчас активно развивает свой новый кроссплатформенный движок [Monkey X](#), который включает собственный объектно-ориентированный язык Monkey, транслирующийся в C++, C#, Java, JavaScript и Flash.

Движок привлекателен внушительным списком поддерживаемых платформ: Windows, Linux, MacOS X, Android, iOS, Windows Phone, XNA, Flash, HTML5, PlayStation Vita и Ouya. Изначально Monkey X предназначен для создания 2D-игр, но для него существует фреймворк miniB3D – легковесный 3D-движок в стиле Blitz3D. Также поддерживается физический движок Vox2D.

Еще одним «открытием» этого лета стал тулkit для разработки игр [Game Develop](#), который теперь тоже свободен и абсолютно бесплатен для любого использования, включая коммерческое. Инструмент интересен своей визуальной IDE, не требующей программирования: вся игровая логика задается при помощи кнопок и контроллеров. При этом логика компилируется в машинный код, как настоящий язык программирования.

Движок Game Develop ориентирован на двумерную графику, поддерживает динамическое освещение, системы частиц, включает физический и сетевой компоненты. Также имеется возможность расширять функциональность при помощи плагинов.



Компания Google, между тем, представила открытый проект Liquid-Fun, в рамках которого развивается физический движок с поддержкой симуляции жидкостей.

Библиотека написана на языке C++, но поддерживается вариант для языка Java, работающий через SWIG и позволяющий использовать Liquid-Fun в том числе и на Android.

Технически, LiquidFun является расширением движка Vox2D, реализующим элементы гидродинамики и позволяющим симулировать течение жидкости, поведение частиц, волновые процессы и трансформацию твердых тел при давлении.



В версии LiquidFun 1.1 добавлена поддержка JavaScript и iOS, библиотека оптимизирована под ARM. Из недавно созданных на базе движка приложений отмечается игра-платформер VoltAir и программа для детского творчества LiquidFun Paint.

А разработчики Unity объявили о открытии исходников некоторых компонентов своего движка. В первую очередь это касается Unity Test Tools и новой системы UI, которые будут открыты под лицензией MIT.

Этим шагом компания стремится достичь более тесного диалога с сообществом пользователей Unity, обеспечив им высокий уровень гибкости и контроля над наиболее часто используемыми компонентами движка.



Подчеркивается, что данная инициатива – это новый шаг в Open-Source-деятельности команды Unity, которая уже включает открытые форки Mono и инструментов для работы с шейдерами (hlsl2glsl и glsl-optimizer), а также различные вспомогательные проекты.

Кстати, компания Sony Computer Entertainment недавно открыла [исходники редактора уровней](#). Программа использует DirectX 11 для рендеринга 3D-графики и предоставляет полноценный WYSIWYG-интерфейс, позволяющий строить уровни для широкого спектра игр.



После выпуска OpenGL 4.5 консорциум Khronos Group приступил к работе над спецификацией нового поколения – OpenGL NG. Ожидается, что этот API не будет обратно совместимым с предыдущими реализациями. Обратной совместимостью придется пожертвовать ради революционных возможностей: разработчики приложений получают полный низкоуровневый доступ к функциональности современных графических процессоров. Данная инициатива встретила одобрение в AMD: на конференции SIGGRAPH компания выступила с предложением взять API Mantle в качестве основы для нового стандарта.



Напомним, в рамках проекта Mantle компания AMD разрабатывает новый программный интерфейс рендеринга графики, ориентированный на разработку игр и графических приложений. Mantle отличается от ныне существующего OpenGL простотой, изначальной ориентацией на переносимость, высокой эффективностью работы с GPU, возможностью прямого обращения к видеопамяти.

Экспериментальная поддержка Mantle уже добавлена в драйвер Catalyst для Windows и ожидается для Linux, OS X, Android и iOS. Mantle не привязан к продуктам AMD и абстрагирован от аппаратного обеспечения, что позволяет реализовать этот API для видеокарт от других производителей.

Кстати, в Mesa, свободной реализации OpenGL для Unix-подобных систем, недавно была реализована поддержка сжатия текстур BPTC (расширение GL_ARB_texture_compression_bptc). Поддержка данного алгоритма сжатия текстур является одним из требований спецификаций OpenGL 4.2.

Ищем авторов!

Если вы – околокомпьютерный журналист (неважно, начинающий или опытный) и хотите расширить свою аудиторию, поделиться опытом и знаниями с тысячами читателей, найти единомышленников или просто высказать свою точку зрения – напишите нам! Мы будем рады опубликовать на страницах «FPS» любой материал, соответствующий тематике журнала.

Это могут быть статьи, уроки, обзоры, интервью, художественная литература, авторские фото, творческие работы, исходный код и т.д.

Вы можете выбрать одну из следующих тем: разработка компьютерных или приставочных игр, программирование, компьютерная графика, рендеринг, шейдеры, цифровой звук и музыка, геймдизайн, написание игровых сценариев, обзоры программ, движков и игровых конструкторов, отзывы и рецензии на игры, вопросы выбора лицензии, издания, дистрибьютинга или продвижения проектов, репортажи с различных конференций, выставок, встреч разработчиков и демо-пати, интересные факты из истории игр, обзоры «железа» и игровых платформ, обзоры полезных сайтов и сервисов.

Материалы принимаются в любое время на почтовый ящик редакции: gecko0307@gmail.com.

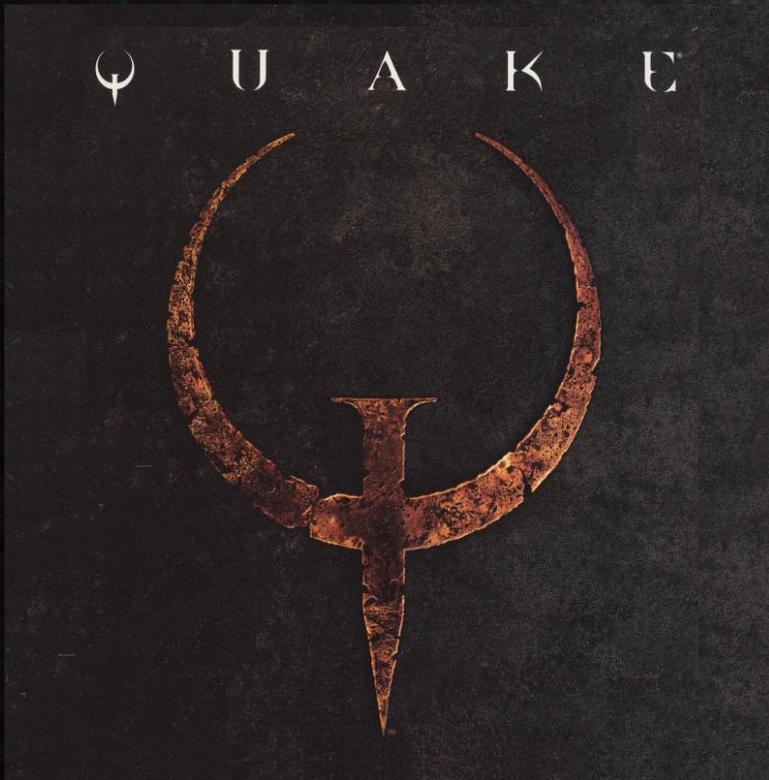
Прикрепленные файлы желательно архивировать в форматах zip, rar, 7z, tar.gz или tar.bz2. В письме или тексте материала укажите Ваши реальные имя и фамилию.

