
Blend4Web. Руководство пользователя

Выпуск 15.04

ООО "Триумф"

29 April 2015

1 Общие сведения	1
1.1 Что такое Blend4Web	1
1.2 “Движок”	1
1.3 Графический движок, трехмерный движок	2
1.4 Что такое WebGL	2
1.5 Поддержка WebGL в браузерах	2
1.6 Преимущества WebGL	3
1.7 Что такое Blender	3
1.8 3D моделирование	3
1.9 Браузерные технологии	4
1.10 Интерактивная графика	5
1.11 Видео-карты и драйверы	5
2 Функционал движка	7
2.1 Общее	7
2.2 Текстуры	7
2.3 Материалы	7
2.4 Освещение	8
2.5 Тени	8
2.6 Система частиц	8
2.7 Рендеринг наружных сцен	9
2.8 Постпроцессинговые эффекты	9
2.9 Анимация	9
2.10 Оптимизация	9
2.11 Звук	10
2.12 Физика	10
2.13 Событийная модель	11
2.14 Визуальное программирование	11
2.15 Прочее	11
3 Экспресс-установка	12
3.1 Установка программы Blender	12
3.2 Установка аддона движка	13
3.3 Экспорт и просмотр сцены	14

3.4 Установка новой версии аддона	14
4 Развёртывание среды разработки	16
4.1 Установка дистрибутива	16
4.2 Подключение аддона	16
4.3 Локальный сервер разработки	19
4.4 Запуск просмотрщика сцен и демо-приложений	21
5 Рабочий процесс	23
5.1 Подготовка сцен	23
5.2 Экспорт сцен	24
5.3 Отображение сцен в просмотрщике	24
5.4 Разработка приложения	27
6 Просмотрщик сцен	28
6.1 Навигация	28
6.2 Боковая панель	28
6.3 Индикаторы	31
7 Веб-плеер	32
7.1 Использование	33
7.2 Навигация	33
7.3 Панель управления	33
7.4 Атрибуты	34
7.5 Название сцены в заголовке	35
7.6 Ошибки сцены	35
8 Аддон	38
8.1 Опции экспорта	38
8.2 Ошибки инициализации	42
8.3 Ошибки совместимости	43
8.4 Критические ошибки экспорта	44
8.5 Некритические ошибки экспорта	48
8.6 Прочие сообщения	51
9 Объекты	53
9.1 Типы	53
9.2 Статические и динамические объекты	53
9.3 Настройка	54
9.4 Управление перемещением объектов	59
9.5 Кватернионы	60
9.6 Перемещение с помощью векторов TSR	62
9.7 Копирование объектов (инстансинг)	63
9.8 Выделение объектов	64
9.9 Морфинг	64
10 Камера	66
10.1 Режимы управления и общие настройки	66
10.2 Скорость движения камеры	67

10.3 Ограничения движения камеры	67
10.4 Управление камерой через API	74
11 Материалы	76
11.1 Параметры освещения	77
11.2 Прозрачность	78
11.3 Зеркальное отражение	80
11.4 Специальные параметры движка	82
11.5 Материалы гало (Halo)	84
12 Текстуры	86
12.1 Типы текстур	86
12.2 Общие настройки	87
12.3 Диффузная текстура (diffuse map)	88
12.4 Карта бликов (specular map)	88
12.5 Карта нормалей (normal map)	89
12.6 Карта высот (height map). Parallax mapping	89
12.7 Карта смешивания (stencil map)	90
12.8 Видео-текстуры	92
12.9 Карта окружения (environment map)	95
12.10 Карта зеркального отражения (mirror map)	96
12.11 Текстура неба (skydome)	97
12.12 Особые типы текстур	98
13 Нодовые материалы	102
13.1 Стандартные ноды	102
13.2 Дополнительные ноды	103
14 Освещение, тени и фон	109
14.1 Типы затенения	109
14.2 Редактор нормалей	110
14.3 Освещение от источников света	116
14.4 Освещение от окружающей среды	117
14.5 Тени	119
14.6 Цвет фона	125
15 Постпроцессинговые эффекты	128
15.1 Размытие при движении	128
15.2 Глубина резкости камеры	129
15.3 Взаимное затенение	130
15.4 Сумеречные лучи	132
15.5 Эффект засветки ярких деталей	133
15.6 Подсветка контура (outline glow)	134
15.7 Анализ стереоизображение	135
15.8 Коррекция цвета	136
15.9 Сглаживание	137
16 Система частиц. Флюиды	138
16.1 Использование	139

16.2 Настройка	140
16.3 Текстуры в системах частиц	144
17 Система частиц. Инстансинг	146
17.1 Настройки системы частиц	147
17.2 Травяной покров	150
17.3 Листья деревьев	152
18 Анимация	155
18.1 Управление анимацией	155
18.2 Объектная анимация	156
18.3 Скиннинг и скелетная анимация	157
18.4 Вертекская анимация	159
18.5 Параметризация источников звука	160
18.6 Анимация ноды Value	161
19 Рендеринг наружных сцен	162
19.1 Вода	162
19.2 Атмосфера	172
19.3 Ветер	175
20 Гамма-коррекция и альфа-композитинг	178
20.1 Общее описание	178
20.2 Человеческое зрение и мониторы	179
20.3 Гамма	180
20.4 Коррекция в нодовых материалах	180
20.5 Альфа-композитинг	181
21 Звуковая подсистема	182
21.1 Настройка звуковых источников	182
21.2 Обработка и кодирование	185
22 Физика	186
22.1 Подготовка к использованию	186
22.2 Статический тип физики	187
22.3 Динамический тип физики	189
22.4 Ограничители (Constraints)	192
22.5 Колесные транспортные средства	193
22.6 Плавающие объекты	196
22.7 Плавающие транспортные средства	197
22.8 Особенности использования в приложениях	198
23 Нелинейная анимация	199
23.1 Редактор NLA	199
23.2 Визуальное программирование (NLA Script)	201
24 Разработчикам приложений	206
24.1 Hello world!	206
24.2 Загрузка сцены в приложение	207
24.3 Быстрое создание приложений	207

24.4	Разработка приложений в составе SDK	208
24.5	Примеры кода	209
24.6	Конвертация ресурсов	210
24.7	Система модулей	212
24.8	Событийная модель	212
24.9	Профили качества изображения	214
24.10	Файловая структура SDK	216
24.11	Текстура типа Canvas	218
24.12	Настройка браузера для загрузки локальных ресурсов	218
24.13	Запуск локального сервера	221
25	Разработчикам движка	223
25.1	Стиль оформления кода	223
25.2	Сборка движка	224
25.3	Сборка аддона	225
25.4	Зависимости	225
25.5	Названия функций и переменных	226
25.6	Отладка	227
25.7	Компиляция шейдеров	227
26	Работа в команде с использованием Git	234
26.1	Общие сведения	234
26.2	Типичный рабочий процесс	234
26.3	Индивидуальные настройки	235
26.4	Проверка статуса	235
26.5	Перед коммитом	236
26.6	Подготовка к коммиту	237
26.7	Коммит	238
26.8	Синхронизация между репозиториями	239
26.9	Разрешение конфликтов	241
26.10	Тэги	243
26.11	Другие полезные команды	244
27	Проблемы и решения	245
27.1	Проблемы при запуске движка	245
27.2	Ошибка инициализации WebGL	246
28	Замечания к релизам	250
28.1	v15.04	250
28.2	v15.03	254
28.3	v15.02	258
28.4	v15.01	261
28.5	v14.12	265
28.6	v14.11	267
28.7	v14.10	269
28.8	v14.09	271
Алфавитный	указатель	275

Общие сведения

1.1 Что такое Blend4Web

Blend4Web - это программная среда для подготовки и интерактивного отображения трехмерного аудиовизуального контента в браузерах, т.е. трехмерный «движок» (жаргонизм от англ. «engine»).

Платформа предназначена для создания визуализаций, презентаций, интернет-магазинов, игр и других “насыщенных” web-приложений.

Фреймворк Blend4Web имеет тесную интеграцию с пакетом 3D моделирования и анимации Blender (отсюда название). Отображение контента осуществляется средствами WebGL и других браузерных технологий, без использования плагинов.

Технически Blend4Web представляет собой программную библиотеку, подключающуюся в web-странице, аддон (дополнение) к программе Blender, а также инструменты для отладки и оптимизации.

3D движок Blend4Web разрабатывается сотрудниками ООО «Триумф» с 2010 г. Первый релиз движка состоялся 28 марта 2014 г.

1.2 “Движок”

Движок - это обособленная часть программного кода, используемая внешними приложениями для реализации нужного им функционала.

Типы движков: сайтовый движок, блоговый движок, движок интернет-магазина, wiki-движок, поисковый движок, игровой движок и т.д. Экономический смысл существования программных движков заключается в многократном использовании одного и того же функционала. Например, используя тот или иной движок, разработчики могут относительно дешево создавать интернет-магазины или игры.

1.3 Графический движок, трехмерный движок

Графический движок выполняет специализированные функции по отображению графики. Он является промежуточным звеном между:

- высокоуровневой прикладной частью (игровой логикой, бизнес-логикой) и
- низкоуровневой системной частью (например, графической библиотекой *WebGL* и нижлежащими *драйверами*).

Графический движок может объединяться со звуковой системой, физическим движком, системой, реализующей искусственный интеллект, сетевой системой, а также редактором сцен и логики, образуя интегрированный инструментарий для создания 3D приложений - **трехмерный движок**.

1.4 Что такое WebGL

WebGL (Web Graphics Library, т.е. графическая библиотека для использования в веб-приложениях) - одна из современных браузерных технологий, позволяющая создавать трехмерные графические веб-приложения. Другими словами WebGL - это “3D в браузере”.

1.5 Поддержка WebGL в браузерах

В настоящий момент технология WebGL в той или иной степени поддерживается во всех браузерах.

1.5.1 Полная поддержка

- Yandex Browser
- Chrome
- Firefox
- Opera
- Safari 8+

1.5.2 Экспериментальная поддержка

- Internet Explorer 11

1.5.3 Мобильные платформы

- Android (на современных устройствах)
- BlackBerry
- Firefox OS
- iOS 8
- Tizen
- Ubuntu Touch
- WebOS

1.6 Преимущества WebGL

- работает в браузерах без установки дополнительных программ (плагинов)
- кроссплатформенный, предназначен для работы во всех стационарных и мобильных системах
- является [открытым стандартом](#), не требует лицензионных отчислений
- поддерживается ведущими участниками рынка IT (Google, Apple, Microsoft, Nvidia, Samsung, Adobe и др.)
- основан на знакомой разработчикам технологии OpenGL
- интегрируется с другими *браузерными технологиями*

1.7 Что такое Blender

Blender - это популярная программа для создания 3D-моделей и анимации, бесплатная и с открытым кодом. Подготовленные в программе модели и сцены могут быть отображены, например, с помощью [трехмерного движка](#) на странице веб-сайта.

1.8 3D моделирование

Создание графических ресурсов требует наличия подготовленных специалистов - 3D-художников.

Типичный рабочий процесс может состоять из следующих этапов:

- подбор фотографий и/или создание концепта и скетчей (“спереди”-“сбоку”-“сверху”) будущей модели или сцены
- моделирование - создается трехмерная модель, состоящая из многоугольников (полигонов)

- текстурная развертка - на модели создается разметка для последующего наложения текстур (плоских изображений)
- текстурирование - на 3D-модель накладываются текстуры
- подбор материалов - назначение различным частям модели материалов и их настройка (например, деревянная дверь с металлической ручкой)
- риггинг (от англ. rigging, т.е. “оснастка”) - к модели прикрепляются управляющие элементы (“кости” “скелета”) с целью дальнейшей анимации
- анимация - модель приводится в движение с целью визуализации действий (например, персонажей)
- экспорт - выполняется на любом этапе с целью отображения 3D-модели в ее конечном виде, например, на веб-странице

Кроме того, в процессе создания 3D-моделей часто используются техники повышения реализма, требующие отдельных этапов:

- создание высокополигональной модели - создается детализированная версия модели
- “запекание” карты нормалей - детали из высокополигональной модели переносятся на основную модель в виде специальной текстуры (карты нормалей)
- создание карты отражения - различным частям модели назначается различный цвет и степень отражения света
- запекание карт окружения - производится с целью реализации эффекта отражения окружающей среды на поверхности модели
- настройка камеры и источников света на сцене
- настройка параметров физической симуляции - частицы, ткань

Затраты времени при изготовлении 3D-моделей и анимации зависят от их сложности и требуемого качества, и могут изменяться от 1-2 дней (например, игровой предмет) до 1-2 недель (например, детализированная модель самолета) и даже нескольких месяцев (реалистичные персонажи с наборами одежды, волос, лиц, с анимацией и настройкой пропорций фигуры).

1.9 Браузерные технологии

Браузер (от англ. “browser”, т.е. “просмотрщик”) - программа для воспроизведения содержимого сети Интернет. На заре развития интернет-технологий роль браузера сводилась к просмотру текстовых страниц с включениями статических изображений (“гипер-текст”). Современные браузеры представляют собой полнофункциональные платформы для создания мультимедийных веб-приложений.

Среди реализованных и перспективных возможностей браузеров, используемых в *Blend4Web*, можно отметить следующие технологии:

- трехмерная графика, *WebGL*

- типизированные массивы, [Typed Array](#)
- временной контроль анимации (`requestAnimationFrame`), [Timing control for script-based animations](#)
- двухмерная графика, [HTML Canvas 2D Context](#)
- обработка звука, [Web Audio API](#)
- загрузка бинарных данных, [XMLHttpRequest Level 2](#)
- полноэкранный режим, [Fullscreen](#)
- захват курсора мыши, [Pointer Lock](#)
- многопоточные вычисления, [Web Workers](#)
- ориентация и перемещение мобильных устройств, [Device Orientation](#)

Другие перспективные технологии:

- векторная графика, [Scalable Vector Graphics \(SVG\)](#)
- безопасный доступ к файлам, [File API](#), [File API: Directories and System](#)
- потоковое соединение между браузерами, [WebRTC](#)
- постоянное сетевое подключение, [The WebSocket API](#)
- игровые пульты, [Gamepad](#)

1.10 Интерактивная графика

Термин “интерактивный” в приложении к компьютерной графике означает, что пользователь имеет возможность взаимодействовать с постоянно меняющимся изображением. Например, пользователь может изменять направление взгляда в 3D сцене, перемещать объекты, инициировать анимацию и выполнять другие действия, обычно ассоциирующиеся с компьютерными играми.

Интерактивность графики достигается за счет частой смены изображений, так что действие пользователя (например, движение курсора или нажатие кнопки) в промежутках между кадрами приводит к изменению изображения в следующем кадре. Изображения должны сменять друг друга так часто, чтобы человеческий глаз не был способен распознать их по отдельности (быстрее 30 кадров в секунду).

Близким по смыслу термином является также “графика реального времени”, или “рендеринг реального времени” (от англ. `rendering`, т.е. “отображение”).

1.11 Видео-карты и драйверы

Интерактивная графика реализуется специализированной аппаратной частью современных компьютеров, называемой графическим процессором, который может быть

выполнен в виде отдельного устройства (видео-карты) или как часть центрального процессора.

Основные производители графических процессоров (в скобках указаны их торговые марки), для настольных компьютеров - NVidia (GeForce, Quadro), AMD (Radeon), Intel (HD), для мобильных устройств - ARM (Mali), PowerVR (SGX), Nvidia (Tegra), Qualcomm (Adreno).

Доступ программ к ресурсам графического процессора осуществляется через программу-посредника, называемого драйвером. Важным условием для корректной работы интерактивных графических программ является наличие в системе драйверов последней версии. Драйверы можно установить (или обновить), загрузив их с соответствующих сайтов производителей графических процессоров. Подробнее в разделе *Ошибка инициализации WebGL*.

Функционал движка

2.1 Общее

- эффективный рендеринг трёхмерных сцен любой сложности и размеров
- формат хранения данных, учитывающий специфику работы WebGL и оптимизированный с целью минимизации времени загрузки
- удобная среда разработки трёхмерного контента на основе плагина к Blender

2.2 Текстуры

- текстурирование (texturing) - покрытие поверхности 3D объекта плоским изображением
- мультитекстурирование (multitexturing) - использование для объекта нескольких текстур
- рендеринг в текстуру (render-to-texture, RTT) для реализации вложения одной сцены в другую и постпроцессинговых эффектов
- анизотропная фильтрация для улучшения качества обзора поверхности под косыми углами (anisotropic filtering, AF, использовано стандартное расширение WebGL)
- поддержка текстурной компрессии (формат S3TC/DXT)
- видео-текстуры - загрузка и воспроизведение видео на текстуре
- Canvas-текстуры - возможность отрисовки двухмерной графики через Canvas API в текстуре

2.3 Материалы

- прозрачность материалов, сортировка по глубине при необходимости (z-sorting)

- улучшенная детализация рельефных поверхностей текстурами (использован метод parallax offset mapping)
- зависимость степени отражения от угла обзора - эффект Френеля (Fresnel)
- динамическое отражение
- поддержка нодовых материалов
- гало материал для рендеринга источников света и звезд (Halo material)

2.4 Освещение

- освещение несколькими источниками света
- типы источников света - прямой (directional), полусферический (hemisphere), точечный (point), конический (spot)
- диффузное (т.е. рассеянное) освещение объектов (diffuse lighting)
- рассеянное освещение от окружающей среды (ambient lighting)
- зеркальное отражение света от поверхности объектов (specular lighting)
- зеркальное отражение окружающей среды (environment mapping)
- дополнительная детализация картами нормалей к поверхности (normal mapping)

2.5 Тени

- статические падающие тени (light mapping)
- динамические падающие тени (использован метод shadow mapping)
- собственные тени — объекты отбрасывают тени сами на себя (self-shadowing)
- каскадные тени для больших сцен (cascaded shadow mapping, CSM)
- мягкие тени

2.6 Система частиц

- система частиц (particle system) для реализации эффектов, таких как огонь, дым, брызги и т.д.
- система частиц для расстановки (инстансинга) однородных объектов: трава, камни, листва деревьев и проч.

2.7 Рендеринг наружных сцен

- туман (fog)
- купол неба/окружающего пространства (skydome)
- эффект линз при направлении камеры на источник света (lens flares)
- рендеринг воды

2.8 Постпроцессинговые эффекты

- размытие при движении (motion blur)
- антиалиасинг - уменьшение зубчатости краев изображения в результате рендеринга в текстуру (использован метод fast approximate anti-aliasing, FXAA)
- стерео-изображение (анаглифное, 3D очки)
- рассеянное затенение от окружающей среды (ambient occlusion, используется метод SSAO)
- глубина резкости для камеры (DOF)
- сумеречные лучи (god rays)
- эффект засветки ярких объектов (bloom)
- подсветка контуров объектов (outlining)

2.9 Анимация

- сканинг (skinning) - деформация объекта с помощью системы костей
- анимация перемещения, вращения, масштабирования объектов, камер и источников света
- скелетная анимация (например, для тела персонажа)
- вертексная анимация (например, для симуляции ткани)
- процедурная анимация (например, изгибание растений на ветру)
- анимация текстурных координат (например, для визуализации волн воды)

2.10 Оптимизация

- оптимизация отсечением по зоне видимости (frustum culling)

- оптимизация уменьшением количества вызовов WebGL — батчинг, текстурные атласы (batching, texture atlases)
- оптимизация уменьшением уровня детализации объектов на удалении (level of detail, LOD)

2.11 Звук

- звуковой движок, основанный на Web Audio API
- поддержка различных форматов файлов с учётом различий браузеров
- гибкое управление воспроизведением, возможность приостановки звука
- позиционирование источников в трёхмерном пространстве
- эффект Допплера для движущихся объектов с возможностью отключения и компенсацией скачков в пространстве
- управление громкостью, скоростью и задержкой воспроизведения
- эффекты плавного перехода громкости (fade-in, fade-out, duck)
- качественное зацикливание звуков (looping)
- рандомизация звуковых параметров для улучшения восприятия повторяющихся звуков
- поддержка кроссфейдерной звуковой анимации
- динамический компрессор
- эффективное хранение и воспроизведение длинных музыкальных композиций
- инструменты для сведения (микширования) звуковой картины в реальном времени

2.12 Физика

- физика жестких тел - определение столкновений, движение, гравитация, определение высоты, опрокидывание
- система соединителей (ограничителей) - жёсткие, гибкие, пружинящие, поворотные, скользящие итд.
- система трассировки лучей
- физика плавания объектов и движения в толще воды
- физика колёсных транспортных средств
- физика плавучих транспортных средств

2.13 Событийная модель

- асинхронный фреймворк для написания логики приложений
- управление анимацией и искусственный интеллект животных и персонажей

2.14 Визуальное программирование

- инструмент NLA Script предоставляет возможность создавать интерактивные приложения, конструируя логические цепочки из простых блоков

2.15 Прочее

- поддержка математических кривых для моделирования удлиненных объектов (дороги, провода, река)
- выбор пользователем объектов на 3D сцене (picking)
- минификация (уменьшение объема) и обfuscация (сокрытие) кода, необходимые для коммерческого использования движка
- модульная структура исходного кода
- мощный шейдерный препроцессор с поддержкой модулей и функциональных блоков (нод)
- удобная система для быстрого развертывания новых 3D приложений
- опции для поддержки работы на широком спектре оборудования
- руководство пользователя и документация для программистов
- взаимодействие с пользователем — управление камерой, персонажем, действиями

Экспресс-установка

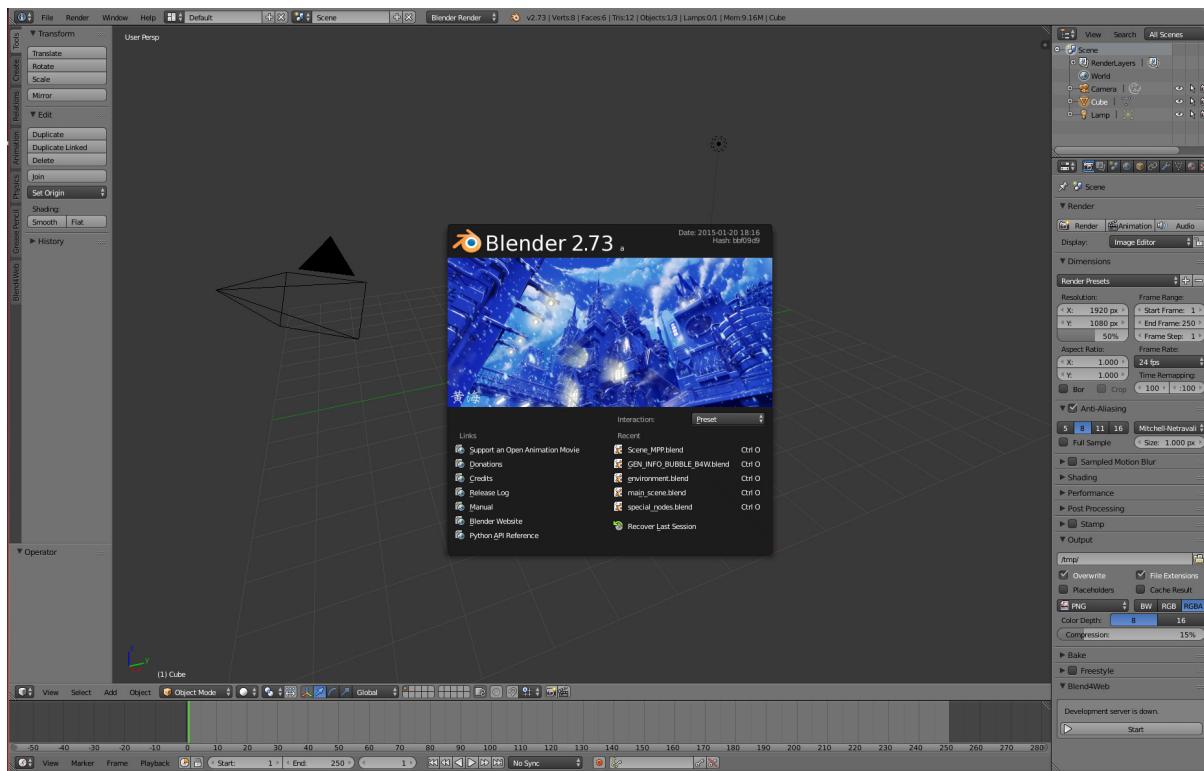
Экспресс-установка аддона Blend4Web подойдет обычным пользователям, которым не требуется разработка полноценных 3D приложений. Основной возможностью в этом случае является экспорт сцены в один HTML-файл с последующим просмотром в браузере, поддерживающем WebGL.

Для более серьезных задач необходима установка *комплекта средств разработки*.

3.1 Установка программы Blender

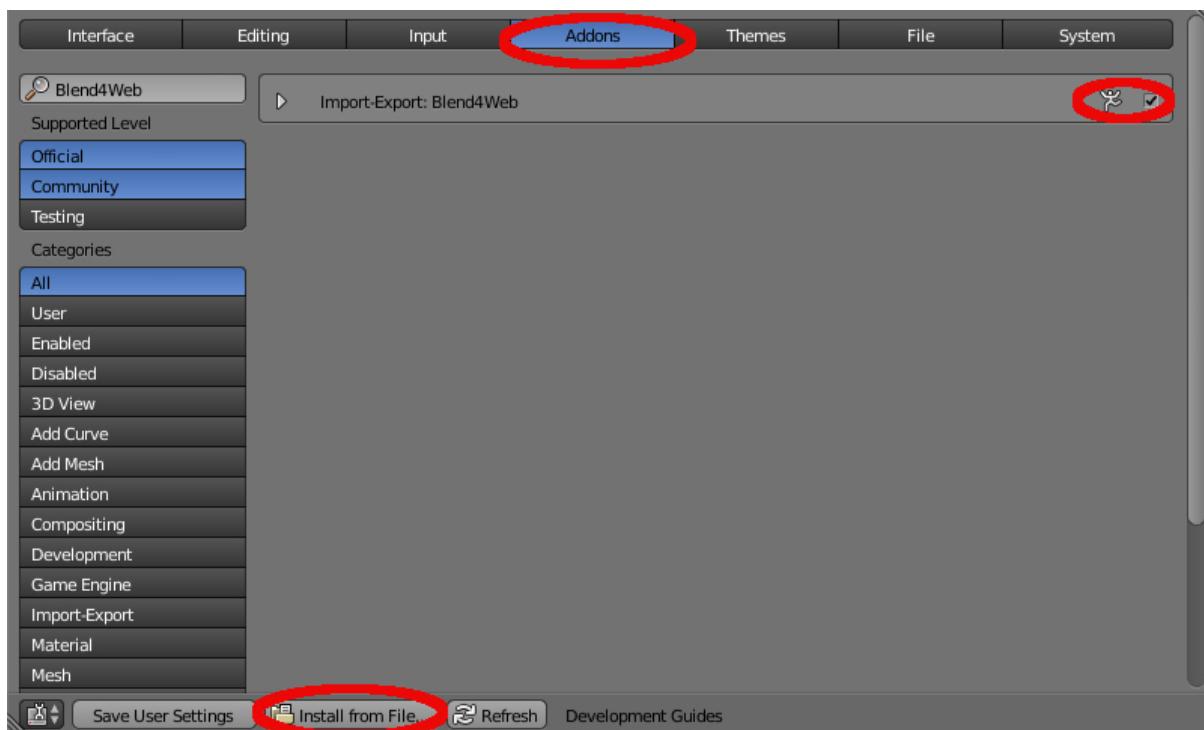
Создание 3D сцен осуществляется в графическом пакете Blender, который является программным продуктом с открытым исходным кодом и распространяется бесплатно.

Должна использоваться текущая стабильная версия Blender. Загрузить Blender можно с [официального сайта](#).



3.2 Установка аддона движка

Запустить Blender, загрузить сцену по умолчанию **File > New**. Вызвать окно пользовательских настроек **File > User Preferences....** Во вкладке **Addons** нажать **Install from File...** и затем выбрать zip-архив с файлами аддона. После этого необходимо отметить галочку напротив **Import-Export: Blend4Web**.



Далее нажать `Save User Settings` и закрыть окно пользовательских настроек.

3.3 Экспорт и просмотр сцены

Созданную сцену можно экспортировать в формате HTML. Для этого нужно выбрать опцию `File > Export > Blend4Web (.html)` и указать путь экспорта. Полученный HTML файл можно открыть любым браузером, поддерживающим технологию WebGL.

См.также:

[Поддерэска WebGL в браузерах](#)

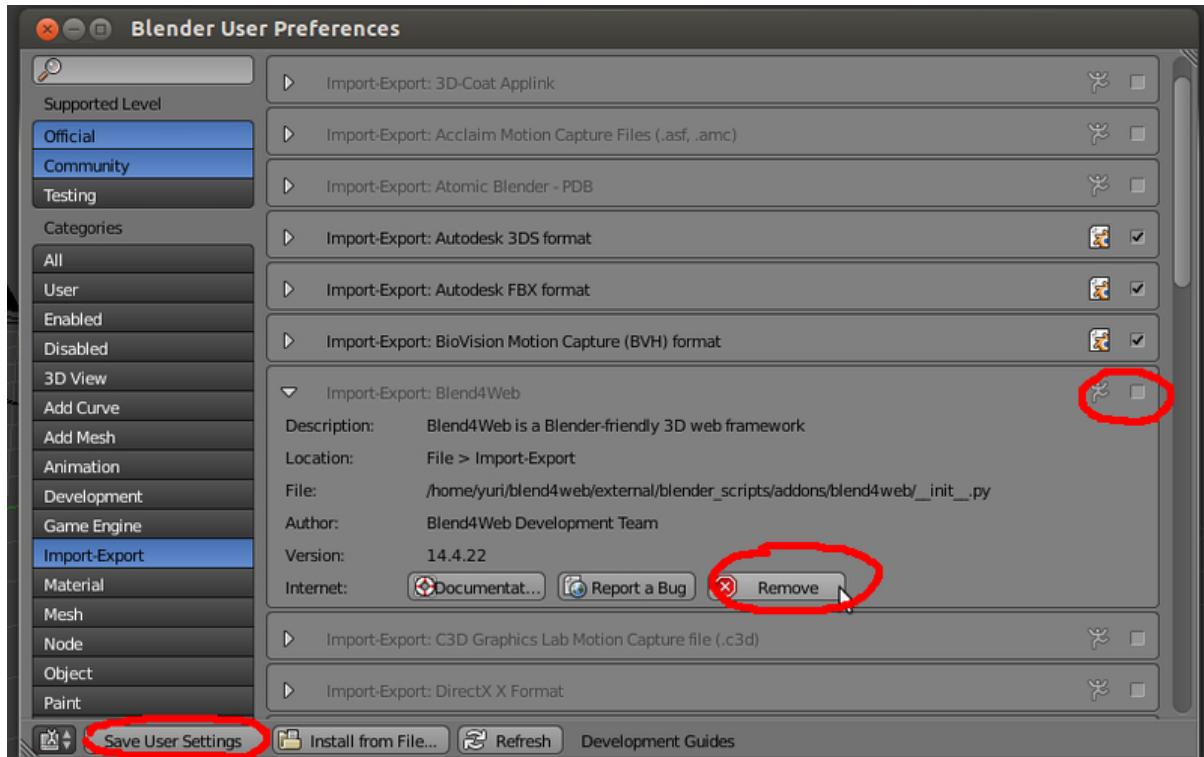
3.4 Установка новой версии аддона

Чтобы установить новую версию аддона, необходимо **сначала отключить, а затем удалить** старую версию.

