

30

2014

BLENDER:
Эффект снега

Сергей Шарыбин:
интервью

Как создать
клон 2048

The Tale
Сказка сказок

Игровой арт-хаус

+ многое другое!

FPS

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





FPS

№30

FPS – бесплатный, свободно распространяемый электронный журнал, посвященный разработке игр и другим видам цифрового творчества.

FPS охватывает широкий круг тем: на страницах журнала рассматриваются вопросы программирования игр с использованием разнообразных движков и графических библиотек, публикуются материалы по двумерной и трехмерной компьютерной графике, включая уроки по популярным графическим пакетам и редакторам, а также различные статьи по теоретическим вопросам, дизайну и философии компьютерных игр.

Журнал издается с января 2008 г. и на данный момент выходит раз в два 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**

Наш сайт: <http://fps-magazine.cf>

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

● Blender

- :: Новости
- :: Интервью с Сергеем Шарыбиным
- :: Dynamic Paint: эффект снега
- :: Обзор дополнений. Выпуск 9

● 2D-графика

- :: GIMP: виньетирование на Python

● Кодинг

- :: Язык D. Новости «с Марса»
- :: Making-of: клон 2048
- :: Как работают великие программисты

● Linux-гейминг

- :: Игровые новости из мира Linux

● The Tale

- :: Сказка сказок

● Культовые игры

- :: Игровой арт-хаус

● RetroGame

- :: Пиратские порты на NES

● Браузеры и Unix-way

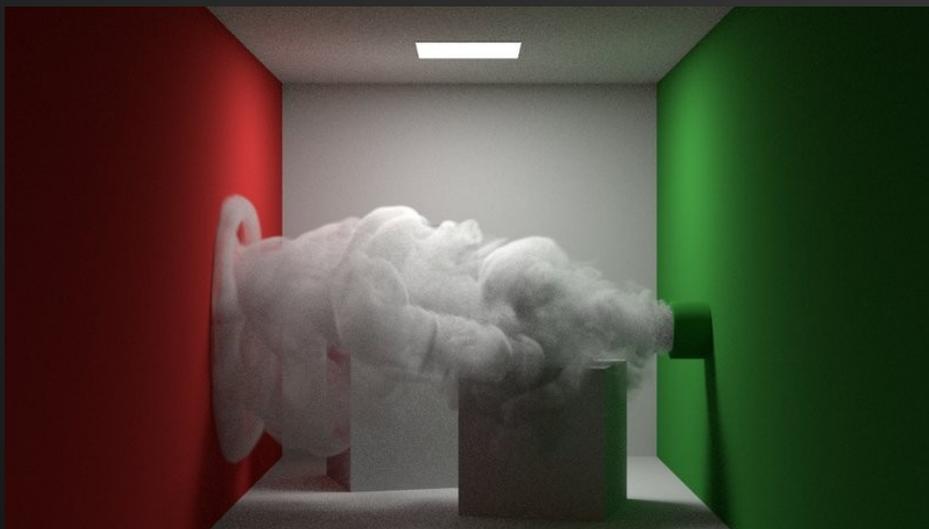
- :: Панацея или эпидемия?



Blender

Новости

Разработчики выложили в свободный доступ тестовую сборку **Blender 2.71**. Ключевые особенности этой версии включают несколько значительных нововведений в рендер-движке Cycles: так, добавлена начальная поддержка рендеринга дыма и огня, поддержка размытия при движении (Motion Blur) для деформирующихся анимированных объектов, запекание рендеров в текстуры (Baking) с поддержкой Ambient Occlusion и много более мелких улучшений. Кроме того, улучшены инструменты моделирования, лепки, анимации, рисования, игровой движок, рендер-движок Freestyle.



Скачать Blender 2.71 Testbuild для Windows, Linux и Mac OS X можно [здесь](#).



В этом году Blender традиционно участвует в **Google Summer of Code**. Недавно были озвучены новые проекты GSoC – среди них интерактивный ремешинг, интеграция движка симуляции жидкостей MantaFlow, работа над улучшением рендер-движка Cycles, рефакторинг инструментов NURBS и многое другое. Напомним, GSoC – это инициативная программа Google, в рамках которой ежегодно проводится отбор открытых проектов, в которых могут принять участие студенты.

Кстати, о новых разработках. Качество работы с высокополигональными моделями в Blender поднимется на уровень выше: недавно разработчики объявили о интеграции в программу небезызвестной технологии **OpenSubdiv** с поддержкой ускорения на GPU. Пока эта инициатива носит экспериментальный характер, и до внесения кода в основную ветку еще далеко, но перспектива видна невооруженным глазом. Напомним, OpenSubdiv – это набор библиотек от Pixar с реализацией subdivision для высокопроизводительного построения гладких поверхностей и кривых. Исходный код проекта был открыт в 2012 году.

В FPS №26 '13 мы уже писали о том, что разработчики коммерческого рендер-движка **OctaneRender** выпустили бета-версию своего плагина для Blender.

Теперь всем пользователям пакета доступна бесплатная демо-версия плагина – она не имеет временных или функциональных ограничений, но имеет предельное разрешение рендеринга 1000x600 и накладывает на полученные изображения водяной знак.

Кстати, есть специальное предложение для учащихся: по академической лицензии можно получить полную версию OctaneRender 1.2 и плагины для Blender, Maya и 3dsMax совершенно бесплатно!

Все подробности – на официальном сайте OctaneRender:

<http://render.otoy.com>.



Напомним, OctaneRender – это профессиональный физически корректный рендер с поддержкой ускорения на GPU и пользовательским интерфейсом на основе узлов. Над этим проектом работал Брехт ван Ломмель, автор Cycles.

Теперь о грустном. В рамках краудфандинга проекту **Gooseberry**, к сожалению, не удалось набрать сумму, необходимую для съемок полнометражного фильма. Однако Blender Institute не отказывается от намеченной цели и, в качестве первого шага, планирует съемку качественной короткометражной пилотной версии картины длительностью 10-15 минут. Работа над проектом начнется в сентябре и продлится до весны 2015 года. Также будет продолжена работа над развитием Blender Cloud и инструментов управления ресурсами, задуманная в рамках Gooseberry.



Еще один перспективный короткометражный фильм с использованием Blender снимается у нас в России: это проект **«Агата»** от казанской студии Propellers под руководством Артура Шамшадинова. Проект почти полностью краудсорсинговый: над ним работают несколько десятков волонтеров, которые рисуют концепт-арт, моделируют, готовят раскадровки и пишут музыку.

В основе сюжета мультфильма – фантастический рассказ Джека Финнея «О пропавших без вести». Окончательная длительность мультфильма пока неизвестна, но заведомо составит не более 12 минут. В процессе создания фильма используется не только СПО: участники работают в Blender, GIMP, MyPaint, Adobe Photoshop, Corel Paint и т. д.

Обсуждение проекта ведется в паблике ВКонтакте: <http://vk.com/public365mult>. Почти все решения принимаются голосованием, в котором участвуют все желающие.



ROAD TO PROVIDENCE

Отличная новость для всех поклонников культового писателя Говарда Лавкрафта: по мотивам «лавкрафтовских ужасов» командой разработчиков из Санкт-Петербурга создается инди-игра **Road to Providence** («Дорога в Провиденс»). Жанр игры – квест; естественно, в стиле хоррор. В процессе разработки используется Blender, Krita, 3D Coat. Код игры пишется на Java с использованием jMonkeyEngine в качестве движка. Поддержать авторов можно **на IndieGoGo**.



В апреле вышел 44 номер журнала **«BlenderArt»** – самого популярного англоязычного электронного издания, посвященного Blender. В этом выпуске читателей ждет много полезной информации: уроки, статьи жара «making-of», галерея работ и другие материалы.

Журнал «FPS» отслеживает все самые свежие новости из мира Blender, моделирования, анимации и рендеринга! В следующем номере ждите очередную подборку новостей. Оставайтесь с нами и держите руку на пульсе последних событий!

Вы разрабатываете перспективный проект? Открыли интересный сайт? Хотите «раскрутить» свою команду или студию? Мы Вам поможем!

Спецпредложение!

«FPS» предлагает уникальную возможность: совершенно **БЕСПЛАТНО** разместить на страницах журнала рекламу Вашего проекта!! При этом от Вас требуется минимум:

- **Соответствие рекламируемого общей тематике журнала.** Это может быть игра, программное обеспечение для разработчиков, какой-либо движок и/или SDK, а также любой другой ресурс в рамках игростроя (включая сайты по программированию, графике, звуку и т.д.). Заявки, не отвечающие этому требованию, рассматриваться не будут.

- **Готовый баннер или рекламный лист.** Для баннеров приемлемое разрешение: 800x200 (формат JPG, сжатие 100%). Для рекламных листов: 1000x700 (формат JPG, сжатие 90%). Содержание — произвольное, но не выходящее за рамки общепринятого и соответствующее грамматическим нормам. Совет: к созданию рекламного листа рекомендуем отнестись ответственно. Если не можете сами качественно оформить рекламу, найдите подходящего художника.«Голый» текст без графики и оформления не принимается.

- Краткое описание Вашего проекта и — обязательно — **ссылка на соответствующий сайт** (рекламу без ссылки не публикуем).

- Заявки со включенными **дополнительными материалами для журнала** (статьи, обзоры и т.д.) не только приветствуются, но даже более приоритетны.

Заявки на рекламу принимаются на почтовый ящик редакции: gecko0307@gmail.com (просьба в качестве темы указывать «Сотрудничество с FPS», а не просто «Реклама», так как письмо может отсеять спам-фильтр).

Прикрепленные материалы (рекламный лист, информация и пр.) могут быть как прикреплены к письму, так и загружены на какой-либо надежный сервер (убедительная просьба не использовать RapidShare, DepositFiles, Letitbit и другие подобные файлообменники — загружайте файлы на свой сайт, блог или ftp-сервер и присылайте статические ссылки). Все материалы желательно архивировать в формате zip, rar, 7z, tar.gz, tar.bz2 или tar.lzma.

У нас в гостях: Сергей Шарыбин



Сергей Шарыбин – один из ключевых разработчиков Blender, автор системы отслеживания движений. Эта технология широко применяется в профессиональном VFX – например, когда необходимо синхронизировать трехмерные объекты в соответствии с положением реальной камеры.

Трекер камеры в Blender был использован при создании открытого фильма «Tears of Steel» от Blender Institute, и Сергей – наш соотечественник, программист из Перми – принимал непосредственное участие в этом проекте. Мы решили взять у него интервью и узнать подробнее о его деятельности.

Здравствуйте, Сергей! Расскажите немного о себе. Программирование – это ваша основная профессия или просто хобби?

Программирование – это у меня давно. Еще со школы, когда начал выступать на олимпиадах по информатике. Во время учебы в университете Blender был просто хобби – либо просто отвлечься, либо иконку какую отрендерить для проекта. Ну а потом программирование для Blender стало основной работой, и на использование его как обычному пользователю уже времени почти нет.

Когда и как вы стали пользователем Blender?

Начинал пытаться освоить после выхода «Elephant's Dream». Точную версию не назову – что-то, что под Debian тогда легко ставилось... А так, чтобы уже результаты рендеринга куда-то использовать можно было – это с версии 2.47, я думаю. Год точный не назову – тем более, что использовал Blender из репозитория Debian, а он там не всегда самый свежий.

Ну и в районе 2009-2010 годов начал его активно агитировать в университете. Сейчас мой товарищ читает курс 3D-графики и использует только Blender.

Широкой публике вы известны, в первую очередь, как автор системы отслеживания движений. Расскажите «на пальцах», как она работает. Какие технологии при этом используются?

Технология называется «компьютерное зрение» (Computer Vision). С точки зрения математики, сюда вовлечены как евклидова так и эллиптическая геометрия.

Если на пальцах, то при 2D-трекинге система ищет такое положение паттерна маркера на следующем кадре, которое лучше всего коррелирует с паттерном на текущем. Для этого очень активно используется библиотека оптимизации задач о наименьших квадратах Ceres. Грубо говоря, эта библиотека деформирует паттерн на следующем кадре таким образом, что результирующая корреляция является максимальной. И делает это быстро :)

Такой подход позволяет с большой точностью производить трекинг объектов, которые движутся относительно медленно. При быстрых движениях, когда включена опция пре-инициализации (Prepass) перед работой Ceres система перебирает всевозможные позиции паттерна в области поиска и ищет такую позицию, которая наиболее похожа на паттерн на текущем кадре. Это даст позицию трека на следующем кадре с точностью в несколько пикселей (обычно 1 пиксель, но может быть и больше).

После такой инициализации методом «грубой силы», Ceres вступает в работу и оптимизирует позицию, что дает субпиксельную точность.

Ну, с 2D-трекингом кое-как разобрались, надеюсь. Теперь 3D-реконструирование. Основано оно на эффекте параллакса и работает примерно так: используя два ключевых кадра и немного математики, связанной как раз с эффектом параллакса, система определяет позиции треков с этих кадров в 3D-пространстве, что дает представление о глубине сцены.

После этого, используя это начальное реконструирование и треки со всех других кадров, система достраивает решение на все кадры.