**Любой Читатель может стать Автором!
Желаем Вам творческих успехов!**

Культовые игры:

QUAKE

Quake – это настоящая легенда в мире компьютерных игр: эталонный шутер 90-х, наряду с Doom и Duke Nukem, являющийся признанной классикой жанра. Quake является родоначальником одной из самых успешных игровых серий: в сумме было продано более 4 миллионов копий Quake, Quake II и Quake III: Arena.



С появлением Quake зародилось понятие «киберспорт», стали проводиться крупные чемпионаты. Благодаря популярности игры Quake, в 1997 году в США появилась первая лига киберспортсменов – Cyberathlete Professional League (CPL). Даже нынешние мастера киберспорта, смахивая слезу, вспоминают, что в плане deathmatch'a первый Quake не превзойден ни одной игрой – разве что только своим сиквелом Quake II.

Сюжет Quake незамысловат. Главный герой – безмянный солдат, которого послали выполнять миссию по уничтожению неведомого врага, который засылает на Землю легионы монстров через телепортационные врата. Вам придется пройти через эти врата в причудливое готическое параллельное измерение и уничтожить Зло...

«Quake» – это кодовое имя главного босса игры, Шуб-Ниггурат, которая попала в игру из вселенной ужасов культового писателя Говарда Лавкрафта. Впрочем, в Quake это существо больше напоминает не саму Шуб-Ниггурат, а одного из ее потомков.

Графическая система Quake была первым в мире полностью полигональным 3D-движком – это был настоящий переворот в мире игровой графики. В это сложно поверить, но, в том или ином виде, движок Quake используется до сих пор: он лег в основу Half-Life и множества различных популярных модов. После того, как id Software открыла исходники Quake и Quake II, появилось немало свободных игр, использующих эти технологии.

Движок Quake (Quake Engine) иногда называют id Tech 1 – как первую в линейке версий фирменного движка id Software. Он был написан Джоном Кармаком в 1996 году на языке C специально для игры Quake. К разработке также был причастен Майкл Абраш, который помогал Кармаку с алгоритмами и оптимизацией кода. Первоначально движок поддерживал DOS и Windows, однако открытый исходный код и использование OpenGL в итоге привели к тому, что Quake был портирован практически на все современные платформы, где возможна хоть какая-то 3D-графика.

Интересна реализация карт в QE: моделирование игровых уровней для движка выполняется при помощи конструктивной сплошной геометрии (Constructive Solid Geometry). Этот подход унаследовали Valve в их движке Source. Стены уровня строятся левелдизайнером из простых выпуклых геометрических объектов, которые называются «кистями». Кисти размещаются и поворачиваются так, чтобы создать закрытые пространства-комнаты. Затем карта пропускается через специальный препроцессор, который убирает «лишнюю» геометрию и оставляет только те полигоны, которые будут видны игроку изнутри карты.



Обработанная таким образом карта имеет существенно меньшее число полигонов, но на отсечение невидимой геометрии уходило много времени: на компьютерах тех лет эта операция занимала до нескольких часов! При этом часто возникали ситуации, когда препроцессор не мог завершить свою работу из-за случайных отверстий, связывающих внутреннее пространство карты с внешним – так называемых «протечек». Найти и ликвидировать эти «протечки», образовавшиеся между несколькими неаккуратно состыкованными объектами, было для дизайнера нелегкой задачей...

Но как же тогда, спросит читатель, реализовано небо с облаками, которое видно на некоторых картах? На самом деле, окна и отверстия в потолке, через которые виден скайбокс неба, образованы просто-напросто невидимыми «кистями» – никаких «утечек» в данном случае нет. Кстати, эту возможность использовали некоторые особо продвинутые читеры, которые редактировали карту, добавляя в нее дополнительные окна – чтобы видеть передвижения противников во время сетевой игры.

Еще одно изобретение id Software, впервые опробованное в Quake – статические карты освещения: вместо того, чтобы вычислять освещенность неподвижных объектов каждый кадр заново, используется текстура с предрасчитанными тенями и световыми пятнами. Эта методика с тех пор используется во многих играх, особенно в шутерах от первого лица – хотя на сегодняшний день она уже слегка устарела и постепенно вытесняется динамическим шейдерным освещением на GPU.



Карты для серии Quake и множества других игр из ее семейства традиционно моделируются при помощи редактора GtkRadiant. Это свободная программа, распространяющаяся по лицензии GNU GPL, но уходящая корнями во внутренние проприетарные инструменты id Software. Одним из них был QuakeEd, первый редактор карт для Quake под операционную систему NeXTSTEP (которая, кстати, наряду с BSD, является прародителем Mac OS X).

Первый прямой предок GtkRadiant – это QE4, инструмент для создания карт Quake II, который поставлялся вместе с Quake II SDK. Когда компания открыла исходники QE4, энтузиасты разработали на ее основе ставший весьма популярным редактор QERadiant. id Software взяла его за основу для создания Q3Radiant – редактора карт для Quake III Arena.

GtkRadiant, появившийся в 2001 году – это форк Q3Radiant, включающий два главных изменения: использование GTK+, благодаря чему обеспечивалась поддержка Linux и Mac OS X, а также независимость от технологий id Software, что позволяло создавать в нем карты для любых игр. Сейчас GtkRadiant поддерживает не только Quake I, II, III, IV, но и Return To Castle Wolfenstein, игры серии Star Wars Jedi Knight, частично Call of Duty и многие другие.

Движок Quake был одной из первых игр на ПК, которые поддерживали аппаратное 3D-ускорение. Первоначально трехмерная графика рассчитывалась чисто программно, но позже Кармак включил поддержку видеокарты Verite 1000 от компании Rendition – эта версия получила название VQuake. Затем движок был переписан с использованием OpenGL – родилась GLQuake, которая стала первым приложением, которое на практике демонстрировало возможности чипсета Voodoo от компании 3DFX.



Помимо революционной на тот момент графической технологии, Quake Engine примечателен также тем, что это был, по сути, первый движок с открытой архитектурой. Это означало, что любой желающий мог модифицировать игру до неузнаваемости, меняя ресурсы и гибко настраивая поведение игровых объектов на специальном скриптовом языке (Quake C). Движок содержал консоль, в которой используется своя система команд – эту идею впоследствии подхватили и другие разработчики.