Для отключения аддона запустить Blender, загрузить сцену по умолчанию `File > New`. Вызвать окно пользовательских настроек `File > User Preferences...`. Перейти на вкладку `Addons` и выбрать категорию `Import-Export`. Снять галочку напротив

Import-Export: Blend4Web. Далее нажать Save User Settings и перезапустить Blender.

Затем для удаления аддона снова вызвать окно пользовательских настроек, раскрыть панель информации об аддоне Blend4Web и нажать кнопку Remove.



Развёртывание среды разработки

Вариант установки с развёртыванием дистрибутива SDK рекомендуется для разработчиков 3D-приложений. Если же нужно быстрое ознакомление с возможностями Blend4Web, то, возможно, будет удобнее [экспресс-установка](#).

Для работы необходим дистрибутив движка, браузер с поддержкой WebGL и Blender.

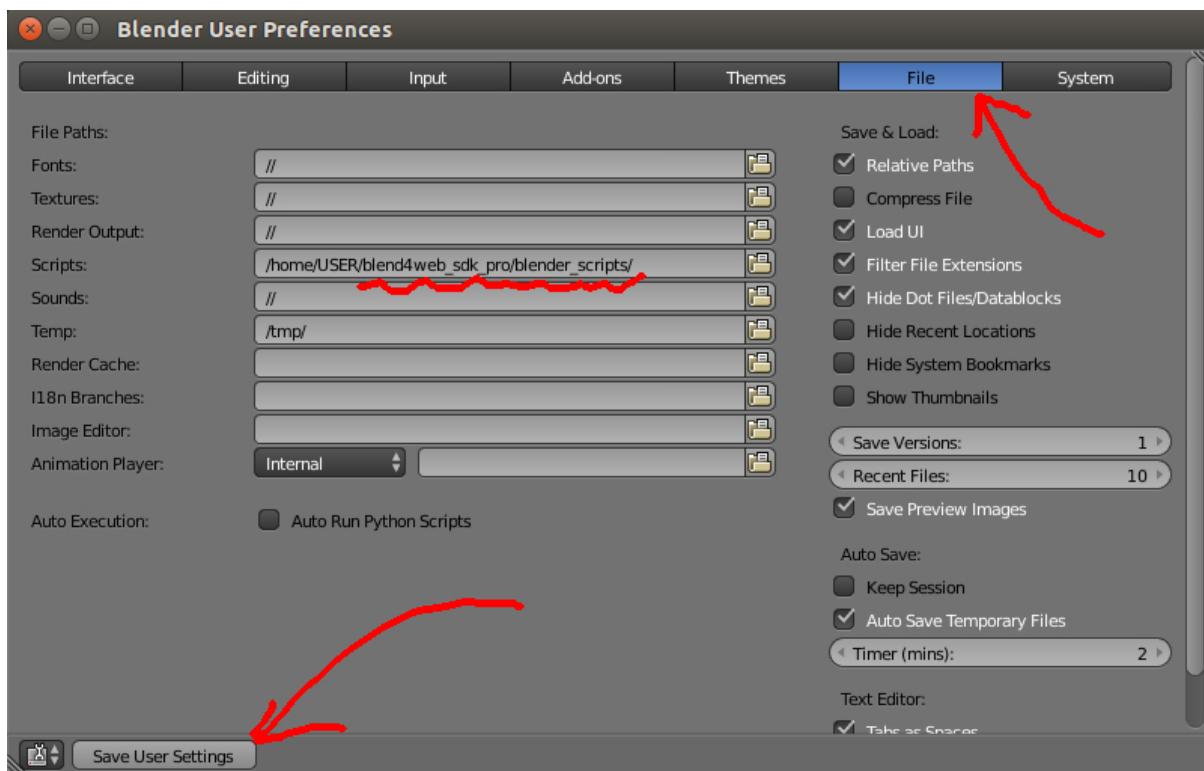
4.1 Установка дистрибутива

Стабильные версии дистрибутива поставляются в виде архива (`blend4web_sdk_free_YY_MM.zip` – бесплатный SDK, `blend4web_sdk_pro_YY_MM.zip` – коммерческий SDK). Для установки достаточно распаковать данный архив в любое место на диске.

4.2 Подключение аддона

Примечание: Если аддон ранее был установлен с помощью [экспресс-установки](#), то рекомендуется прежде его удалить.

Запустить Blender, загрузить сцену по умолчанию `File > New` (горячие клавиши `Ctrl-N`). Вызвать окно пользовательских настроек `File > User Preferences...` (горячие клавиши `Ctrl-Alt-U`). Во вкладке `File` в поле `Scripts` выбрать путь к директории `blender_scripts`.

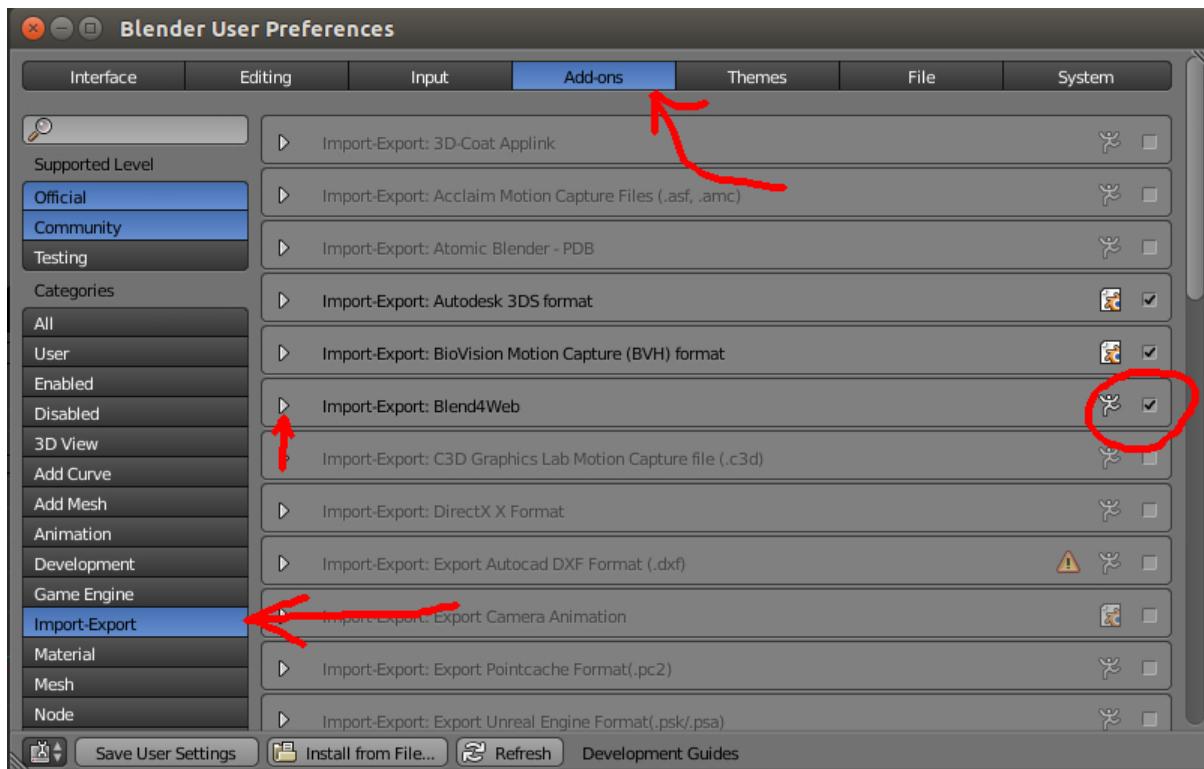


Нажать **Save User Settings** и **перезапустить** Blender.

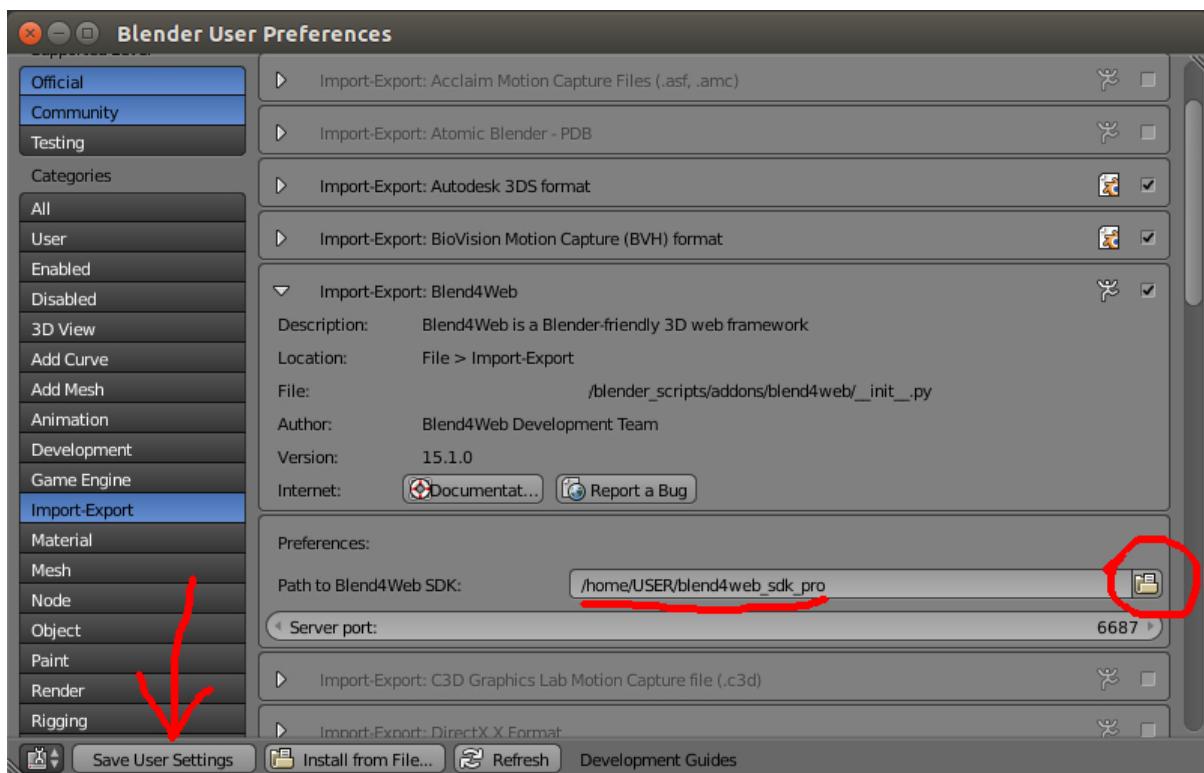
Примечание: Вместо этого можно скопировать директорию со скриптами `blender_scripts/addons/blend4web` в уже используемую пользовательскую директорию для скриптов или даже в установочную директорию, например:

`C:\Program Files\Blender Foundation\Blender\2.70\scripts\addons\blend4web.`

Повторно загрузить сцену по умолчанию, вызвать окно пользовательских настроек, перейти на вкладку **Addons** и выбрать категорию **Import-Export**. Отметить галочку напротив **Import-Export: Blend4Web**.



Раскрыть вкладку аддона и указать путь к директории с SDK в поле “Path to Blend4Web SDK”. Здесь также можно изменить номер порта, используемый при создании локального сервера разработки (6687 по умолчанию).



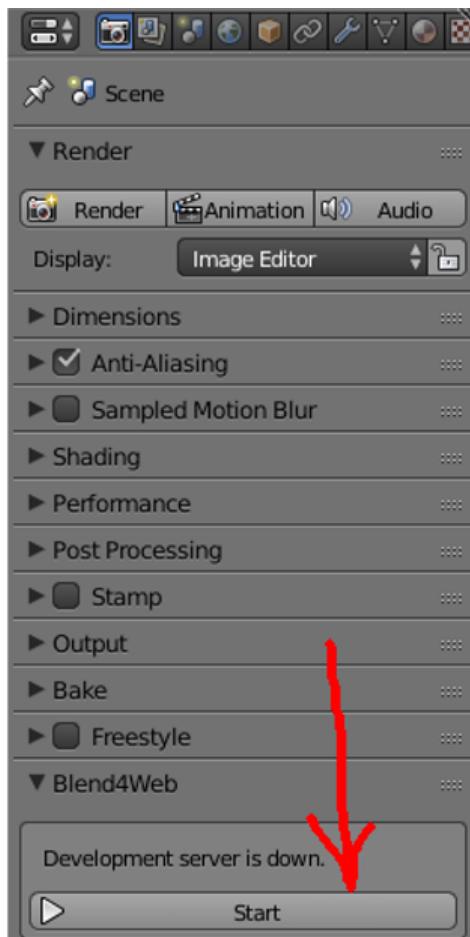
Нажать **Save User Settings**. Перезапуск Blender не требуется.

Для проверки:

В меню **File > Export** должны появиться опции **Blend4Web (.json)** и **Blend4Web (.html)**. Кроме того должны появиться операторы при выполнении поиска по “B4W” (горячая клавиша **ПРОБЕЛ**).

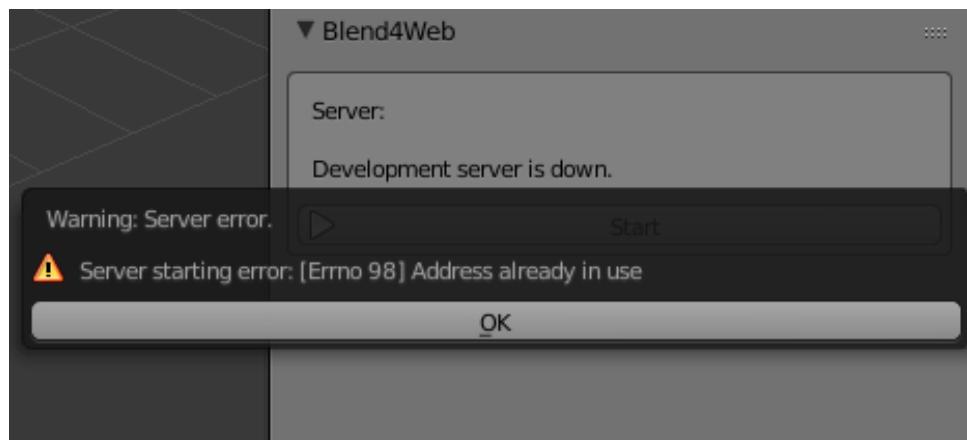
4.3 Локальный сервер разработки

Для запуска сервера перейдите во вкладку **Render** и нажмите кнопку **Start** на панели Blend4Web:

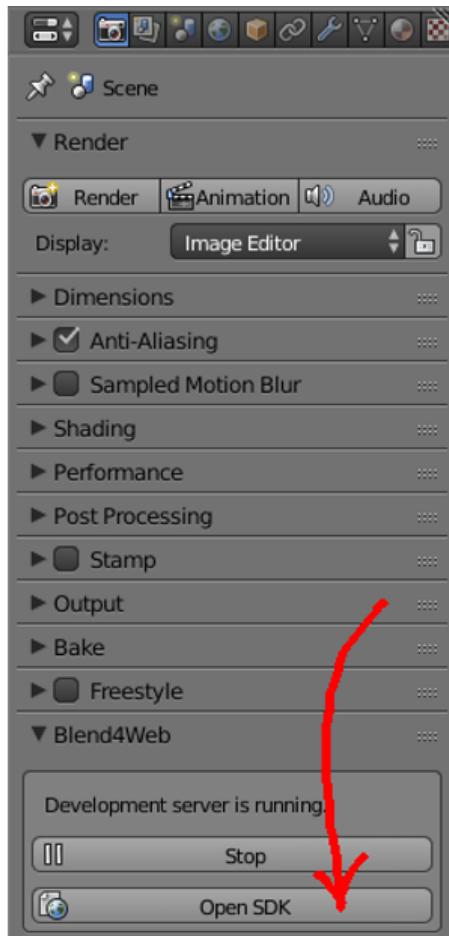


Примечание: Если путь к Blend4Web SDK не был указан, запуск локального сервера невозможен. При этом вместо кнопки **Start** будет выведено соответствующее сообщение.

В случае ошибки запуска сервера будет показано окно с описанием причины:



Нажмите кнопку **Open SDK** для запуска в браузере обзорной веб-страницы Blend4Web SDK, доступной по адресу <http://localhost:6687>.

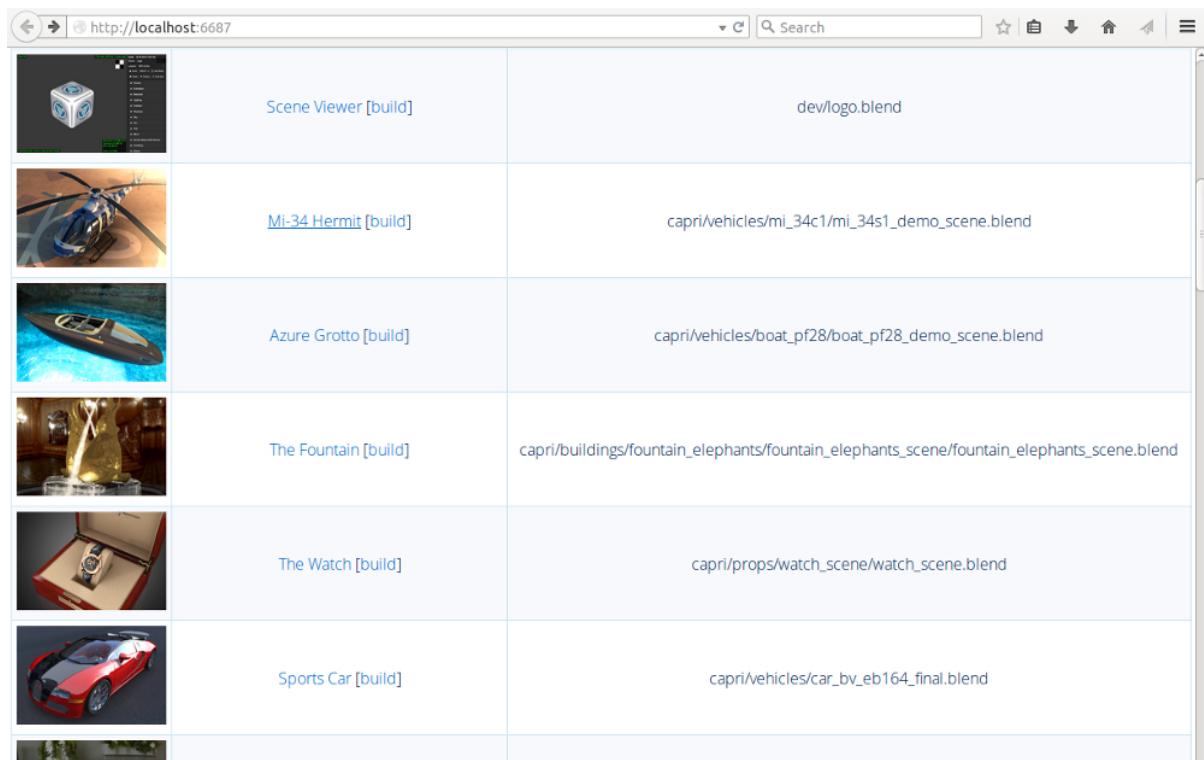


При этом будет автоматически запущен браузер, используемый в операционной системе по умолчанию.

Сервер может быть остановлен по нажатию на кнопку **Stop**. Сервер также прекращает работу при закрытии программы Blender.

4.4 Запуск просмотрщика сцен и демо-приложений

На обзорной странице имеются ссылки для запуска просмотрщика сцен и демо-приложений. Для их запуска необходим *браузер с поддержкой WebGL*.



Примечание: Если приложения SDK не отображаются корректно, или появляются сообщения об ошибках, необходимо предпринять действия, описанные в разделе *Проблемы при запуске движка*.

Рабочий процесс

Создание любого продукта является творческим процессом, в котором могут участвовать множество людей, с различными навыками и опытом. Однако вне зависимости от его сложности и конечного результата, всегда можно выделить стадию производства, на которой создаётся основной объём ресурсов (ассетов) и исходного кода.

При использовании Blend4Web, производственный процесс можно представить следующим образом:

1. Подготовка трёхмерной сцены в программе Blender.
2. Экспорт ресурсов в формате, пригодном для использования движком.
3. Запуск, настройка и отладка сцены в программе-просмотрике.
4. Создание целевого приложения.

5.1 Подготовка сцен

Помимо обычных операций по моделированию, текстурированию, анимации и т.д. должна быть осуществлена подготовка сцены для работы в движке.

Общие рекомендации:

1. Blend-файлы должны находиться в директории `blender/имя_проекта`.
2. Файлы текстур и звуков должны быть внешними и находиться в директории `deploy/assets/имя_проекта`.
3. Вспомогательные файлы, не предназначенные для загрузки в движок (например, референсы), должны находиться в директории `blender/имя_проекта`.
4. Файл, из которого будет осуществляться экспорт, должен содержать только необходимые в разрабатываемом приложении модели.
5. Объект, меш, материал, текстура, арматура должны иметь отличающие названия (на англ. языке). Они не должны называться “Cube.001”, “Material”, “Armature”.

6. Допускается добавление по ссылке (linking) компонентов из других файлов (библиотек).

5.2 Экспорт сцен

Для загрузки сцен, созданных с помощью пакета Blender, в движок, необходимо перевести их в формат, пригодный для чтения браузером. На данный момент используются текстовые файлы с расширением `.json`, в которые сохраняются экспортируемые структуры данных в формате JSON (JavaScript Object Notation). Данный файл, в свою очередь, ссылается на один бинарный файл с расширением `.bin`, содержащий массивы данных моделей, и на внешние ресурсы - текстуры и звуковые сэмплы.

Файлы `.json` и `.bin` создаются при экспорте, файлы текстур и звуков, как правило, должны быть размещены вручную (имеется исключение: внедренные в `.blend`-файл ресурсы размещаются автоматически).

Экспорт производится выбором в меню `File > Export` опции `Blend4Web (.json)`. Быстрый доступ - поиск по `b4w export` (горячая клавиша ПРОБЕЛ).

Экспортные файлы рекомендуется размещать в директории, предназначеннной для развертывания приложения, например `deploy/assets/имя_проекта`.

Необходимо использовать относительные пути для изображений (как правило, это происходит по умолчанию). В случаях, когда это не так, необходимо выполнить команду `File > External Data > Make All Paths Relative` (т.е. сделать все пути относительными). Использование абсолютных путей вместо относительных может приводить к ошибкам при попытках загрузки `.blend` и `.json` файлов на других компьютерах.

В момент экспорта происходит проверка сцены на предмет использования не поддерживаемых движком возможностей Blender'a. В таких случаях генерируется сообщение об ошибке. Перечень возможных ошибок экспорта перечислен в [соответствующем разделе](#).

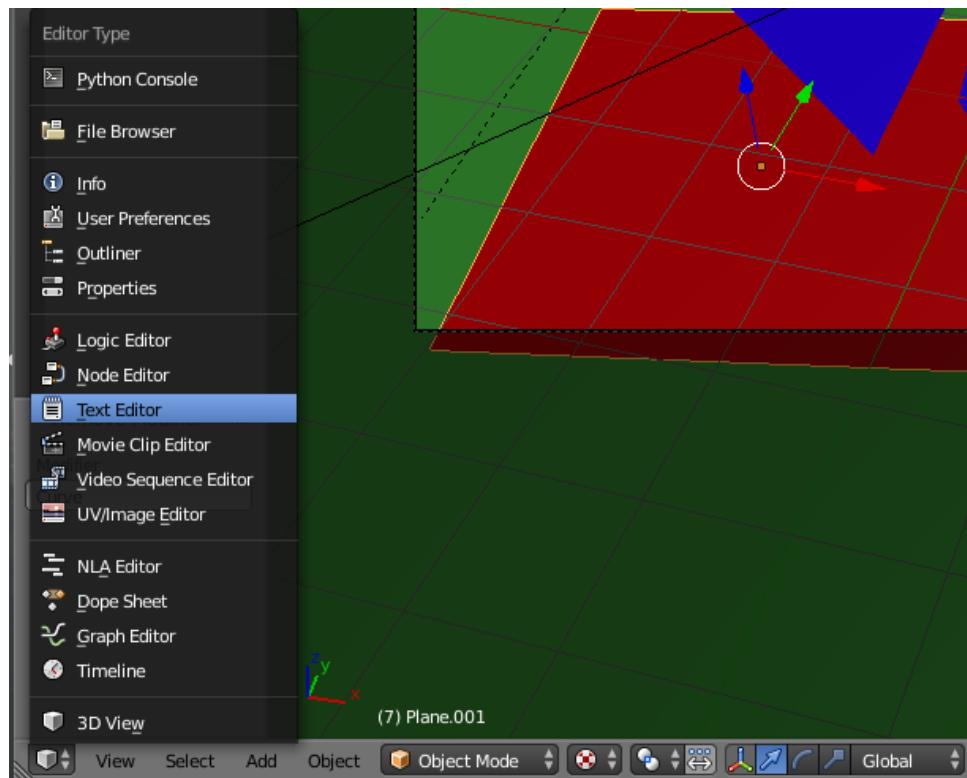
Опции экспорта подробно описаны в [соответствующем разделе](#).

5.3 Отображение сцен в просмотрщике

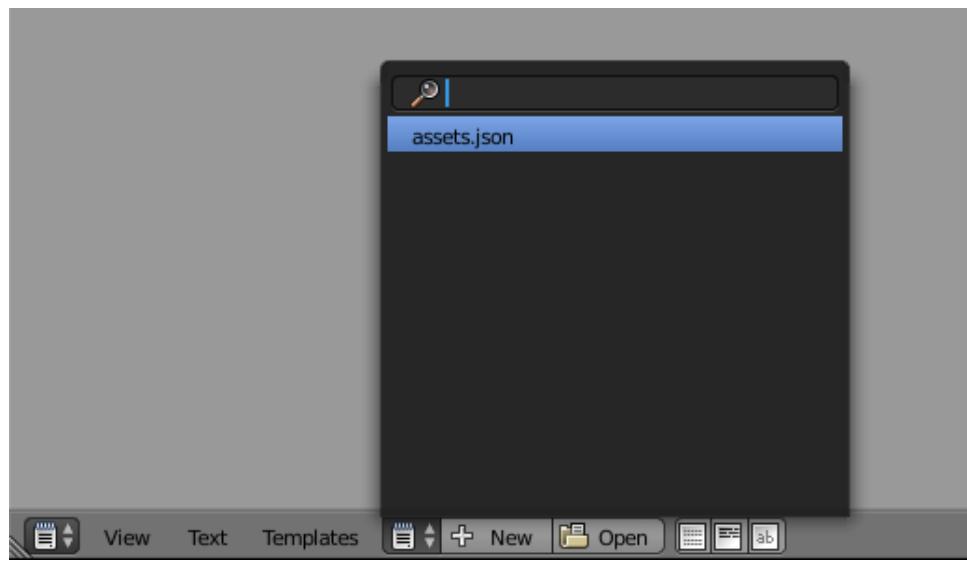
При использовании [локального сервера разработки](#) и опции экспорта `Run in Viewer` сцены показываются в просмотрщике сцен сразу после экспортации.

При необходимости долгосрочного хранения сцены в списке сцен просмотрщика, нужно вручную добавить запись в текстовой файл `apps_dev/viewer/assets.json`, который доступен при открытии Blender'a при использовании Blend4Web SDK.

Для редактирования файла откройте “Text Editor”:



В появившемся окне вы сможете выбрать файл assets.json и отредактировать его:



Для добавления новой сцены нужно знать категорию, в которой она должна отображаться. Категория обычно соответствует названию проекта и имени директории,

где хранятся соответствующие файлы.

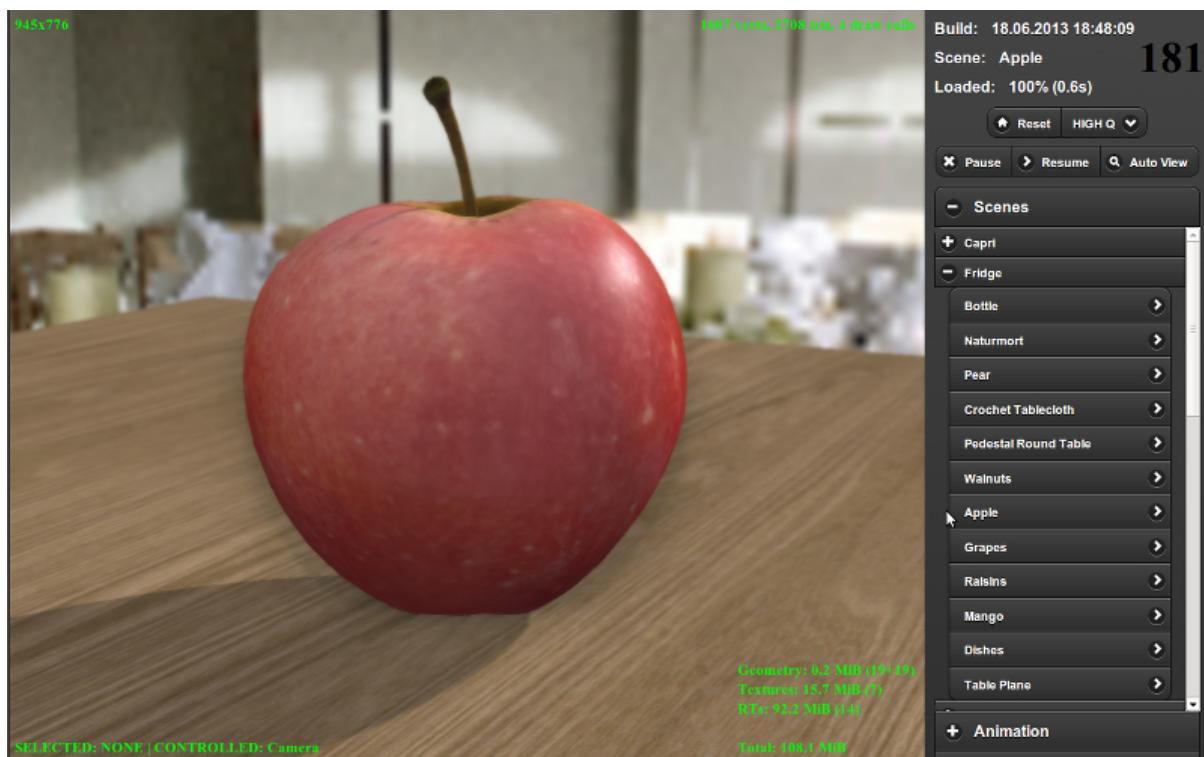
5.3.1 Пример

Ниже приведена примерная часть файла `assets.json`, в которой находятся два проекта “Capri” и “Fridge” с соответствующими сценами в каждом проекте:

```
{  
    name: "Capri",  
    items: [  
        {  
            name: "Baken",  
            load_file : "capri/props/baken/baken.json"  
        },  
        {  
            name: "Terrain",  
            load_file : "capri/landscape/terrain/terrain.json"  
        }  
    ]  
},  
{  
    name: "Fridge",  
    items: [  
        {  
            name: "Apple",  
            load_file : "fridge/fruits/apple/apple.json"  
        },  
        {  
            name: "Mango",  
            load_file : "fridge/fruits/mango/mango.json"  
        }  
    ]  
}
```

Добавление можно осуществить копированием и вставкой описания похожей сцены в нужной категории и последующим редактированием ее названия и пути к экспортному файлу.