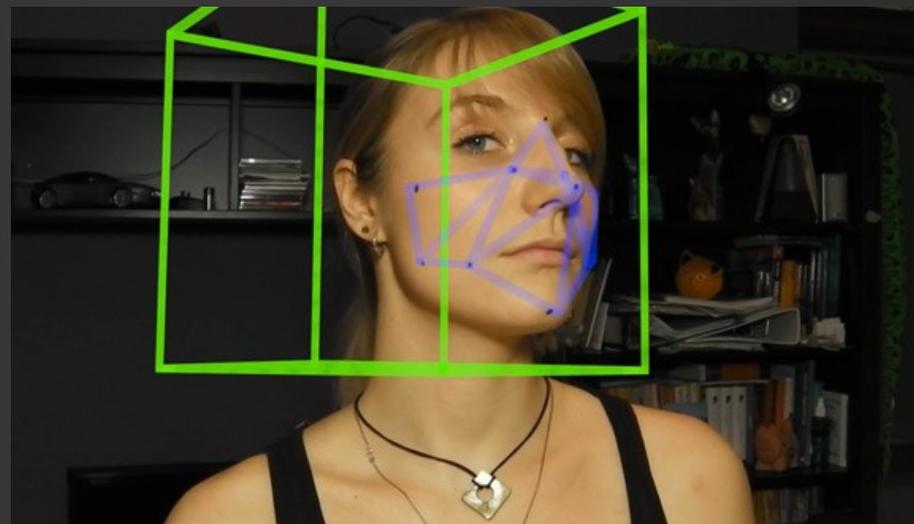
В связи с тем, что треки могли быть оттречены не очень точно, решение сцены тоже может быть не очень точным. Для минимизации ошибки, используется этап так называемого Bundle Adjustment, который, грубо говоря, двигает камеру на каждом кадре и каждый трек в 3D-пространстве с тем, чтобы минимизировать ошибку ре-проецирования. Как-то так.

Можно ли считать трекинг в Blender полным аналогом коммерческим решениям? Например, MatchMover от Autodesk?

Не знаю, не использовал коммерческие решения. Но как-то во время проекта Mango (фильм «Tears of Steel») было несколько моментов, когда Blender успешно построил решение в ситуациях, когда я этого не ожидал.

Например, в сценах с капитаном его наглазная повязка была оттречена и даже не потребовалась ручная корректировка при добавлении эффектов на нее.

Могу лишь предполагать, что в Blender не хватает лишь большой зеленой кнопки «Autotrack».



Вы были на Blender Conference, лично общались с Тоном Розендалем и другими членами команды Blender. Также, насколько нам известно, вам официально предложили парт-тайм работу в Blender Foundation. Как с этим сейчас обстоят дела? Какова атмосфера внутри организации? Легко ли было найти общий язык с европейскими коллегами?

С Тоном и другими участниками проекта я общался через IRC задолго до того, как в первый раз съездил на конференцию в 2010 году. Общий язык найти было достаточно легко, несмотря даже на то, что для большинства из нас английский не является основным языком.

Насчет атмосферы – это хороший вопрос. Мы в основном работаем по удаленке – общение лишь через IRC. Даже во время съемок открытых фильмов не так часто бывает, когда вся команда разработчиков собирается вместе в Blender Institute. Но когда даже часть разработчиков собирается вместе с художниками и работает над фильмом – атмосфера очень хорошая. Не обходится без стрессовых ситуаций, но это скорее предсказуемо.

Всем известно о вашем вкладе в создание «Tears of Steel». Какие у вас остались впечатления от работы над этим проектом?

Aweseome. In space. Как-то так :) Было очень здорово работать в студии вместе с художниками, видеть как они используют Blender, оптимизировать его под конкретный рабочий процесс. Ну и забавнее всего было видеть прогресс художников на недельных сборах в пятницу вечером, где они показывали, что сделали за неделю.



Сейчас все говорят о Gooseberry. Планируется чисто анимационный фильм без задействования реальных съемок, и трекинг, очевидно, использоваться не будет – но все-таки интересно узнать подробности о проекте. Не поделитесь ли какой-нибудь информацией «из-за кулис»?

Я сам знаю не многим больше того, что написано в **блоге**, так что ничего такого закулисного поведать не могу. Следите за нами в блоге :)

Каковы ваши ближайшие планы относительно разработки для Blender? Планируете ли что-то новое?

Вкратце – да. Если чуть подробнее, то трекинг камеры – это не единственное, чем я занимаюсь. Есть несколько областей, которые нужно улучшить: например, сделать композитор более отзывчивым.

Также я работал над новым графом зависимостей, но сейчас этот проект полностью подхвачен Лукасом (Lukas Toenne). Но скорее всего, я вскоре присоединюсь к нему в этом проекте.



Еще в данный момент я трачу некоторое время на интеграцию OpenSubdiv в Blender. Это библиотека подразделения Кэтмулла-Кларка от Pixar, которая позволяет производить подразделение мешей на GPU. Дает очень значительный прирост скорости и интерактивности вьюпорта.

Мы сделали небольшое видео с демонстрацией текущего состояния проекта:

https://www.youtube.com/watch?v=dzIL_S-qHIQ

Насколько, по вашим впечатлениям, Blender востребован в России?

Если честно – не знаю. Удивительно, но здесь не так много студий, в которых используют Blender. Может, просто скрываются. Хотя последнее время началась какая-то движуха, и в Сети появляется все больше работ от российских студий, сделанных в Blender. Что, естественно, радует.

Как считаете, есть ли будущее у СПО в сфере профессионального коммерческого кинематографа?

Я несколько скептически отношусь к мысли, что Blender сможет вытеснить какой-то продукт из рабочего процесса в крупных студиях. Есть мнение, что, если студия достаточно крупная, и в ней есть явно выраженные подразделения трекинга камеры, моделлинга и т.д., то им проще использовать специализированный софт.

Но это не значит, что Blender не используется в крупных студиях. Насколько я помню, Rhythm & Hues Studios использовали Blender для симулирования океана в фильме «Жизнь Пи». В новом «Капитане Америка» Blender использовался для препросмотра сцены, когда летательный аппарат врезается в мост. И это лишь несколько примеров, которые сразу вспомнились. Фактов использования Blender в студиях достаточно много, стоит только немного погуглить :)

Так что да, будущее есть – весь вопрос в нише этого софта в рабочем процессе.

Огромное спасибо за интервью! Желаем вам успехов в вашей работе!

*Интервью взял:
Тимур ГАФАРОВ*

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.



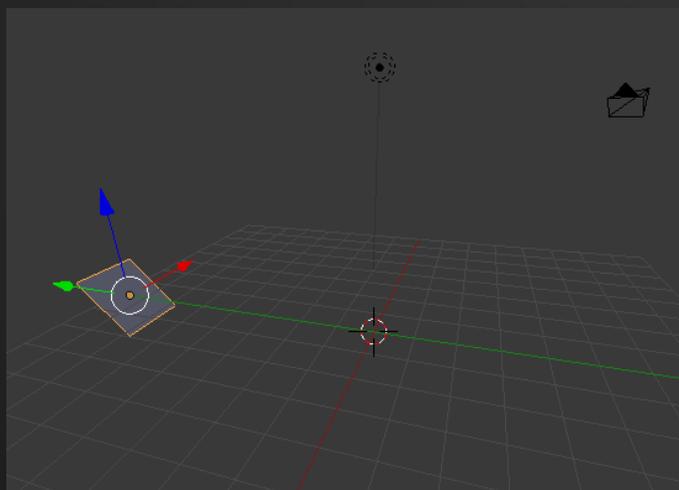


Dynamic Paint: Эффект снега

Система **Dynamic Paint** (динамическое закрашивание) впервые появилась в Blender 2.61. Это модификатор, позволяющий превратить трехмерные объекты в своеобразные «холсты» и «кисти»: при их соприкосновении во время анимации, на «холсте» остаются следы, которые влияют на материал или геометрию.

Эта система значительно облегчает динамическое создание сложных эффектов – таких, например, как появление дождевых луж, грязи, пыли и т.д.

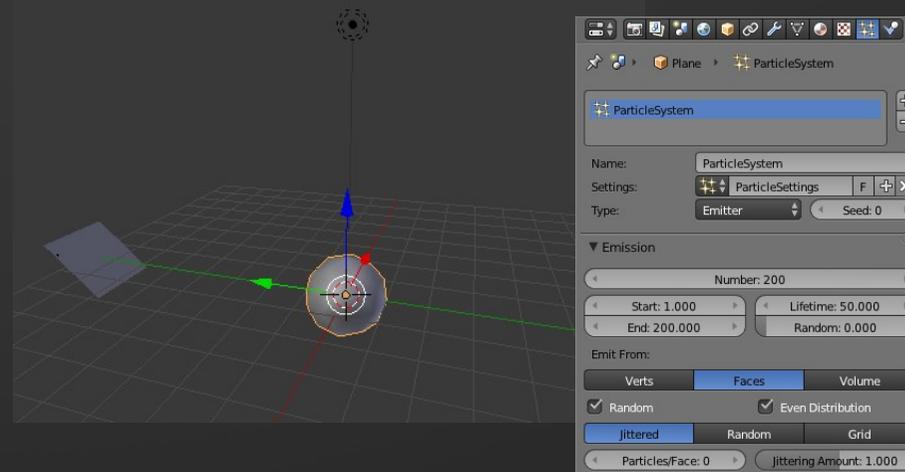
Предлагаю вашему вниманию технику использования Dynamic Paint для имитации следов на снегу.



1. Эффект будет работать при помощи системы частиц, которая будет «стрелять снежками» – но его можно адаптировать для любого типа взаимодействий. Создайте объект-эмиттер – у меня это обычная плоскость.

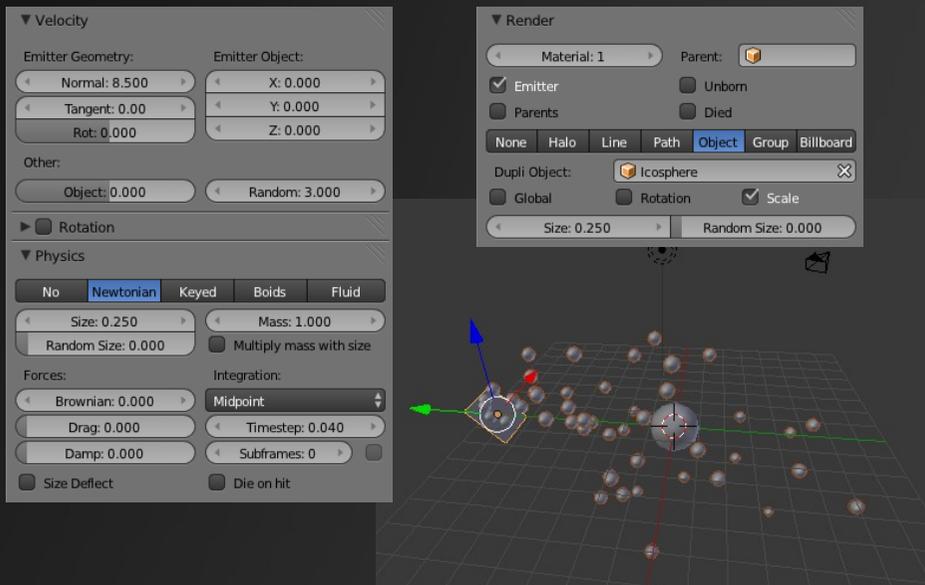
Поверните ее на 45 градусов, чтобы частицы летели не отвесно вверх, а под наклоном.

2. Добавьте мастер-объект, который будет использован для создания клонов в системе частиц. В данном случае это может быть просто икосфера.

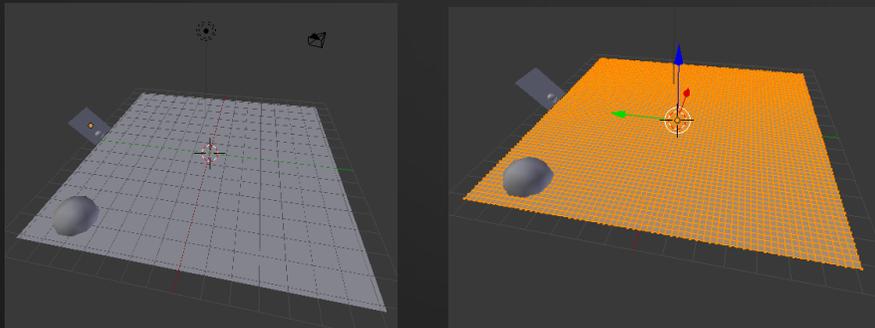


3. Снова выберите плоскость и добавьте ей систему частиц. Выставьте настройки как на скриншотах: количество частиц – 200, скорость в направлении нормали поверхности – 8.5, случайный фактор скорости – 2.0, размер физического объекта – 0.25. На вкладке **Render** укажите режим **Object** и выберите мастер-объект. Размер копий укажите равным 0.25.

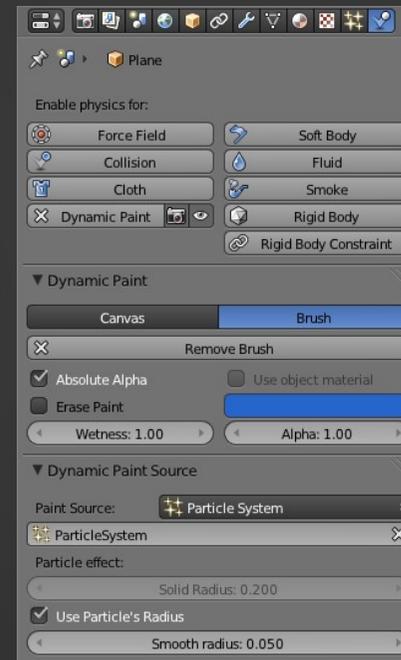
Если теперь запустить воспроизведение анимации, вы увидите, как из плоскости-эмиттера будут вылетать «снежки».