Культ моддинга во многом обязан своему возникновению именно Quake – это одна из первых игр, для которой были выпущены сторонние модификации, добавлявшие новые локации, оружие, персонажей и вводявшие новые правила в игровой процесс. Например, в мультиплеерном моде HeadHunters голову убитого врага необходимо было донести до алтаря – чем больше одновременно сданных голов, тем больше фрагов...

Многие разработчики модов для Quake впоследствии основали собственные компании на рынке. В качестве примера можно привести студию Ritual Entertainment, разработавшую официальное дополнение для первой части Quake, а впоследствии и другие самостоятельные проекты, такие как шутер SiN.

Что еще родилось как мод для Quake? Это, конечно, всем известный многопользовательский шутер Team Fortress. Его первая версия была создана австралийцами Робинот Уокером, Джоном Куком и Яном Кофли в 1996 году. Мод оказался настолько успешен, что Уокер, Кук и Кофли организовали собственную компанию Team Fortress Software и задумали создать продолжение Team Fortress 2 – коммерческий мод для Quake II.

Однако в 1998 году их поглотила Valve, и разработка Team Fortress 2 была продолжена на движке Half-Life. В конечном счете, игра была выпущена под именем Team Fortress: Brotherhood in Arms. Выход Team Fortress 2 был отсрочен, а его движок был переработан для создания Half-Life 2. Позже этот движок получил название Source Engine. TF2 вышел только в 2006 году в рамках сборника The Orange Box, в который также вошли такие культовые тайтлы, как Portal и Half-Life 2: Episode Two.

Что же касается самого Half-Life, то он возник не как мод, а как игра на основе лицензированного движка Quake. Авторы игры – бывшие программисты Microsoft Гейб Ньюэлл и Майкл Харрингтон – мечтали создать страшный 3D-шутер о вторжении пришельцев. Первоначально разработка носила название Quiver и создавалась по мотивам повести Стивена Кинга «Туман». На концепцию будущей игры также повлиял телесериал «Секретные материалы».

Окончательное название Half-Life было взято из терминологии физиков: оно переводится как «период полураспада» (а не как «полуживой» или «полужизнь», как считают некоторые). Это название подошло к сюжету – игра повествует о приключениях ученого Гордона Фримена в исследовательском центре Black Mesa, подвергшейся нападению монстров из параллельного мира.

Движок Half-Life получил название GoldSource (или GoldSrc). В его основе лежит код QuakeWorld – обновленной версии Quake с улучшенным многопользовательским режимом. Использовать движок Quake, кстати, основателям Valve предложил не кто иной, как Майкл Абраш, один из главных разработчиков Quake.

Силами Гейба и его коллег движок был сильно улучшен и дополнен: появилась поддержка Direct3D, революционная на тот момент скелетная анимация и многое другое. Впоследствии Valve открыла исходники GoldSource – его можно найти на [GitHub](#). Преемником GoldSource стал движок Source, на котором построены Half-Life 2 и множество других игр от Valve и сторонних компаний.

Если говорить о наиболее известных модах Half-Life, то на первом месте стоит, безусловно, культовый Counter Strike. Его создателями были моддеры Quake II, которые задумали создать собственную многопользовательскую онлайн-игру. Первый вариант Counter Strike зародился именно как мод для Quake II и назывался тогда Action Quake 2 (AQ2). Он был разработан командой A-Team в 1998 году – после этого команда разделилась, и бывшие ее участники работали над модами Action Half-Life и Counter Strike для Half-Life.



Будучи бесплатной, игра стремительными темпами завоевала аудиторию, успешно конкурируя с коммерческими Quake III и Unreal Tournament. Вскоре к проекту подключилась сама Valve, выкупив игру и наняв разработчиков в свой штат.

В том же 1999 году id Software открыла исходники движка Quake, и это привело к созданию огромного количества разнообразных форков и модификаций, а также позволило энтузиастам портировать движок на другие платформы, в том числе мобильные – например, Symbian и J2ME.

Наиболее известными бесплатными игровыми движками, основанными на коде Quake Engine, стали DarkPlaces и Tenebrae. Первый приобрел известность как движок, на котором построен популярный свободный многопользовательский шутер Nexuiz. Изначально эта игра тоже создавалась как мод для Quake, но затем была перенесена на самостоятельную основу.

По сравнению со своим оригиналом, DarkPlaces имеет ряд серьезных улучшений: например, поддержку динамических теней и эффектов пост-обработки, различных методов рельефного текстурирования, карт формата id Tech 3 и скелетной анимации для персонажей. DarkPlaces поддерживает Windows, Linux и Mac OS X.



Еще одной свободной игрой, использующей DarkPlaces, стала Xonotic – форк Nexuiz. Форк возник как реакция сообщества на коммерциализацию Nexuiz: в 2010 году игра была куплена фирмой 3MFonic, которая отказалась делиться своими наработками с сообществом свободного ПО.

Из прочих игр, так или иначе основанных на Quake Engine, отдельного упоминания заслуживает Nexen II, разработанный студией Raven Software и изданный id Software. Она является третьей игрой в своей серии, после Heretic и его сиквела Nexen, основанных на движке Doom – это шутеры в средневековом антураже и с фэнтезийными элементами в духе RPG. От своих предшественников Nexen II отличается полноценной трехмерной графикой и еще большим упором на ролевые элементы и решение головоломок.



Технологии, составлявшие Quake Engine, легли в основу последующих движков id Software – в первую очередь, id Tech 2, на базе которого создан Quake II (1997 год). С этим можно поспорить, но эта игра, пожалуй, даже круче своей предшественницы. По стилистике она не имеет ничего общего с первым Quake и выполнена в научно-фантастическом стиле: главным героем является земной десантник, высадившийся на планету враждебной расы пришельцев – строггов. Главной целью этой операции было закрытие портала, по которому строгги переправляют свои войска на Землю.

Quake II официально вышла не только на ПК, но и на PlayStation, Nintendo 64 и других платформах. В основу движка id Tech 2 вошли наработки сразу нескольких версий Quake Engine – GLQuake, Quake World и WinQuake. Наряду с программным рендерингом, игра изначально включала поддержку OpenGL. Технологически движок близок к оригинальному Quake Engine, но имеет ряд существенных отличий – например, модульную архитектуру: рендерер был вынесен в отдельную DLL-библиотеку, что позволило подключать к игре альтернативные рендер-движки и переключаться между ними без перекомпиляции самой игры.



В 2001 году исходники id Tech 2 были выпущены под свободной лицензией GPL. Из самых известных игр, использующих эту технологию, можно назвать Heretic II, SiN, Daikatana и отечественный «Спецназ: Антитеррор».

Третья часть серии вышла в 1999 году. Quake III: Arena был изначально задуман как мультиплеерный шутер и не имел полноценного однопользовательского режима – цель одиночной игры состояла в поэтапном сражении с ботами.

Зато развитый многопользовательский режим, основной козырь Q3, предлагал игрокам аж четыре разных варианта игры по сети: классический deathmatch в духе «каждый сам за себя», командный deatchmatch, турнир и «захват флага». Дополнительные моды реализуют и другие режимы – например, «Freeze tag», «Rocket arena», «Excessive», «ExcessivePlus». Существует также «Defrag» – мод для тренировки техники передвижения и точности стрельбы.