В случае успешного добавления сцена должна появиться в списке сцен просмотрика в нужной категории.



5.4 Разработка приложения

На этой стадии создаётся приложение, в котором на языке JavaScript пишется логика по загрузке сцены и взаимодействию с ней пользователя. Документация для разработчиков приложений приведена в [соответствующем разделе](#).

Просмотрщик сцен

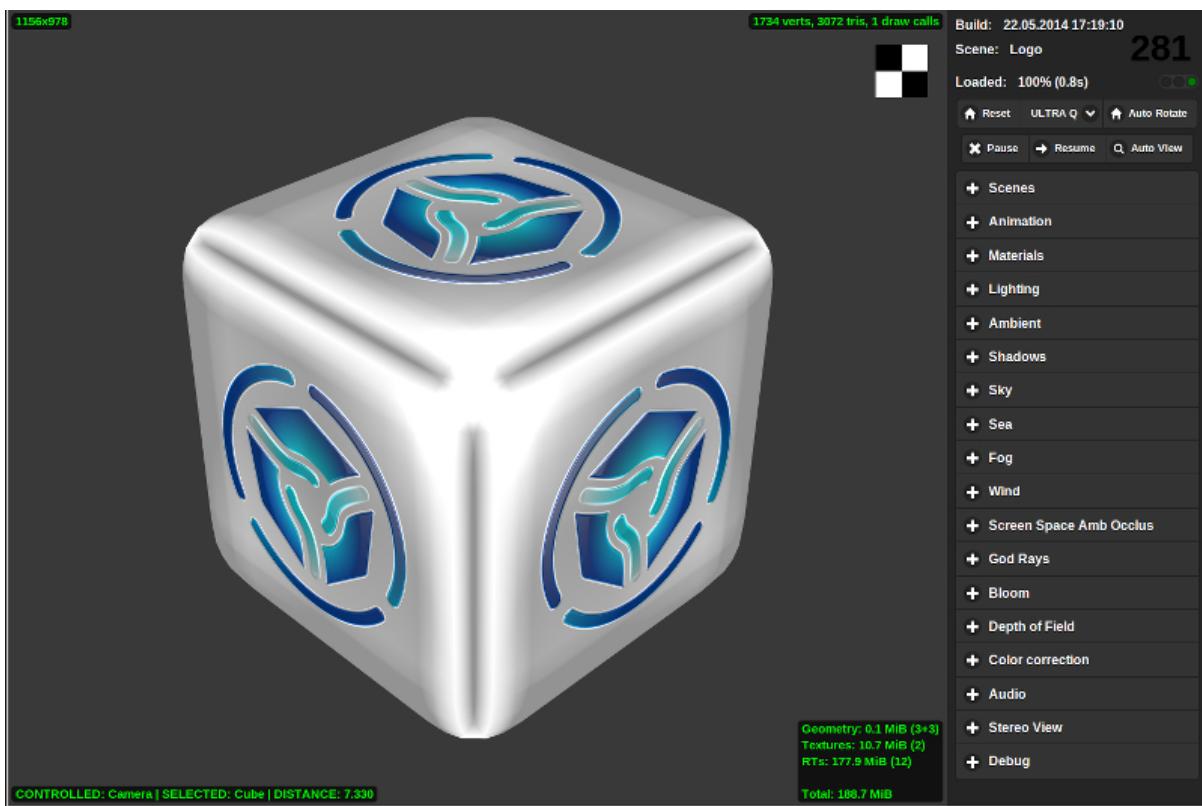
Запуск просмотрщика сцен.

6.1 Навигация

Управление камерой осуществляется мышью с нажатой кнопкой, а также клавишами W, A, S, D, R, F: вперед, влево, назад, вправо, вверх, вниз. Также поддерживаются стрелки и клавиши `numPad`. В режиме камеры `Target` возможно фокусирование на выделенный объект посредством нажатия клавишей Z или . (точка).

6.2 Боковая панель

Боковая панель содержит в себе три области: информационное табло, базовые кнопки управления и список выпадающих панелей, содержащий дополнительные элементы управления, разделённые по функциональному признаку.



6.2.1 Список элементов управления сверху вниз

Build Дата и время сборки движка. В версии для разработки показывает время загрузки страницы.

Scene Название загруженной сцены, взятое из файла assets.json. При наведении курсора мыши всплывает путь к файлу.

Loaded Процент и время загрузки.

Reset Кнопка удаляет сохраненное название последней просмотренной сцены и перезагружает страницу, возвращая к сцене по умолчанию.

LOW Q - HIGH Q - ULTRA Q Выпадающее меню выбора профиля работы движка.

См.также:

[Профили качества изображения](#)

Pause Приостановка рендеринга.

Resume Возобновление рендеринга.

Auto View Активация режима автоматического переключения сцен по списку, пауза между просмотрами 1 секунда.

Scenes Двухуровневый список категорий и сцен из файла `assets.json`.

Animation Управление анимацией. При просмотре анимированных моделей можно выбирать объект и переключать для него анимацию с помощью выпадающего меню, включать режим циклической анимации, останавливать и возобновлять анимацию, выставлять нужный кадр (при этом анимация должна быть остановлена).

Materials Настройка свойств материалов. Материал выбирается с помощью выпадающего меню. В настоящий момент поддерживается изменение ограниченного ряда свойств.

Lighting Настройка параметров прямого освещения. Источник света выбирается с помощью выпадающего меню. Поддерживается изменение цвета и интенсивности. Также на этой панели осуществляется настройка параметров времени суток и солнечного освещения.

Ambient Настройка параметров освещения от окружающей среды. Поддерживается изменение цветов в полусферической модели освещения и интенсивности.

Shadows Настройка параметров падающих теней, включая параметры каскадов теней и параметры смягчения краев теней.

Sky Настройка параметров динамического неба, таких как цвет, параметры рассеивания солнечного света и т.д.

Sea Настройка параметров рендеринга воды, включая переходы цвета по глубине и удаленности от берега, параметры пены, подповерхностного рассеивания, динамики волн и т.д.

Fog Настройка параметров тумана, включая плотность и цвет.

Wind Настройка параметров ветра, включая направление и силу.

Screen Space Amb Occlus Настройка параметров взаимного затенения.

God Rays Настройка параметров эффекта сумеречных лучей.

Bloom Настройка параметров эффекта засветки ярким светом.

Depth of Field Настройка параметров эффекта глубины резкости камеры.

Color correction Настройка параметров цветовой коррекции, включая яркость, контрастность, экспозицию и насыщенность.

Anti-aliasing Выбор метода сглаживания изображения (антиалиасинга).

Audio На панели находится переключатель режима микширования. После включения на сценах со звуковыми источниками появляется интерфейс микшера.

Stereo View На панели находится переключатель режима стерео-изображения.

Debug На панели находится ряд инструментов отладки сцен, в частности переключатель режима просмотра сетки, переключатель режима просмотра постпроцессинговых стадий.

6.3 Индикаторы

Счетчик количества кадров в секунду Находится в правом верхнем углу. Выводит усредненное и округленное значение за последние 1,5 секунды.

Размер области рендеринга Находится в левом верхнем углу. Выводит размер области рендеринга в пикселях.

Выбранный объект и контролируемый объект Находится в левом нижнем углу. Выводит название выбранного объекта и контролируемого объекта. Выбор объекта осуществляется мышью. Для получения прямого контроля над объектом (обычно в целях проверки физики) нужно нажать Q и выбрать объект. Движение объекта осуществляется клавишами W, A, S, D. Для выхода из режима контроля нужно нажать Q и “кликнуть” на пустом пространстве. Индикатор также показывает расстояние до выбранного объекта в единицах Blender'a (эквивалент метра).

Индикатор сложности сцены Находится в правом верхнем углу области рендеринга. Выводит количество вершин, треугольников и WebGL вызовов на основной рендер-сцене (т.е. не учитываются, например, вызовы по отрисовке динамических теней).

Индикатор видео-памяти Находится в правом нижнем углу области рендеринга. Выводит количество видео-памяти, занимаемой геометрией, текстурами, буферами с результатами рендеринга (render targets), а также суммарное количество занимаемой памяти.

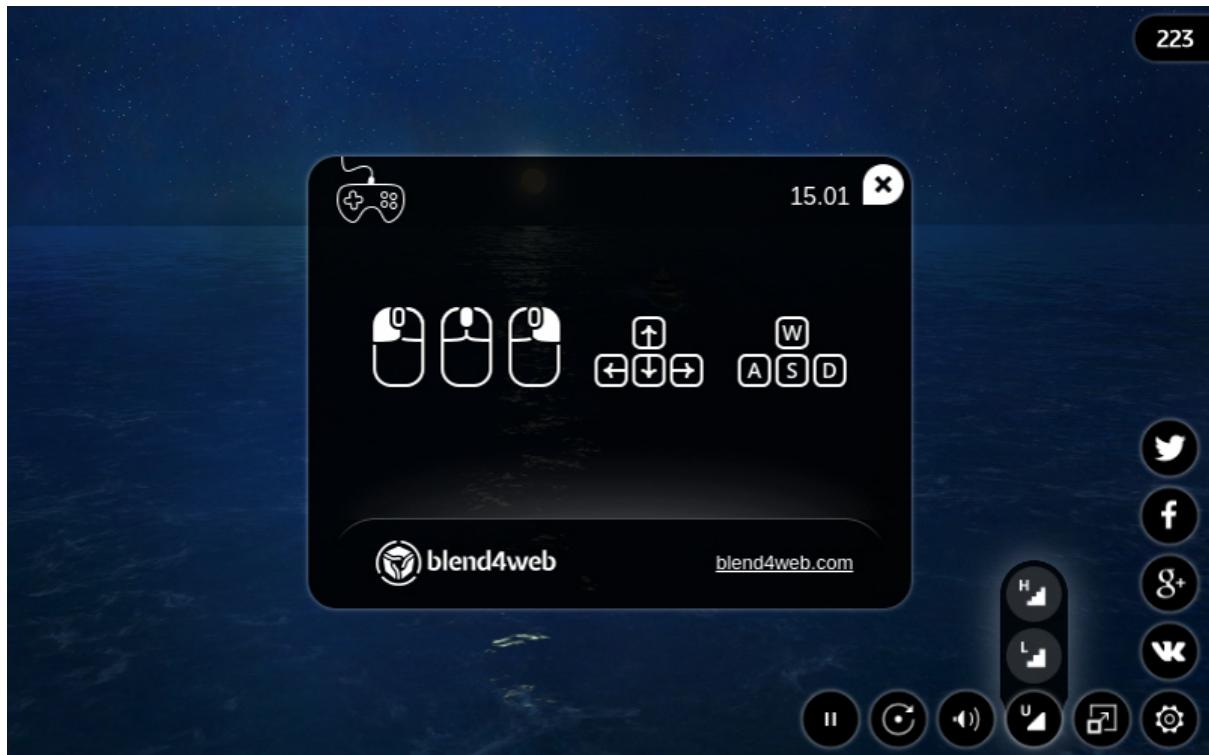
Индикатор ошибок при загрузке сцены Находится под счетчиком количества кадров. Выводит ошибки и предупреждения при загрузке сцены. Красный свет означает ошибки, желтый - предупреждения, зеленый - загрузка сцены прошла успешно.



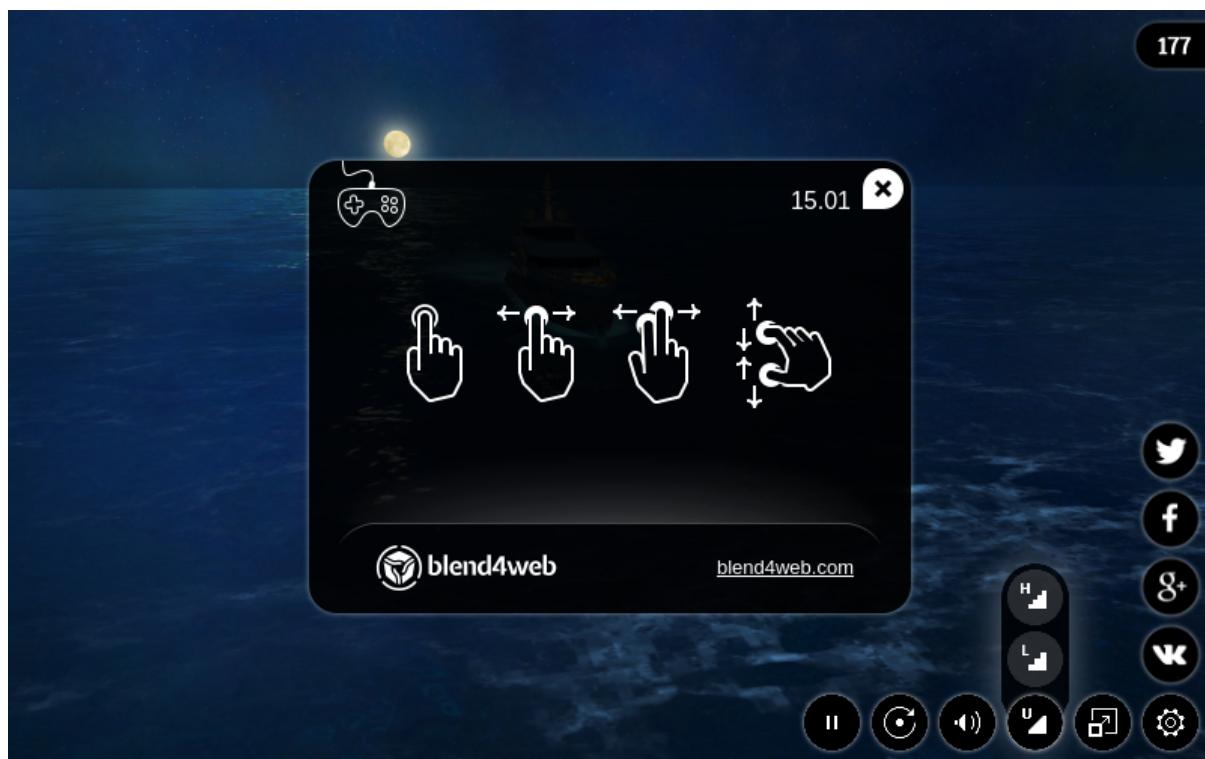
Веб-плеер

Веб-плеер представляет собой специализированное приложение для отображения моделей и сцен в режиме демонстрации.

Десктопная версия:



Мобильная версия:



7.1 Использование

Директорию с файлами веб-плеера `deploy/apps/webplayer` можно скопировать из дистрибутива Blend4Web SDK и разместить на сайте. Полученные в результате экспорта файлы сцен можно разместить на сайте и указать путь к ним (абсолютный или относительный) с помощью параметра веб-плеера `load`.

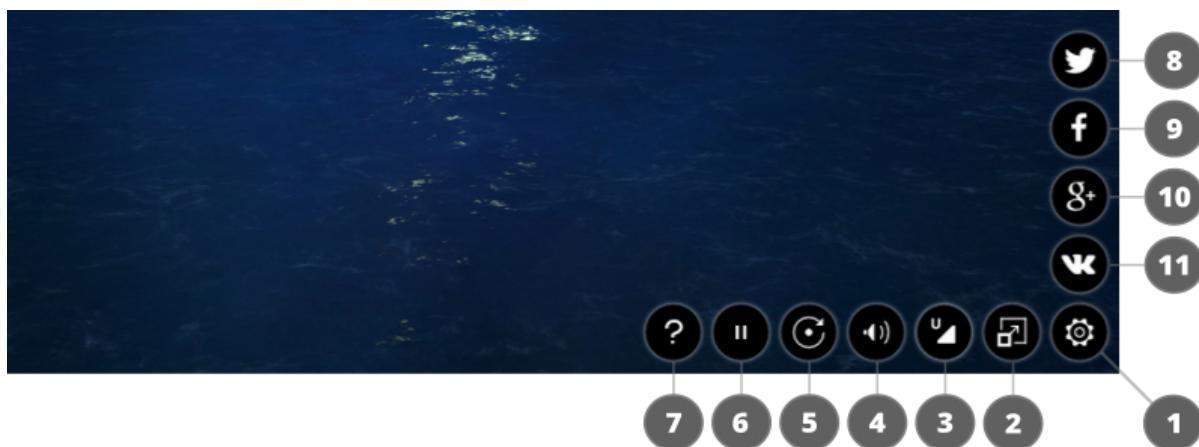
При экспорте в единый HTML файл интерфейс веб-плеера интегрируется автоматически.

7.2 Навигация

Управление камерой (в режимах `Target` и `Eye`) осуществляется мышью с нажатой кнопкой, а также клавишами `W`, `A`, `S`, `D`, `R`, `F`: вперед, влево, назад, вправо, вверх, вниз. Также поддерживаются стрелки и клавиши `numpad`.

7.3 Панель управления

Ниже показана панель управления веб-плеера.



1. показ / скрытие панели управления;
2. включение / выключение полноэкранного режима;
3. установка качества сцены;
4. включение / выключение звука;
5. включение / выключение автоматического поворота камеры вокруг сцены;
6. запуск / остановка движка;
7. открытие окно помощи;
8. кнопка Твиттер;
9. кнопка Facebook;
10. кнопка Google+;
11. кнопка ВКонтакте.

7.4 Атрибуты

Веб-плеер принимает атрибуты из адресной строки браузера:



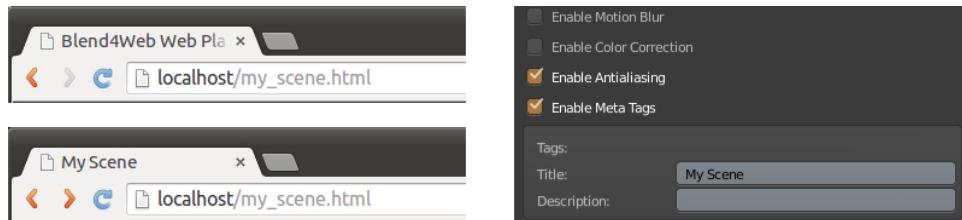
1. для загрузки сцены используется обязательный атрибут `load`, в который помещается относительный путь к JSON файлу.
2. необязательный атрибут `fallback_image` используется для установки фона, заменяющего сцену при возникновении ошибки WebGL.

3. необязательный атрибут `fallback_video` используется для установки видео, заменяющего сцену при возникновении ошибки WebGL. Может использоваться неоднократно с целью добавления видео файлов различных форматов.
4. необязательный атрибут `show_fps` применяется для отображения счетчика кадров в секунду в правом верхнем углу плеера.
5. необязательный атрибут `autorotate` используется для включения автоматического вращения камеры сразу после загрузки сцены.
6. необязательный атрибут `compressed_textures` используется для включения загрузки уменьшенных текстур и текстур в формате с компрессией (DDS).

Примечание: Если указаны параметры `fallback_image` и `fallback_video`, приоритет имеет `fallback_image`.

7.5 Название сцены в заголовке

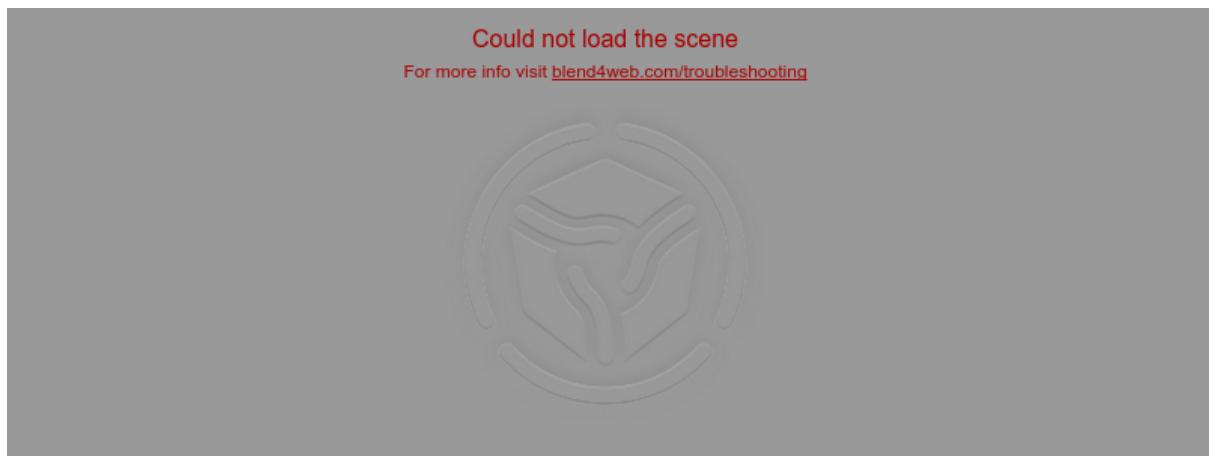
По умолчанию веб-плеер устанавливает тег `title` равным `Blend4Web Web Player`. С помощью мета-тега `title` в настройках сцены можно изменить значение по умолчанию.



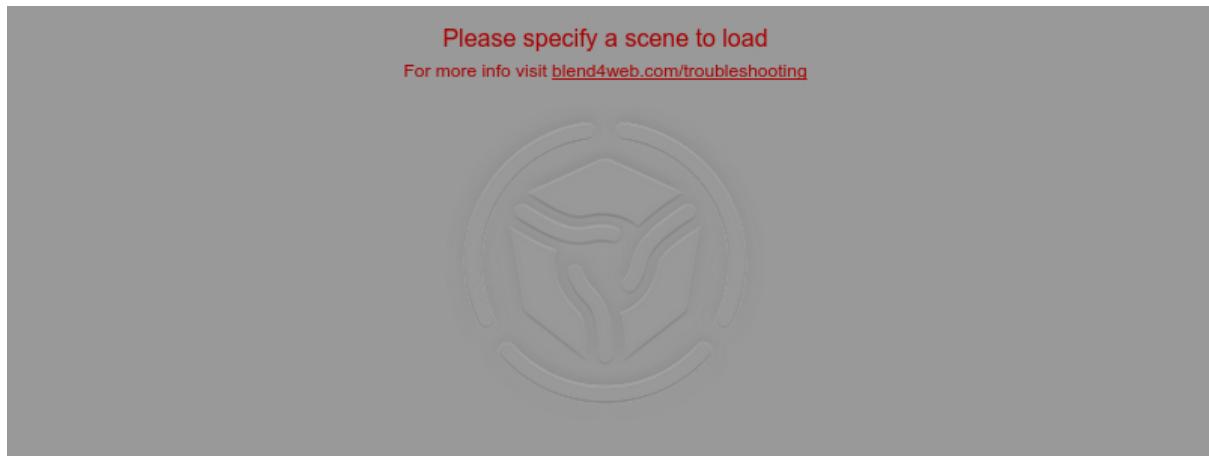
7.6 Ошибки сцены

В случае некорректного использования веб-плеера выводятся сообщения об ошибках.

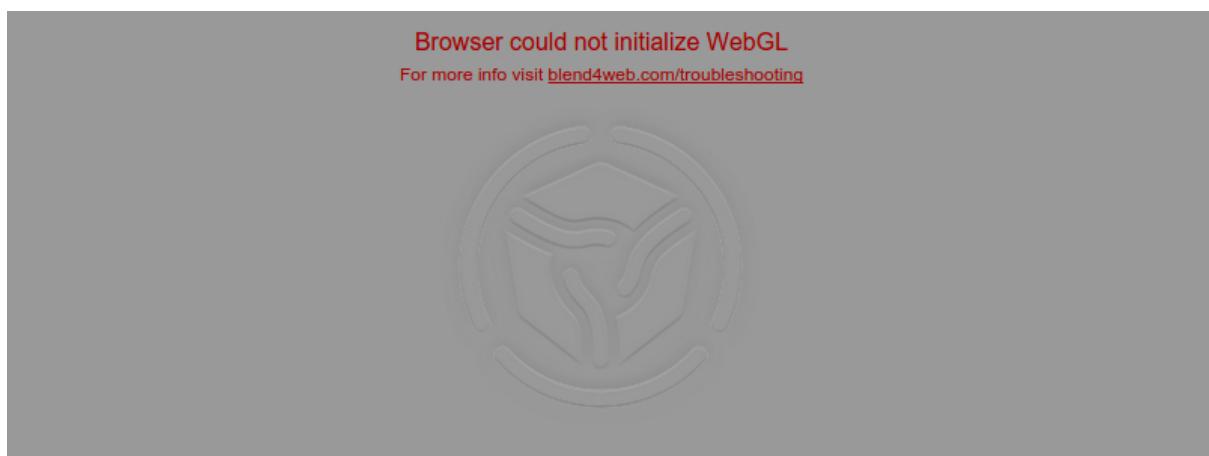
1. атрибутом `load` указан неверный путь к JSON файлу или данный файл неисправен;



2. атрибут `load` не найден или его значение пустое;



3. ошибка инициализации WebGL.

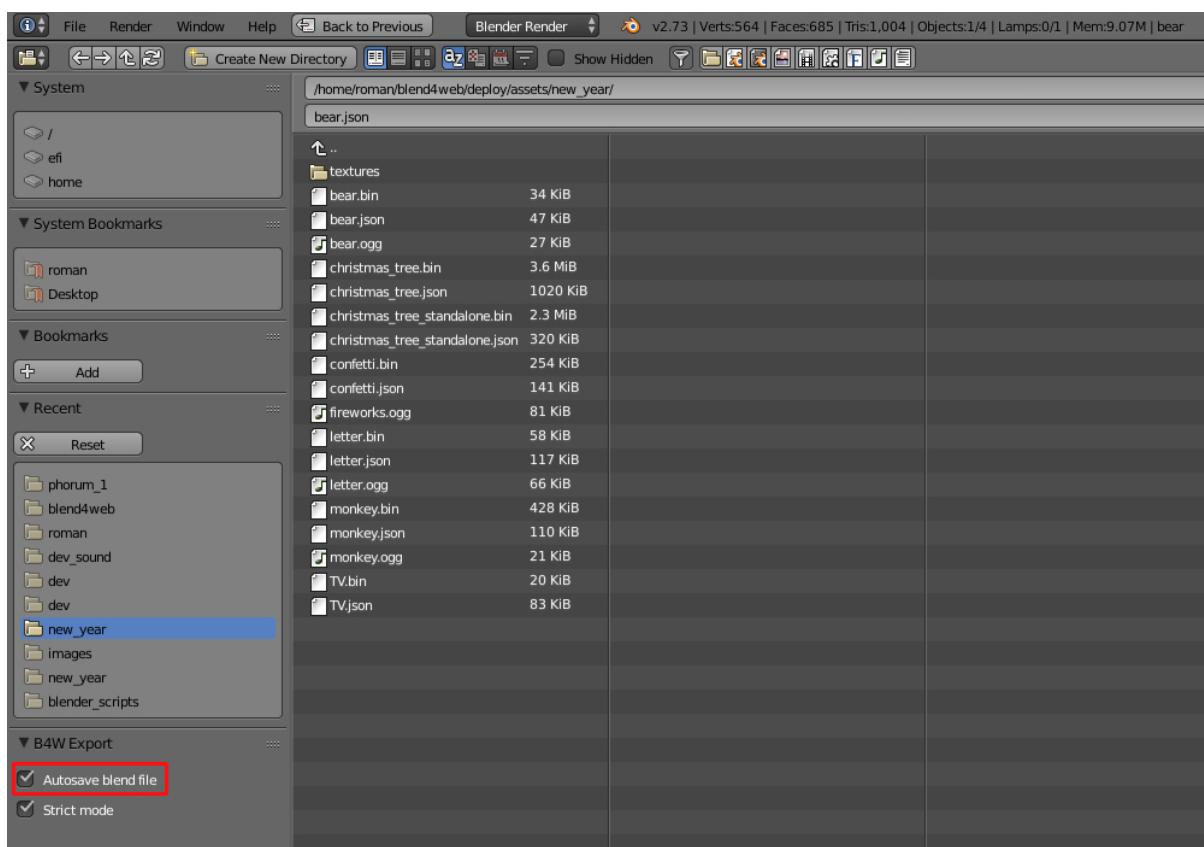


Аддон

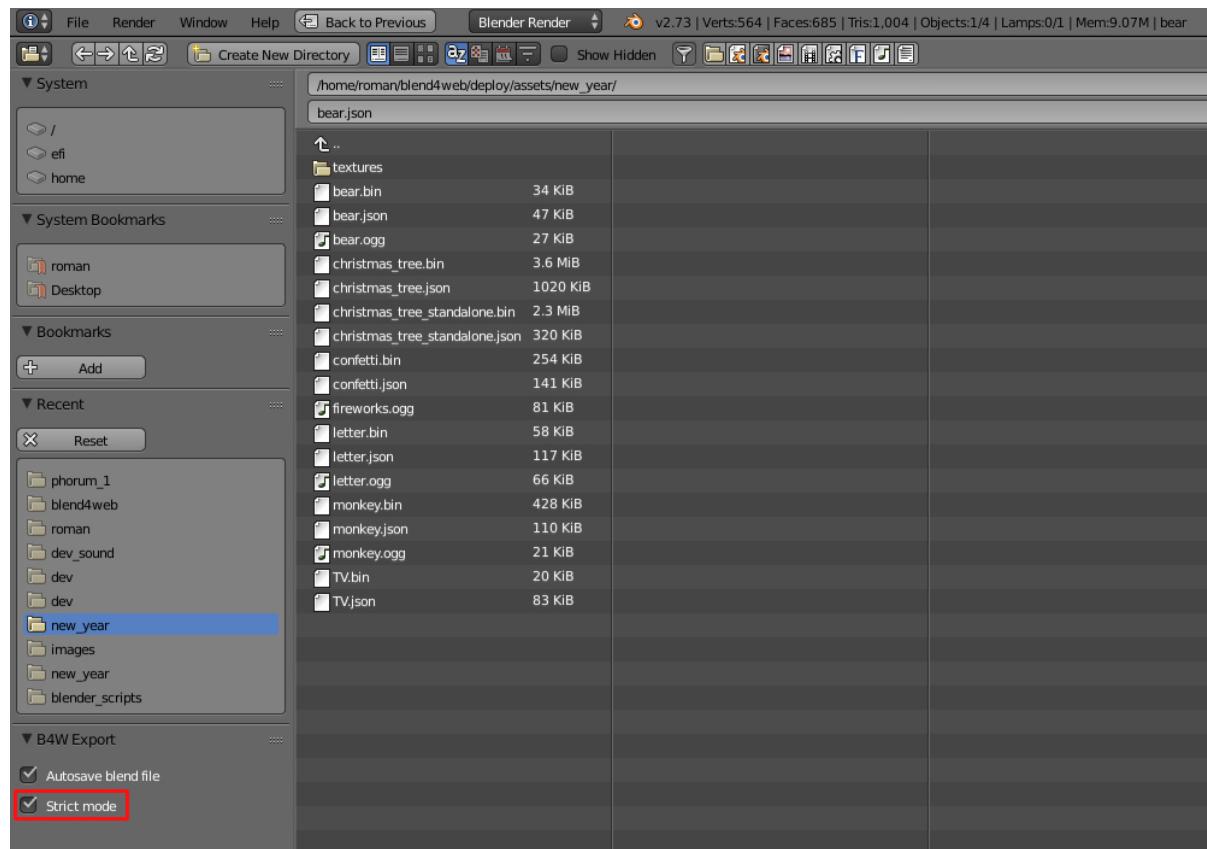
8.1 Опции экспорта

Autosave blend File Автосохранение файла, из которого осуществляется экспорт.