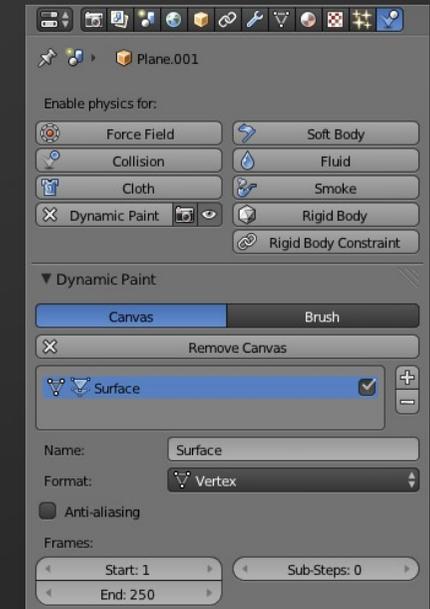
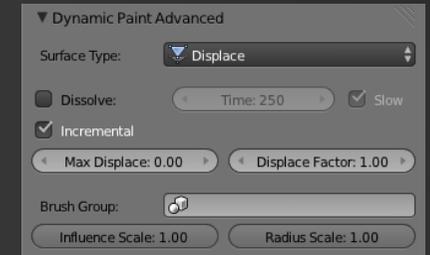
4. Займемся непосредственно эффектом Dynamic Paint. Создайте большую плоскость-пол и подразбейте ее несколько раз, чтобы она состояла из множества мелких ячеек.



В настройках **Physics** для этой плоскости включите свойство **Dynamic Paint**. На появившейся вкладке **Dynamic Paint** выберите **Canvas** (Холст). Добавьте новый холст (**Add Canvas**). На вкладке **Dynamic Paint Advanced** в качестве **Surface Type** выберите **Displace** и поставьте галочку напротив **Incremental**.

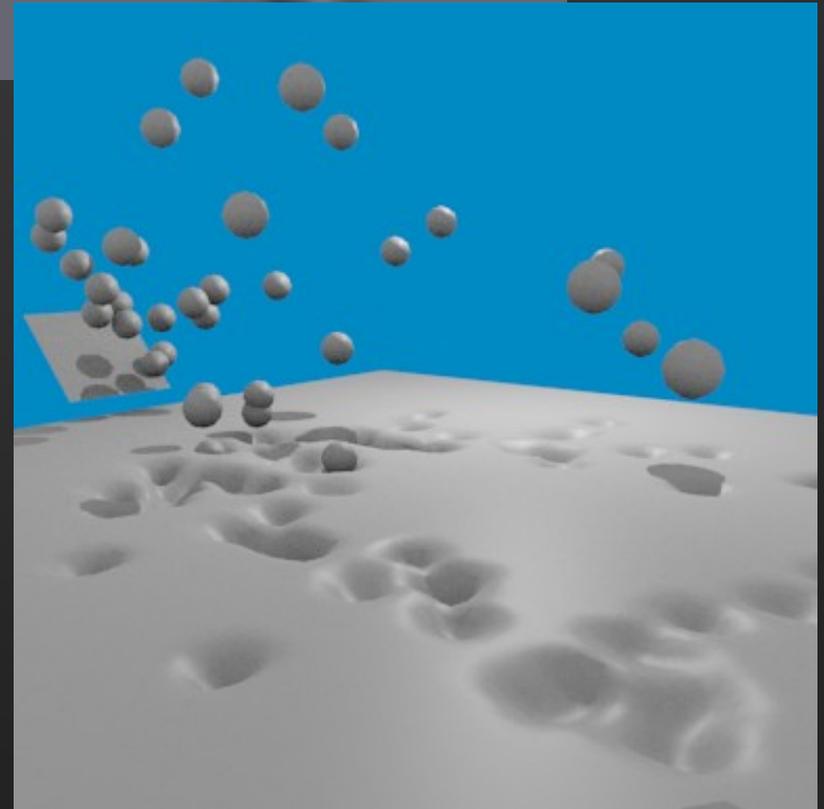
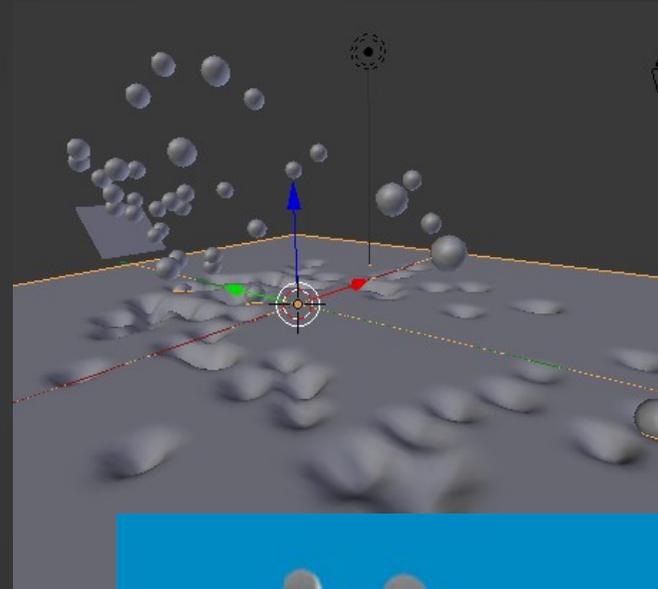


5. Теперь включите Dynamic Paint для плоскости-эмиттера. На вкладке **Dynamic Paint** выберите **Brush** (кисть). Создайте новую кисть (**Add Brush**).

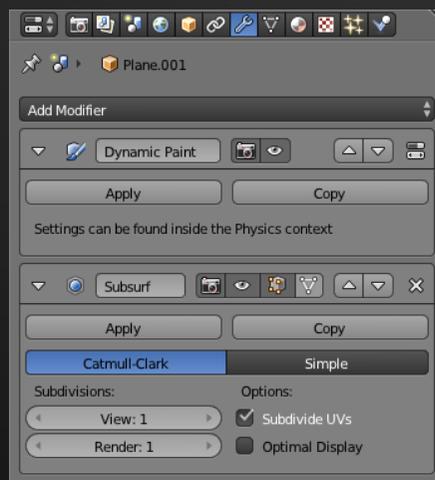


На вкладке **Dynamic Paint Source** в качестве **Paint Source** выберите **Particle System** и укажите нашу систему частиц. Поставьте галочку напротив **Use Particle's Radius**.

Запустите анимацию, и, если все было сделано правильно, «снежки» будут пробивать в плоскости отверстия.



6. Чтобы сгладить полученную геометрию, стоит добавить плоскости-полу модификатор **Subdivision Surface**.





Обзор дополнений Blender

Выпуск 9

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.

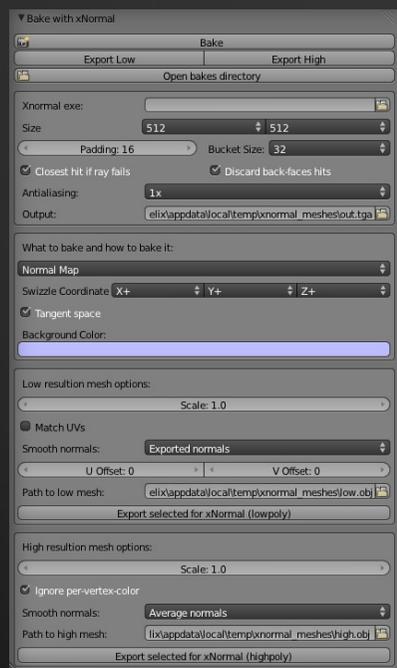
Если вы разрабатываете собственное дополнение или просто нашли в Интернете чей-то интересный проект, будем очень рады, если вы напишете нам об этом и поделитесь ссылкой. Пишите на gecko0307@gmail.com, либо в наше сообщество:

http://gplus.to/fps_community

Blender xNormal

xNormal – это популярная программа для генерирования карт нормалей, Ambient Occlusion и других видов текстур на 3D-моделях. Данное дополнение позволяет использовать xNormal напрямую из интерфейса Blender, не тратя времени на переключение между двумя программами. По заверению автора, интеграция покрывает около 95% функциональности xNormal.

Автор: Феликс Шлиттер
[Ссылка](#)

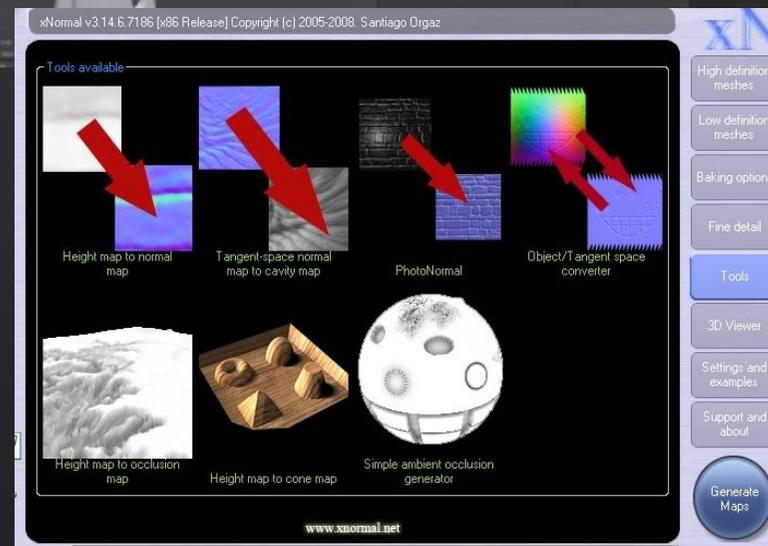


BoolTool

Дополнение, обеспечивающее упорядоченную организацию работы с булевыми операциями и моделирования твердых поверхностей, добавляя новые элементы интерфейса.

Автор: Витор Бальбио

[Ссылка](#)

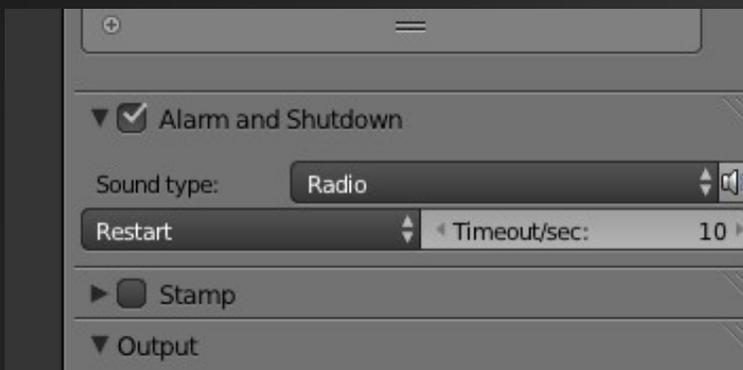


Alarm and Shutdown

Небольшой аддон, добавляющий звуковой сигнал при завершении длительного рендеринга. Также он позволяет автоматически выключить компьютер при завершении задачи, что полезно, когда рендеринг происходит в отсутствие пользователя.

Автор: Сергей Метельский

[Ссылка](#)



Music Player

Аддон, добавляющий в Blender функциональность музыкального плеера. Есть плейлист, поддержка выбора позиции в треке и возможность воспроизведения звуковых дорожек из видеофайлов.

Автор: Никита Городецкий

[Ссылка](#)



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.Денег](#), любой суммой на ваше усмотрение. Для нас важен любой, даже маленький вклад!

Наш WMR-кошелек: **R120156543694**

Номер кошелька Яндекс.Денег: **410012052560079**

Адреса Bitcoin:

16PSGbj5foeqMN8isdoyiKvWYGM9V5idFk
16XaSt1U5eXWG7EAkuEMpFE6M6fPia5o4F
1PdNHTL5nJsZJGHyNJ4c5xPW3eApEoB9pQ

Адрес PayPal:

gecko0307@gmail.com

Заранее благодарны!
Редакция



Секреты мастерства Python-Fu

Виньетирование

Виньетирование – это артефакт фотографии, при котором изображение затемняется по краям кадра. Этот эффект появляется, если установить на камеру объектив, предназначенный для аппаратов с меньшим размером кадра: возникает разница в углах поля зрения, и края фотографии оказываются затемнены «рамкой» кадра объектива.

Часто виньетирование используется в художественных целях: оно концентрирует внимание зрителя на объекте, который находится в центре кадра.

Искусственно добавить эффект виньетирования в GIMP очень легко: достаточно просто нарисовать круговой градиент от центра к одному из углов. Поскольку операция простая и часто используемая, можно оформить ее в виде скрипта.