Игра использовала движок id Tech 3, который был значительно более требовательным к ресурсам, нежели его предшественники. Это оттолкнуло от игры часть фанатов – игроки Quake II с большой неохотой переходили на третью часть, мотивируя это неудобностью, непривычностью и ресурсоемкостью.

id Tech 3 не основывался на второй версии движка и был написан с нуля. Технологическим прорывом стало использование программных шейдеров для задания материалов.



В 2005 году вышел Quake IV, который продолжил историю сражения землян и строггов – игрок выступает в роли десантника, участвующего в контратаке на планету-столицу строггов. Сюжет начинается через несколько недель после окончания Quake II: земное командование, узнав о смерти Макрона, делает вторую попытку вторжения на планету Строггос.

Quake IV использует движок id Tech 4, на котором также основан Doom 3 и много других игр от id Software. Это был очередной крупный технологический прорыв – на время выхода id Tech 4 был одним из самых инновационных игровых движков. Изначально он планировался как расширение id Tech 3, но по ходу разработки программисты решили перейти с языка C на C++, и в результате движок был почти полностью переписан.



Движок отличался очень высокой по тем временам требовательностью к ресурсам – изначально он нуждался в видеокарте с поддержкой OpenGL 2.0, широко использовал вершинные и фрагментные шейдеры. Кармак предостерегал игроков от покупки видеокарт GeForce 4 MX, так как они поддерживали только OpenGL 1.x, но в итоге, «по просьбам трудящихся», id Software все-таки добавила в движок поддержку этих карт.

Главными новшествами движка были динамическое попиксельное освещение на шейдерах, эффект bump mapping, тени на основе теневых объемов, а также мегатекстуры (которые, впрочем использует лишь одна игра на id Tech 4 – это Enemy Territory: Quake Wars). Doom 3 и Quake IV задали планку коммерческим продуктам – все эти новые технологии быстро приобрели статус стандартных для игр AAA-класса. Кстати, после выхода id Tech 5 четвертая версия движка была выпущена как свободное ПО под лицензией GPL.

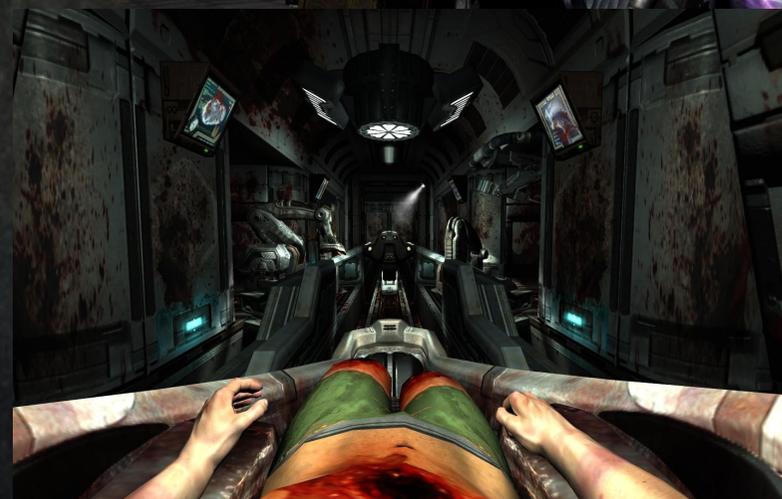
После Quake IV компания выпустила еще одну игру, которую можно отнести к серии Quake – тактический шутер Enemy Territory: Quake Wars (2007 год). Как и третья часть, она также ориентируется на мультиплеер, а по сюжету она является приквелом Quake II – действие игры происходит за 50 лет до событий второй части.

На этом линейка версий легендарного шутера заканчивается – но, надеемся, не навсегда. В 2011 году Джон Кармак в интервью одному из игровых порталов рассказал о том, что история противостояния землян и строггов завершена, и, если Quake 5 и выйдет когда-нибудь, то его сюжет будет в той или иной мере продолжать классический Quake с его вселенной лавкрафтовских чудовищ: «Продолжать делать что-то в стиле Quake IV абсолютно бессмысленно. Оригинальная атмосфера Quake нам кажется намного интереснее. Мы хотим забыть о вселенной строггов и, скорее всего, мы вернемся к причудливому миру первого Quake...»

Однако в 2013 году поклонников ожидала печальная новость: Кармак покинул id Software, присоединившись к команде разработчиков шлема виртуальной реальности Oculus Rift. С тех пор о судьбе Quake 5 практически ничего не известно.

Сейчас id работает над Doom 4, релиз которого, по неофициальным данным, намечен на 2015 год. Очень может быть, что Quake 5 станет следующим большим проектом компании – ведь с годами серия не потеряла популярности, и количество ее поклонников продолжает расти.

Тимур Гафаров



Telltale Games:

СКАЗКИ ДЛЯ ВЗРОСЛЫХ

С развитием игровой индустрии и увеличением компаний, штампующих похожие друг на друга шутеры и квесты, казалось бы, ничего нового уже придумать нельзя. Но в 2004 году появляется на игровом рынке компания Telltale Games, решившая перевернуть наше представление об играх.

Компания занялась выпуском так называемых эпизодических игр. Каждая игра разделяется на эпизоды, которые выпускаются в разные временные промежутки. Согласитесь, что это похоже на выпуски сериала? Так и есть - типичные игры TTG очень похожи на теленовеллы.

С 2004 года TTG выпустила довольно много игр, большинство из которых имеют разделение на эпизоды. Компания тесно сотрудничает с киностудиями - было произведено несколько игр фильм, например, «Назад в будущее» и «Парк Юрского периода».



В 2007 вышла игра в жанре квест «Sam and Max», основанная на комиксах Стива Перселла про антропоморфного пса и кроликоподобного существа, которые вместе совершают расследования. Игра снискала довольно большую популярность, было выпущено два сезона игры с 6 сериями в каждом.



Позже компания берется за еще несколько проектов - таких, как игра по английскому мультфильму «Wallace & Gromit's Grand Adventures», пиратские приключения «Tales of Monkey Island» (пятая часть легендарного квеста от LucasArts) и несколько игр по известным фильмам. Однако большой успех у геймеров обрела игра по популярным комиксам «Ходячие мертвецы» - «The Walking Dead: The Game».



«The Walking Dead: The Game» увидели свет в 2012-ом, в этом же году ее объявили игрой года несколько известных игровых изданий.

Сам сюжет «Мертвецов» построен по мотивам комиксов Роберта Киркмана: центральным персонажем является Ли Эверетт, который был под арестом, пока на Земле наступил зомби-апокалипсис.