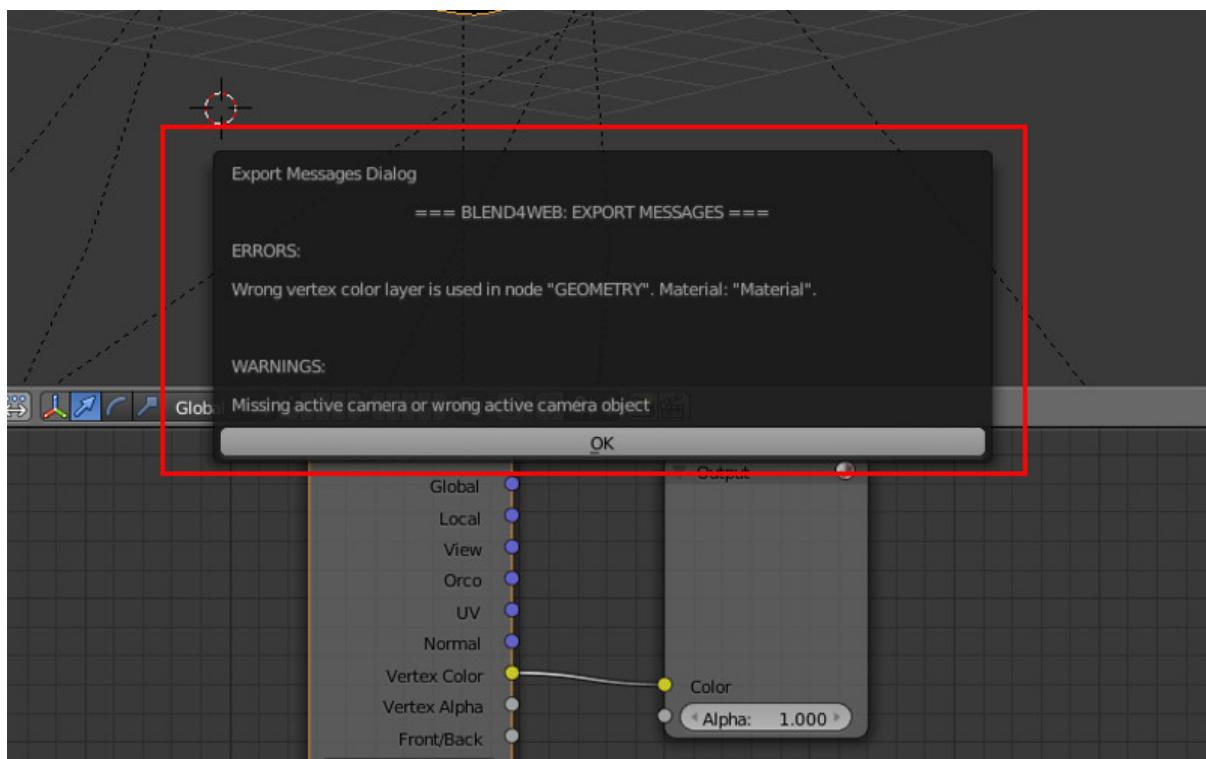
Включено по умолчанию. Осуществляется непосредственно после экспорта с целью поддержки соответствия между текущим содержимым blend-файла и экспортного файла. Кроме того, для удобства в blend-файле сохраняется относительный путь к экспортному файлу.



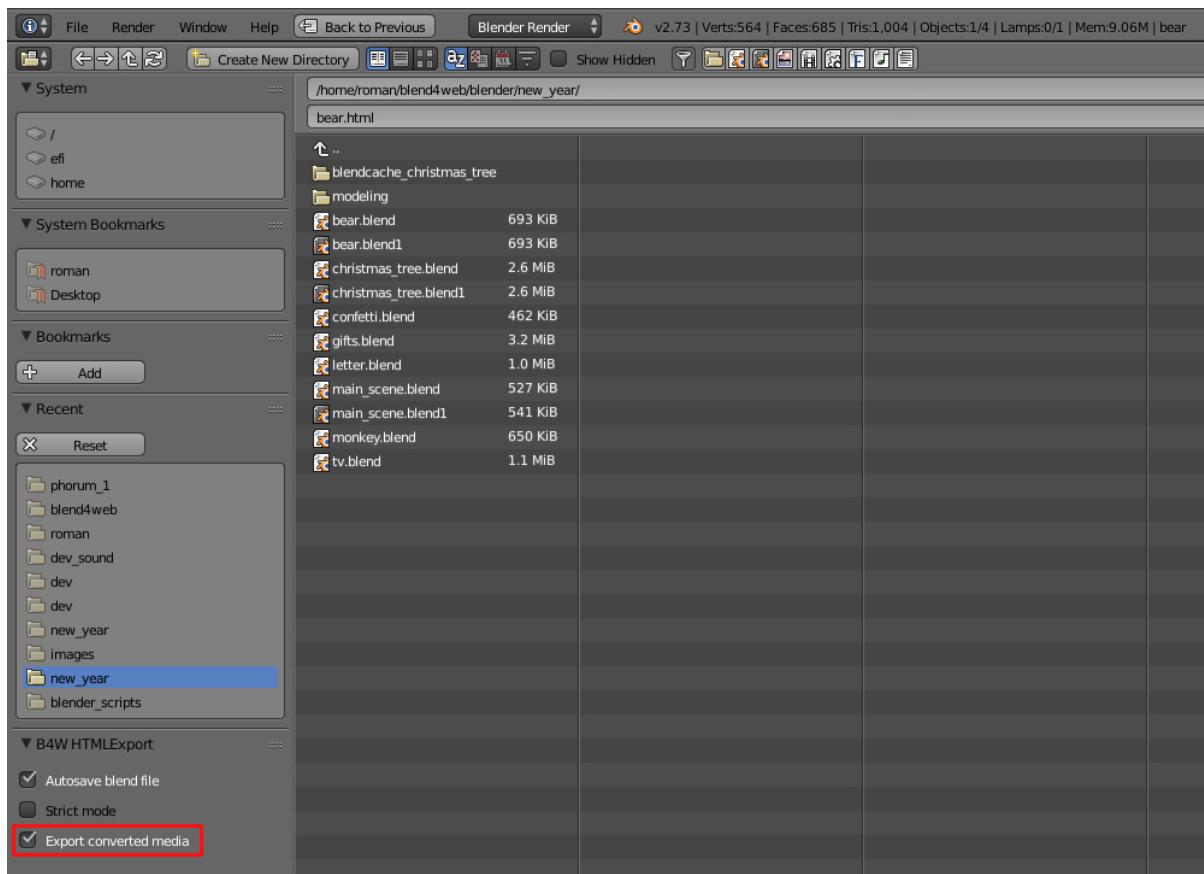
Strict Mode Данный режим блокирует экспорт при наличии ошибок и сообщений, требующих внимания пользователя. Режим включается при выставлении опции Strict Mode в меню экспорта:



При наличии некритических ошибок экспорта или сообщений, требующих внимания пользователя, вашему вниманию будет представлено диалоговое окно вида:



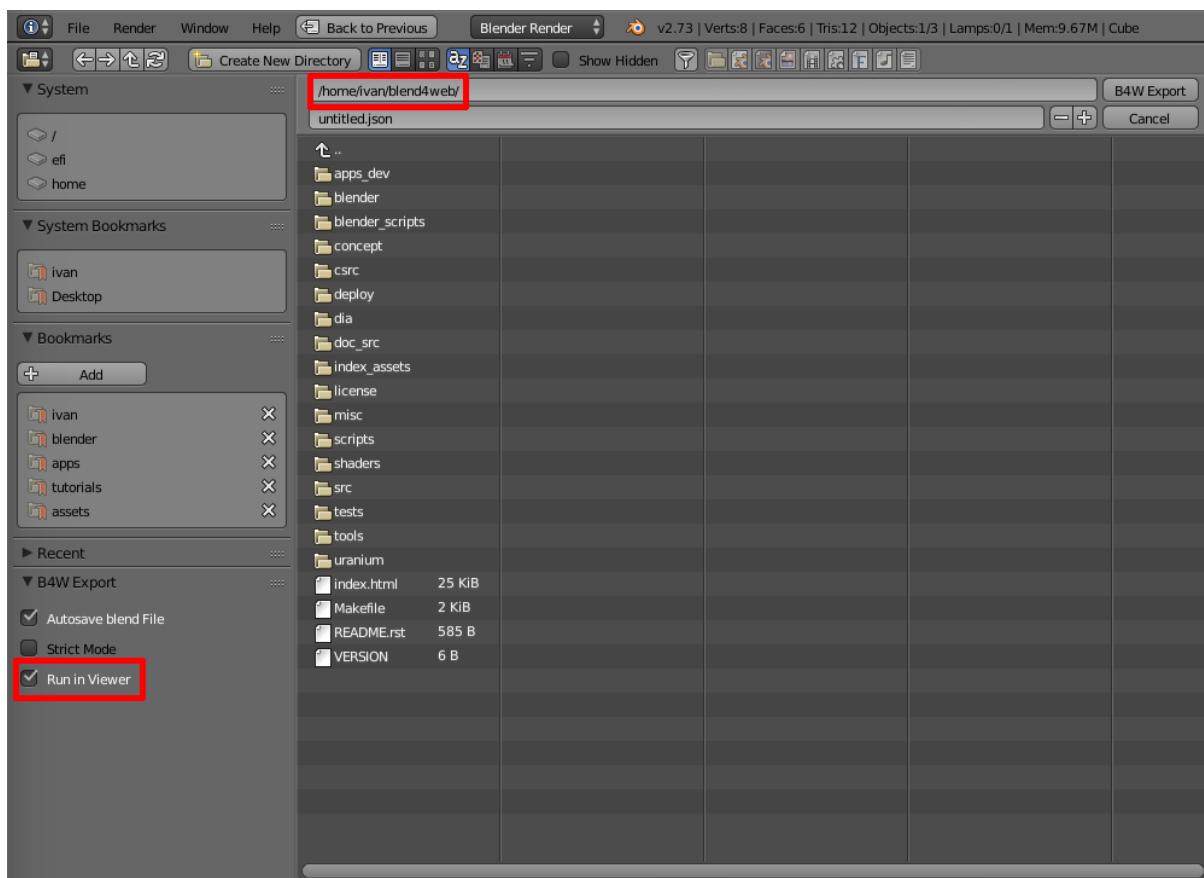
Export Converted Media Опция доступна при html-экспорте. Включение данного режима экспорта позволяет записать в HTML файл конвертированные медиафайлы разных форматов. Это необходимо использовать при создании кросбраузерных и кросплатформенных приложений при html-экспорте. При этом в html-файл будут записываться файлы, созданные с использованием [нашего конвертера](#).



Run in Viewer Автоматически запустить просмотрщик сцен и добавить в него экспортируемую сцену.

При использовании *локального сервера разработки*, имеется возможность открыть сцену, экспортированную в формате .json, в просмотрщике сцен. Для этого при экспорте необходимо выбрать любой путь, лежащий внутри файловой структуры Blend4Web SDK.

В качестве директории для экспорта может использоваться созданная пользователем директория внутри Blend4Web SDK. При несоблюдении этого условия опция не будет отображаться в меню экспорта. Также опция не будет отображаться если локальный web-сервер не запущен.



8.2 Ошибки инициализации

Ошибки инициализации могут проявляться при инициализации аддона, либо при загрузке сцены в Blender. При возникновении появится диалоговое окно с описанием ошибки.



Сообщение об ошибке	Причина
Blend4Web initialization error! Addon is not compatible with the PLATFORM platform.	Ошибка загрузки аддона. Аддон не совместим с платформой PLATFORM.
Warning: Blender version mismatch. Blender VER_REQUIRED is recommended for the Blend4Web addon. Current version is VER_CURRENT.	Предупреждение о возможной несовместимости с текущей версией Blender. Для работы рекомендуется версия Blender'a VER_REQUIRED. Текущая версия - VER_CURRENT.

8.3 Ошибки совместимости

Ошибки совместимости версий могут проявиться при просмотре сцены в браузере, если друг другу не соответствуют версия аддона, которым был произведен экспорт сцены, и версия движка Blend4Web, который пытается эту сцену загрузить.

Сообщение об ошибке	Причина
Version mismatch. File version: VER_OLD, required: VER_NEW. Some compatibility issues can occur. Reexport scene with the latest B4W addon to fix it.	Частичная несовместимость версий. Версия экспортированной сцены - VER_OLD, движком требуется - VER_NEW. Сцена будет загружена, однако, возможны различные ошибки. Рекомендуется переэкспортировать сцену аддоном последней версии, а также использовать последнюю версию движка.
Incompatible file version: VER_OLD, required: VER_NEW. Reexport scene with the latest B4W addon to fix it.	Полная несовместимость версий. Версия экспортированной сцены - VER_OLD, движком требуется - VER_NEW. Сцена не будет загружена. Рекомендуется переэкспортировать сцену аддоном последней версии, а также использовать последнюю версию движка.

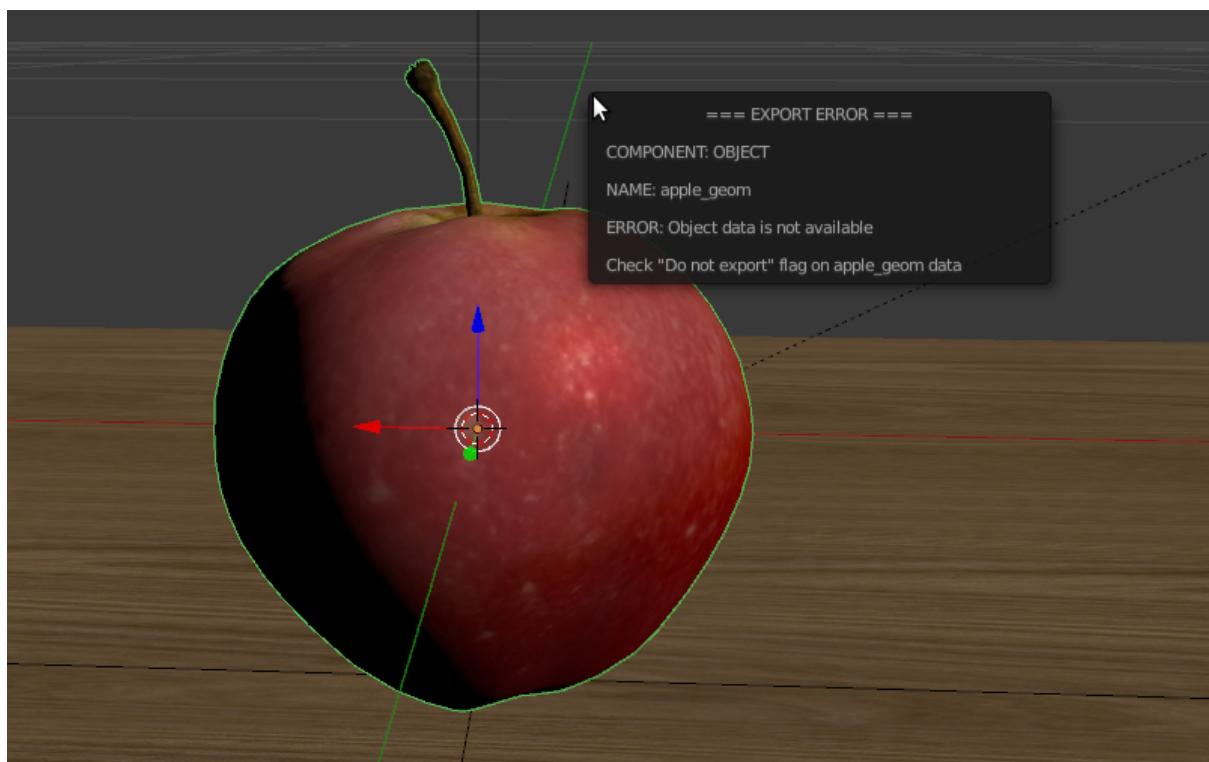
8.4 Критические ошибки экспорта

При возникновении ошибок во время экспорта появляется диалоговое окно BLEND4WEB EXPORT ERROR с описанием проблемы:

COMPONENT - тип компонента (объект, меш, материал, текстура и т.д.), при экспорте которого произошла ошибка.

NAME - имя компонента.

ERROR - краткое описание возникшей проблемы на англ. языке.



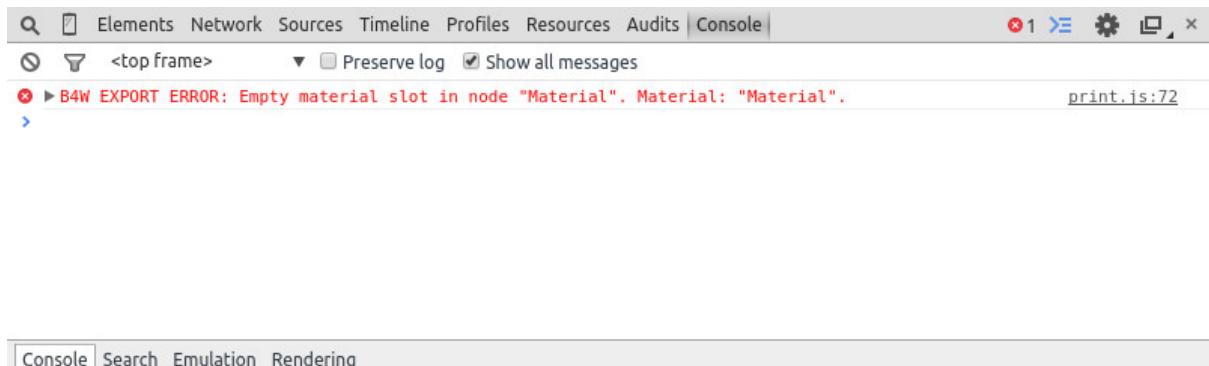
Сообщение об ошибке	Причина
Dupl group error; Objects from the GROUP_NAME dupli group on the OBJECT_NAME object cannot be exported	Ни один из объектов группы GROUP_NAME, выбранной для дублирования на объекте OBJECT_NAME, не экспортируется. Требуется разрешить экспорт хотя бы одного из объектов группы, либо убрать дублирование группой.
Export to different disk is forbidden	Не разрешен экспорт в директорию, находящуюся на другом диске
Incompatible objects with a shared mesh; The OBJECT_NAME object has both vertex groups and a shared mesh	Несовместимые объекты с общим мешем. Не допускается экспорт объекта с общим мешем и вертексными группами. Исключения: экспорт возможен, если на объекте включены опции <code>Apply modifiers</code> , <code>Export vertex animation</code> , <code>Export edited normals</code> , <code>Apply scale</code> (т.к. в этом случае при экспорте происходит полное копирование мешей).
Incomplete mesh; Material slot is empty	Неполный меш: пустой слот материала.
Incomplete vehicle. The NAME vehicle doesn't have any chassis or hull	Моделируемое средство передвижения NAME является незавершенным: оно должно содержать один элемент <code>Chassis</code> или <code>Hull</code> .
Incomplete vehicle. The NAME vehicle requires at least one bob	Моделируемое средство передвижения NAME является незавершенным: оно должно содержать хотя бы один элемент <code>Bob</code> .
Incomplete vehicle. The NAME vehicle requires at least one wheel	Моделируемое средство передвижения NAME является незавершенным: оно должно содержать хотя бы один элемент <code>wheel</code> .
Incorrect mesh; Wrong group indices	Меш содержит вершины, привязанные к несуществующей группе.
Incorrect vertex animation; Object has no vertex animation	Включен экспорт вертексной анимации для объекта, но ни одной анимации не имеется.
Incorrect vertex animation; Unbaked "ANIM_NAME" vertex animation	Включен экспорт вертексной анимации для меша, но анимация ANIM_NAME не содержит ни одного кадра.
Loading of resources from different disk is forbidden	Не разрешен экспорт ресурсов из директории, находящейся на другом диске.
The material has a normal map but doesn't have any material nodes	Нодовый материал использует <code>Normal Mapping</code> , но не имеет ноды <code>Material</code> .
The mesh has a UV map but has no exported material	Меш имеет текстурную развертку, но не имеет материала, который бы экспортировался.

The mesh has a vertex color layer but has no exported material	Меш имеет слой вертексного цвета, но не имеет материала, который бы экспорттировался.
Missing lamp	На сцене должен быть хотя бы один источник света.
No such file or directory	Данная директория не существует.
Node material invalid; Check sockets compatibility: FROM_NODE with TO_NODE	Ошибка нодового материала. Типы входа и выхода связи между нодами FROM_NODE и TO_NODE не соответствуют друг другу.
Object constraint has no target	Для ограничителя объекта (вкладка Object Constraints) не установлено свойство Target Object.
Particle system error; Dupli group isn't specified	Ошибка системы частиц. Не выбрана группа, используемая в качестве частицы.
Particle system error; Dupli object isn't specified	Ошибка системы частиц. Не выбран объект, используемый в качестве частицы.
Particle system error; Dupli object OBJECT_NAME doesn't export	Ошибка системы частиц. Объект OBJECT_NAME, выбранный в качестве частицы, не экспортируется (на нем выбрана опция Do not export).
Particle system error; The GROUP_NAME dupli group contains no valid object for export	Ошибка системы частиц. Ни один подходящий объект из группы GROUP_NAME, выбранной в качестве частицы, не экспортируется. Либо на таких объектах выбрана опция Do not export, либо объекты имеют неподходящий тип. Поддерживаемые типы: MESH.
Particle system error; Wrong dupli object type TYPE_NAME	Ошибка системы частиц. В качестве частицы выбран объект неподходящего типа. Поддерживаемые типы: MESH.
Permission denied	Нет прав доступа к текущей директории.
Wrong edited normals count; It doesn't match with the mesh vertices count	Число редактируемых нормалей не совпадает с числом вершин меша. Требуется сделать Clean Up либо Save в панели B4W Vertex Normals Editor.
Wrong overridden bounding box; Check the mesh's bounding box values	Указаны неверные размеры при переопределении BoundingBox для меша: минимальное значение больше максимального для хотя бы одного из измерений.
Wrong vertex animation vertices count; It doesn't match with the mesh vertices count for "ANIM_NAME"	Включен экспорт вертекской анимации, но число вершин покадрово в анимации ANIM_NAME не совпадает с числом вершин меша. Возможное решение - "перезапекание" анимации.

8.5 Некритические ошибки экспорта

В отличие от критических ошибок экспорта, рассмотренных ранее, данные ошибки не препятствуют экспорту, однако могут приводить к некорректному отображению сцен. Сообщения выводятся в консоли браузера (горячая клавиша F12) при загрузке сцены. Сообщение имеет вид:

B4W EXPORT ERROR: Сообщение об ошибке



Сообщение об ошибке	Причина
Canvas texture ID NAME already exists. Texture NAME. Empty canvas texture ID for texture NAME. Empty material slot in node “NAME”. Material: “NAME”. Exported UV-layer is missing in node “GEOMETRY”. Material: NAME. Exported UV-layer is missing in texture NAME. [Material: NAME.] Ignoring LODs after empty LOD for the NAME object.	Данный идентификатор для объекта типа Canvas уже существует. Пустое поле идентификатора для объекта типа Canvas . Не задан материал в ноде: “NAME” В ноде типа GEOMETRY указан не экспортируемый UV-слой для текстурных координат типа UV. В текстуре указан не экспортируемый UV-слой для текстурных координат типа UV.
Incomplete mesh NAME; Dynamic grass vertex colors required by material settings	В списке LOD объектов, настроенных для объекта NAME, были проигнорированы все LOD объекты, следующие за пустым. Неполный меш: специальный материал для ландшафта использует опции Dynamic grass size и/или Dynamic grass color , но у меша нет слоев вертексного цвета с такими именами.
Incomplete mesh; No UV in mesh with UV-textured material	Неполный меш: в материале меша используются текстуры с типом координат UV , но у меша нет текстурной развертки.
Incomplete mesh; Material settings require vertex colors	Неполный меш: материал меша имеет включенную опцию вертексного цвета (Vertex Color Paint), но у меша нет слоя вертексного цвета.
Incorrect NLA script, falling back to simple sequential NLA. Invalid link found in node material. Material: “NAME”. No image in the NAME texture. [Material: NAME.] No texture for the NAME particle settings texture slot. No texture in the NAME world texture slot. No texture in the texture slot. Material: NAME. Object “NAME” has the mesh with shape keys. The property “Relative” of mesh has been enabled.	Некорректный NLA-скрипт. Вместо него будет использоваться стандартная NLA-анимация. Нодовый материал с именем “NAME” содержит некорректные связи между нодами. У текстуры отсутствует изображение. В текстурном слоте системы частиц отсутствует текстура. В текстурном слоте объекта World отсутствует текстура. В текстурном слоте материала отсутствует текстура. Объект с именем “NAME” имеет меш с опорными фигурами. На меше было включено свойство “Relative”.

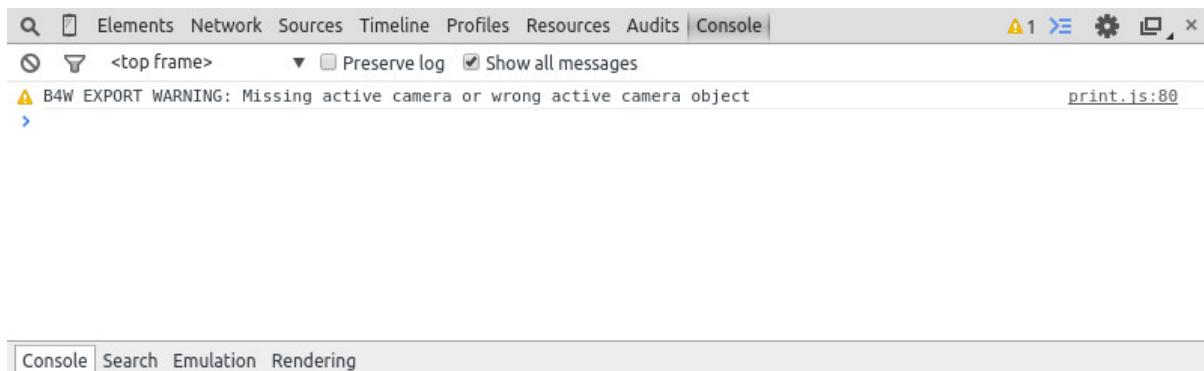
Only 2 UV textures are allowed for a mesh; The mesh has N UVs.	Движком поддерживаются только до 2 UV текстур на каждый меш. Меш содержит UV текстуры в количестве N. Ошибка системы частиц. Вертексный цвет NAME указанный в поле <code>from</code> , отсутствует в эмиттере OBJECT_NAME.
Particle system error for “NAME”; The “NAME” vertex color specified in the <code>from</code> field is missing in the last of the “OBJECT_NAME” object’s vertex colors Particle system error for “NAME”; The “NAME” vertex color specified in the <code>to</code> field is missing in the list of the “OBJECT_NAME” object’s vertex colors	Ошибка системы частиц. Вертексный цвет NAME указанный в поле <code>to</code> , отсутствует в объекте OBJECT_NAME, выбранном в качестве частицы.
Particle system error for “NAME”; The “NAME” vertex color specified in the “ <code>to</code> ” field is missing in the “OBJECT_NAME” object (“GROUP_NAME” dupli group)	Ошибка системы частиц. Вертексный цвет NAME указанный в поле <code>to</code> , не присутствует в объекте OBJECT_NAME группы GROUP_NAME, выбранной в качестве частицы.
The NAME action has decimal frames. Converted to integer. The NAME armature modifier has a proxy object as an armature. Modifier removed. The NAME armature modifier has no armature object or it is not exported. Modifier removed. The NAME curve modifier has no curve object. Modifier removed. The NAME curve modifier has unsupported curve object. Modifier removed. The NAME LAMP node has no lamp object. Material: NAME. The NAME node is not supported. The NAME material will be rendered without nodes. Material: NAME.	Анимация NAME содержит дробные значения кадров. Округлено до целых. Модификатор арматуры имеет прокси объект в качестве арматуры. В модификаторе NAME типа <code>Armature</code> не указан объект, либо объект не экспортируется. Модификатор удален. В модификаторе NAME типа <code>Curve</code> не указан объект. Модификатор удален. В модификаторе NAME типа <code>Curve</code> указан неподходящий объект. Модификатор удален. В node NAME типа <code>LAMP</code> не указан подходящий объект. Нода с данным именем не поддерживается движком, поэтому нодовый материал будет отключён. Чаще всего проблемы подобного рода возникают при использовании нод <code>Cycles</code> . Объект должен находиться в той же группе, что и арматура или оба объекта должны явно присутствовать на сцене. Используется нодовый материал с неправильно заданным свойством Alpha Blend. Допускается значение <code>Alpha sort</code> , <code>Alpha blend</code> и <code>Add</code> при использовании ноды “REFRACTION”.
The NAME object has NAME armature modifier which references the wrong group. Modifier removed. Using B4W_REFRACTION node NODE_NAME with incorrect type of Alpha Blend. Material: NAME.	