Импортируем GimpFu:

```
from gimpfu import *
```

Объявляем функцию, которая будет совершать виньетирование. На вход она принимает цвет и прозрачность:

```
def python_fu_vignette(image, layer, color, alpha):  
    pdb.gimp_image_undo_group_start(image)  
  
    # Весь последующий код – здесь  
  
    pdb.gimp_image_undo_group_end(image)
```

Запоминаем позицию текущего слоя в топке слоев и создаем новый:

```
pos = pdb.gimp_image_get_layer_position(  
    image, layer)  
vignette = pdb.gimp_layer_new(image,  
    image.width, image.height,  
    RGBA_IMAGE,  
    "vignette",  
    100,  
    NORMAL_MODE)
```

Добавляем созданный слой над текущим и делаем его активным:

```
pdb.gimp_image_add_layer(image, vignette, pos - 1)  
pdb.gimp_image_set_active_layer(image, vignette)
```

Создаем градиент:

```
gradient = "Vignette"  
pdb.gimp_gradient_new(gradient)  
pdb.gimp_gradient_segment_set_left_color(  
    gradient, 0, color, 100)  
pdb.gimp_gradient_segment_set_right_color(  
    gradient, 0, color, 0)  
pdb.gimp_context_set_gradient(gradient)
```

Вычисляем точки начала и конца рисования градиента (x1, y1 и x2, y2):

```
x1 = image.width * 0.5
y1 = image.height * 0.5
x2 = 0
y2 = image.height
```

Рисуем градиент на созданном слое:

```
pdb.gimp_edit_blend(vignette,
    CUSTOM_MODE, NORMAL_MODE,
    GRADIENT_RADIAL,
    alpha,
    60,
    REPEAT_NONE,
    TRUE,
    FALSE,
    0, 0, TRUE,
    x1, y1, x2, y2)
```

Удаляем градиент из памяти:

```
pdb.gimp_gradient_delete(gradient)
```

Объединяем слой с предыдущим, чтобы окончательно применить виньетирование:

```
pdb.gimp_image_merge_down(image, vignette, 0)
```

Осталось зарегистрировать функцию и вызвать main():

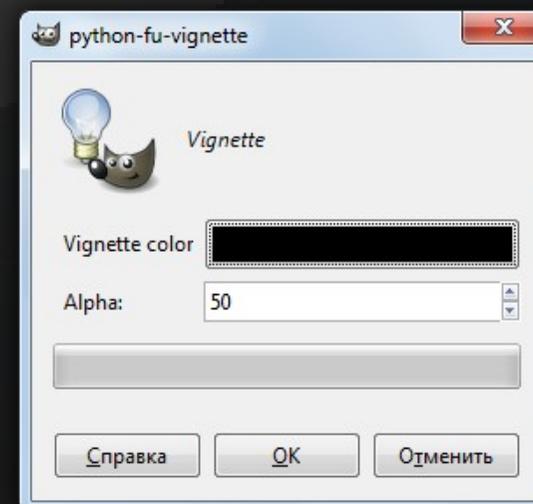
```
register(
    "python-fu-vignette",
    "Vignette",
    "Vignette 0.1",
    "Timur Gafarov",
```

```
"(c) Copyright 2014 Timur Gafarov",
"03-05-2014",
"Vignette...",
"RGB*",
[
    (PF_IMAGE, "image", "Target image", None),
    (PF_DRAWABLE, "drawable", "Target layer", None),
    (PF_COLOR, "color", "Vignette color", (0, 0, 0)),
    (PF_SPINNER, "alpha", "Alpha:", 50, (0, 100, 1)),
],
[],
python_fu_vignette,
menu = "<Image>/Python-Fu/Vintage")
```

```
main()
```

Скрипт будет доступен в меню **Python-Fu -> Vintage -> Vignette...** При запуске будет выведено окно с настройками, в котором можно выбрать цвет и прозрачность виньетирования.

Тимур ГАФАРОВ





Язык



Новости «с Марса» свежие релизы и обновления

Проекту Derelict – 10 лет

8 мая 2004 года программист Майк Паркер опубликовал в digitalmars.D.announce первое сообщение о Derelict – проекте по созданию коллекции динамических привязок к мультимедийным библиотекам.

Несмотря на название («derelict» в переводе с английского означает «заброшенный»), проект отнюдь не был заброшен, как многие другие D-библиотеки того времени. Напротив, Derelict активно развивается и имеет солидную аудиторию пользователей – он применяется практически во всех играх, написанных на D. Сейчас актуально уже третье поколение Derelict, ориентированное на самые последние версии OpenGL, SDL и других популярных библиотек.

<https://github.com/DerelictOrg>

Игровой движок Dash

Dash – профессиональный игровой движок, написанный на D разработчиками из Circular Studios. Движок использует OpenGL, работает под Windows и Linux. Есть поддержка скелетной анимации, ресурсов на YAML, сетей, а также веб-интерфейс с использованием Awesomium.

На основе Dash сейчас создается трехмерная пошаговая стратегия Spectral Robot Task Force, публичную альфа-версию которой обещают в конце 2014 года.

<https://github.com/Circular-Studios/Dash>
<http://circularstudios.com>

DBox

DBox – это полный порт популярного двумерного физического движка Box2D с C++ на D2. Основан на последнем релизе Box2D 2.3.1, включает тестовое приложение.

<https://github.com/d-gamedev-team/dbox>
<http://box2d.org>

dlib 0.3.1

Состоялся релиз коллекции библиотек dlib 0.3.1. В этой версии улучшен пакет dlib.image, добавлена поддержка интерполированного чтения пикселей. В dlib.math добавлены операторы суммирования и вычитания матриц, а также тензорное произведение векторов любых размерностей. Исправлено много ошибок.

<https://github.com/gecko0307/dlib>
<https://github.com/gecko0307/dlib/releases/tag/v0.3.1>



Tkd

Tcl/Tk обязан своей популярностью тому, что позволяет быстро и легко создавать графические приложения. Теперь данный тулkit доступен и для D – в лице проекта Tkd.

<https://github.com/nomad-software/tkd>

DGUI

Возобновлена разработка DGUI – графического тулкита для D под Windows с API, близким к .NET. Уже сейчас по своим возможностям DGUI значительно превосходит DFL. В будущем планируется поддержка Linux.

<https://bitbucket.org/dgui/dgui>

DIMGUI

Анонсирован DIMGUI – D-порт библиотеки IMGUI, легковесного графического тулкита на основе OpenGL. Этот тулkit используется в тестовом приложении, поставляемом с DBox.

<https://github.com/d-gamedev-team/dimgui>

DlangUI

Последние месяцы ознаменовались ростом интереса к GUI-приложениям на D – это уже четвертый анонс графического тулкита в нашем обзоре! DlangUI – кроссплатформенная библиотека построения GUI, использующая SDL, Win32 или XCB в качестве бэкенда (есть возможность добавлять новые). Имеется поддержка отрисовки при помощи OpenGL. Поддерживается интернационализация и Юникод.

<https://github.com/buggins/dlangui>

DAuth 0.6

DAuth – система аутентификации для D – обновилась до версии 0.6. В этой версии добавлена поддержка SHA-2 и генератора случайных чисел Hash_DRBG, проведены различные мелкие улучшения и исправлены баги.

<https://github.com/Abscissa/DAuth>

TitaniumD

Анонсирован TitaniumD – биндинг к библиотеке Botan.

Botan – это открытая библиотека криптографии, написанная на C++11. Она включает реализацию стандартов SSL/TLS, X.509, алгоритмов ECDSA, AES, RSA, SHA-2, SHA-3 и многих других.

<https://github.com/ellipticbit/titanium>
<https://github.com/randombit/botan>

Regal

Regal – библиотека реляционной алгебры (генератор SQL-запросов). Может работать с любыми БД. API библиотеки близок к Arel для Ruby.

<https://github.com/dymk/regal>

Duml

Генератор UML-дескрипторов из классов D. Полученные данные предназначены для программы PlantUML, которая, в свою очередь, формирует графические представления UML-диаграмм.

Для справки: UML (Unified Modeling Language) – это язык графического описания абстрактных моделей каких-либо систем. Чаще всего UML используется для проектирования программного обеспечения.

<https://github.com/rikkimax/Duml>
<http://plantuml.sourceforge.net>

Mono-D 2.0

Mono-D – плагин для среды MonoDevelop/XamarinStudio, предоставляющий поддержку языка D в этой IDE – обновился до версии 2.0. Главное новшество релиза – совместимость с XamarinStudio 5.0.

<http://mono-d.alexanderbothe.com>

Visual D 0.3.38

Visual D – проект по интеграции D в среду разработки Microsoft Visual Studio – обновился до версии 0.3.38. Улучшена совместимость с VS 2013, проведены различные мелкие исправления.

<http://rainers.github.io/visuald>

wave-d

Мини-библиотека для чтения/записи аудиоформата WAV.

<https://github.com/p0nce/wave-d>

Книга «Программирование на D» переведена на 97%

712 из 732 страниц переведены с турецкого на английский. Книга доступна для чтения онлайн и скачивания в виде PDF.

<http://ddili.org/ders/d.en/index.html>

Наши проекты

Cook

Программа автоматизации сборки проектов на языке D. В отличие от аналогичных инструментов (Make, CMake, Scons, Jam, DSSS и др.), Cook не требует конфигурационного файла: всю информацию о проекте она получает самостоятельно, сканируя модули (файлы *.d). При этом программа отслеживает прямые и обратные зависимости между модулями: если модуль был изменен, необходимо скомпилировать заново не только его, но и все модули, которые от него зависят (это важно, если был изменен внешний интерфейс модуля: объявления классов, семантика шаблонов и т.д.). Для этого Cook производит лексический анализ модулей - но не всех, а только тех, которые были изменены со времени последнего анализа. Данные анализа кэшируются в файл для повторного использования (кэш автоматически обновляется при пересборке). Cook работает в Windows и Linux.

<http://github.com/gecko0307/cook>
<http://github.com/gecko0307/cook2>

dlib

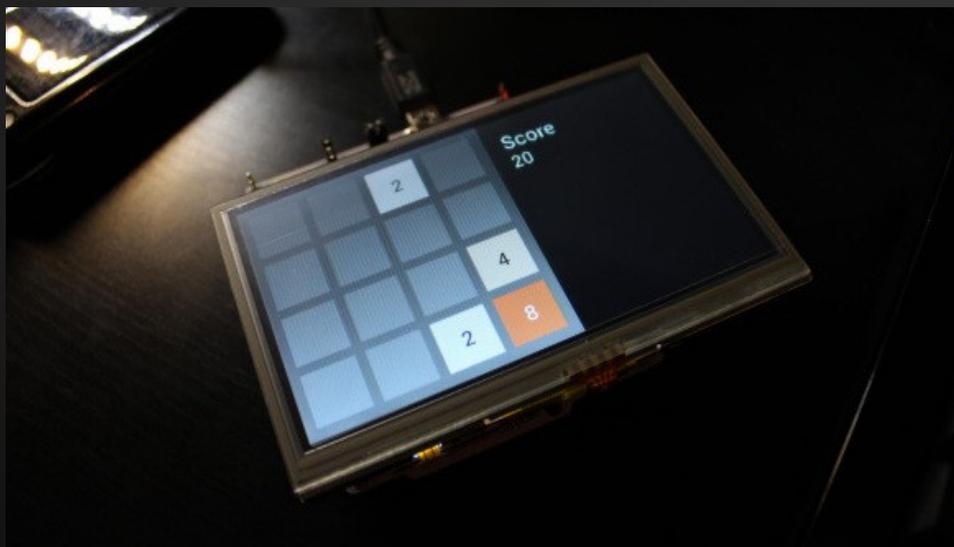
Коллекция библиотек «на все случаи жизни» для D, которая может быть использована в игровых движках и других мультимедийных приложениях. Написана на D2 с использованием Phobos, не имеет никаких других внешних зависимостей. Разработка dlib пока находится на ранней стадии - API нестабилен и может измениться в любой момент, если появится возможность улучшить общую архитектуру.

<http://github.com/gecko0307/dlib>

Making-of Клон игры 2048

«2048» – новый игровой феномен, пришедший на смену Flappy Bird. Изначально созданная итальянским программистом Габриэлем Чирулли на основе концепции другой аналогичной головоломки Threes, нехитрая «2048» намертво приковала к себе внимание как игроков, так и разработчиков. Сейчас в Интернете можно найти десятки ее клонов – и это только на JavaScript и HTML!

Разумеется, игра портирована на все мобильные платформы, на Flash и на PC. Есть клоны «2048» в стиле Tetris, в трех измерениях, с поддержкой мультиплеера, с использованием дополненной реальности, с сотовыми ячейками, с квантовыми состояниями, с поддержкой отмены, с римскими цифрами, с цветами вместо чисел... Существуют версии игры в консольном режиме, версии для NES, SNES, GameBoy, SEGA Mega Drive, Atari 2600 – этот «вирус» захватил все компьютеры мира!



Кстати, с солидным списком клонов можно ознакомиться на сайте <http://get2048.com>.

Я тоже решил не оставаться в стороне и написал собственную версию игры – а если конкретнее, десктопный вариант «2048» для двух локальных игроков.

Экран делится пополам: левое поле управляется WASD, правое – клавишами-стрелками. Игровой процесс, теоретически, может продолжаться бесконечно – пока не заполнятся оба поля. Побеждает тот, кто к концу своей игры наберет больше очков.

Скачать игру можно [здесь](#).

Правила

Я следовал оригинальным правилам, которые вы можете найти на Википедии в статье, посвященной «2048»:

- В каждом раунде появляется плитка номинала 2 (с вероятностью 90%) или 4 (с вероятностью 10%).
- Нажатием стрелки игрок может скинуть все плитки игрового поля в одну из 4 сторон. Если при сбрасывании две плитки одного номинала «налетают» одна на другую, то они соединяются в одну, номинал которой равен сумме соединившихся плиток. Если при нажатии кнопки местоположение плиток или их номинал не изменится, то ход не совершается.
- За каждое соединение игровые очки увеличиваются на номинал новообразовавшейся плитки.
- Игра заканчивается, если после очередного раунда невозможно совершить действие.

Детали реализации

Игра написана на языке D. Вывод графики осуществляется через OpenGL, за пользовательский ввод и создание окна отвечает SDL, шрифт загружает FreeType.

Также были использованы следующие компоненты:

- **dlib** – библиотека общего назначения (математика, геометрия, обработка изображений и т. д.)
- **Derelect** – динамический биндинг к OpenGL, SDL и FreeType
- **DGL** – движок для создания двумерных и трехмерных игр на D

Игра поддерживает локализацию (все тексты кодируются в UTF-8). Есть английская и русская локаль, пользователь может добавлять новые.

16	32	128	4				4
	4	8	16	2		2	4
			2	8	16	32	16
2				16	256	8	2
Счет: 968				Счет: 2040			

Алгоритм сбрасывания

Я намеренно не заглядывал в исходный код оригинальной «2048» – какой смысл копировать чужой код для создания клона чужой же игры? Поэтому мной был разработан собственный стековый алгоритм сбрасывания плиток.