На данный момент выпущено два сезона «Ходячих», планируется третий. Чем же так полюбилась эта игровая серия геймерам? Может, из-за нестандартной механики? Или из-за того, что в центре игры - не кровавое месиво из зомби, а переживания и чувства людей? А, может, из-за необычного графического оформления (разработчики пытались воссоздать в игре атмосферу комикса)?



Скелетом геймплея являются отношения между людьми, диалоги. Вам приходится выбирать ответ на вопрос очень быстро, что похоже на реальный разговор.

В игру полностью погружаешься, она похожа на сериал, но ты принимаешь в нем участие и можешь влиять на ход развития сюжета - разве это не самое интересное?..

Компания продолжала выпускать новые игры - в 2013-ом появился «The Wolf Among Us». За основу опять-таки взят комикс - на этот раз «Fables». «Волк среди нас» - первая эпизодическая игра в стиле нуар с тем же привычным интерфейсом «point-and-click», в которой вам предстоит сыграть за детектива, раскрывающего убийство проститутки.

Кроме детектива в игре много героев, с которыми мы постепенно знакомимся. Это и Белоснежка, Красавица и Чудовище, Дровосек... Все герои не так просты, как кажется. Многие положительные герои оказались плохими, многие в этом мире страдают. Страшно, когда Русалочка идет на панель, а Прекрасный Принц вышибает себе мозги от безысходности...



Все было бы слишком избито, будь он обычным детективом, но он - Серый Волк, а весь мир - это место, куда сбежали герои из известных сказок в поисках лучшей жизни.

Но все оказалось не так просто: героям пришлось заниматься черной работой, проституцией, появились алкоголики и наркоманы. Все Сказания погрузились в серую действительность, из которой они не могут сбежать.

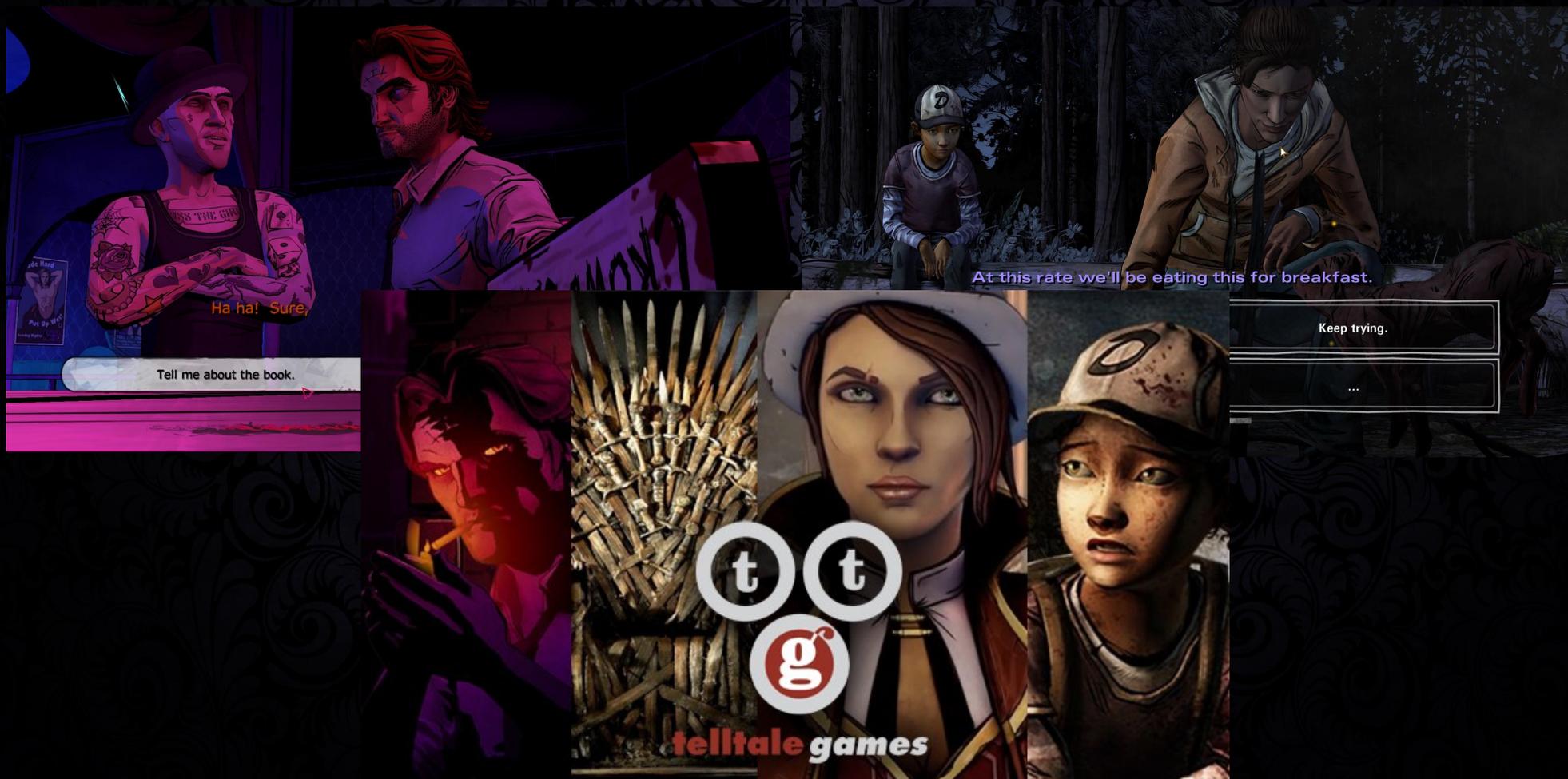
Тут-то и совершается убийство одной из проституток, раскрытием которого стал заниматься детектив Серый Волк.



Следует отметить, что «The Walking Dead: The Game» и «The Wolf Among Us» имеют общую стилистику и графическую оболочку. Также их роднит одинаковая игровая механика и важность человеческих (и не совсем человеческих) отношений. Обе игры были высоко оценены критиками и геймерами.

Telltall Games серьезно взялись за переложение известных выдуманных миров в игры. Сейчас кипит работа над «The Game of Thrones» и «Tales from the Borderlands», которыми «сказочники» обещают порадовать геймеров уже в этом году.

Наталья Чумакова



RetroGame

Коллекция одскульного геймера

Мы продолжаем обзор игр на NES, которые и по прошествии многих лет не теряют привлекательности и шарма – их приятно проходить и сейчас (хотя бы на эмуляторе), в эпоху навороченной 3D-графики и игр AAA-класса...

Rainbow Island

Эта игра от легендарной компании Taito, разработчика таких шедевров, как Bubble Bobble и его сиквел Bubble Bobble II, а также Newzeeland Story и прочих.

Кажется, что фантазия разработчиков безгранична – они в очередной раз доказали это, создав продолжение игры BVII.

Игра рассказывает о приключениях одного из персонажей предыдущих игр серии, который таки превратился из дракончика обратно в человека. Теперь ему (то есть, вам в его роли) придется спасти волшебные Радужные острова от злых чар неких злодеев...