Wind bending: not all vertex colors exist for “NAME”. Properties were set to default values.	Настройки процедурной анимации деревьев: должны существовать все указанные слои вертексных цветов.
Wind bending: vertex colors weren’t properly assigned for “NAME”. Properties were set to default values.	Настройки процедурной анимации деревьев; должны быть указаны названия всех слоев вертексных цветов (Main stiffness (A), Leaves stiffness (R), Leaves phase (G), Overall stiffness (B)), либо только главного (Main stiffness (A)), либо ни одного из них.
Wrong “Height Map” input for the “NAME” B4W_PARALLAX node. Only link from the TEXTURE node with a non-empty texture is allowed.	На вход “Height Map” ноды NAME типа B4W_PARALLAX подаются некорректные данные. Разрешено подавать только выход из ноды типа TEXTURE с заполненной текстурой.
Wrong texture coordinates type in texture NAME. [Material: NAME.]	Для текстур с изображением (image) поддерживаются следующие типы координат: UV, Normal и Generated.
Wrong vertex color layer is used in node “GEOMETRY”. [Material: NAME.]	В ноде “GEOMETRY” используется неправильно заданный вертексный слой.

8.6 Прочие сообщения

Сообщения выводятся в консоли браузера (горячая клавиша F12) при загрузке сцены. Сообщение имеет вид:

B4W EXPORT WARNING: Сообщение экспорта, требующее внимания пользователя



Сообщение об ошибке	Причина
Missing active camera or wrong active camera object	На сцене отсутствует активная камера (свойство Camera на вкладке Scene).
Missing world or wrong active world object	На сцене должен быть хотя бы один мир.
The “NAME” camera has unsupported PANORAMIC type. Changed to PERSPECTIVE type.”	Панорамная камера не поддерживается. Будет использована перспективная камера.

Объекты

Объекты служат целям размещения компонентов различного типа (мешей, камер, ламп и т.д.) в пространстве 3D сцены.

9.1 Типы

Движком поддерживаются объекты следующих типов:

- меш (mesh)
- камера (camera)
- лампа (lamp)
- пустой (empty)
- скелет (armature)
- источник звука (speaker)
- кривая (curve)
- текст (text)
- метасфера (metaball)
- поверхность (surface)

9.2 Статические и динамические объекты

Все объекты типа MESH можно разделить на статические и динамические.

Статические объекты - объекты, меши которых могут быть объединены друг с другом, если они имеют одинаковый материал.

Динамические объекты - объекты, меши которых не могут быть скомбинированы друг с другом.

Объединение статических объектов - т.н. батчинг (batching) - осуществляется автоматически при загрузке сцены в целях оптимизации количества вызовов отрисовки. Конвертация происходит, даже если на сцене присутствует всего один объект. Центр получившегося в результате конвертации меша находится в начале координат.

Среди объектов другого типа динамическими являются **CAMERA** и **ARMATURE**. Остальные - статические.

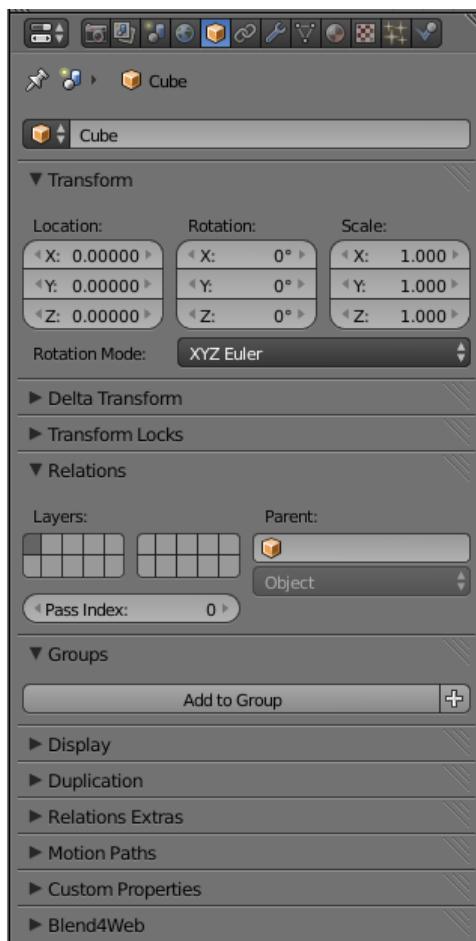
Объекты, имеющие анимацию, физику или родителя, являющегося динамическим объектом, рассматриваются как динамические.

Перемещение посредством API возможно только в случае динамических объектов. Чтобы обеспечить движение объекта, не имеющего настроек динамического объекта, в его настройках следует активировать опцию **Blend4Web > Force Dynamic Object**.

9.3 Настройка

Для объектов всех типов поддерживаются расположение в пространстве, указатель на блок данных, родительский объект, принадлежность к группе и ряд свойств, специфичных для Blend4Web.

9.3.1 Вкладка Object



Transform > Location Координаты местоположения.

Transform > Rotation Углы вращения. При повороте объекта доступны все режимы **Rotation Mode**, но при создании *объектной анимации* поддерживаются только Quaternion (WXYZ) и XYZ Euler.

Transform > Scale Масштабирование. Все 3 компоненты (x, y, z) должны быть одинаковы. Для физических объектов масштабирование не поддерживается.

Relations > Parent Указатель на родительский объект.

Groups Группы объектов, которым принадлежит данный объект.



Blend4Web > Do Not Export Не экспортовать.

Blend4Web > Apply Scale Применить масштабирование объекта при экспорте.

Blend4Web > Apply Modifiers Применить модификаторы объекта при экспорте.

При наличии модификатора SKIN рекомендуется применить его перед экспортом, т.к. он сбрасывает слои вертексных цветов и UV-текстур, что может привести к ошибкам экспорта.

Blend4Web > Export Vertex Animation Экспортить предварительно созданную и сохраненную вертексную анимацию. Актуально только для объектов типа MESH.

Blend4Web > Export Shape Keys Экспортить ключи деформации. Актуально только для объектов типа MESH.

Blend4Web > Export Edited Normals Экспортить предварительно отредактированные и сохраненные нормали. Актуально только для объектов типа MESH.

Примечание: Свойства *Apply Scale*, *Apply Modifiers*, *Export Vertex Animation*,

Export Edited Normals и *Export Shape Keys* взаимоисключают друг друга.

Blend4Web > Animation > Use Default Начать воспроизведение в движке связанный с объектом анимации.

Blend4Web > Animation > Animation blending Только для арматурных объектов. Разрешить плавный переход от одной скелетной анимации к другой.

Blend4Web > Animation > Behavior Поведение анимации при достижении последнего кадра: **Finish Stop** - остановить, **Finish Reset** - остановить и сбросить в нулевой кадр, **Loop** - циклически повторять.

Blend4Web > Do Not Batch Принудительно сделать объект **динамическим**.

Blend4Web > Force Dynamic Geometry Разрешить изменение геометрии объекта через API Blend4Web.

Blend4Web > Disable Frustum Culling Отключить оптимизацию отсечением по зоне видимости.

Blend4Web > Disable Fogging Отключить туман для объекта.

Blend4Web > Do Not Render Отключить рендеринг объекта (например, вспомогательный объект физики).

Blend4Web > Shadows: Cast Объект будет отбрасывать тени.

Blend4Web > Shadows: Cast Only Объект будет отбрасывать тени, при этом сам будет оставаться невидимым. Становится доступным после включения **Blend4Web > Shadows: Cast**.

Blend4Web > Shadows: Receive Разрешить затенение от прилегающих объектов (получение теней) на объекте.

Blend4Web > Reflections: Reflexible При включении объект будет отражаться от зеркальных поверхностей.

Blend4Web > Reflections: Reflexible only При включении объект будет отражаться, при этом сам будет оставаться невидимым. Становится доступным после включения **Blend4Web > Reflections: Reflexible**.

Blend4Web > Reflections: Reflective При включении объект будет отражать своей поверхностью другие объекты.

Blend4Web > Reflections: Reflection Plane Текстовое поле для названия пустого объекта, задающего плоскость отражения. Становится доступным после включения **Blend4Web > Reflections: Reflective**.

Blend4Web > Caustics На объекте будут видны эффекты от прилегающей к нему воды (каустика).

Blend4Web > Wind Bending Включить процедурную анимацию под действием ветра.

Blend4Web > Selectable Разрешить *выделение объекта* с помощью мыши или другого устройства ввода.

Blend4Web > Enable Outlining Разрешить подсвечивание объекта (*эффект Outline*).

Blend4Web > Billboard Использовать объект в качестве билборда (т.е. автоматически ориентировать его относительно камеры).

Blend4Web > Preserve Global Orientation and Scale Билбординг объекта с учетом его ориентации и масштабирования в мировом пространстве. Объект всегда будет обращен к камере той стороной, которая видна при взгляде на него вдоль направления оси Y в системе координат Blender'a. Становится доступным после включения Blend4Web > Billboard.

Blend4Web > Billboard Type Способ ориентирования билборда. *Spherical* (по умолчанию) - объект всегда ориентирован к камере одной стороной независимо от угла обзора, *Cylindrical* - аналогично *Spherical*, только вращение ограничено осью Z мирового пространства в Blender. Становится доступным после включения Blend4Web > Billboard.

Blend4Web > LOD Transition Ratio Параметр для сглаживания переключения между LOD объектами. Определяет, на каком дополнительном расстоянии LOD объекты будут видны до замены последующими LOD объектами. Назначается для главного объекта. Измеряется в долях от радиуса охватывающей сферы объекта.

Blend4Web > Enable Meta Tags Интерфейс добавления мета-тегов на объект:

Blend4Web > Tags > Title Заголовок объекта.

Blend4Web > Tags > Category Категория объекта.

Blend4Web > Tags > Description Описание объекта. В зависимости от значения опции *Description Source* в это поле заносится либо текст описания непосредственно, либо имя файла, в котором содержится описание.

Blend4Web > Tags > Description Source Тип (источник) описания объекта: текст либо файл с текстом.

Blend4Web > Anchor > Enable Anchor Опция доступна только для объектов типа EMPTY. Интерфейс добавления графических меток (якорей) к объектам:

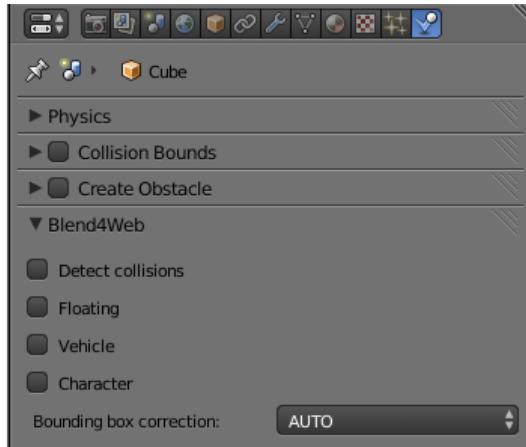
Blend4Web > Anchor > Type Тип якоря. Аннотация (*Annotation*) - информация из мета-тегов объектов отображается в раскрывающихся информационных окнах стандартного дизайна. Элемент (*Custom Element*) - в качестве якоря можно назначить произвольный HTML-элемент с текущей веб-страницы. Общий (*Generic*) - якорь, положение которого можно определить с помощью API модуля *anchors*.

Blend4Web > Anchor > Detect Visibility Определять видимость якоря, т.е. реагировать на перекрытие объекта-якоря другими объектами. Активация опции приводит к падению производительности рендеринга и должна рассматриваться только в случае реальной необходимости.

Blend4Web > Anchor > Max Width Используется для якорей-аннотаций. Ограничить всплывающее информационное окно аннотации выбранным

значением (в CSS пикселях).

9.3.2 Вкладка Physics (в режиме Blender Game)



Blend4Web > Detect Collisions Активировать связанную с объектом физику.

Blend4Web > Floating Сделать объект плавучим. Подробное описание настроек плавучих объектов можно найти в разделе *физика*.

Blend4Web > Vehicle Активировать использование объекта в качестве составной части транспортного средства. Подробное описание настроек транспортных средств можно найти в разделе *физика*.

Blend4Web > Character Активировать использование объекта в качестве физического каркаса игрового персонажа. Подробное описание настроек персонажа можно найти в разделе *физика*.

9.4 Управление перемещением объектов

Примечание: Необходимо обеспечить, чтобы объект, над которым выполняется преобразование, был *динамическим*.

Для управления перемещением объектов в движке предусмотрены следующие базовые функции модуля *transform*:

get_translation Получить координаты центра объекта. Вариант с одним параметром возвращает новый вектор (неоптимизированный вариант), вариант с двумя требует отдельного вектора для записи результата.

get_rotation Получить кватернион поворота объекта. По аналогии с *get_translation* имеется два варианта вызова функции.

get_scale Получить значение коэффициента увеличения объекта.

set_translation, set_translation_v Переместить центр объекта в указанное место. Первая функция принимает в качестве параметров отдельные координаты, вторая - трёхмерный вектор (Array или Float32Array).

set_rotation, set_rotation_v Установить кватернион поворота объекта. Первая функция принимает в качестве параметров отдельные координаты, вторая - четырёхмерный вектор (Array или Float32Array).

set_scale Установить коэффициент увеличения объекта. Единица соответствует исходному состоянию. Значение меньше единицы - уменьшение. Значение больше единицы - увеличение. Не все объекты могут быть увеличены. В частности, увеличение невозможно для физических объектов.

set_rotation_euler, set_rotation_euler_v Установить поворот объекта с помощью углов Эйлера. Используется **внутренняя** (intrinsic) система поворота YZX (то есть углы следуют в последовательности YZX, сама система отсчитывается при этом вращается и занимает каждый раз новое положение).

9.5 Кватернионы

Кватернионы представляют собой четырёхмерные векторы, используемые для осуществления поворотов. Использование кватернионов обладает рядом преимуществ перед другими способами представления поворотов:

- Не имеет неоднозначности и зависимости от порядка применения поворотов, которые имеют место в случае использования углов Эйлера.
- Более эффективное использование памяти (от 2-х до 4-х раз меньше в зависимости от типа используемой матрицы).
- Высокая эффективность вычисления серии поворотов, чем при использовании матриц.
- Нейтрализация ошибок умножения, возникающих вследствие неточности чисел с плавающей запятой.
- Удобный метод интерполяции.

Кватернионы имеют ряд недостатков:

- Поворот вектора с помощью кватерниона более сложная в вычислительном плане операция чем поворот с использованием матрицы.
- Использование кватернионов для представления отличных от поворота преобразований (перспективная или ортогональная проекция) затруднено.

Для удобства работы с кватернионами в движке имеется ряд функций:

quat.multiply Умножение кватернионов. Умножение кватерниона А на кватернион Б слева А*Б является поворотом на А. То есть у объекта уже имеется некий поворот Б, который мы дополняем новым поворотом на А.

quat.setAxisAngle Кватернион представляет собой иную форму записи поворота относительно произвольной оси (вектора) на произвольный угол. Положительное направление поворота отсчитывается против часовой стрелки, если смотреть с конца вектора. Например вызов `quat.setAxisAngle([1, 0, 0], Math.PI/3, quat)` сформирует кватернион, который может быть использован для осуществления поворота относительно оси X на 60 градусов (против часовой стрелки, если смотреть с конца оси X).

quat.slerp Сферическая интерполяция кватернионов. Используется для осуществления плавного разворота объектов и анимации.

util.euler_to_quat, util.quat_to_euler. Преобразование из углов Эйлера и наоборот.

9.5.1 Пример работы с кватернионами

Требуется повернуть объект на 60 градусов в горизонтальной плоскости вправо. В программе Blender имеется соответствующая модель с названием “Cessna”.



Сохраним ссылку на объект в движке в переменной `aircraft`:

```
var aircraft = m_scenes.get_object_by_name("Cessna");
```

О我们将 осуществим поворот:

- Расположение координатных осей в Blender'е и в движке различается, и при экспорте произойдет преобразование: [X Y Z] (Blender) -> [X -Z Y] (движок). Поэтому поворот нужно осуществлять относительно оси Y, а не Z.
- Повороту вправо соответствует поворот по часовой стрелке, т.е. в отрицательном направлении.
- 60 градусов = $\pi/3$ радиан.

В итоге получаем:

```
// compose quaternion
var quat_60_Y_neg = m_quat.setAxisAngle([0, 1, 0], -Math.PI/3, m_quat.create());

// get old rotation
var quat_old = m_transform.get_rotation(aircraft);

// left multiply: quat60_Y_neg * quat_old
var quat_new = m_quat.multiply(quat_60_Y_neg, quat_old, m_quat.create());

// set new rotation
m_transform.set_rotation_v(aircraft, quat_new);
```

Оптимизированный вариант, не приводящий к созданию новых объектов:

```
// cache arrays as global vars
var AXIS_Y = new Float32Array([0, 1, 0])
var quat_tmp = new Float32Array(4);
var quat_tmp2 = new Float32Array(4);
...
// rotate
m_quat.setAxisAngle(AXIS_Y, -Math.PI/3, quat_tmp);
m_transform.get_rotation(aircraft, quat_tmp2);
m_quat.multiply(quat_tmp, quat_tmp2, quat_tmp);
m_transform.set_rotation_v(aircraft, quat_tmp);
```

9.6 Перемещение с помощью векторов TSR

Удобным способом управления перемещением объектов является использование специальных векторов, имеющих следующий формат:

$$[T_x, T_y, T_z, S, R_x, R_y, R_z, R_w]$$

Здесь T_x, T_y, T_z - компоненты вектора перемещения, S - масштаб, R_x, R_y, R_z, R_w - компоненты кватерниона поворота. Отсюда следует название вектора: TSR или TSR-8.

Для работы с данными векторами существует специальный модуль *tsr*, а также методы *set_tsr()*/*get_tsr()* модуля *transform*.

9.7 Копирование объектов (инстансинг)

Часто возникает необходимость копирования объектов (инстансинг) во время работы приложения.

Существует ряд ограничений на копируемые объекты:

- тип объекта должен быть MESH
- объект должен быть *динамическим* (включить настройку Blend4Web > Force Dynamic Object)
- объект должен принадлежать активной сцене

9.7.1 Простое копирование

В случае простого копирования для объекта-копии будет использоваться геометрия объекта, с которого осуществлялось копирование. При изменении геометрии на исходном объекте, геометрия изменится и на объекте-копии.

9.7.2 Глубокое копирование

При глубоком копировании новый объект будет обладать уникальными для него свойствами, а именно, он будет иметь свою собственную геометрию. То есть при изменении геометрии на объекте-оригинале у объекта-копии геометрия изменяться не будет. Для такого типа копирования на исходном объекте нужно дополнительно выставить опцию Blend4Web > Force Dynamic Geometry.

Для копирования объектов в модуле `objects.js` предусмотрена функция `copy`, в которую надо передать идентификатор копируемого объекта, уникальное имя нового объекта и булеву переменную, определяющую тип копирования: `true`- глубокое, `false` - простое. По умолчанию будет осуществлено простое копирование.

Созданный при копировании объект необходимо добавить на сцену. Для этого в модуле `scenes.js` существует функция `append_object`, куда нужно передать объект, полученный копированием.

```
// ...
var new_obj = m_objects.copy(obj, "New_name", true);
m_scenes.append_object(new_obj);
m_transform.set_translation(new_obj, 2, 0, 2);
// ...
```

9.7.3 Удаление объектов

Для удаления объектов, созданных копированием, в модуле `scenes.js` предусмотрена функция `remove_object`, куда надо передать объект. Возможно удаление только объектов-копий.

```
// ...
m_objects.remove_object(new_obj);
// ...
```

9.8 Выделение объектов

Чтобы разрешить выделение конкретного объекта, необходимо установить флаг `Selectable` в панели `Object > Blend4Web`.

Примечание: Следует убедится в том, что в панели `Scene > Blend4Web` поле `Enable Object Select` установлено в состояние `YES` или `AUTO`.

Выделение объекта возможно программно через API, например, в модуле `scenes.js` имеется функция `pick_object`, которая по координатам в canvas'е выделяет объект,

```
// ...
var x = event.clientX;
var y = event.clientY;

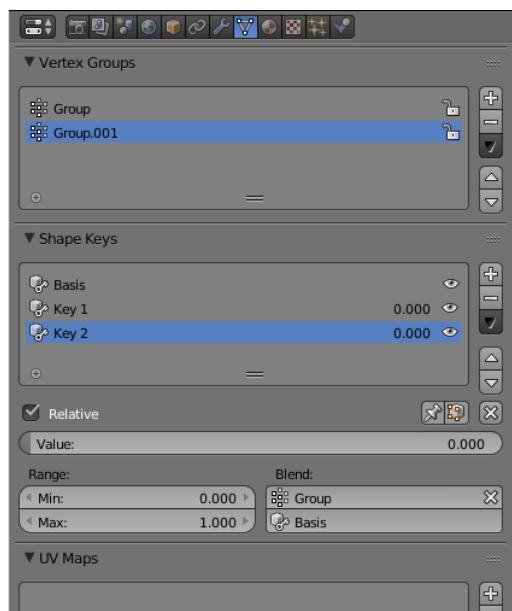
var obj = m_scenes.pick_object(x, y);
// ...
```

или с использованием *NLA-скрипта*.

Если для выделяемого объекта в панели `Object > Blend4Web` установлены флаги `Enable Outlining` и `Outline on Select`, то вызов функции `pick_object` приведет к активации *Outline-анимации*.

9.9 Морфинг

Для создания ключей деформации (опорных фигур) перейдите во вкладку `Mesh > Shape keys`.



Движком поддерживаются все опции типа “Relative” для ключей деформации.

Для изменения значения ключа деформации, существует функция `apply_shape_key` в модуле `geometry.js`.

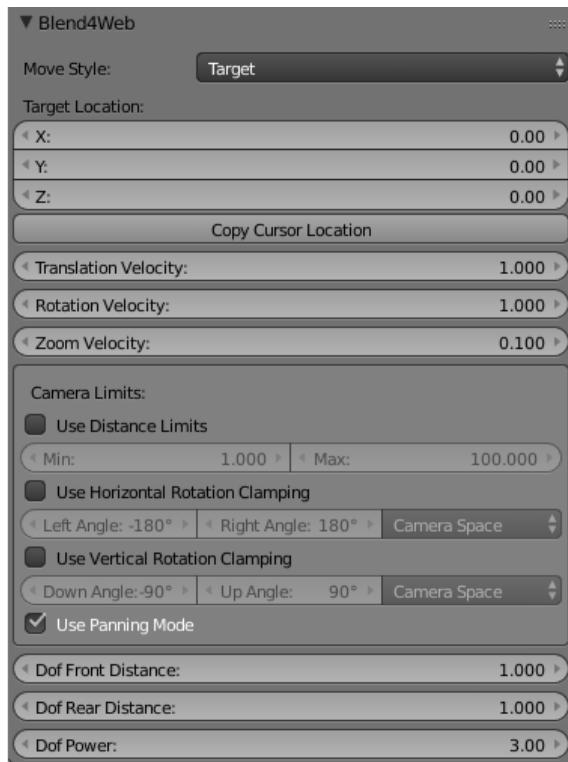
Примечание: На объекте должно стоять свойство *Export Shape Keys*.

```
// ...
var obj = m_scenes.get_object_by_name("Object");
m_geometry.apply_shape_key(obj, "Key 1", 0.5);
// ...
```

Камера

10.1 Режимы управления и общие настройки

Настройки камеры выставляются в панели Properties на вкладке Object Data.



Blend4Web > Move Style Режим управления камерой:

- **Target** По умолчанию камера находится в режиме вращения вокруг некоторой точки (target). Положение точки вращения может быть изменено (см. [панорамирование](#)).

- **Eye** Режим взгляда позволяет осуществлять вращение и перемещение от первого лица.
- **Hover** В режиме нависания камера движется параллельно горизонтальной плоскости.
- **Static** В статическом режиме изменение положения камеры осуществляется посредством анимации или через API.

Blend4Web > Target Location Доступно в режиме Target. Позиция точки, относительно которой будет вращаться камера. Кнопка **Copy Cursor Location** позволяет скопировать текущее положение курсора.

Blend4Web > Dof Front Distance Описано в разделе *Глубина резкости камеры*.

Blend4Web > Dof Rear Distance Описано в разделе *Глубина резкости камеры*.

Blend4Web > DOF Power Описано в разделе *Глубина резкости камеры*.

10.2 Скорость движения камеры

Для камеры доступно несколько настроек скорости движения.

Blend4Web > Translation Velocity Доступно в режимах Target, Eye, Hover. Задаёт скорость перемещения камеры.

Blend4Web > Rotation Velocity Доступно в режимах Target, Eye, Hover. Задаёт скорость вращения камеры.

Blend4Web > Zoom Velocity Доступно в режимах Target, Hover. Задаёт скорость приближения камеры к опорной точке.

Значение параметров по умолчанию: Translation Velocity, Rotation Velocity: 1; Zoom Velocity: 0.1.

Допустимые значения параметров: Translation Velocity, Rotation Velocity: $[0, \infty)$; Zoom Velocity: $[0, 1]$.

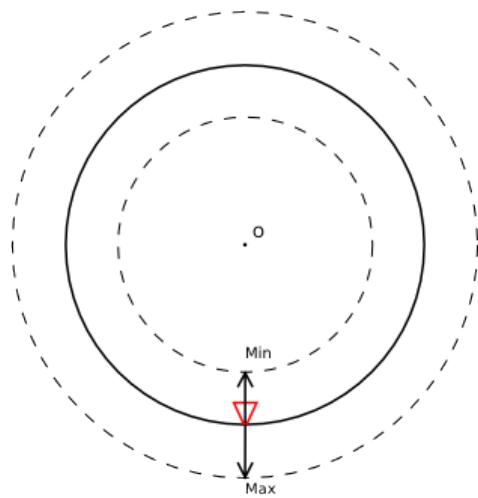
10.3 Ограничения движения камеры

Для камеры доступно несколько настроек, ограничивающих/изменяющих её движение тем или иным образом. Они объединены в группу **Camera Limits**.

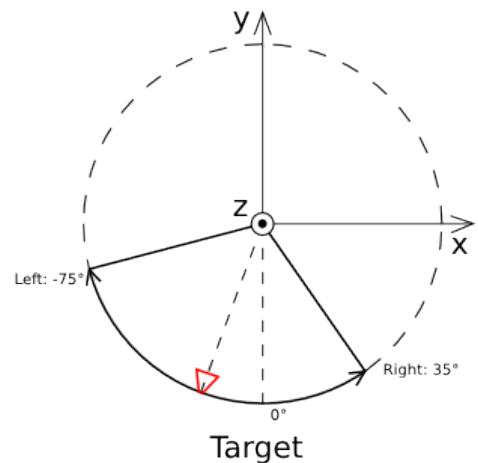


Камера типа Target

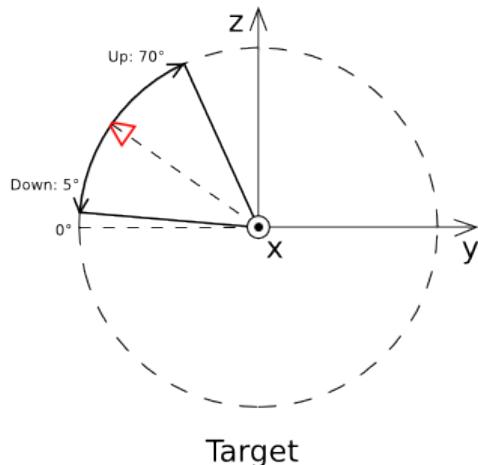
Blend4Web > Use Distance Limits Задание минимального и максимального расстояний от камеры до точки вращения. Допустимые значения: $Min \leq Max$. По умолчанию отключено.



Blend4Web > Use Horizontal Rotation Clamping Ограничение горизонтального (вокруг мировой оси Z в Blender'e) вращения камеры относительно соответствующей точки. Вращение происходит по дуге окружности определяемой значениями **Left Angle** и **Right Angle**. Дуга вращения соответствует движению из **Left Angle** в **Right Angle** против часовой стрелки. По умолчанию отключено.



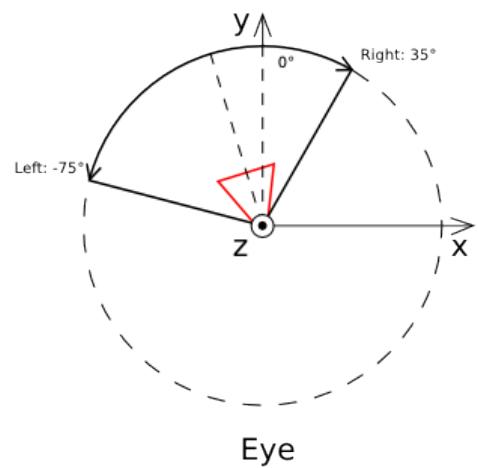
Blend4Web > Use Vertical Rotation Clamping Ограничение вертикального (вокруг локальной оси X камеры в Blender'e) вращения камеры относительно соответствующей точки. Вращение происходит по дуге окружности определяемой значениями **Down Angle** и **Up Angle**. Дуга вращения соответствует движению из **Down Angle** в **Up Angle** по часовой стрелке. По умолчанию отключено.



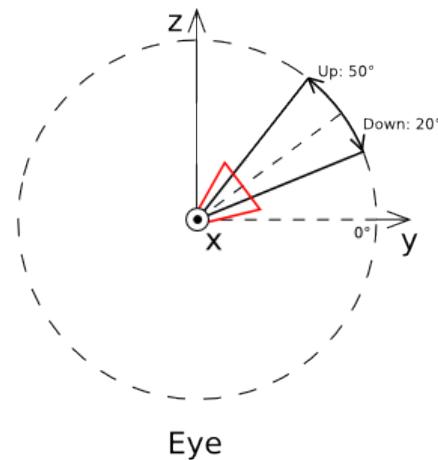
Blend4Web > Use panning mode Разрешить панорамное перемещение камеры.

Камера типа Eye

Blend4Web > Use Horizontal Rotation Clamping Ограничение горизонтального (вокруг мировой оси Z в Blender'e) вращения камеры относительно своего местоположения. Вращение происходит по дуге окружности определяемой значениями Left Angle и Right Angle. Дуга вращения соответствует движению из Left Angle в Right Angle по часовой стрелке. По умолчанию отключено.



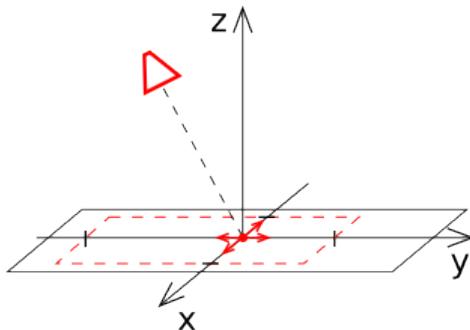
Blend4Web > Use Vertical Rotation Clamping Ограничение вертикального (вокруг локальной оси X камеры в Blender'e) вращения камеры относительно своего местоположения. Вращение происходит по дуге окружности определяемой значениями Down Angle и Up Angle. Дуга вращения соответствует движению из Down Angle в Up Angle против часовой стрелки. По умолчанию отключено.