Игровое поле представлено матрицей 4x4 с возможностью доступа к столбцам и строкам как к векторам.

Дополнительно добавляется возможность получить вектор указанного индекса в одном из четырех направлений сбрасывания (влево, вправо, вверх, вниз). При этом начало вектора всегда соответствует крайнему элементу в данном направлении – векторы «вправо» и «вверх» будут инвертированы. Иными словами, если вектор «влево» – это обычная строка матрицы, то «вправо» – это обратный вариант вектора «влево».

Благодаря такому подходу можно сбрасывать плитки (элементы вектора) всегда в одном направлении (влево, к началу массива), не заботясь далее о направлениях.

Естественно, матрица возвращает не сами элементы, а указатели на них – чтобы мы могли их модифицировать.

Когда пользователь совершает сбрасывание в каком-либо направлении, мы просто проходим циклом по четырем векторам в этом направлении. Для хранения промежуточного результата используем стек.

Перебираем элементы текущего вектора: если мы обнаружили элемент с номиналом, равным номиналу верхушки стека, то суммируем эти значения и заменяем верхушку стека этой суммой. В противном случае добавляем элемент на верхушку стека.

В конце перебора на стеке получается результирующий вектор (дно стека соответствует началу вектора) – мы просто записываем его по указателям в соответствующую строку или столбец матрицы, с инверсией или без.

Анимация

Вышеописанный алгоритм – дискретный: он принимает на вход одно состояние игрового поля и дает на выходе измененное, со смещенными до предела плитками. Но на практике это выглядит не очень эффектно: пользователь не успевает понять, что куда переместилось. Поэтому имеет смысл перемещать плитки не резким скачком, а постепенно, с некой заданной скоростью.

Но для этого совсем необязательно отказываться от дискретного алгоритма и вводить проверку столкновений. Я решил проблему путем разделения игры на два чередующихся состояния – статическое и динамическое. Статическое состояние отображает данные матрицы в промежутках между ходами пользователя. Динамическое отображает процесс сбрасывания: дискретный алгоритм просто создает движущиеся плитки в специальном пуле – в момент создания плитка запоминает свой номинал, позицию начала движения и позицию конца (эти позиции нетрудно вычислить, зная координаты соответствующих элементов матрицы). Когда все плитки достигают своих конечных позиций, динамическое состояние сменяется статическим, и все начинается заново.

Случайность

Отдельного упоминания заслуживает алгоритм генерации случайной плитки. Сначала я нахожу все незанятые ячейки поля путем заполнения в цикле специального буфера: если ячейка имеет номинал 0, ее индекс помещается в буфер. Затем генерирую случайный номинал – 2 или 4 с вероятностью 90% и 10%. Здесь помог модуль `std.random` из стандартной библиотеки D:

```
uint nominal = (dice(90, 10) + 1) * 2;
```

Затем получаю случайный индекс из буфера незанятых ячеек и записываю по нему номинал:

```
uint i = buffer[uniform(0, 16)];  
boardMatrix[i] = nominal;
```

Тимур ГАФАРОВ

OpenSource Report Card

Каждый день на GitHub обновляются многие тысячи репозиторииев с OpenSource-проектами. При этом данные об этих коммитах доступны всем желающим через публичный API – в GitHub Archive хранится огромная база истории активности всех пользователей! На основе этих данных сторонние разработчики могут создавать различные веб-приложения – и одним из самых интересных является проект OpenSource Report Card.

Данный ресурс предоставляет владельцам аккаунтов на GitHub уникальную возможность проанализировать статистику их активности «на человеческом языке». На основе информации о коммитах и другой активности, OSRC формирует нечто вроде профессионального профиля пользователя: описание его стиля разработки, умений и привычек, языковых предпочтений, взаимоотношений с другими пользователями. При этом результаты настолько «живые», что, читая досье на самого себя, может на первый взгляд показаться, что его составил живой человек, отслеживающий вашу работу!

На самом деле, конечно, информация чисто машинная. Автор, во избежание каких-либо претензий, так и подчеркивает: OSRC не является вашим резюме – это всего лишь забавный «игрушечный» сервис, который не стоит воспринимать слишком серьезно.

<http://osrc.dfm.io>

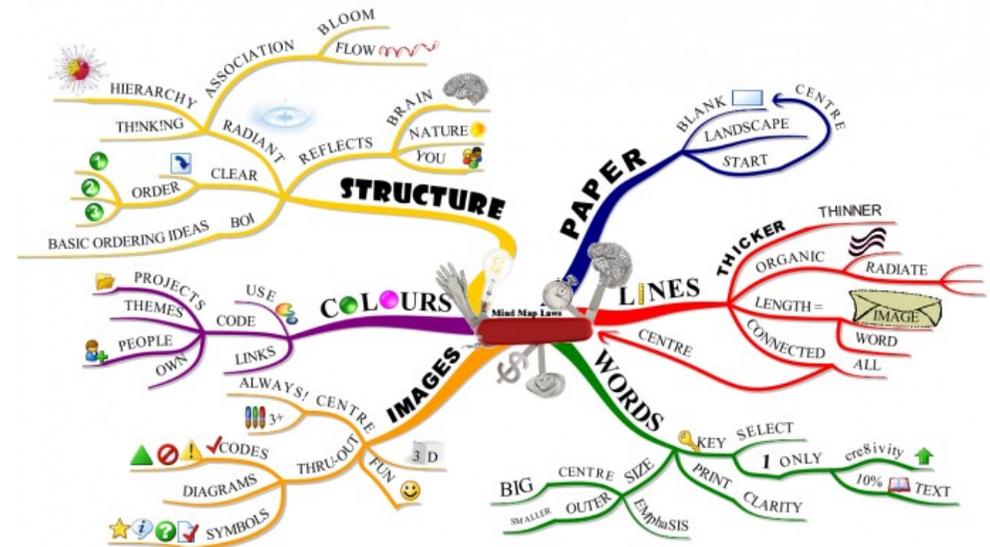
Как работают великие программисты

Во-первых, они **НЕ** делают определенные вещи:

- **Они не изобретают велосипед.** Нет никакого смысла тратить время на создание того, что другие уже сделали до вас – и сделали гораздо лучше, чем вы когда-либо смогли бы. Если есть готовые библиотеки, куски кода, примеры и т.д., они используют этот материал и вкладывают силы в создание чего-то нового, вместо того, чтобы в сотый раз переделывать старое.
- **Они не пишут код, который им не нужен прямо сейчас.** Добавлять новую функциональность, считая, что она пригодится в будущем – это неверный путь. Никто не может знать, что ему понадобится в будущем, поэтому хороший программист добавляет новую функциональность только по мере необходимости.

Что они делают:

- **Они держат в голове мысленный план проекта** – «ментальную карту». Перед тем, как написать хотя бы строчку кода, хороший программист уже знает, как она будет выглядеть – так же, как художник представляет в своем воображении будущую картину перед тем, как положить мазок.
- **Они работают над проектом гораздо больше, чем кажется окружающим.** Лежа в постели перед сном, принимая душ, сидя за рулем, они продумывают архитектуру проекта: «ментальная карта» эволюционирует беспрерывно!



- **Они тестируют свой код** по мере написания. Даже гении не застрахованы от ошибок.

- **Они накапливают опыт.** Блестящий ум или прочтение хорошей книги еще не делают вас великим программистом – как не делают и 30 лет однообразной работы на одной должности.

Великие программисты пишут много кода – они пишут на разных языках, они участвуют в разных проектах. Они накапливают в памяти сотни и тысячи приемов, паттернов и шаблонов, которые применяли на практике – и умеют применить их повторно там, где они больше всего уместны.

Роза Углы, Parkside Software, LLC

Linux-гейминг

Игровые новости из мира Linux

Прошедшие месяцы оказались невероятно богаты событиями из мира разработки игр под Linux. Начнем с того, что компания Leadwerks Software, которая в прошлом году на Kickstarter собрала 42 тысячи долларов на создание Linux-версии своего игрового движка Leadwerks Engine, опубликовала первый демонстрационный выпуск этой версии!

Сборки движка подготовлены для Ubuntu 14.04, для работы необходимы проприетарные видеодрайверы NVIDIA и ATI и видеокарта с поддержкой OpenGL 4.0. Для разработки 3D-моделей в Leadwerks обеспечена интеграция со свободным пакетом 3D-моделирования Blender. Интерфейс среды разработки построен с использованием GTK+. Разработка игр осуществляется на языках C++ и Lua. Компоненты движка Leadwerks поставляются под проприетарной лицензией.



А компания GarageGames, в 2012 году открывшая под свободной лицензией MIT исходные тексты игрового движка Torque, представила релиз Torque 2D 3.0, примечательный обеспечением полноценной поддержки Linux и Android, а также компиляции игр в JavaScript-представление для Web-браузеров с поддержкой WebGL.



Изначально Linux поддерживался только в серверной части продукта, а клиентские компоненты были доступны только для Windows, Mac OS X и iOS. В прошлом году была организована краудфандинг-кампания по сбору средств для портирования движка для Linux, но в ее рамках удалось собрать чуть больше половины требуемой суммы.

В конечном счете, компания GarageGames при участии энтузиастов смогла выполнить портирование без дополнительного финансирования.

Из других улучшений Torque 2D 3.0 можно отметить начальную поддержку скелетной анимации (Parametric Skeletal Animation), возможность создания 64-разрядных сборок, поддержку хранения объектов и параметров сцен в формате JSON, поддержку Leap Motion и контроллеров Xbox 360.

Компания Crytek, развивающая игровой движок CryEngine, на основе которого построены такие игры, как Crysis, Far Cry, Warface и Ryse: Son of Rome, объявила об открытии исходных текстов инструментария Renderdoc, предназначенного для отладки графических и игровых приложений, использующих средства вывода 3D-графики. Код открыт под свободной лицензией MIT и опубликован на GitHub.



Renderdoc предоставляет графический интерфейс для захвата, анализа, глубокого инспектирования, повторного кадрового повтора графических операций и исследования использования графического API.

Несмотря на поддержку сборки Renderdoc в Windows и Linux, в текущем состоянии поддерживается только отладка приложений, использующих Direct3D 11, но в будущих выпусках обещают добавить поддержку OpenGL.

Напомним, что ранее компания Valve открыла под лицензией MIT код похожего отладчика VOGL, нацеленного на работу с OpenGL.

Кстати, Valve недавно анонсировала новый выпуск клиента Steam для Linux, а также обновление SteamOS. Новая версия Steam примечательна исправлениями множества ошибок и новой функцией «In-Home Streaming», которая позволяет организовать потоковую трансляцию мультимедиа-контента и игр на другие домашние ПК, на которых запущен клиент Steam. Кроме того, в режиме Big Picture для платформ, отличных от Windows, по умолчанию теперь выводится не общий список игр, а только игры, работающие на текущей платформе.

Режим «In-Home Streaming» доступен для всех поддерживаемых платформ: можно запустить игру на мощном ПК, а непосредственно играть в нее со слабой системы. Или можно организовать «проброс» игры с Windows-системы на компьютер с Linux или на приставку с SteamOS. Функция предоставляет полноценные средства удаленного доступа: подключившись к другому ПК, пользователь может выполнять на нем все типовые операции Steam – в том числе устанавливать, запускать и играть в игры. Весь вывод и ввод с клавиатуры, мыши и игрового контроллера прозрачно транслируются на удаленную систему.



Компания Epic Games объявила о поддержке Linux в будущем шутере Unreal Tournament 4. Как и недавно выпущенный движок Unreal Engine 4, код игры будет размещен в закрытом репозитории на GitHub и будет доступен разработчикам по специальной подписке. Любой желающий может бесплатно зарегистрировать аккаунт и участвовать в обсуждениях разработки и дизайна игры.

Что самое интересное, Unreal Tournament 4 будет распространяться бесплатно! В качестве источника дохода рассматривается создание каталога-магазина, в котором сторонние разработчики будут продавать контент и моды, а Epic Games получит определенный процент от продаж.



Также компания опубликовала обзор планов по развитию игрового движка Unreal Engine 4 для платформы Linux, а также по предоставлению нативных средств для разработки игр в Linux. Подобно Valve, Epic Games рассматривает Linux как первичную игровую платформу, стоящую в одном ряду с Windows и OS X, и позиционируемую не только для игровых серверов, но и для клиентских систем.

Кстати, для Linux теперь доступна выпущенная в 2011 году ролевая игра The Witcher 2: Assassins of Kings, которая в свое время получила премии «Лучшая европейская игра» и «Игра 2011 года». Игру можно установить через Linux-версию клиента Steam.



Компании 4A Games и Deep Silver анонсировали игры Metro 2033 Redux и Metro: Last Light Redux, в рамках которых будут предложены переработанные и улучшенные варианты игр Metro 2033 и Metro: Last Light, созданные по мотивам романа Дмитрия Глуховского «Метро 2033». Серия Metro Redux будет официально поставляться в версиях для Linux и SteamOS.

Компания Feral Interactive объявила о планах выпуска нативного варианта стратегической игры XCOM: Enemy Unknown для платформы Linux. Игра будет распространяться через Steam (заявлено о поддержке Ubuntu 14.04). Для Linux также будут выпущены все доступные обновления к игре, в том числе XCOM: Enemy Within. Кроме того, появится Linux-версия стратегии Stronghold 3 Gold, изначально выпущенной осенью 2011 года для платформ Windows и OS X.