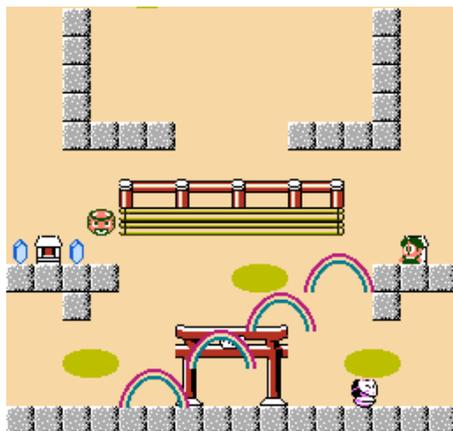
Игра сделана по образу и подобию серии Bubble Bobble – вы перемещаетесь по платформам в многоэтажных комнатах, которые населяют полчища врагов. Основное движение – вверх: в этом как раз и помогают волшебные радужные мосты. На последних уровнях платформ практически нет, и вам придется продвигаться вверх только с помощью радуги.

Чего не отнять у игр Taito – это количество и разнообразие бонусов. Вам встретятся драгоценности, кристаллы, фрукты, тортики – словом, все как в Bubble Bobble. Призы лежат на земле и выбиваются из врагов. Есть также бонусные уровни, где необходимо собрать все призы за выделенное время.

Пройдя несколько уровней, вы попадаете к боссу – в первом мире это большой паук. Боссы побеждаются теми же радугами, никакого другого оружия в Rainbow Islands нет. Что немного расстраивает, потому что в том же Bubble Bobble, например, были такие спецсредства, как волна воды, сносящая на своем пути врагов.

Кстати, главари получились весьма неоднозначными. Складывается впечатление, что над игрой работали два разных дизайнера: первые миры с их боссами получились очень оригинальные и вполне соответствующие общей атмосфере игры и ее посылу. Но вторая половина боссов была либо сделана в спешке, либо вовсе создавалась для какого-то другого проекта.

Когда я впервые проходил Rainbow Island на консоли, я предвкушал какого-то супер-босса и, соответственно, шикарную концовку, но меня ждало разочарование: главарем космического мира была какая-то совершенно неуместная огромная летающая рыба. То же самое с уровнями: если в начале все миры выстроены более-менее в соответствии с законами жанра (замок, мир ковбоев, мир в восточной стилистике и т.д.), то ближе к концу герой неожиданно попадает в некие сюрреалистические пространства вроде мира геометрических фигур.



Графика в игре выполнена в духе Bubble Bobble – не совсем пиксель-арт, а, скорее, нечто схожее с векторной графикой. Первые уровни, как уже было упомянуто, выполнены блестяще и со вкусом, но к концу, по всей видимости, фантазия художников исчерпалась.

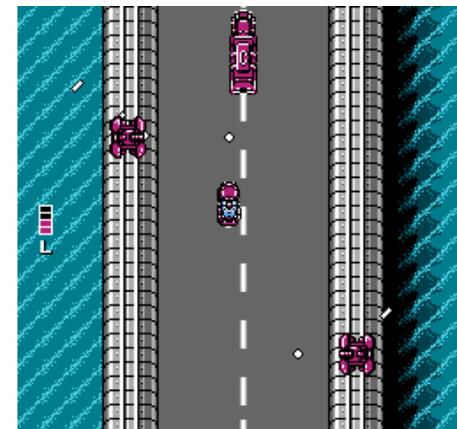


Наверное, самая слабая часть игры – музыка. Есть неплохие мелодии – в основном, опять-таки, в начале игры – и я бы промолчал о музыке, если бы не то, что в конце! Это откровенный кошмар – их тоже будто делал другой композитор, и притом совершенно некачественно. Что касается звуковых эффектов, то они не слишком запоминающиеся – я бы сказал, самые обыкновенные.

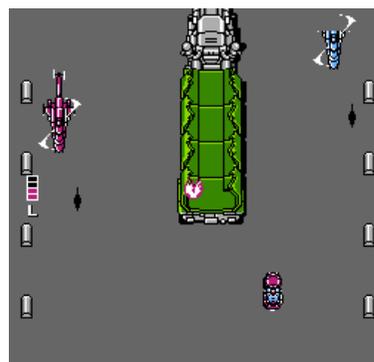
Но, несмотря на все недостатки, игра определенно удалась. Из большинства игр NES эта – одна из моих любимых. Настоятельно рекомендуется поклонникам Bubble Bobble.

Super Spy Hunter

NES – это настоящий аркадный рай, ведь очень многие игры перекечевали сюда с игровых автоматов. Наряду с платформером, один из самых важных аркадных жанров – скроллинг-шутер. Начиная с легендарного Space Invaders, этот жанр породил множество культовых серий и настоящих игрсимволов. Со временем менялись идеи, менялась графика и тематическая направленность. Но одно осталось неизменным: вы управляете каким-либо транспортным средством и уничтожаете врагов, которые надвигаются на вас сверху. Просто и ясно.



Super Spy Hunter – не исключение. Вы управляете шпионским автомобилем, который умеет все (мечта всех мальчишек 90-х!) Не буквально, конечно, но поверьте – оружия в игре хватает. Игра является продолжением аркадной Spy Hunter, которая, впрочем, кроме автомобильной тематики, имеет мало общего со своим сиквелом.



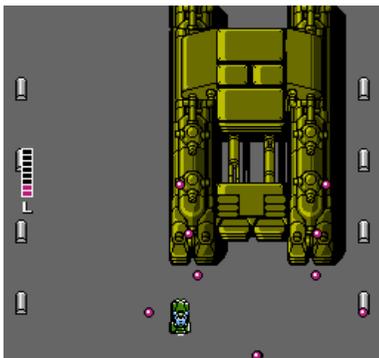
Геймплей основан на стандартной концепции скроллинг-шутера: вы едете вперед и уничтожаете все, что движется. Однако даже здесь нашлось место интересным фишам: например есть грузовики, которые при уничтожении дают приз, изображенный на крыше. Причем приз каждую секунду меняется. Среди призов – дополнительные жизни, оружие, вспомогательные средства, такие, как сброс масла на дорогу.

Вам придется прокатиться по множеству уровней-трасс, на которых вас встретят армии вражеских объектов – вертолеты, истребители, танки, гоночные машины со сверлами на бампере, лодки, монорельсы с пушками и даже космические корабли... А боссы – целые комплексы боевых машин на колесах, с каждым уровнем становящиеся все более и более громадными. Вас ожидает фейрверк взрывов, перестрелок и бешеной езды – в этой игре сосредоточено все лучшее от скроллинг-шутеров того времени.



Графика для NES безупречная: японцы действительно умеют выжать из приставки максимум и сделать игру графическим шедевром. С музыкой все тоже самое – мелодии и звуки как нельзя кстати подходят к атмосфере игры, используется один из самых мощных звуковых движков на NES. Единственный минус этой игры – исконно японская высокая сложность. Чтобы пройти хотя бы первый уровень, нужно очень много тренироваться.