Камера типа Hover

Blend4Web > Use Horizontal Translation Limits Ограничение перемещения опорной точки вдоль оси X в мировых координатах в Blender'e. Допустимые значения: $Min \leq Max$. По умолчанию отключено.

Blend4Web > Use Vertical Translation Limits Ограничение перемещения опорной точки вдоль оси Y в мировых координатах в Blender'e. Допустимые значения: $Min \leq Max$. По умолчанию отключено.

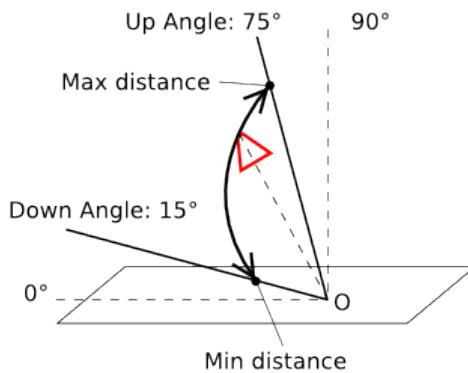


Blend4Web > Use Distance Limits Задание минимального и максимального расстояний от камеры до точки пересечения (опорная точка) направления взгляда камеры с горизонтальной плоскостью (по умолчанию с плоскостью XOY мировых координат в Blender'e). Допустимые значения: $Min \leq Max$. При отключении камера будет свободно двигаться вдоль оси Z мировых координат в Blender'e, опорная точка не будет задана, а настройки Blend4Web > Use Horizontal Translation Limits и Blend4Web > Use Vertical Translation

Limits будут ограничивать позицию самой камеры. По умолчанию отключено.

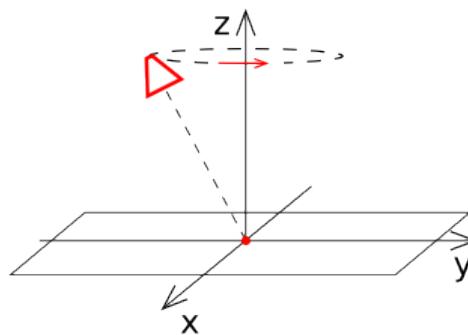
Blend4Web > Camera Angle Limits Ограничения угла подъема камеры (угла между направлением взгляда камеры и горизонтальной плоскостью). Ставятся доступными при включении опции Blend4Web > Use Distance Limits. Допустимые значения: $0 \leq \text{Down Angle} \leq \text{Up Angle} \leq 90$. Значения по умолчанию: *Down Angle = 0*, *Up Angle = 90*.

При включении настройки Blend4Web > Use Distance Limits лимиты по расстоянию и углу подъема будут действовать одновременно, задавая траекторию движения камеры в вертикальной плоскости.



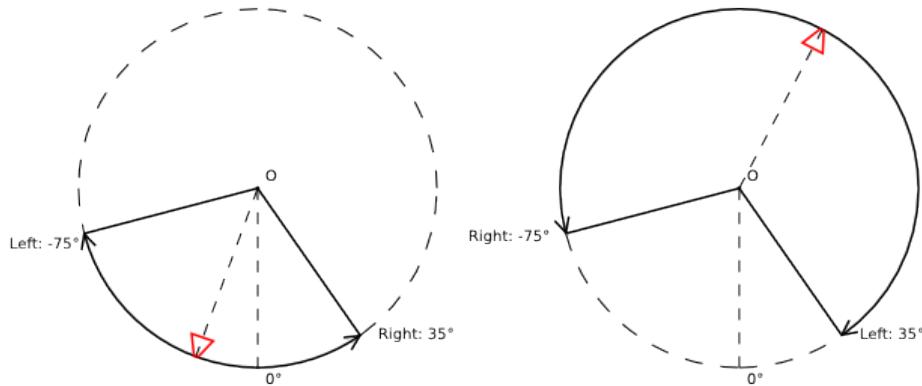
Некорректное задание лимитов по расстоянию либо углу подъема приведет к отключению опции Blend4Web > Use Distance Limits.

Blend4Web > Use Horizontal Rotation Разрешить вращение камеры в плоскости XOY Blender'a относительно опорной точки. Ставятся доступными при включении опции Blend4Web > Use Distance Limits. По умолчанию включено.



Особенности задания ограничений

- Для EYE/TARGET камеры перемена мест значений *Left/Right* или *Down/Up* приводит к движению по противоположной дуге окружности вращения.



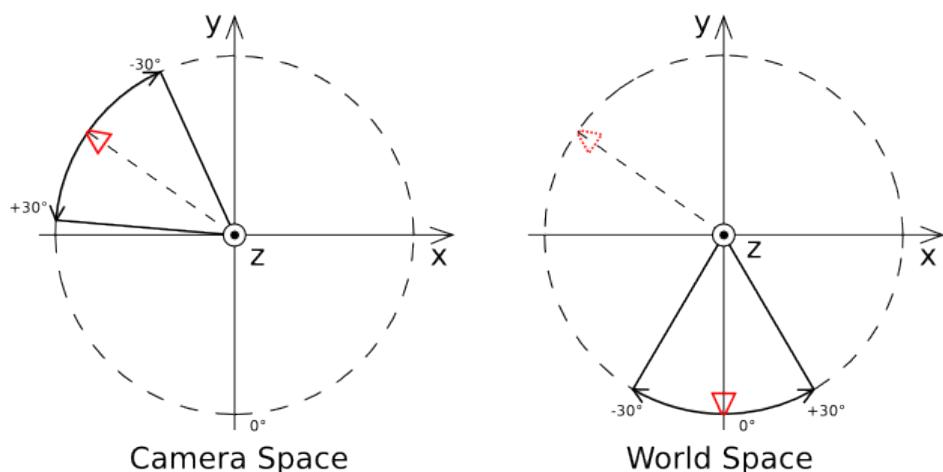
- При ограничении горизонтального и вертикального вращения камеры можно выбрать пространство координат:

Camera Space Все углы отсчитываются относительно начального положения и ориентации камеры.

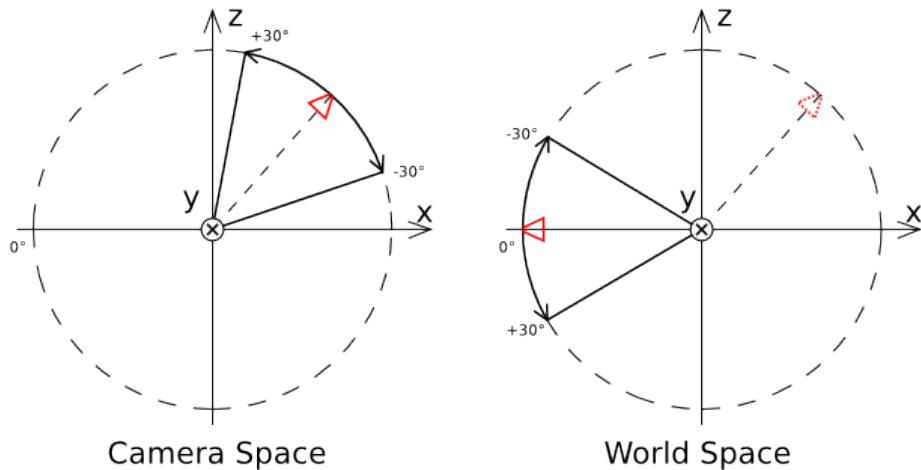
World Space Горизонтальные углы отсчитываются относительно направления оси Y в пространстве мировых координат, вертикальные - относительно плоскости XOY в мировом пространстве в Blender'e.

Значение по умолчанию: Camera Space.

Ограничения по горизонтали на примере TARGET камеры:



Ограничения по вертикали на примере TARGET камеры:

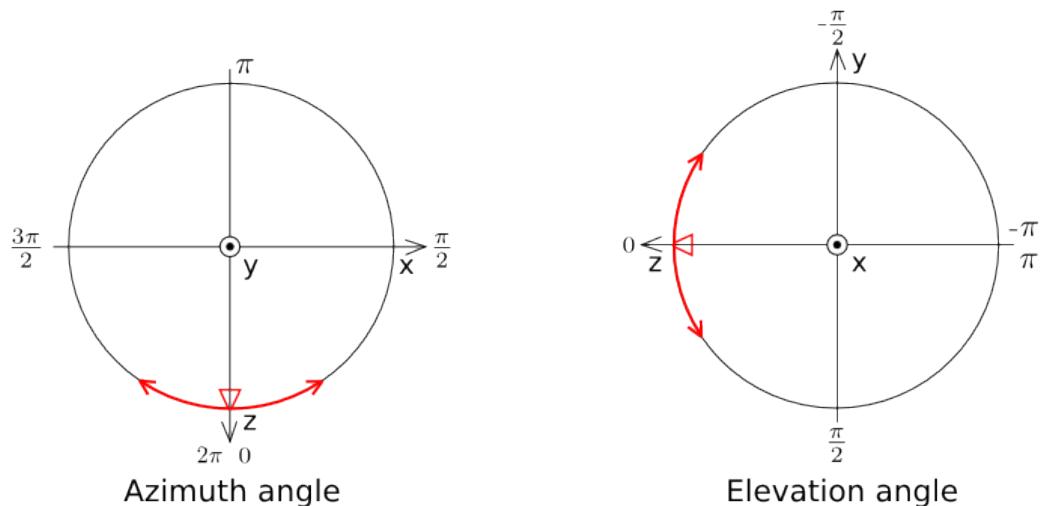


10.4 Управление камерой через API

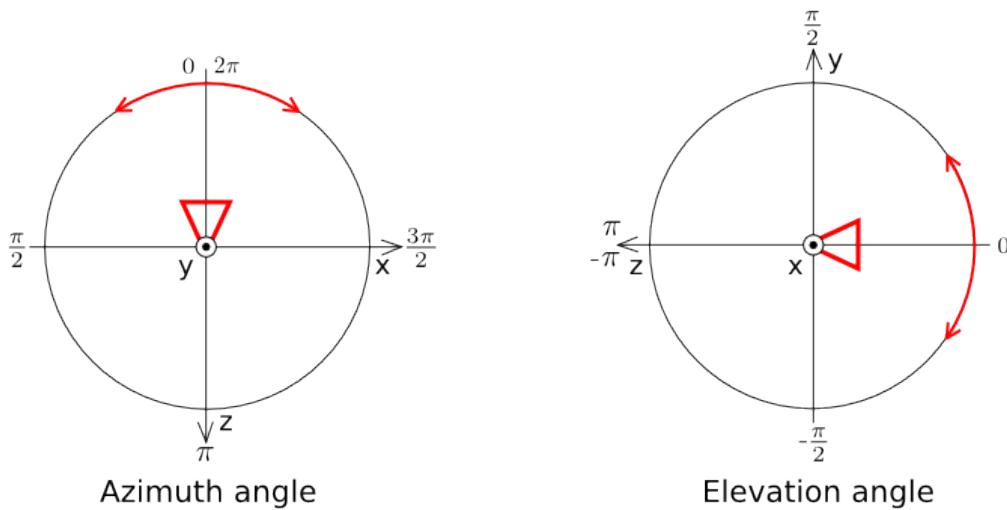
Основные функции управления камерой находятся в модуле `camera.js`.

При работе с камерой через API (вращение, выставление ограничений) отсчет угловых координат ведется в системе координат движка следующим образом:

Для типов TARGET/HOVER:

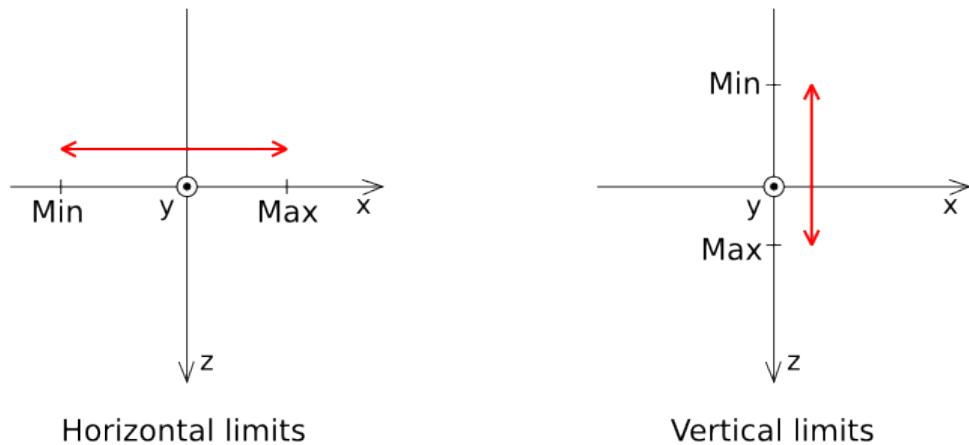


Для типа EYE:



При выставлении лимитов перемещения для камеры типа `HOVER` значения лимитов будут соответствовать осям системы координат движка:

- Horizontal Translation Limits - ось X
- Vertical Translation Limits - ось Z



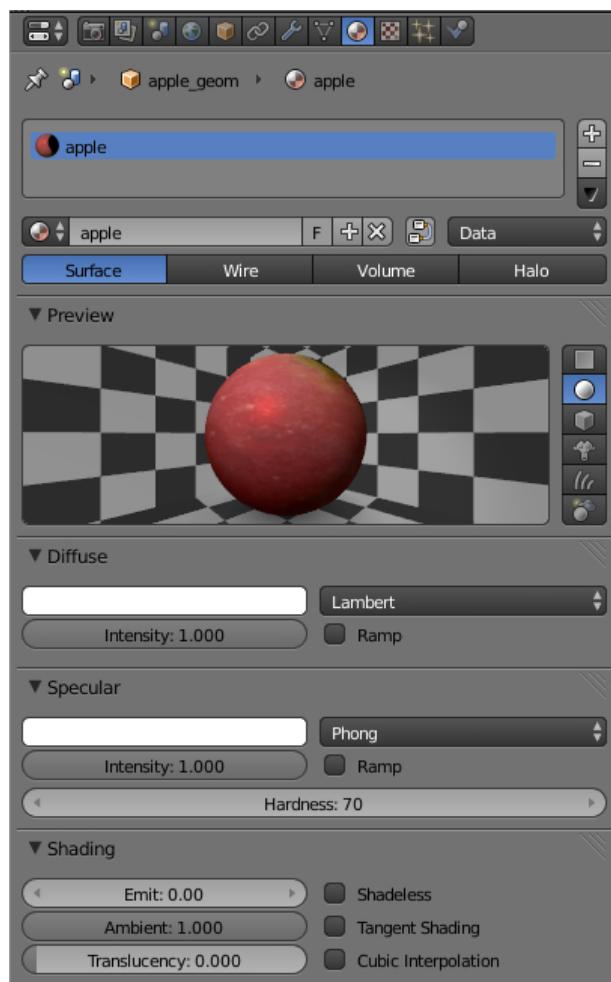
Материалы

Материалы описывают реакцию поверхности объекта на освещение, а также содержат информацию о ее прозрачности, отражающей способности, физических и других параметрах.

Меши могут использовать один или несколько материалов. В случае использования нескольких материалов назначение их различным полигонам происходит в режиме редактирования **Edit Mode**. Для этого нужно выделить нужные полигоны, выбрать материал из списка и нажать кнопку **Assign**.

Поддерживаются следующие типы материалов: **Surface** (поверхность), **Halo** (гало).

11.1 Параметры освещения



Diffuse > Color Цвет диффузного (рассеянного) света. Значение по умолчанию (0.8, 0.8, 0.8). Может взаимодействовать с цветом диффузной текстуры.

Diffuse > Intensity Интенсивность диффузного (рассеянного) света. Значение по умолчанию 0.8.

Diffuse > Shader Алгоритм расчета диффузного (рассеянного) освещения. Движок поддерживает следующие алгоритмы: Ламберта (Lambert), Орена-Найара (Oren-Nayar), Френеля (Fresnel). Значение по умолчанию Lambert.

Specular > Color Цвет отраженного света. Значение по умолчанию (1.0, 1.0, 1.0). Может взаимодействовать с цветом карты бликов.

Specular > Intensity Интенсивность отраженного света. Значение по умолчанию 0.5.

Specular > Hardness Степенный показатель в формуле расчета отраженного света ("жесткость" блика). Значение по умолчанию 50. Алгоритм применения в движке отличается от алгоритма применения в Blender'e.

Specular > Shader Алгоритм расчета отраженного освещения. Движок поддерживает следующие алгоритмы: Кука-Торранса (CookTorr), Фонга (Phong) - пове-

дение обоих одинаковое, а также Уарда анизотропный (*WardIso*). Значение по умолчанию *CookTorr*.

Shading > Emit Интенсивность эмиссии (излучения). Значение по умолчанию 0.0.

Shading > Ambient Фактор влияния освещения от окружающей среды на материал. Значение по умолчанию 1.0.

Shading > Shadeless При включении материал не реагирует на освещение. По умолчанию выключено.

Game Settings > Backface Culling При включении обратная сторона полигона не отображается движком. По умолчанию включено.

Options > Vertex Color Paint Опция включает использование вертекского цвета меша вместо диффузного цвета.

11.2 Прозрачность

11.2.1 Типы

Тип реализации прозрачности выбирается в меню *Alpha Blend* на панели *Materials > Game Settings* (в режиме *Blender Game*).

Движком поддерживаются следующие типы реализации прозрачности, перечисленные в порядке увеличения производительности:

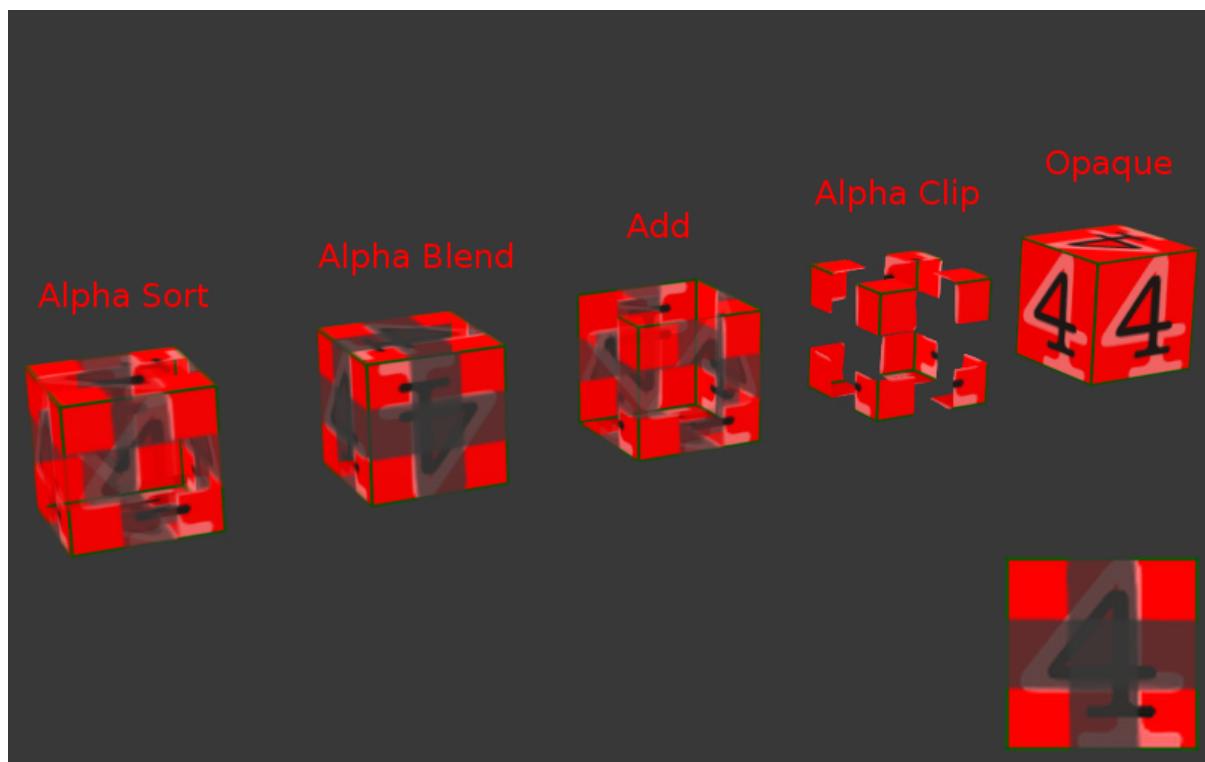
Alpha Sort Прозрачный с градиентом. Для корректного отображения перекрывания одних прозрачных поверхностей другими движком производится сортировка треугольников по дальности от камеры. Операция требует дополнительных затрат вычислительных ресурсов. Рекомендуется применять для замкнутых прозрачных объектов (бутилка, стекла автомобиля и т.д.).

Alpha Blend Прозрачный с градиентом. Сортировка треугольников не производится. Рекомендуется применять для незамкнутых прозрачных объектов (поверхность воды, декали).

Add Прозрачный с градиентом. Сортировка треугольников не производится. Движок отключает запись в буфер глубины, что приводит к произвольному порядку отображения прозрачных поверхностей. Рекомендуется применять для создания эффектов (системы частиц, светящиеся лучи).

Alpha Clip Прозрачный без градиента. Движок отбрасывает пиксели (*discard*) с прозрачностью менее 0.5. Сортировка треугольников не производится. Рекомендуется применять с текстурой в качестве маски для визуализации множества мелких деталей (листва деревьев, трава).

Opaque Непрозрачный. Альфа-канал игнорируется. Значение по умолчанию.



11.2.2 Дополнительные настройки

Transparency Опция включения прозрачности требуется для отображения прозрачных объектов в Blender'e. Движок игнорирует эту опцию, используя вместо нее Alpha Blend.

Transparency > Alpha Уровень прозрачности материала. При наличии диффузной текстуры движок (в отличие от Blender'a) игнорирует этот параметр, используя вместо него значение прозрачности текстуры.

Options > Z Offset, смещение по глубине Используется для явного указания расположения прозрачных объектов с **разными** материалами относительно друг друга с целью сортировки по глубине. Может принимать отрицательные и положительные значения. Для корректного отображения дальние объекты должны иметь меньшее значение параметра, чем ближние. Значение по умолчанию 0.0.

Transparency > Fresnel Степень Френеля для прозрачности. Экспортируется, но в настоящее время не используется.

Transparency > Blend Фактор Френеля для прозрачности. Экспортируется, но в настоящее время не используется.

11.3 Зеркальное отражение

11.3.1 Статическое отражение

Поверхность отражает одно и то же изображение вне зависимости от изменения окружающей среды. Для активации достаточно использовать карту зеркального отражения.

См.также:

Эффект Френеля для отражения

11.3.2 Динамическое отражение

Поверхность отражает текущее расположение определенных объектов. Поддерживается только отражение от плоскости.

Активация

1. Включить опцию `Render reflections` на панели `Scene > Blend4Web`.
2. Добавить пустой объект для задания плоскости отражения `Add > Empty > Single Arrow`. Переименовать для удобства.
3. Для отражающих объектов на панели `Object > Blend4Web` выставить опцию `Reflective` и указать имя пустого объекта в поле `Reflection plane`.
4. Для нужных материалов отражающих объектов выставить значение отражающей способности `Mirror > Reflectivity`.
5. Для отражаемых объектов на панели `Object > Blend4Web` выставить опцию `Reflexible`.

Примечание: Рекомендуется также включить использование освещения от окружающей среды `World > Environment Lighting`.

Ограничения

В отраженном изображении игнорируется карта нормалей, тени.

См.также:

Эффект Френеля для отражения

11.3.3 Эффект Френеля для отражения

Эффект Френеля проявляется в зависимости интенсивностей проходящего и отраженного света от угла падения. Если угол падения близок к нулю (т.е. свет падает почти перпендикулярно поверхности), доля проходящего света велика, а отраженного мала. И наоборот, если угол падения близок к 90 градусам (т.е. свет падает почти параллельно поверхности), отражается почти весь свет.

Движок использует приближенную формулу Шлика:

$$R = R_0 + (1 - R_0)(1 - \cos \theta)^N, \text{ где}$$

R - коэффициент отражения,

R_0 - коэффициент отражения в случае обзора под прямым углом к поверхности (т.е. при $\theta = 0$),

θ - угол падения (равный углу отражения, под которым свет попадает в камеру), рассчитывается движком в реальном времени,

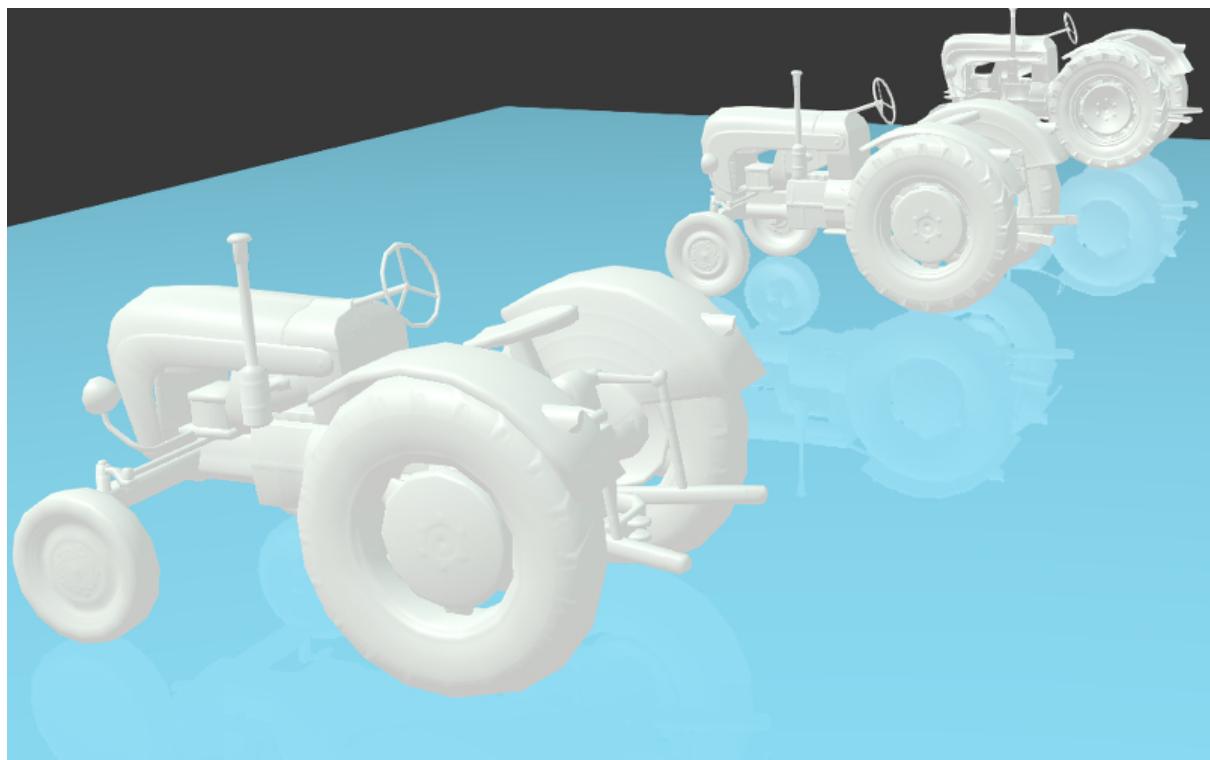
N - показатель степени.

Настройка

Эффект Френеля применяется как для статического, так и для динамического отражения.

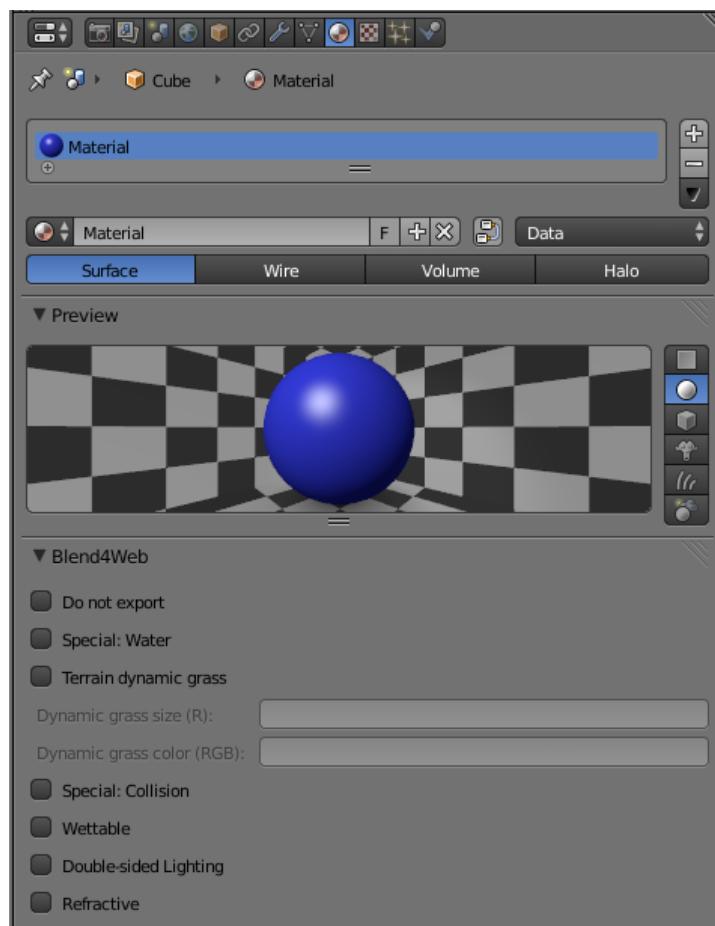
Mirror > Fresnel Степень Френеля для отражения. Показатель степени N в формуле Шлика. В пакете Blender ограничен значениями от 0 до 5. Если этот параметр равен нулю, эффект Френеля не проявляется, происходит *полное отражение* на всех углах. Если этот параметр больше нуля, при обзоре поверхности под углами, близкими к прямому (почти перпендикулярно поверхности), материал становится менее отражающим. Чем больше этот параметр, тем больше отклонение угла от прямого, для которого наблюдается такой эффект.

Mirror > Blend Фактор Френеля для отражения. Приводится к R_0 в формуле Шлика: $R_0 = 1 - Blend / 5$. В пакете Blender ограничен значениями от 0 до 5. Этот параметр показывает интенсивность проявления эффекта Френеля: чем больше фактор Blend, тем сильнее влияние эффекта Френеля. Если он равен нулю, эффект Френеля не проявляется.



11.4 Специальные параметры движка

Располагаются в панели Blend4Web.



Do not export Не экспортовать.

Render above all Материал отрисовывается поверх всех объектов на сцене. Требует тип прозрачности с градиентом (Add, Alpha Blend или Alpha Sort)

Special: Water Специальный материал для [рендеринга воды](#).

Terrain dynamic grass Материал используется для [рендеринга травяного покрова](#).

Special: Collision Специальный материал для физического объекта.

См.также:

[Физика](#)

Wettable Для материала активируется эффект смачивания водой.