The Tale. Сказка сказок

Можете себе представить игру... без игрока? В противовес распространенной игровой схеме с участием одного или нескольких персонажей/юнитов, подконтрольных играющему, существует концепция Zero Player Game, в которой нет персонажа, которым вы могли бы напрямую управлять. Именно к такому типу относится проект, о котором пойдет речь ниже – The Tale.

Это браузерная многопользовательская RPG с непрямым управлением персонажем и изменяемым от действий игроков миром. Вы играете за ангела-хранителя у сказочного героя, который самостоятельно занимается своими геройскими делами. Влиять на героя можно только косвенно: выбирая способности, настраивая характер и помогая делать выбор в нелинейных заданиях. Также иногда можно помочь напрямую, например, подлечив подопечного.

В каждом задании у героя есть выбор поведения, от которого зависит изменения «влияния» участвующих NPC. Те, в свою очередь, определяют параметры городов, в которых живут, тем самым изменяя весь игровой мир. За счет этого игроки могут координировать свои усилия для изменения мира в нужном направлении.

Ключевые особенности:

- Два уровня геймплея:
- «Тамагочи»: игроки «растят» своего героя, помогая ему в приключениях;
- «Стратегия-песочница»: игроки координируют свои усилия для получения необходимого влияния на мир;
 - Герои обладают характером, который влияет на их поступки;
 - Состояние мира игры полностью зависит от действий игроков;

Сайт проекта: <http://the-tale.org>



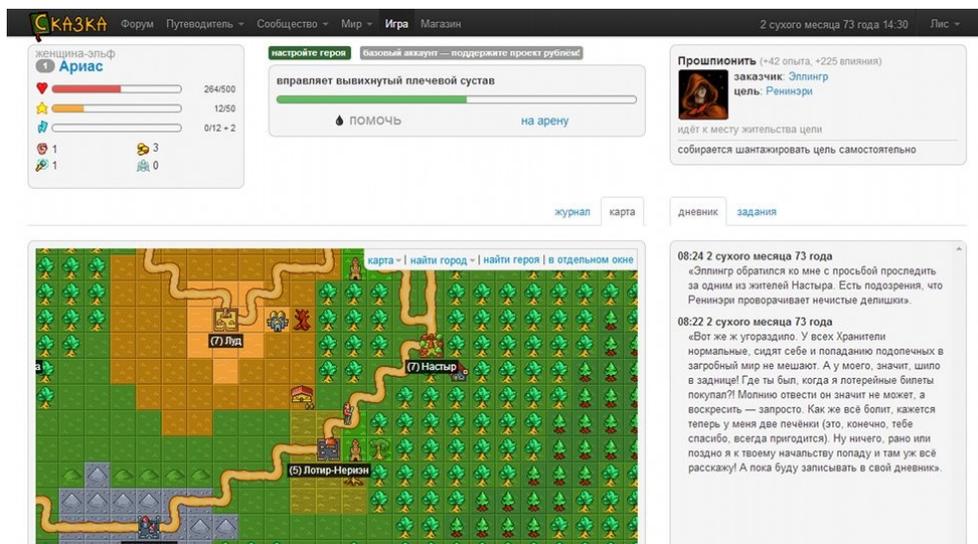
Журнал «FPS» связался с автором игры, программистом **Алексеем Елецким**, чтобы узнать подробности об этом необычном проекте.

Здравствуйте, Алексей! Не могли бы вы ответить на несколько вопросов?

С радостью.

Как к вам пришла идея создания вашей «Сказки»? Подтолкнула ли на создание проекта игра «Годвилль»?

Сделать что-то похожее я хотел еще со школы, но нельзя сказать что идея «Сказки» появилась в одно мгновение – скорее она выкристаллизовалась из множества небольших экспериментов, которые я проводил в свободное от учебы (а позже – от работы) время.



Больше всего на «Сказку» повлияло две игры:

- **Majesty: The Fantasy Kingdom Sim** (1-я часть) – это, наверно, первая масштабная компьютерная игра, основанная на непрямом управлении самостоятельными героями.
- **«Годвилль»** – в первую очередь, повлиял фактом своего существования. Насколько я знаю, его тоже разрабатывает всего несколько человек. Успешный пример придал уверенности для начала собственного проекта.

На саму игру влияние оказал не столько сам «Годвилль», сколько все существующие ZPG игры (которых не так много), итогом развития которых он является.

Вами ли одним проработан сюжет? Или здесь трудится какая-то команда сказочников?

На текущий момент у нас в команде 3 человека. Кроме меня, это еще семейная пара: Саша и Лена Дедковы – наши геймдизайнеры и авторы большинства художественных текстов.

В формировании мира я принимаю мало участия: сформулировал основные идеи и слежу, чтобы про них не забывали. Сейчас Пандора полностью под ответственностью Саши и Лены.

Текущий сюжет у нас полностью определяется игроками, с чем они успешно справляются. Правила игры создают множество точек соприкосновения интересов как отдельных игроков, так и гильдий, поэтому скучать не приходится. За 74 года игрового времени были и войны, и предательства, и великие союзы...

Наиболее важные события автоматически записываются в летопись (<http://the-tale.org/game/chronicle/>).

Кроме того, их можно отследить по творчеству игроков в разделе фольклора и по истории принятия законов в политике. Игроки любят roleplay, поэтому наблюдать за развитием событий становится все интереснее.

Как вы можете оценить перспективы развития проекта? Как быстро он развивается?

Перспективы у нас, на мой взгляд, хорошие. Конкуренции почти нет. Да и какая конкуренция у ZPG – одновременно можно хоть в 100 штук играть :)

Мы стараемся регулярно выпускать обновления: обычно в игре появляется что-то новое раз в 2-4 недели. Кстати, часть статистики игры (включая доход) находится в открытом доступе: <http://the-tale.org/statistics/>

Продолжится ли развитие «Сказки»? Чем вы собираетесь привлечь новых пользователей?

Безусловно продолжится. Привлекаем людей мы оригинальным геймплеем, дружным сообществом и нетребовательностью ко времени: 10 минут в день достаточно, чтобы не отставать в развитии своего героя.

На **странице с планами разработки** можно узнать подробнее о том, что будет с игрой в будущем.

Может ли поклонник вашей игры поучаствовать в ее улучшении?

Да, и для этого есть множество способов:

- Мы тесно общаемся с сообществом на нашем форуме и всегда открыты для новых идей и критики.
- Игроки могут предлагать новые фразы для журнала и дневника героя.
- Для части функционала игры мы открыли API и всячески поддерживаем любые начинания игроков. Добровольцы уже реализовали информер для форумов, клиент для устройств на Android и расширение для популярных браузеров.
- Часть кода игры открыта для всех желающих (например, генераторы имен и квестов, карты). Эти библиотеки можно не только улучшать, помогая игре, но и использовать в своих проектах.

Со временем мы планируем открыть весь код игры, но для этого нужно проделать еще много работы.



Что ж, пожелаем «Сказке» долгих лет жизни и развития, а ее авторам – успехов в их интересном деле! Подобные проекты доказывают, что в игровой индустрии все еще возможны открытия новых жанров и свежих идей. Кто знает – возможно, за ZPG будущее ролевых игр?..

Наталья ЧУМАКОВА

Памятка читателю

В Интернете часто можно встретить вопросы о том, где скачать старые номера нашего журнала. Отвечаем. Архив всех номеров «FPS» (с 2008 по 2014 гг.) можно найти сразу на нескольких сервисах:

На файловом хостинге **DropBox**:

https://www.dropbox.com/sh/b7lgxxh6nxbxre9/uVvzqU8_j-

В **Документах Google** (для скачивания файлов нужен аккаунт Google):

<https://docs.google.com/folderview?id=0B1BlzRb1uMv-bnpHNDhwZTI4eHc>

В электронном издательстве **Issuu.com**:

<http://issuu.com/tgafaroff/docs>

Для тех, кто предпочитает скачивать с торрентов – журнал также есть на **PyТреке**:

<http://rutracker.org/forum/viewtopic.php?t=4403193>



КУЛЬТОВЫЕ ИГРЫ

Игровой арт-хаус

Что такое артхаусные игры и как они отличаются от обычных? Артхаусом принято называть то, что выходит из рамок обыденного, вносит какое-то разнообразие и неоднозначность.

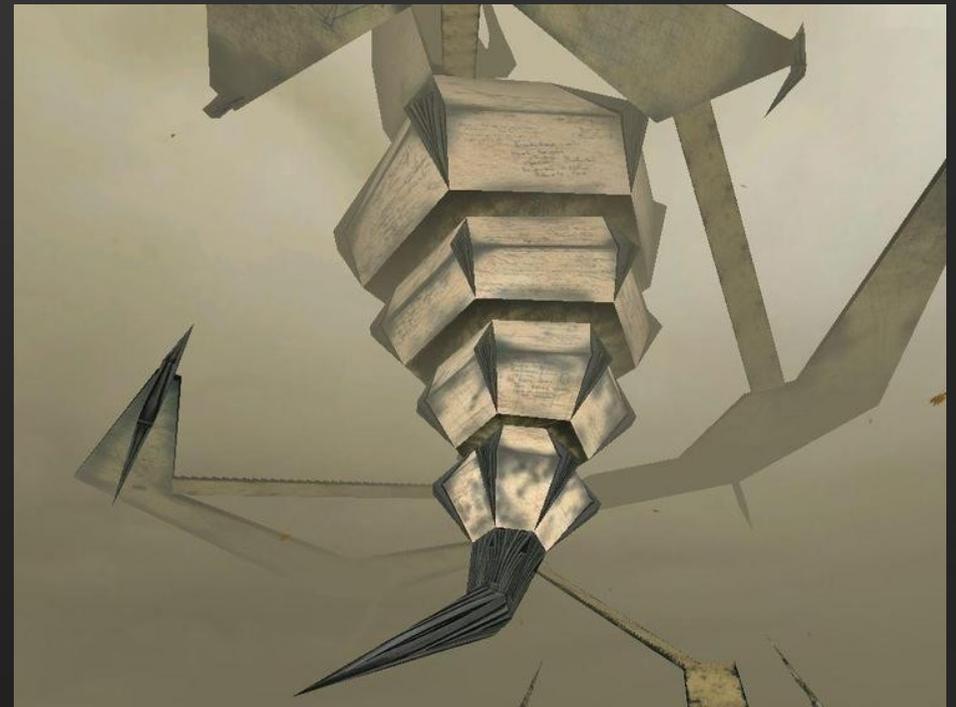
Артхаусные игры – это иной игровой процесс, иной взгляд на геймплей, на структуру игры и дизайн. Это всегда игры, которые не пользуются огромной популярностью, в них главное – это подача материала разработчиком, некая особая зрелищность, посыл и новая идея.

В «Море» вам предстоит играть за одного из трех персонажей, которые оказались в закрытом городке во время эпидемии. Главная цель в игре – это выжить, несмотря ни на что.

Отдельно хочется отметить проработанный мир игры, в которую попадает ваш персонаж. Маленький городок живет своей жизнью, в нем запутанные улочки, он окружен болотами, жертвенными алтарями... Особенно впечатляют в городе два огромных здания: это скотобойня и Многогранник, необычная постройка, возвышающаяся над городом.



Интересно, что в России данный жанр развивается довольно активно. Основными разработчиками является московская компания Ice-Pick Lodge. Студия была создана в 2002 году, а в 2005-м вышла их первая игра – «Мор. Утопия», которая вдохнула новую жизнь в российский рынок видеоигр.



Днем город живет, но, как только сгущается тьма, на охоту выходят грабители и убийцы, которых следует опасаться. Кроме людей, город населяют странные существа – Черви. Также в этом мире особая мифология и свои суеверия. Сложно сказать, в каком веке он затерялся – в одном из дней в городе сжигают женщину, обвиненную в колдовстве, хотя люди носят вполне современную одежду. Мор – это мир вне времени и пространства, просто некий островок отчаяния...



Игра имеет несколько концовок – все зависит от поведения игрока, кого он в этой игре спасет, кого оставит погибать в жестоком мире «Мора». Могу сказать, что после игры остается очень неприятное «послевкусие»: она необычна и интересна, но таит в себе в огромном объеме мрачность, тягучесть и чувство безысходности.

Необычный сюжет, игровой процесс и особая атмосфера позволила «Мору» получить отзыв как об игре «одной из лучших на свете, но совсем не для всех». Игра не провалилась в продажах, но и не имела особый успех. Чем не игровой артауас?..

Ice-Pick Lodge продолжили радовать поклонников «особых» игр своими творениями. В 2008 увидела свет игра «Тургор», очередной образец артауаса.

Разработчики заранее предупредили покупателей игры, что она является не развлечением, а способом донести до человека свои особые мысли, построить с ним диалог. В «Тургоре» игрок перемещается по особому миру – Промежутку, который является олицетворением творческого застоя. Игрок борется с разными существами при помощи Цвета – это духовная пища персонажа: только в его присутствии он может жить. Цель игры – выбраться из этого странного мира.



Здесь обитают загадочные эротизированные персонажи – Сестры, а так же Охотники – такие же создания, как и вы, пытающиеся высвободиться из Промежутка.



Загадочный мир, сложный геймплей и продуманная сюжетная линия – это игра «Тургор», получившая награду «Самый нестандартный проект» на КРИ в 2007 году.

Теперь расскажем кратко о ярких представителях данного жанра за рубежом. В 2013 году студия Krillbite Studio, в промежутке между напряженной работой над другими своими проектами, разработала... симулятор мухи, получивший название **The Plan**.

Игра удивительно здорово сделана – она красива и реалистична. Вы попадаете в вечерний час в мир, полный тумана, ласкового ветерка: легко покачиваются деревья, играет классическая музыка.