Кстати, есть здесь и секретная мини-игра – классический «понг» под названием Hyper Tennis. Как ее открыть – догадайтесь сами, так будет интересней! =)



Track & Field

На олдскульных платформах очень мало симуляторов олимпийских видов спорта. Игры такого типа действительно сложно реализовать: ведь нужна высокая точность расчетов механического движения, чтобы происходящее на экране реалистично имитировало прыжки в длину или метание копья.

Данная игра является неплохой попыткой создать реалистичный спортивный симулятор в далеком 1987 году. Во всяком случае, геймплей вас не разочарует: вам понадобятся только глазомер и быстрота реакции.

До того, как мне в руки попался картридж Track & Field, я уже успел поиграть в разные олимпийские симуляторы, но во всех чего-то всегда нехватало – то управление слишком сложное, то еще что. Эта игра, напротив, порадовала практически идеальным балансом: все реализовано предельно просто, но играть, тем не менее, интересно.

В игру включены 8 самых популярных видов легкой атлетики:

100-meter dash (100-метровка). Ваша цель – как можно чаще нажимать кнопку A и обогнать вашего соперника. Самое главное – не стартовать раньше сигнала, иначе придется начинать сначала. Всего дается три попытки.

Long jump (прыжок в длину). Разбегайтесь той же кнопкой A – когда увидите линию, прыгайте кнопкой «влево». Секрет успеха – не держать кнопку слишком долго: чем быстрее вы ее отпустите, тем дальше прыгнете.

110-meter hurdles (бег с препятствиями) – похоже на 100-метровку, только с барьерами, которые перепрыгиваются нажатием кнопки «влево». Поначалу трудно привыкнуть к тому, когда нужно нажимать кнопку, но потом все становится легко.

Javelin Throw (метание копья) – здесь тоже нужно сначала бежать, затем, не заступая за линию, резко нажать кнопку «влево». Чем быстрее отпустите, тем дальше полетит копье.

Skeet Shooting (стрельба) – управление тут несколько другое. Суть такова: вылетают тарелки, у вас есть два указателя, которые отслеживают полет тарелки по вертикали и горизонтали. От вас требуется лишь вовремя нажать кнопку стрельбы, когда тарелка попадает прямо в прицел обоих указателей. Чем больше стреляете без промахов, тем больше становится ваш прицел и больше шансов попасть. Но если вы промахнетесь, то прицел становится снова маленьким и скорость тарелок снова снижается.

Triple Jump (тройной прыжок) – практически то же, что и прыжки в длину, но прыгать нужно три раза.



Archery (стрельба из лука) – этот уровень тоже отличается от других. Арена показана сверху: ваша цель – стрелять из лука по движущейся мишени. Точность стрельбы зависит от множества факторов, включая даже силу ветра.

High Jump (прыжки в высоту) – здесь тоже нужно сначала бежать, затем нажать и удерживать кнопку прыжка. В отличие от других уровней, где нужно резко нажимать кнопку, здесь нужно удерживать нажатие. Цель, как нетрудно догадаться – перепрыгнуть планку, не задев ее.

В Track & Field можно играть бесконечно, так как есть стимул – можно каждый раз пытаться побить свой предыдущий рекорд (впрочем, это не относится к прыжкам в высоту, где цель – просто не задеть планку). Эта одна из немногих игр на NES, которые не надоедают со временем.

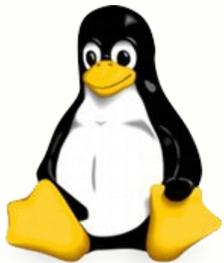
Конечно, графика здесь не самая лучшая – даже для NES – но для симулятора все выглядит вполне прилично. Спортсмены выглядят очень живо – они либо радуются победе, либо плачут из-за неудавшейся попытки. То же самое и со зрителями – они могут ликовать при виде хорошего выступления. Стоит упомянуть один небольшой секрет: если вы три раза выступите положительно, то может появиться какой-нибудь неожиданный сюрприз!



О звуке. Порадовало, что на главном экране играет олимпийская мелодия. В самой игре музыки как таковой нет – есть просто мелодии, звучащие время от времени.

Существует и вторая часть этой игры: в ней гораздо больше видов спорта – есть, например, тхэквондо, фехтование и еще много чего. А также большее разнообразие игровых режимов – чемпионат, тренировка и т. д.

Фархад Гафаров



Linux: ПОЛЕЗНЫЕ КОМАНДЫ

Мы продолжаем наш «марафон» полезных консольных команд для ОС Linux, начатый в предыдущих номерах.

Начнем с простого: пакетная конвертация всех изображений в каталоге из PNG в JPG (требуется установленный ImageMagick):

```
$ mogrify -format jpg *.png
```

ImageMagick также умеет создавать многостраничные PDF из набора отдельных изображений:

```
$ convert *.jpg mydoc.pdf
```

Найти и заменить все вхождения «Foo» на «Bar» в текстовом файле:

```
$ sed "s/Foo/Bar/g" list.txt > list2.txt
```

Продолжить загрузку недокачанного файла:

```
$ wget -c http://site.com/file
```

Вывести все активные Unix-сокеты:

```
$ netstat -lx
```

Зашифровать и расшифровать файл:

```
$ gpg -c filename
```

```
$ gpg filename.gpg
```

Запустить простой HTTP-сервер на 0.0.0.0:8000:

```
$ python -m SimpleHTTPServer 8000
```

Отправить сообщение по Jabber/XMPP:

```
$ purple-remote 'xmpp:goim?screenname=user@mail.com&message=text'
```

Отслеживать входящие сообщения XMPP (в одну строку):

```
$ dbus-monitor "type=signal,sender=im.pidgin.purple.PurpleService, interface=im.pidgin.purple.PurpleInterface,member=ReceivedImMsg"
```

Добавить все файлы в текущем каталоге в Git:

```
$ git add *
```

Воспроизвести любой файл как звук:

```
$ cat file.jpg | aplay
```

ВосПИ напоследок – скрипт для конвертации шрифта из формата OTF в TTF при помощи FontForge:

```
#!/usr/local/bin/fontforge  
  
Print("Opening "+$1);  
  
Open($1);  
  
Print("Saving "+$1:r+".ttf");  
  
Generate($1:r+".ttf");  
  
Quit(0);
```

#GamerGate

Истинное лицо информационной войны

«Чувство собственной важности - главнейший и самый могущественный из врагов человека. Его уязвляют и обижают действия либо посягательства со стороны ближних, и это делает его слабым. Чувство собственной важности заставляет человека всю его жизнь чувствовать себя кем-то или чем-то оскорбленным...»

Карлос Кастанеда, «Огонь изнутри»

Разработчики игр, игровые журналисты и геймеры в очередной раз оказались втянуты в громкий политический скандал. Но на этот раз речь идет не о «пропаганде насилия» и прочих расхожих клише, как можно было подумать - последние два месяца ознаменовались ожесточенными интернет-баталиями между игровым сообществом и... феминистками.