См.также:

[Вода](#)

Double-sided Lighting Включить двухстороннее освещение. Опция полезна для однослойных непросвечивающих объектов.

Refractive Включить рефракцию. Коэффициент возмущения задаётся опцией **Reftaction bump** на панели **Reftaction Settings**. Значение по умолчанию 0.001.

Примечание: Для отображения эффекта необходимо включить опцию **Render refraction** на панели **Scene > Blend4Web**. Объект должен быть с типом прозрачности **Alpha Blend**.

См.также:

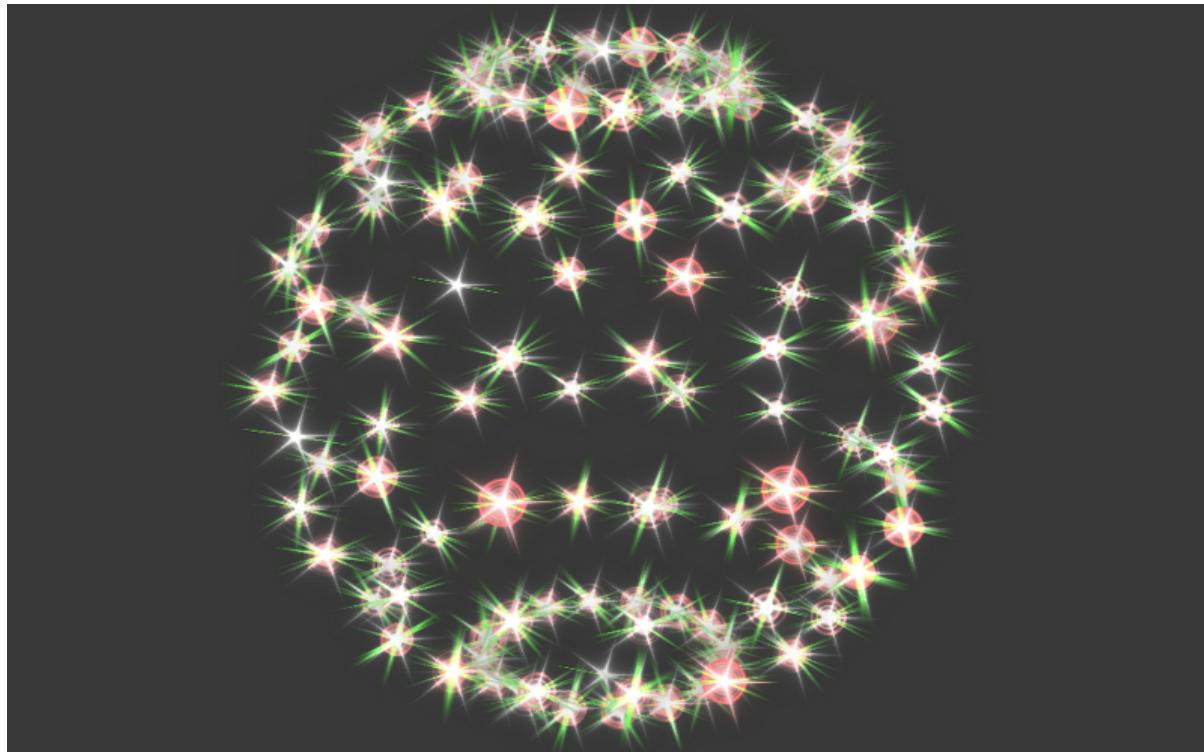
[Прозрачность](#)

11.5 Материалы гало (Halo)

Используются в системах частиц и в статических мешах. Ниже рассматривается использование гало на статических мешах.

11.5.1 Активация

Выставить тип **Halo** во вкладке **Materials**. Рекомендуется также выставить тип прозрачности с градиентом (**Add**, **Alpha Blend** или **Alpha Sort**).



11.5.2 Дополнительные настройки

Halo > Alpha Параметр прозрачности материала. Значение по умолчанию 1.0 (непрозрачный).

Halo > Color Цвет материала. Значение по умолчанию (0.8, 0.8, 0.8) (почти белый).

Halo > Seed Не используется.

Halo > Size Размер частиц. Значение по умолчанию 0.5.

Halo > Hardness Показатель степени при расчете градиента. Влияет на видимый размер частиц. Значение по умолчанию 50.

Halo > Add Не используется.

Halo > Rings Использовать кольца. Настраивается относительное количество и цвет.

Halo > Lines Использовать линии. Настраивается относительное количество и цвет.

Halo > Star Tips Использовать звезды. Настраивается количество концов.

Blend4Web > Special: Stars Включает режим рендеринга звездного неба, при этом меш неподвижен относительно камеры. Для лампы необходимо также выставить опцию Blend4Web > Dynamic intensity. Приложения должны установить ночное время суток, используя API.

Blend4Web > Blending Height Диапазон высот, на котором происходит затухание яркости звезд.

Blend4Web > Stars Minimum Height Минимальная высота в локальном пространстве объекта, на которой видны звезды.

Текстуры

Текстуры - подготовленные вручную или процедурно сгенерированные изображения, накладываемые на поверхность моделей с целью их детализации. Для соотнесения точек 3D поверхности и пикселей плоского изображения, как правило, используются текстурные развертки. По этой причине текстуры иногда называют картами.

Текстуры обычно помещаются в текстурные слоты *материалов*, также возможно их использование для параметризации *систем частиц* и визуализации *небесного свода*.

12.1 Типы текстур

Опция выбора типа текстуры Туре расположена во вкладке **Textures**. Дважды поддерживаются текстуры следующих типов:

1. Image or Movie, изображение или фильм

- *диффузная (diffuse map)*
- *карта бликов (specular map)*, может также содержаться в альфа-канале диффузной текстуры
- *карта нормалей (normal map)*
- *карта высот (height map)*, может содержаться только в альфа-канале карты нормалей, используется для *реализации рельефной поверхности (parallax mapping)*
- *карта смешивания (stencil map)*
- *видеотекстура*

2. Environment Map, карта окружения

- *карта зеркального отражения (mirror map)*
- *текстура неба (skydome)*
- используется для реализации одной из методик *освещения от окружающей среды*

3. None, пустая

- применена на кубе в стартовой сцене Blender'a. Также используется для *рендеринга сцены в текстуру*, а также для *рендеринга canvas-текстуры*.

4. Blend, градиент

- используется в *системах частиц*

5. Voronoi, процедурная текстура с разбиением Вороного

- используется для *рендеринга воды* с целью настройки каустики

12.2 Общие настройки

Размер Размер растром для текстур-изображений (длина и ширина изображения в пикселях) должен быть числом 2^N , т.е. 4, 8, 16, 32, 64, 128, 256, 512, 1024, 2048, 4096 пикселов. Использование текстур других размеров (т.н. NPOT) поддерживается, но не рекомендуется. Для корректной работы компрессии текстур размер должен составлять не менее 4 пикселов. Как правило, используются изображения квадратной формы (например, 512 x 512 px), однако могут использоваться и прямоугольные (например, 4 x 128 px). Использование изображений размером более 2048 пикселов не рекомендуется.

Image Mapping > Extension Режим интерпретации текстурных координат (в WebGL - Wrap Mode). Доступен для текстур типа Image or Movie. В случае значения Repeat движок устанавливает для текстуры режим REPEAT. При этом целочисленная часть текстурных координат игнорируется, используется только дробная часть. Во всех остальных случаях (например, Extend) движок устанавливает CLAMP_TO_EDGE. При этом происходит ограничение текстурных координат отрезком [0, 1]. Значение по умолчанию Repeat.

Mapping > Coordinates Тип текстурных координат. Поддерживаются UV (использовать развертку), Normal (использовать направление на камеру, только для диффузных текстур, применяется для создания материалов в стиле material capture, matcap) и Generated (исходные недеформированные координаты объекта). Значение по умолчанию Generated.

Mapping > Offset Не поддерживается.

Mapping > Size Масштабирование развертки по соответствующим осям. Значения по умолчанию 1.0.

Blend4Web > Do Not Export Не экспортировать текстуру.

Blend4Web > Anisotropic Filtering Фактор анизотропной фильтрации для индивидуальной текстуры. Имеет приоритет перед аналогичной настройкой для сцены. Значение по умолчанию DEFAULT (т.е. использовать настройки сцены).

Blend4Web > Water Foam Текстура пены. Используется материалом для рендераинга воды.

Blend4Web > Disable Compression Отключить использование текстурной компрессии для данной текстуры. Применяется в случае, когда компрессия приводит к ухудшению качества изображения. В частности, рекомендуется отключать для текстур-масок, использующихся для смешивания различных характеристик материалов.

Примечание: Для карт нормалей текстурная компрессия не применяется.

12.3 Диффузная текстура (diffuse map)

Диффузная текстура применяется для указания распределения цвета рассеянного света (модель Ламберта).

12.3.1 Активация

Выставить опцию Diffuse > Color на панели Textures > Influence.

12.3.2 Дополнительные настройки

Influence > Diffuse > Color Степень влияния текстуры на диффузный цвет.
Значение по умолчанию 1.0.

Influence > Blend Тип взаимодействия с цветом материала (Material > Diffuse > Color), или с вертексным цветом, если включена опция Vertex Color Paint. Поддерживаются Mix (смешивается с цветом), Multiply (умножается на цвет). Значение по умолчанию Mix.

12.4 Карта бликов (specular map)

Карта бликов применяется для указания распределения цвета отраженного света (модель Фонга).

12.4.1 Активация

Выставить опцию Specular > Color на панели Textures > Influence (опция Specular > Intensity не поддерживается).

12.4.2 Дополнительные настройки

Influence > Specular > Color Степень влияния текстуры на цвет отраженного света. Значение по умолчанию 1.0.

Influence > Blend Тип взаимодействия с цветом отраженного света материала (**Material > Specular > Color**). Поддерживается только **Mix** (смешивается с цветом). Значение по умолчанию **Mix**.

Карта бликов может быть упакована в альфа-канал диффузной текстуры в целях оптимизации. В этом случае для текстуры необходимо одновременно выставить опции **Diffuse > Color** и **Specular > Color**. Цветовой диапазон ограничен оттенками серого цвета.

12.5 Карта нормалей (normal map)

Карта нормалей применяется для указания распределения нормалей (перпендикуляров) к поверхности с целью увеличения уровня детализации ее рельефа. Информация о нормалях должна храниться в текстурном пространстве координат. Карты нормалей в объектном пространстве не поддерживаются.

12.5.1 Активация

Выставить опцию **Geometry > Normal** на панели **Textures > Influence**.

12.5.2 Дополнительные настройки

Influence > Geometry > Normal Степень участия карты в расчетах нормалей. Значение по умолчанию 1.0.

12.6 Карта высот (height map). Parallax mapping

Карта высот содержит информацию о распределении относительных высот рельефа. Более высокий уровень поверхности обозначается более светлым цветом. Карта высот в сочетании с картой нормалей требуются в качестве входящих данных для реализации рельефной поверхности (parallax mapping). Карта высот должна содержаться в альфа-канале карты нормалей.

12.6.1 Активация

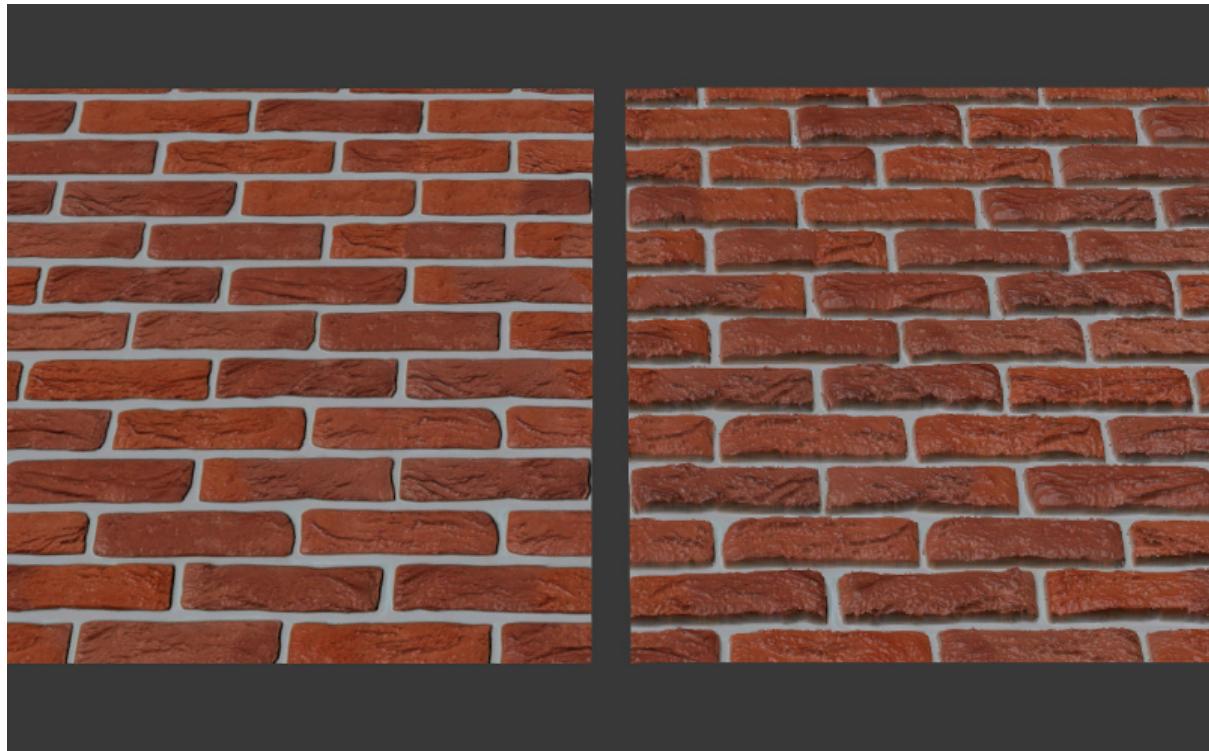
Для карты нормалей дополнительно к опции **Geometry > Normal** на панели **Textures > Influence** выставить опцию **Parallax** на панели **Textures > Blend4Web**.

12.6.2 Дополнительные настройки

Blend4Web > Parallax Scale Фактор влияния эффекта рельефной поверхности. Значение по умолчанию 0.03.

Blend4Web > Parallax Steps Количество итераций в расчетах рельефной поверхности. Большее значение приводит к лучшему качеству и к большим затратам вычислительных ресурсов. Значение по умолчанию 10.

Blend4Web > Parallax LOD distance Расстояние на котором виден эффект параллакса.



12.7 Карта смещивания (stencil map)

Специальная текстура (цветная или оттенков серого), содержащая информацию о распределении других текстур по поверхности.

12.7.1 Активация

1. В случае нодовых материалов карта смещивания должна использоваться соответствующим образом в нодовой структуре.
2. В случае обычных материалов карта смещивания должна располагаться в текстурном слоте между двумя смешиаемыми диффузными текстурами. Для текстуры смещивания необходимо одновременно выставить опции **RGB to Intensity** и **Stencil** на панели **Textures > Influence**.

12.7.2 Дополнительные настройки

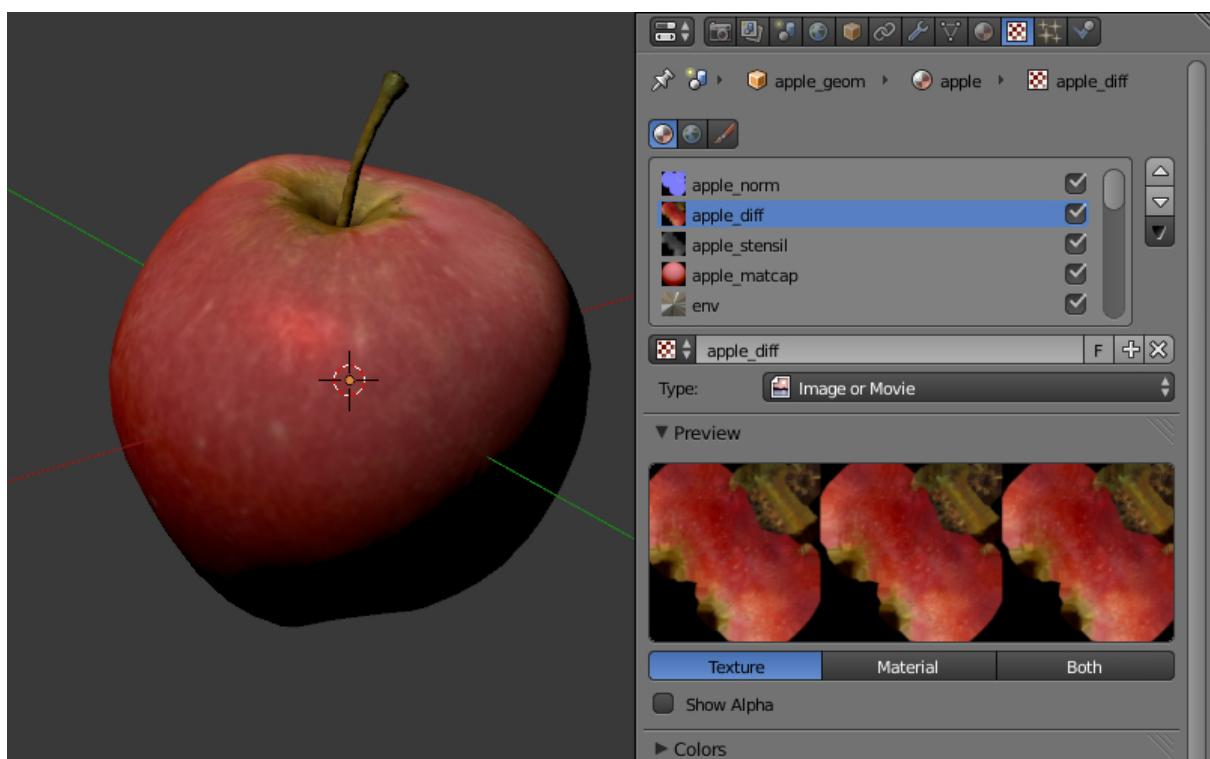
В случае обычных материалов для одной из смешируемых диффузных текстур поддерживается тип текстурных координат **Normal** (“matcap”).

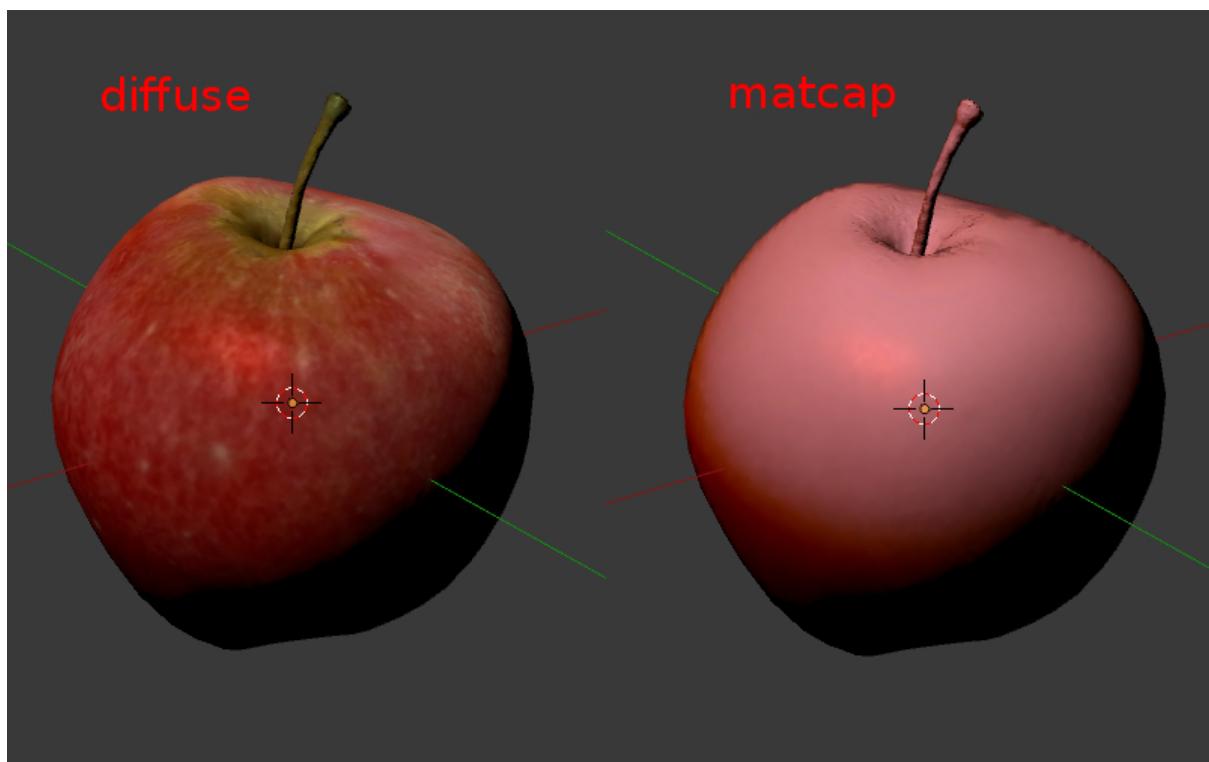
12.7.3 Ограничения

В случае обычных материалов движком интерпретируется только красный канал текстуры смешиания. Карта бликов или карта нормалей при их наличии смешианию не подвергаются. Настройка масштабирования **Mapping > Size** извлекается из первой текстуры и применяется ко всем остальным текстурам.

12.7.4 Пример

Материал яблока имеет текстуры: карту нормалей, диффузную текстуру с картой бликов в альфа-канале, карту смешиивания, диффузную карту “matcap”, карту зеркального отражения.





12.8 Видео-текстуры

В качестве текстуры может применяться видеозапись. Для этого должен быть выбран тип текстуры **Image or Movie**. Также, если была включена опция **Scene > Blend4Web > Use NLA**, то воспроизведение видеотекстур будет происходить и в рамках NLA-сценария.

Примечание: Видеотекстура поддерживает только воспроизведение видеоряда. Звуковое сопровождение следует накладывать, используя объект типа **SPEAKER**.

12.8.1 Поддерживаемые форматы (контейнеры):

- webm, кодек VP8 (Chrome, Firefox)
- m4v, кодек H.264 (Chrome, Safari, IE)
- ogv, кодек Theora (Chrome, Firefox)

В качестве базового формата рекомендуется использовать *WebM*, который является открытым стандартом, поддерживается многими браузерами, обеспечивает хорошее качество изображения.

Примечание: Файлы в форматах mp4 и ogg имеют различное расширение для звуковой и видео-информации: расширения *.mp4* и *.ogg* используются для звуков, *.m4v* и *.ogv* - для видео.

Конвертация ресурсов в различные форматы описывается в *соответствующем разделе*.

12.8.2 Настройка текстуры

При использовании видео-текстуры можно настроить следующие опции на панели Texture > Image:

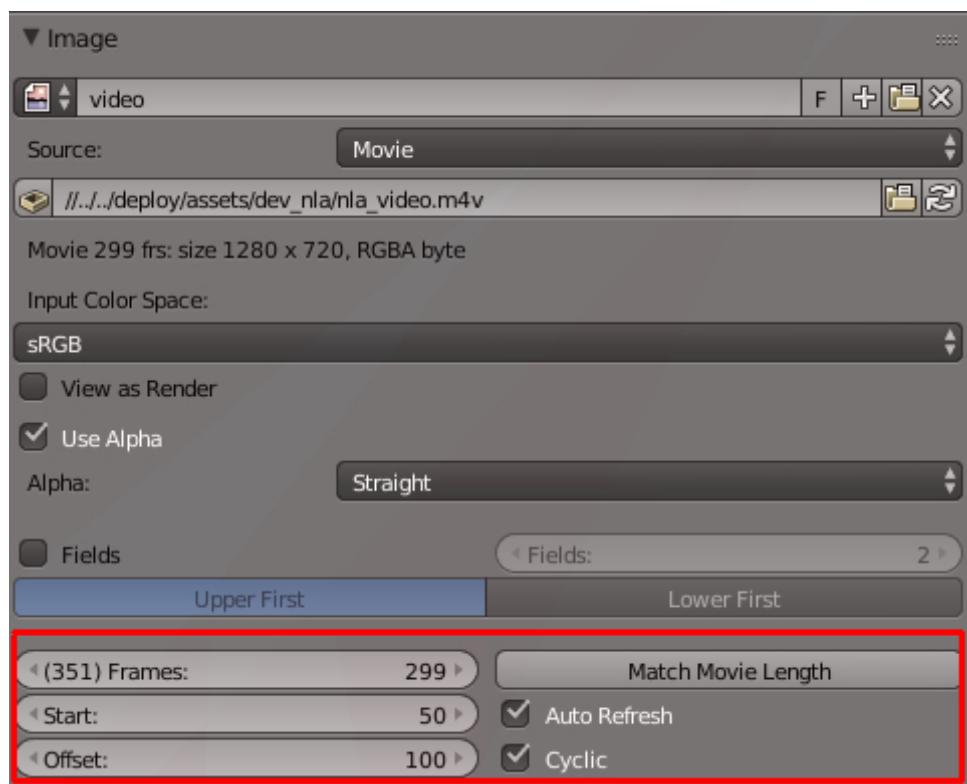
Image > Frames Длина воспроизводимого фрагмента видеозаписи в кадрах.

Image > Start Задержка воспроизведения видеозаписи в кадрах при использовании нелинейной анимации (опция Scene > Blend4Web > Use NLA).

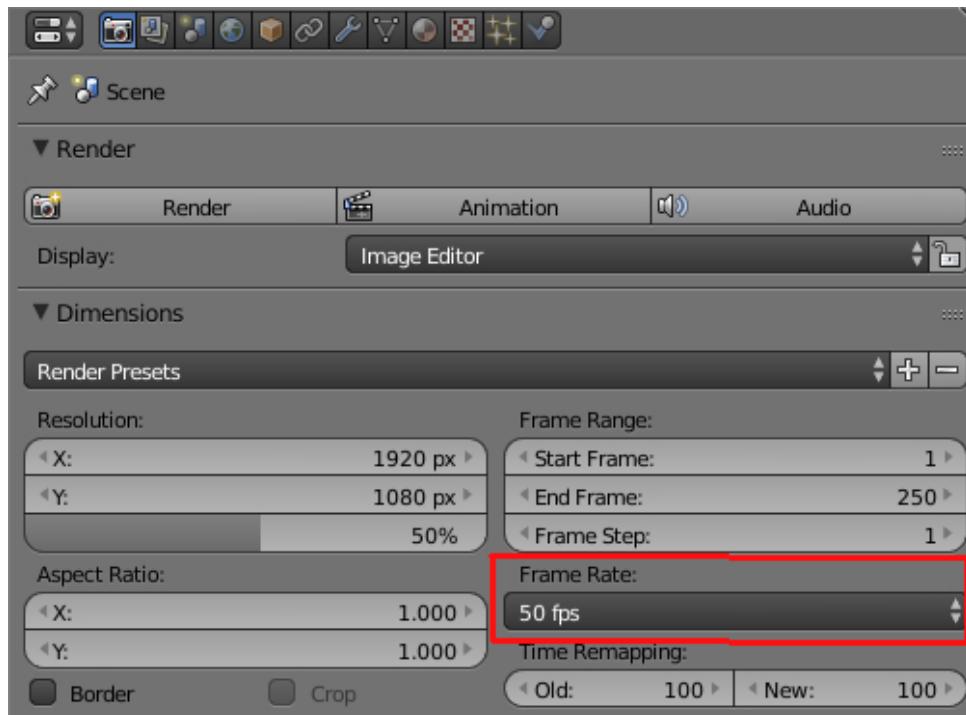
Image > Offset Номер кадра, с которого будет начинаться воспроизведение видеозаписи.

Image > Auto Refresh Воспроизводить видеозапись сразу после загрузки. Не актуальна при использовании нелинейной анимации: воспроизведение видео определяется остальными параметрами.

Image > Cyclic Начинать воспроизводить видеозапись заново каждый раз после ее завершения.



Существует возможность ускоренного воспроизведения видеозаписей. Для этого необходимо задать значение свойства **Scene > Dimensions > Frame rate** отличным от значения **fps** (кадров в секунду) для видеозаписи. При этом скорость воспроизведения видеозаписи расчитывается как отношение **Frame rate** сцены к **fps** видеозаписи.



12.8.3 Особенности работы на мобильных устройствах

Присутствуют следующие особенности реализации на мобильных устройствах:

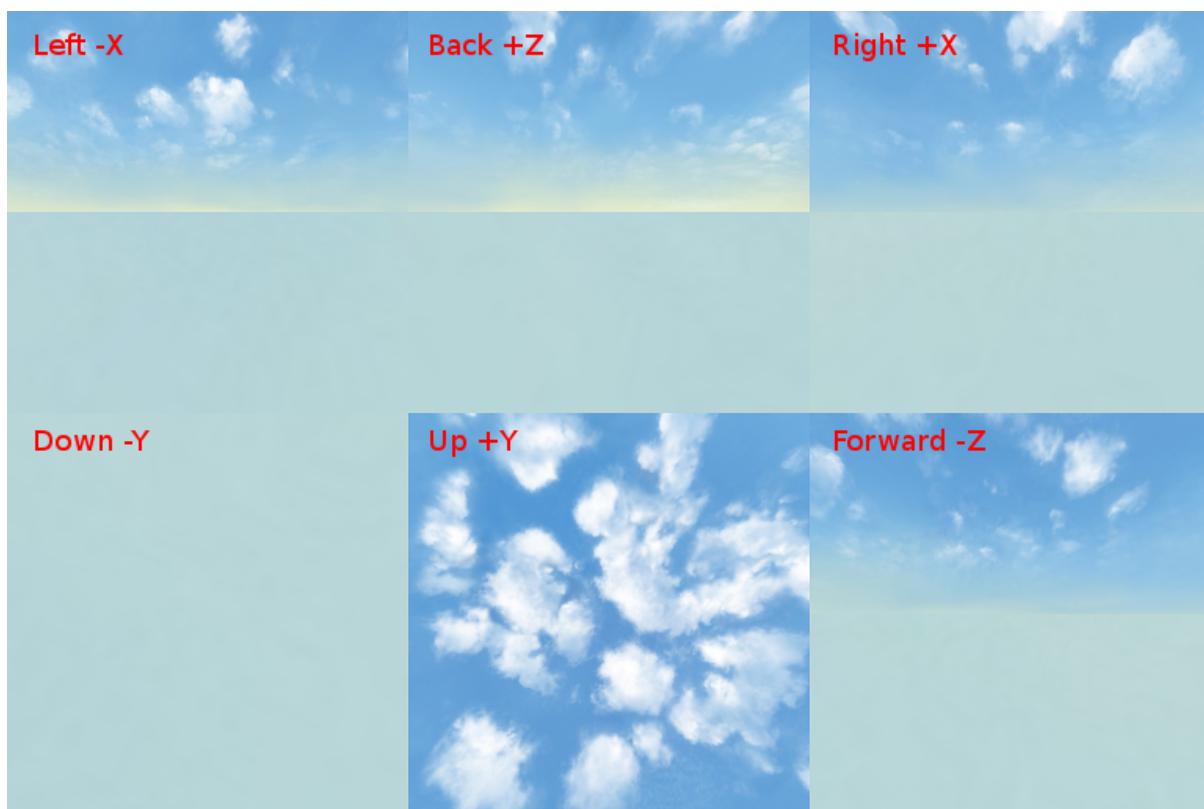
- на iPhone используется специально разработанный видеоформат **.seq**, потому что данное устройство открывает все стандартные видеозаписи через свой видеопроигрыватель. Видео надо сконвертировать заранее, используя [наш конвертер](#).
- на некоторых мобильных устройствах имеется поддержка воспроизведения только одного видеофайла.
- не гарантируется стабильная работа при выставлении опции **Offset** в ненулевое значение.
- не на всех устройствах поддерживается ускореное воспроизведение видео.
- на iPad и iPhone отсутствует возможность управления громкостью звука видеофайла, поэтому следует убрать аудио дорожку из видео перед добавлением файла в Blender.