Игра не имеет какой-то цели или смысла, ее задача – это дать игроку насладиться ощущением покоя и умиротворения. Найти ее можно в Steam:

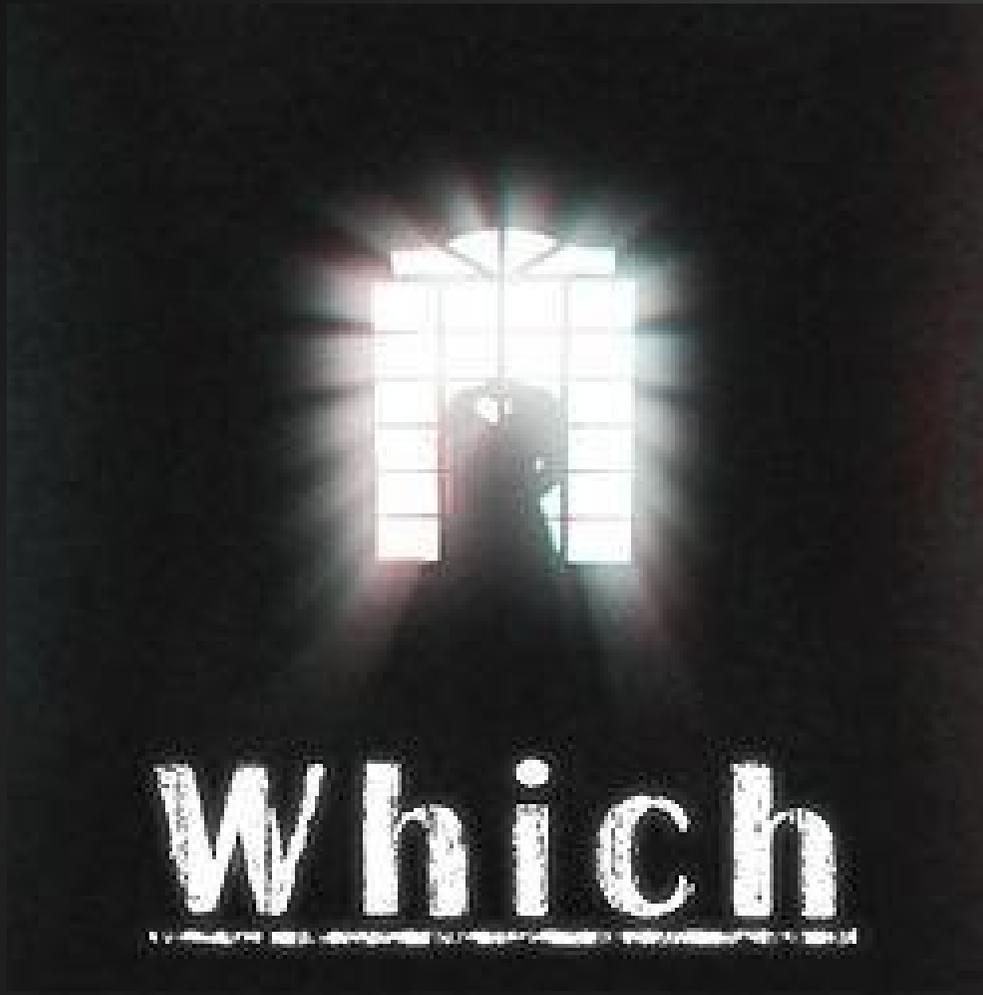
<http://store.steampowered.com/app/250600/>



Разработчик Mike Inel в 2010 году создал особую игру, ставшую одной из ярких представителей архауса. Это игра **Which**.

Ее жанр можно описать как квест с элементами хоррора. Вы оказываетесь в закрытом доме, цель игры – выйти наружу, подобрав ключи. Но все не так просто: в поисках герой находит тело женщины без сердца и головы, которая держит в руках тот самый нужный ключ.

Игроку следует найти недостающие части тела жуткой незнакомки – в зависимости, от того, что именно герой отдаст ей, он либо выживет и выберется из дома, либо в нем погибнет...



Палитру игрового мира составляют серые оттенки, игра проходится очень быстро, но при этом волнует и пугает, несет в себе заряд определенных эмоций и мрачное настроение. Скачать ее можно здесь:

<http://gamejolt.com/games/adventure/which/1523>

В список арthouse-видеоигр можно включить и шутер **.kkrieger**. Что же в шутере арthouse, спросите вы? А то, что данная игра занимает всего... 97281 байт. Игра содержит один игровой уровень, который нужно пройти, уничтожив всех врагов и босса. Игра включена в данный список из-за своего новаторства – особого способа генерации всей графики и звука «на лету», без загрузки ресурсов из внешних файлов.



Также в наш список можно отнести **Dear Esther** – игру без особой цели, которая наполнена прекрасными видами необитаемого острова и сопровождается голосом рассказчика, неспешно читающего письмо.

Нельзя недооценивать игровой арthouse – ведь он приносит в мир видеоигр новые технологии игрового процесса, свежий воздух и незаезженные сюжеты. Однако жаль, что такие игры находят мало ценителей, что довольно странно: ведь иногда хочется отложить какой-нибудь поднадоевший экшн и поиграть во что-то новое и необычное!..

Наталья ЧУМАКОВА

ReroGame: пиратские порты на NES

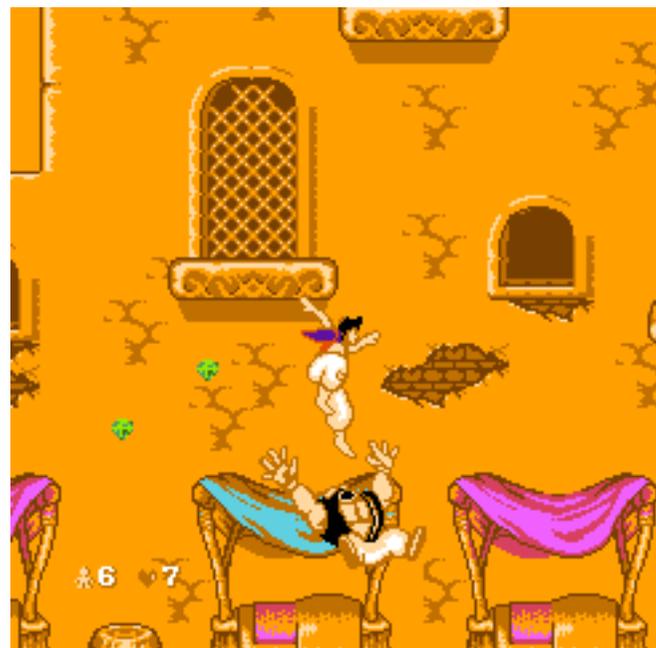
В этом выпуске рубрики «RetroGame» мы подготовили обзор нескольких примечательных пиратских портов на NES – известных тайтлов, перенесенных приставочными пиратами с более мощных платформ без разрешения правообладателей.

Что характерно, зачастую такие «нелегальные» порты гораздо играбельнее официальных. Именно так сложилось с игрой **Aladdin** по мотивам знаменитого диснеевского мультфильма – с ней связано такое запутанное хитросплетение версий от самых разных разработчиков, что далеко не каждый знает, какие из них официальные, а какие – «пиратки». Попробуем разобраться, что к чему...

Оригинальная **Disney's Aladdin** была создана в 1993 году компанией Virgin Interactive для Sega Mega Drive/Genesis. Впоследствии она была официально портирована на NES, Amiga, DOS, GameBoy и некоторые другие платформы. Одновременно с версиями от Virgin, аналогичная игра для Disney была разработана компанией Сарсом для платформы SNES. Что же касается пиратских версий, то на NES существует не один, а несколько неофициальных портов «Аладдина». Самый известный, о котором речь пойдет ниже, является портом именно Сарсом'овской игры, созданный китайской фирмой Hummer Team в 1995 году.

Порт выполнен очень качественно – даже с трудом верится, что вы играете на 8-битной консоли. Графика почти SNES'овская, и, что самое главное, игра не потеряла свой дух, атмосферу мультфильма! Как нетрудно догадаться, вы играете за Аладдина – главного персонажа известной арабской сказки «Волшебная лампа Аладдина» и мультфильма «Аладдин». Сюжет в игре не совсем ясен – нет даже заставки. Вы должны пройти 16 опасных уровней, чтобы сразиться с главным врагом – Джафаром: он, как всегда, затекает что-то очень плохое, и вам нужно его победить...

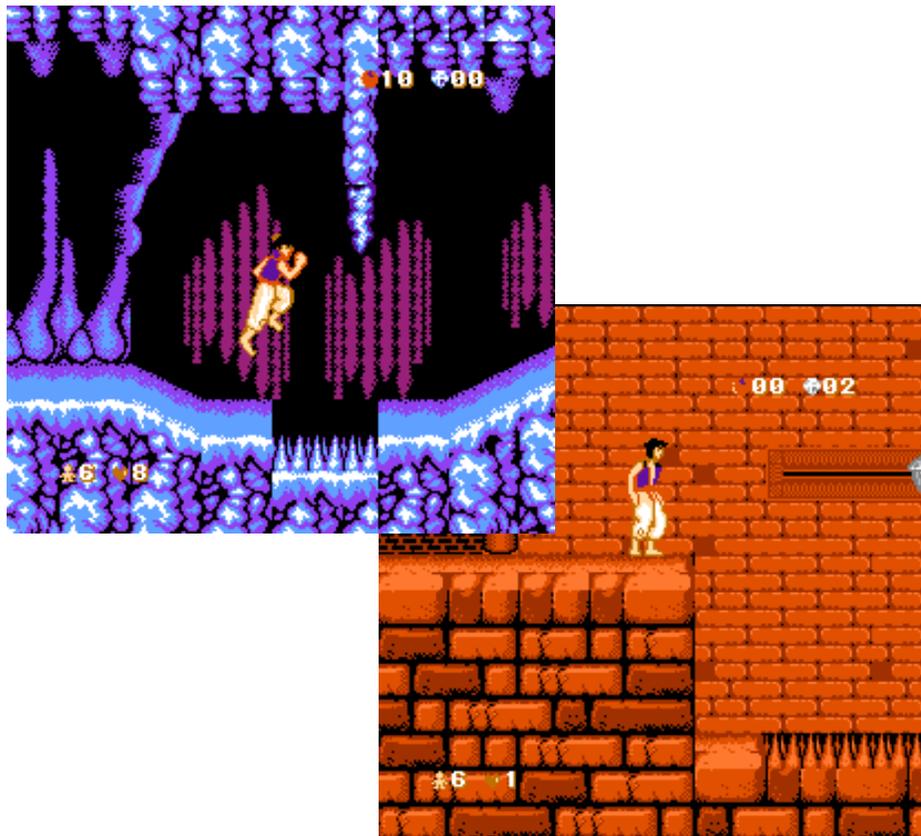
Геймплей в игре достаточно прост и типичен для игры подобного плана. Врагов Аладдин атакует, запрыгивая им на голову. Также можно использовать оружие – яблоки. Для мелких врагов оно смертельно, а больших – например, стражников дворца на 1 уровне – яблоки просто временно оглушают, чтобы можно было быстро пробежать мимо врага или запрыгнуть ему на голову.



Из подручных средств есть также полотно, которое используется как парашют. Если в прыжке нажать Select, Аладдин раскроет полотно и будет приземляться медленнее. Это полезно, когда нужно куда-то плавно упасть – но в игре практически нет таких моментов, и потому целесообразность полотнища вызывает сомнение.

Жареные курицы, встречающиеся на пути, восстанавливают здоровье героя. А если подобрать механического жука, вылетевшего из разбитой вазы, в конце этапа вам предлагается сыграть в мини-игру – своеобразную рулетку, где разыгрываются бонусные жизни и продолжения.

В игре большое разнообразие врагов: стражники, змеи, лучники, жуки, вазы с крыльями или ножками. Все они разделяются на две категории: мелкие (змеи, жуки) и большие (стражи, лучники). Уровни тоже довольно оригинальны и разнообразны: сначала вы проходите несколько улиц города Аграбы, затем оказываетесь в пещере, в песчаном дворце и других интересных местах. Удастся и прокатиться на коврейном самолете. Моим любимым уровнем является мир Джинна внутри лампы – его довольно сложно пройти, но это стоит того!



Боссов в игре всего два – начальник дворцовой стражи, с которым вам предстоит сразиться после 3-го уровня, и сам Джафар – после 15-го. Когда я изначально проходил игру на Dendy, то победить Джафара было очень сложно.

Когда я потратил ему две жизни он в очередной раз вызвал летающие вазы, на которых я напоролся и потерял жизнь – по сути, я должен был начать этап сначала, но в игре произошел глюк. Я автоматически потратил жизнь не только себе, но и Джафору, так что с помощью этого глюка я прошел игру.

Теперь немного о графике и звуке. Графика в игре максимально приближена к оригиналу – в тайлах практически не видно швов. Сам Аладдин отлично анимирован – анимация такого качества на NES встречается крайне редко.

Музыка и звуки, в целом, не уступают графике: саундтрек подобран очень неплохо и соответствует атмосфере. Можно сказать, что Aladdin – одна из лучших игр на NES в своем жанре. Она лишена большинства (если не всех) недостатков, присущих подобного рода портам.

Еще одна популярная серия, из которой пираты, как говорится, выжали все, что могли – Mortal Kombat. Официально этот культовый файтинг не выходил на NES – но пиратских портов существует тьма.