Началось все с того, что в августе этого года вскрылся секс-скандал с американской инди-девелопершей Зоей Квинн: выяснилось, что она вступила в половые отношения с несколькими игровыми журналистами (с пятью, если верить некоторым источникам) за положительные отзывы о ее игре - среди них даже был корреспондент Kotaku Натан Грейсон.

Естественно, подобное кумовство не могло остаться незамеченным интернет-активистами: на YouTube незамедлительно появилась серия расследований коррупции в игровой прессе, а сама Квинн с ее проектами стала подвергаться всяческой критике. Это движение получило название GamerGate - по названию хэштега, которым активисты помечали свои сообщения в Twitter.

Естественно, к этой информационной войне подключились феминистки всех мастей: мол, права женщин в геймдеве ущемляют, кругом неравноправие, геймеры не соблюдают идеалы демократии. Канадская феминистка Анита Саркисян, которая выступила в защиту Зои Квинн, обвинила геймеров в сексизме и опубликовала документальный ролик о нелегкой доле женских персонажей в видеоиграх. Буквально на следующий день Анита была вынуждена покинуть свой дом в связи с угрозами убийства!

Тему немедленно подхватили в прессе: многие игровые сайты (включая Gamasutra.com, Kotaku, Polygon и др.) практически одновременно выпустили статьи, в которых реакционеров обвинили в женоненавистничестве, называли «террористами» и «насилъниками».

1 сентября журналисты и инди-разработчики, участвующие в этом антигеймерском движении, опубликовали открытое письмо к игровому сообществу с требованием остановить деятельность GamerGate и прекратить нападки на индустрию - видимо, все эти «срывы покровов» серьезно ударили по их бизнесу.

Участники GamerGate, в свою очередь, обвинили Gamasutra и другие медиа в необъективности и неуважении к геймерам - по их мнению, издания используют тему сексизма лишь как маневр, отвлекающий от проблем с продажностью игровых рейтингов.

Активисты написали рекламодателям, что больше не будут покупать продукцию фирм, рекламирующихся на сайтах, где их в открытую унижают.

Но как все эти дразги могут касаться разумной части человечества, не участвующей в политических разборках? Не торопитесь с выводами: политика сегодня проникла туда, где ее, казалось бы, не может быть в принципе.

После получения жалоб от GamerGate, сама Intel приняла решение прекратить размещение рекламы на сайте Gamasutra. Ответный бойкот последовал от Мэтью Гаррета, известного разработчик ядра

Linux (в свое время он получил от Фонда СПО премию за достижения в области обеспечения загрузки Linux на системах с UEFI) - Гаррет в знак протеста отказался от работы над исправлением ошибок в ядре, проявляющихся на системах Intel.

Трудно сказать, чем это обернется для пользователей Linux - кто-то уже считает, что x86 уже отжила свое и для линуксоидов настало время порвать историческую связь с Intel и переключиться, наконец, на ARM.



Но главный вопрос даже не в этом. Возмущает, в первую очередь, то, что на развитие свободные проекты влияют (и очень сильно влияют!) личные политические убеждения вовлеченных в эти проекты людей. Почему сообщество СПО должны волновать вопросы гомофобии, женоненавистничества и борьба за права меньшинств? Почему в это должны быть вовлечены программисты и пользователи программ?

Вот вам еще один яркий пример. Как показали события еще одного скандала, в Mozilla, оказывается, весьма сильное ЛГБТ-лобби. Под давлением представителей сексуальных меньшинств с поста руководителя организации ушел Брендан Айк - создатель JavaScript, один из разработчиков Netscape и отцов-основателей Mozilla. Геи оказались недовольны тем, что Айк является приверженцем традиционных взглядов на семейные отношения - так, он пожертвовал тысячу долларов на продвижение поправки в конституцию Калифорнии, определяющей брак только как союз между мужчиной и женщиной.



Несмотря на неоднократные заявления о том, что политика никак не связана с миссией Mozilla, и человек имеет право на выражение своей личной позиции, нападки со стороны ЛГБТ не прекратились. Травля дошла до того, что некоторые подконтрольные ЛГБТ сайты (в том числе один из крупнейших сайтов знакомств OKCupid), стали блокировать доступ пользователей Firefox, предлагая сменить браузер!

Что же касается феминисток, то не так давно произошел еще один неоднозначный инцидент, касающийся их движения. После жалоб со стороны многочисленных «оскорбленных», руководство GitHub удалило с сервиса репозиторий шуточного «языка программирования феминисток» C++, созданного группой каких-то троллей. Проект после этого переехал на Gitorious, где и доступен по сей день: <https://gitorious.org/c-plus-equality>.



Надо сказать, что исходники в репозитории содержат не прямые оскорбления в чей-либо адрес, а просто остроумные пародии на убеждения феминисток (например, «#define womain main»). Исходники распространяются по шуточной свободной лицензии FemSFPL - модификации MIT, которая дает право всем женщинам и женоподобным транссексуалам использовать, модифицировать и распространять код при условии сохранения оригинального авторства и оригинальной лицензии.

Смех смехом, однако тот факт, что чей-то юмор может стать поводом к цензуре в пользу лоббистов, особой радости не вызывает.

Есть и такая сторона вопроса: постепенно культивирующееся в нашем обществе чувство оскорбленности, нагнетание комплекса обиды: «Скажи мне, что тебя оскорбляет, и я скажу тебе, кто ты». Закон и государство, как правило, всегда на стороне этих «униженных и оскорбленных» - чуть ли не каждый день появляются новые репрессивные законы для защиты чьих-то чувств.

Человек со всех сторон окружен противоборствующими сторонами, вынужден ежеминутно делать мучительный выбор, постоянно вставать на чью-то сторону - как же, в противном случае его обвинят в отсутствии гражданской позиции! Интернет в этом отношении только добавляет масла в огонь: вы теперь сильны не собственным мнением, а приверженностью «генеральной линии» всевозможных политиков, партий и церквей - свобода слова заменена свободой лайка и репоста. Но где же в этом своеобразном «голосовании» пункт «против всех»?..

В нашу информационную эпоху, когда создание человека задвинуто обрушивающимся на него плюрализмом идей и мнений, главная ценность - это сохранить индивидуальность. Именно индивидуальность, а не идентичность по половому, расовому, национальному или религиозному признаку.

Патриотизм и чувство солидарности с кем-либо вас не спасет - все это уже в прошлом. Время «священных войн», знамен и лозунгов, время всеобщей ненависти и оскорбленности неизбежно должно пройти, и этот момент уже близок...



Это все!

Надеемся, номер вышел интересным. Если Вам нравится наш журнал, и Вы хотели бы его поддержать – участвуйте в его создании! Отправляйте статьи, обзоры, интервью и прочее на любые темы, касающиеся игр, графики, звука, программирования и т.д. на gecko0307@gmail.com.



<http://fps-magazine.blogspot.ru>