Mortal Kombat 5: 30 People. Разработчик этого творения – китайская компания Yoko. Качество данной игры оставляет желать лучшего. Перенесены далеко не все персонажи из МК: в этой игре почему-то нет Горо и Джонни Кейджа, а Шанг Цунг здесь присутствует в виде... Скорпиона, превращающегося в других бойцов. С именами разработчики совсем все напутали: Flash вместо Raiden, Sova место Sonya Blade, Yoko вместо еще кого-нибудь...

На аренах то и дело встречается откровенная самореклама Yoko – причем в таких местах, где вообще надписи не нужны: например на огромной статуе Горо.

Геймплей далек от оригинального Mortal Kombat'овского. Неприятное впечатление оставляет неудобное управление: А – блок; В – удар. Чтобы сделать удары ногами, нужно зажимать В + стрелки. Однако даже в такой игре создатели умудрились вставить спецприемы бойцов – в основном, стрельбу разными файерболлами и коронные спецалы бойцов типа подката у Саб-Зиро и полета у Рейдена и Лю Канга. Кстати в игре, как ни странно, есть «Finish Him» – то есть, по идее, должны быть и Fatality, но их, к сожалению, нет, и можно просто добить противника каким-нибудь ударом.



Графика сделана на типичном уровне NES'овского хака – персонажи нарисованы весьма примитивно и даже не думают походить на оригинальных. Задние планы взяты из оригинального МК1, но обработаны они тоже не очень хорошо. Музыка и звуки еще терпимы, хотя можно было сделать и лучше. Хотя заставочная мелодия мне немного понравилась за определенный шарм.

Итог: вышла игра, отдаленно напоминающая МК на 16-битках. Из-за некрасивой графики, малокачественного звука и посредственного геймплея, в целом, в нее неинтересно играть.

Mortal Kombat 3: 56 People. Не прельщайтесь: это на самом деле ни какой ни МК3, а МК2 с перерисованным тайтлом. Сама игра является портом МК2 на основе движка Street Fighter. Однако этот продукт заметно лучше предыдущего – и геймплеем, и графикой, и звуком.



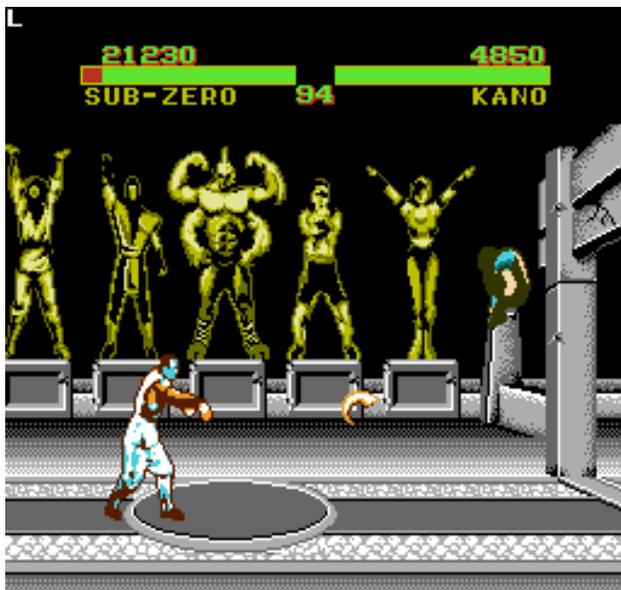
Бойцов здесь целых 14. Не 56, конечно, как было заявлено, но все же присутствуют все бойцы из оригинальной игры + Шао Кан и Кинтаро, за которых в МК2 нельзя было играть! В игре есть все спецприемы бойцов из оригинального МК, и схема из выполнения такая же, как и в оригинале – то есть, заморозка Саб-Зиро действительно замораживает. Бывает, правда, что многие спецприемы не отнимают здоровье, а лишь на время обезвреживают противника. Управление тоже вменяемое – есть отдельные кнопки для ударов руками и ногами, а блок делается двойным нажатием назад, как в Street Fighter.

Конечно, геймплей все-таки далек от оригинального – не хватает Fatality и прочих фиш, полюбившихся игрокам. Сам процесс боя не такой динамичный, но для китайской поделки это очень неплохой уровень.

Графика, честно говоря, просто отличная для 8-битки. Может, конечно, и не предел мечтаний, но все же выполнена в своем неповторимом стиле. Радуют и локации, и бойцы. Что примечательно, в игре есть некоторые арены из первого МК, что дополняет игру и придает ей индивидуальность. Что весьма похвально – в игре есть кровь (хоть и не такая живописная, как в оригинале).

Музыка и звуковые эффекты – средней паршивости. Звуковой движок – самый обычный для китайских игр подобного плана. Музыка не похожа на оригинальные мелодии из МК, звуки тоже не в стиле «Мортала». В общем, в плане звукового сопровождения до шедевра, конечно, далеко – но и не совсем ужасно. В итоге имеем очень неплохой порт второго «Мортала», который мог бы даже сойти за оригинал.

Mortal Kombat II. Может показаться, что эта игра сделана на основе МК2, но на самом деле это не так – в ее основе лежит первый Mortal Kombat. В отличии от «собрата» МК5 от Yoko, игра сделала большой прорыв – она в очередной раз доказывает, что даже на движке другой игры (все того же Street Fighter) можно сделать нечто хорошее.



Итак, вы в очередной раз вместе с бойцами из МК отправились на восточный остров, где проходят турниры «Смертельной битвы». Главными злодеями здесь являются Шанг Цунг и его фаворит – чемпион турнира Горо. На этот раз присутствуют все бойцы – включая Джонни Кейджа, четырехрукого Горо и, что важно, самого Шанг Цунга!

Движок игры несколько хуже предыдущей. Но для МК1, думаю, хватает. Что важно: здесь есть все оригинальные движения бойцов – включая знаменитый удар в пах Джонни Кейджа. Можно играть вдвоем. Второй игрок активируется нестандартно: нажатием на втором джойстике кнопок А или В.

Графика не такая плохая, как можно было ожидать. Фоны максимально похожи на оригинальные, персонажи тоже вышли неплохо и лишь отдаленно напоминают Street Fighter'овских.

Кто бы мог подумать, что о 8-битных консолях вспомнят в эпоху PlayStation 3 и Xbox 360? И уж совсем нереальным казалось появление под NES такого крупного тайтла современности, как Grand Theft Auto. Однако это чудо все-таки свершилось!

В начале «нулевых» двое энтузиастов начали разрабатывать под разные приставки (SMD и NES) адаптацию **GTA 3**.

Один из них – Брайан Провинсиано – также известен разными другими интересными разработками в области геймдева на NES. Например, он создал Devtendo – приставку, совмещенную с компьютером, для создания игр на NES. Соответственно, его проект получил название **Grand Theftendo**.

Релиз был намечен на 2005 год – Провинсиано заявил, что игра будет бесплатной, чтобы не нарушать авторские права Rockstar, создателей оригинальной серии GTA. Проект был закончен на 90%, когда сайт игры внезапно был закрыт, а автор куда-то пропал. По некоторым данным, к этому приложили руку сами Rockstar'овцы..



Как бы там ни было, игра наделала много шума. Про нее ходило немало слухов, появилось много ложных скриншотов и фотографий – якобы она даже продавалась на картриджах. Многие в «GTA на NES» не верили, считая, что это просто фейк.

Но прошло несколько лет, и история получила продолжение: в 2011 году Провинсиано объявился на Independent Games Summit в Сан-Франциско, где анонсировал свой новый проект **Retro City Rampage** – по сути, продолжение идеи Grand Theftendo: игра на ту же тему и с той же «восьмибитной» графикой, но уже под современные платформы. И, естественно, без лицензионных коллизий с пресловутой Rockstar. Игра была выпущена в 2012 году для PlayStation 3, PlayStation Vita и PC, а годом позднее – для Xbox 360. Однако NES'овская версия отнюдь не канула в лету: Провинсиано добавил ее в виде бонуса в Steam-версию Retro City Rampage под именем **ROM City Rampage**. Этот бонусный уровень в игре лежит в виде настоящего ROM-файла и запускается через встроенный эмулятор. В итоге этот исторический прототип оказался в Интернете, и теперь любой может скачать его на сайтах с ROM'ами!

В игре полностью присутствует остров Portland – а это для NES очень даже немало. Город представляет собой карту в 7168x4672 пикселей, при размере людей 8x12. Считается, что это самая большая игра такого жанра для NES из когда-либо созданных – даже больше, чем Metroid.

Вы можете ходить пешком или ездить на автомобиле, свободно перемещаться по городу, стрелять, убивать граждан и полицейских, угонять машины и еще много чего. В игре довольно крупный арсенал оружия. Правда, миссии, в отличие от оригинальной GTA, линейны и идут одна за другой.

Графика выполнена вполне профессионально и со вкусом – у игры в определенном смысле есть своя уникальная стилистика. Конечно, до атмосферы полноценного «симулятора угонщика» времен ранних GTA ей далековато – но и это, по сути, можно назвать шедевром, ведь ничего подобного на 8-битных приставках никогда раньше не выходило!

Второй известный порт-демейк GTA должен был выйти для Sega Mega Drive. О данной игре почти ничего не известно – одно время на разных сайтах ходила информация о картридже и фотографии. Судя по скриншотам на боксе от картриджа, игра сильно смахивает на GTA Advance с платформы GBA, но со значительными различиями. Вполне вероятно, впрочем, что скриншоты и были взяты именно с GBA-версии.

Фархад ГАФАРОВ

Интерес к ретро-играм захватил всех без исключения: в прошлом году Xbox Live провела уникальную в своем роде археологическую кампанию по вскрытию... захоронения игровых картриджей!

Эта «сокровищница» долгое время считалась легендой – началось все в 1982 году, после провала игры Extra-Terrestrial для Atari 2600. Она была признана одной из худших игр в истории. Не совсем ясно, чего хотели руководители компании, но все невостребованные копии этого продукта, согласно легендам, решено было закопать. Естественно, вся информация, связанная с этим, была засекречена и, скорее всего, утеряна.

И только недавно это захоронение наконец-то было найдено – рядом с городом Аламогордо в штате Нью-Мексико. Археологам удалось найти некоторые игры, руководства и геймпады, однако главной своей задачей они считают найти уникальный контроллер Midlink для Atari 2600, управление которым происходит с помощью лобных мышц...

Браузеры и Unix-way

Философия Unix, благодаря которой родилась громадная инфраструктура качественных, продуманных программ и архитектурных решений, предполагает создание небольших программ, взаимодействующих между собой посредством текстовых потоков и сокетов (в противовес разделению памяти и динамической компоновке). Этот принцип простоты и модульной декомпозиции порождает высокий уровень надежности.

Изначально архитектура той части Интернета, которая называется Вебом, прекрасно соответствовала философии Unix. Предками современных веб-приложений были CGI-программы: они просто печатали текст в стандартное устройство вывода, а веб-сервер отвечал за запуск этих программ и отправку текста клиентам по их запросу. Веб до сих пор приблизительно так и работает. Но – только на стороне сервера. Веб-клиент – браузер – давно уже перестал следовать философии Unix и стал эдакой «программой-богом»: ни о каком минимализме и вытекающей из него надежности речь давно уже не идет...

1. «Пусть программа делает что-то одно, но хорошо»

Браузеры нарушают это правило на каждом шагу. Если в эпоху становления Веба они действительно делали что-то одно (а именно – просто осуществляли загрузку и рендеринг HTML-страниц), то сейчас браузер – это одновременно средство веб-серфинга, аудио- и видео-плеер, радио, менеджер загрузок, почтовый клиент, IM-клиент, FTP-клиент, видеочат, игровая платформа и бог знает что еще... При этом каждая из функций, естественно, не доведена до ума и является некой полумерой. Браузер пытается быть всем, но это у него получается из рук вон плохо.

2. «Красиво – небольшое»

Современные браузеры трудно назвать небольшими. Конечно, есть программные пакеты и покрупнее, однако их объемы, как правило, вполне оправданы. Но оправдать ресурсоемкость браузеров...

Это, по сути, программы-монстры, пожирающие оперативную память и захламляющие жесткий диск кэшем. Mozilla Firefox содержит свыше 20 миллионов строк кода – комментарии, думаю, излишни.

3. «Разрабатывайте программы так, чтобы их можно было соединить с другими программами»

Соединить браузер с другими программами? Wut? Раньше это было возможно с использованием плагинов, но теперь политика всех главных браузеров – постепенный отказ от бинарных плагинов в пользу дополнений на основе JavaScript и HTML5.

С одной стороны, мы имеем меньше проблем с безопасностью, но с другой – браузер становится «вещью в себе».

4. «Если можно не добавлять новую функциональность, не добавляйте ее»

Разработчики браузеров уже лет 10 как наращивают функциональность – вместо того, чтобы фиксировать баги в уже существующей.

Классический пример браузера, перегруженного функциональностью – Opera.



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.

5. «Предпочитайте переносимость эффективности»

Переносимость веб-приложений, как ее преподносят интернет-пиарщики, конечно, соблазнительна. Но так ли Веб надежен в качестве стандарта на вычислительную среду для разработки программ? Начнем хотя бы с того, что браузеров существует довольно много, и реализация веб-стандартов в них зачастую сильно различается.

HTML5 реализован не везде и не полностью. WebRTC? WebGL? Забудьте, если вы обладатель старой машины, неспособной потянуть последний Firefox или Chrome. Про трагикомедию с кодеками и H.264 и говорить нечего...

При этом в любой другой программной среде и кодеки, и OpenGL, и любые другие средства, доступные в ОС, задействовать вполне легко на любом аппаратном обеспечении – достаточно только перестать слушать медовые речи и оставить, наконец, в покое JavaScript, признав, что браузеры – далеко не панацея переносимости.

Тимур ГАФАРОВ

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

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.cf>