12.9 Карта окружения (environment map)

Применяется в качестве *карты зеркального отражения* (*mirror map*), в качестве статической *текстуры неба* (*skydome*), а также для реализации одной из методик *освещения от окружающей среды*.

В движке представлена кубической текстурой. Раstry для карт окружения должны содержать 6 спроецированных изображений окружающей среды, упакованных в 2 ряда по 3 (формат, используемый в Blender'e). Размер растрор для каждого из изображений должен подчиняться правилу 2^N (512, 1024 и т.п.).

Во избежание проявления швов рекомендуется использовать формат без потери качества (PNG).



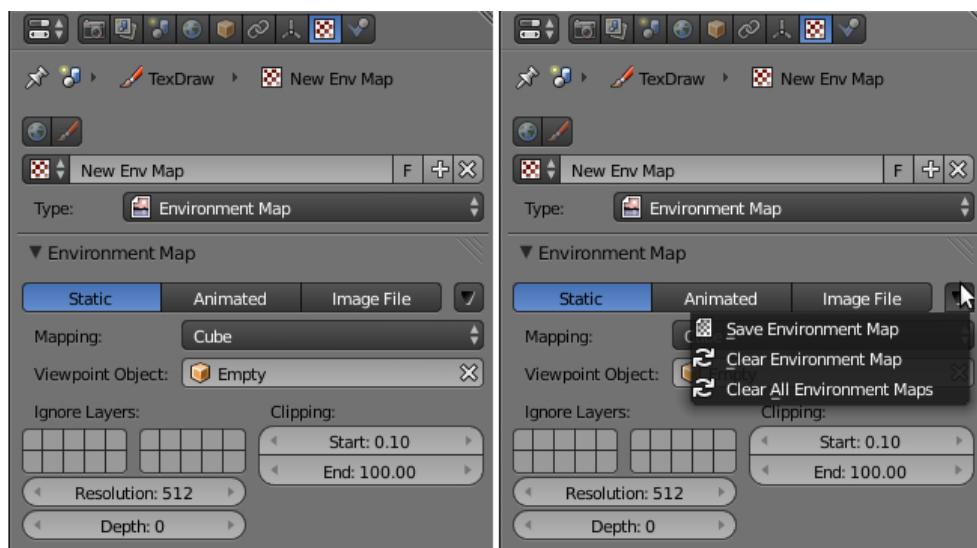
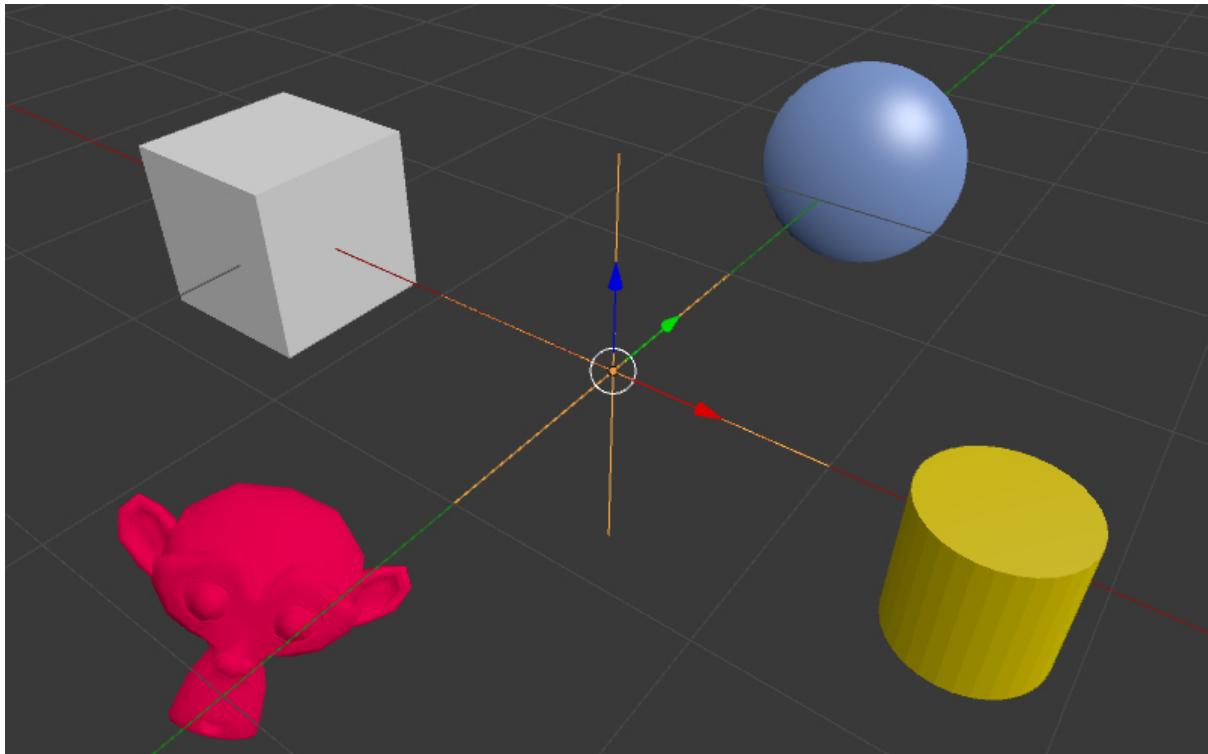
12.9.1 Создание карты окружения

Blender позволяет запекать сцену в карту окружения. Для этого:

1. Создать сцену для запекания.
2. Добавить пустой объект в предполагаемом центре обзора (Add > Empty).
3. Перейти во вкладку **World**, затем перейти во вкладку **Textures**, создать новую текстуру, выбрать тип **Environment Map**.
4. На панели **Environment Map** выбрать источник **Static**, выбрать созданный пустой объект в поле **Viewport Object**, установить разрешение 2^N (512, 1024 и т.п.).

5. Выполнить рендеринг сцены F12 (требуется наличие камеры).

6. Сохранить карту окружения в файл.



12.10 Карта зеркального отражения (mirror map)

Применяется для визуализации отражающей способности поверхности. Представляет собой [карту окружения](#).

12.10.1 Активация

Выбрать тип текстуры (Type) Environment Map. Выставить опцию Shading > Mirror на панели Textures > Influence.

12.10.2 Дополнительные настройки

Influence > Shading > Mirror Степень влияния карты зеркального отражения.
Значение по умолчанию 1.0.

См.также:

Статическое отражение.

12.11 Текстура неба (skydome)

Применяется для визуализации бесконечно удаленного окружения (например, небесного свода). Представляет собой *карту окружения*.

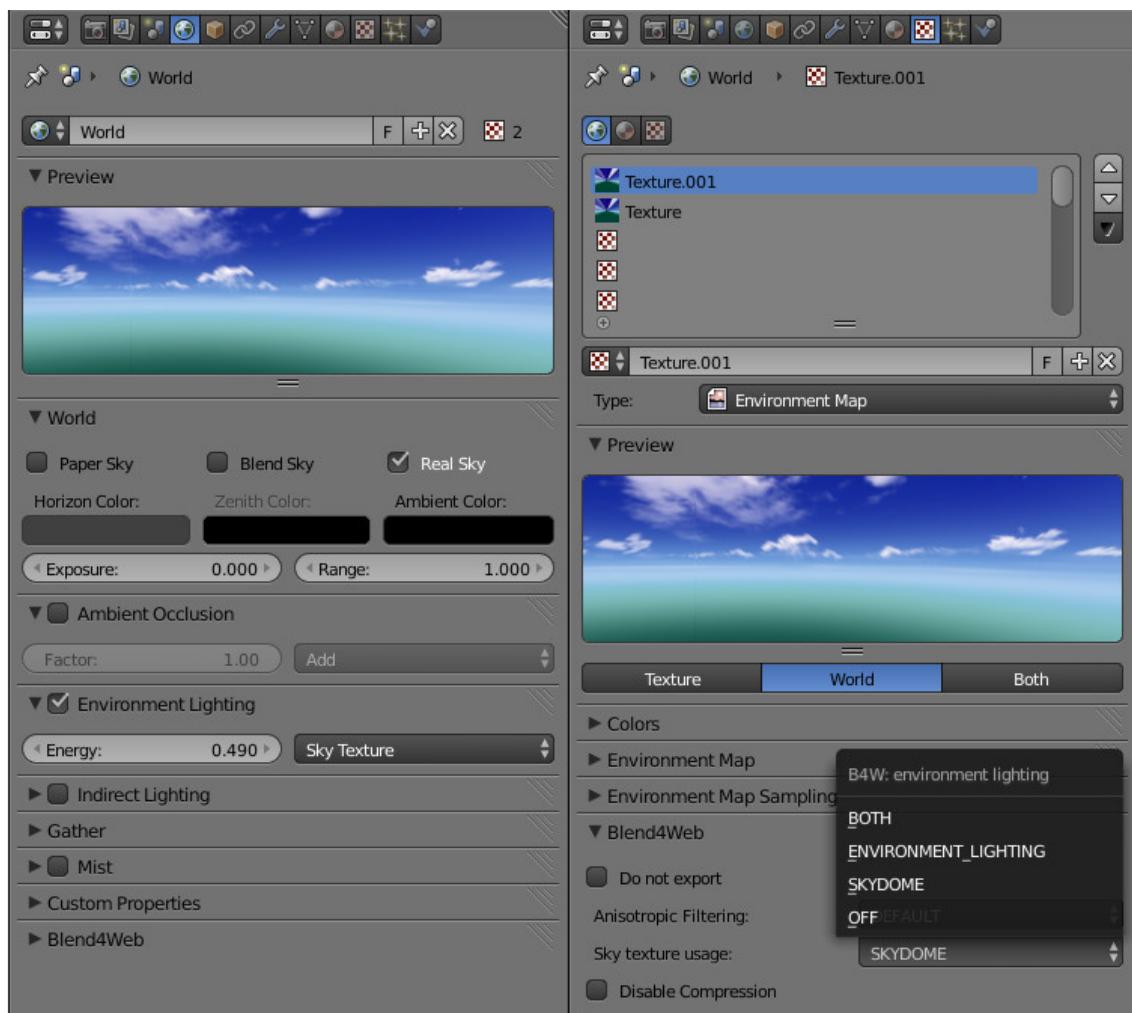
Также может применяться для реализации одной из методик *освещения от окружающей среды*.

12.11.1 Активация

Создать текстуру мира (world texture) с типом “Environment Map”. Выбрать опцию Blend4Web > Sky Texture Usage > SKYDOME, предварительно выставив опцию Sky Settings > Render Sky на вкладке World.

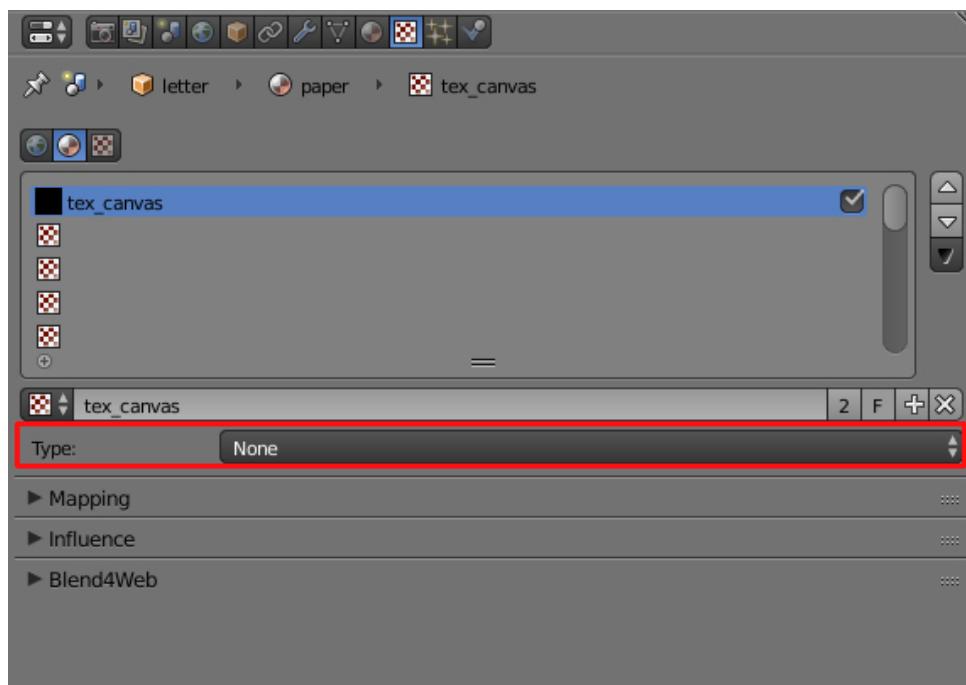
Примечание: Для имитации *освещения от окружающей среды* можно воспользоваться опцией Blend4Web > Sky Texture Usage > ENVIRONMENT_LIGHTING. При этом необходимо также выбрать соответствующую опцию в настройках мира: Environment Lighting > Sky Texture.

Для создания обоих эффектов от одной текстуры необходимо выставить опцию Blend4Web > Sky Texture Usage > BOTH.



12.12 Особые типы текстур

Для использования особых типов текстур необходимо выставить тип текстуры **None** на панели **Textures**.



В меню **Textures > Blend4Web** имеется возможность задать основные свойства для данного типа текстуры:

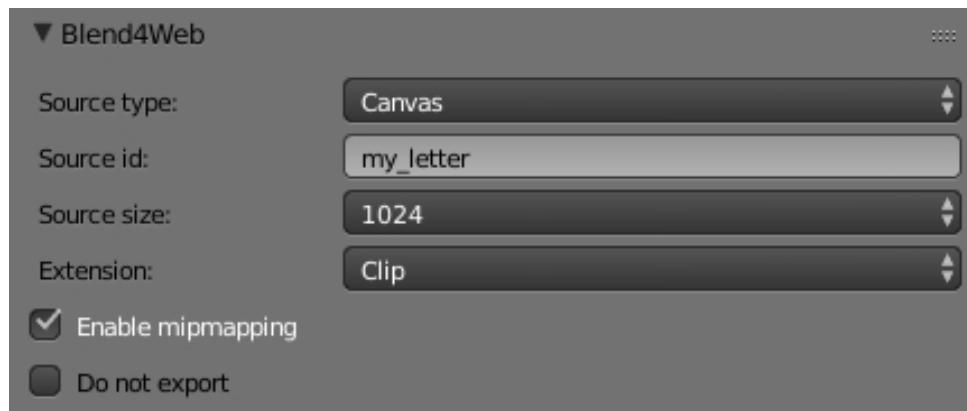
Blend4Web > Source Type Выбор особого типа текстуры: **Scene** - для рендеринга 3D сцены в текстуру, **Canvas** - использование HTML-элемента `<canvas>` и **None** - отсутствие текстуры.

Blend4Web > Source ID Имя сцены, которая будет отображена в текстуре, в случае 3D сцены либо идентификатор HTML-элемента `<canvas>` в случае Canvas-текстуры.

Blend4Web > Source Size Разрешение текстуры.

Blend4Web > Extension *Режим интерпретации текстурных координат.* Значение по умолчанию **Repeat**.

Blend4Web > Enable Mipmapping Включение МIP-текстурирования для Canvas-текстуры. По умолчанию включено.



12.12.1 3D сцена

Изображение 3D сцены может быть использовано в качестве текстуры на объекте другой (“главной”) сцены.

1. Создать дополнительную сцену-источник, переименовать для удобства, создать **World**, добавить нужные объекты, настроить вид из камеры.
2. В главной сцене для текстуры целевого объекта выставить тип **None**, в поле **Blend4Web > Source Type** указать тип **Scene**. В поле **Blend4Web > Source ID** указать название сцены-источника. В поле **Blend4Web > Source Size** указать размер текстуры в пикселях.



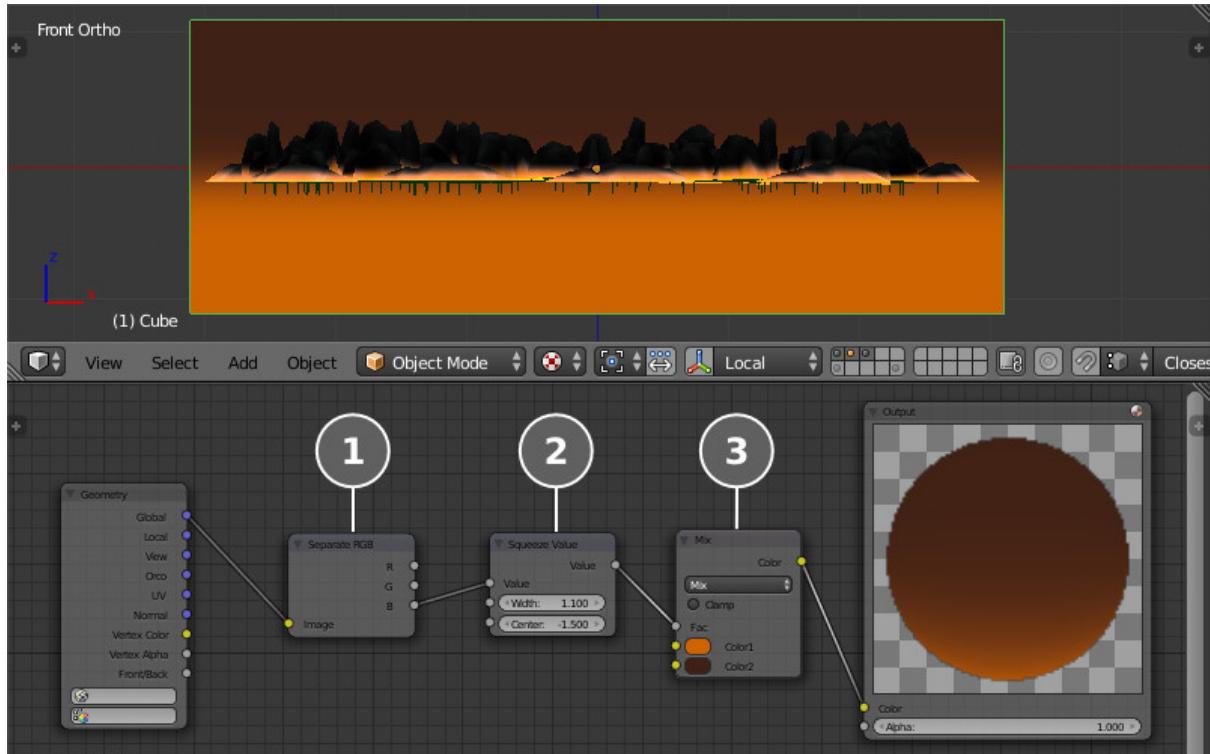
12.12.2 Canvas

В качестве текстуры используется HTML-элемент <canvas>. Его редактирование осуществляется при помощи [API](#).

В главной сцене для текстуры целевого объекта выставить тип **None**, в поле **Blend4Web > Source Type** указать тип **Canvas**. В поле **Blend4Web > Source ID** указать идентификатор объекта типа **Canvas**. В поле **Blend4Web > Source Size** указать размер текстуры в пикселях.

Нодовые материалы

Шейдерные ноды (Shader Nodes) существенно расширяют возможности стандартных материалов Blender, позволяя представить освещение как серию базовых преобразований.



13.1 Стандартные ноды

Полностью поддерживаются все возможности Blender, за исключением следующих случаев:

- **Geometry** - не поддерживаются выходы **Local**, **Orco** и **Vertex Alpha**.
- **Material, Extended Material** - допускается не больше одной ноды на материал, не поддерживаются входы **Refl**, **Ambient**, **SpecTra** и выход **AO**.

- RGB Curves - не поддерживается.
- Vector Curves - не поддерживается.

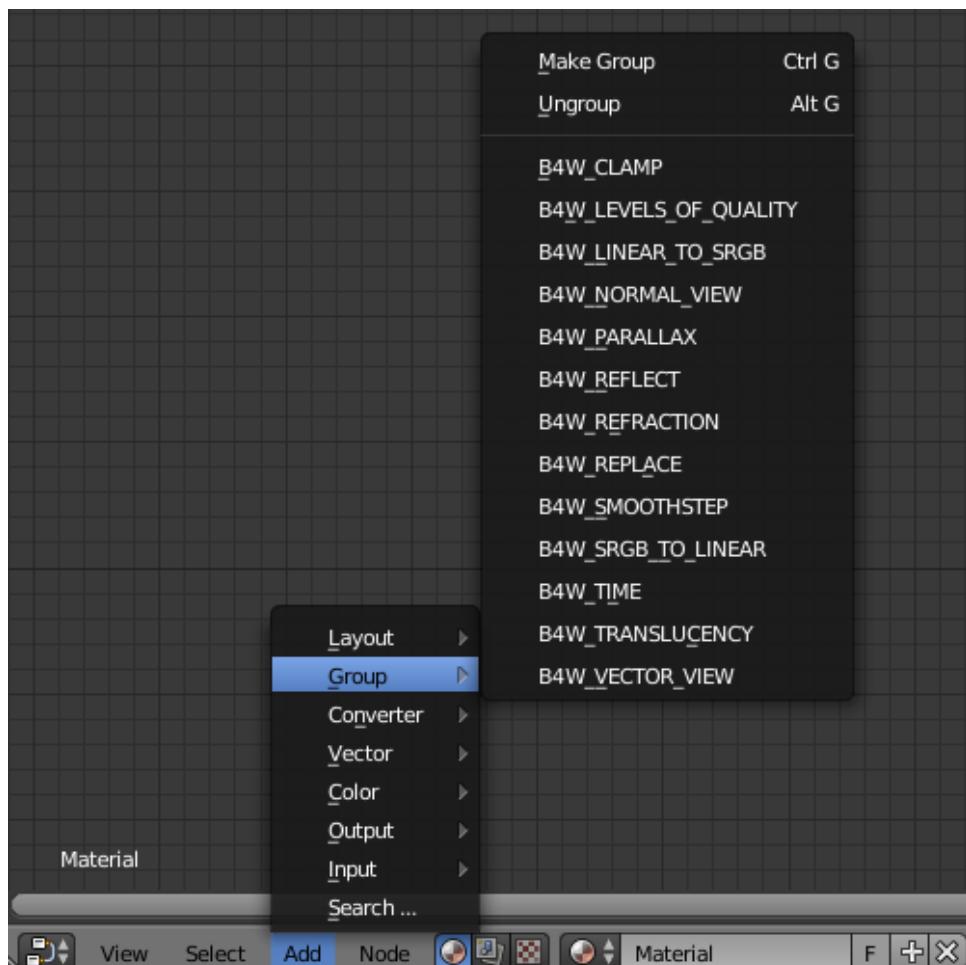
Кроме того, в контексте рендеринга в реальном времени, следует учитывать низкую производительность некоторых нод. Не рекомендуется к использованию:

- Hue/Saturation
- MixRGB типы Burn, Dodge, Value, Saturation, Hue, Color

Не рекомендуется создавать сложные материалы, особенно использующие большое количество нод Geometry и Texture.

13.2 Дополнительные ноды

Дополнительные ноды расширяют функционал стандартных с учётом специфики работы движка. Ноды оформляются в виде нодовых групп (*Node groups* или *Node tree*) со специально выбранным именем и форматом входов. Для удобства, все дополнительные ноды добавляются в blend-файл при его открытии.

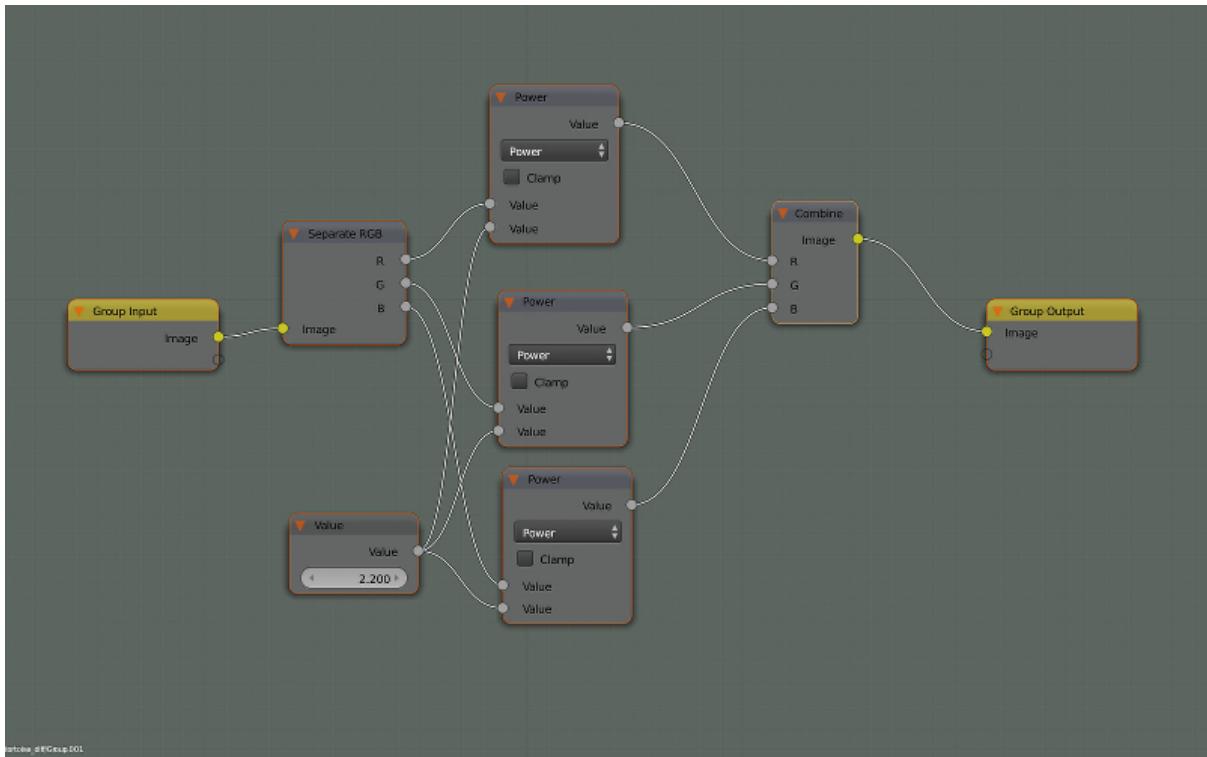


См. также:

[Анимация ноды Value](#)

13.2.1 B4W_LINEAR_TO_SRGB и B4W_SRGB_TO_LINEAR

Преобразование цвета из линейного цветового пространства в пространство sRGB и наоборот.

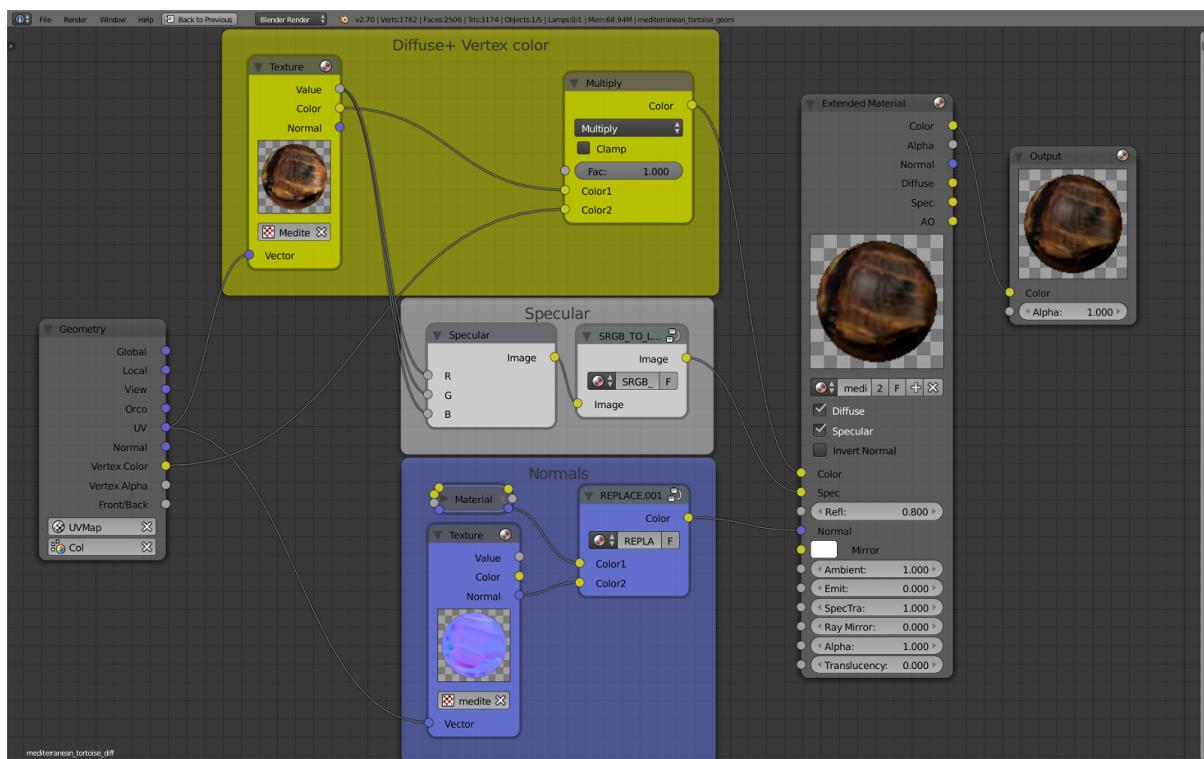


См. также:

Коррекция в нодовых материалах

13.2.2 B4W_REPLACE

Осуществляет замену входов в зависимости от того, в какой среде (viewport Blender'a или движок) в данный момент работает текущая сцена. При работе в Blender вход Color1 подключается к выходу Color, вход Color2 игнорируется. При работе в движке входы меняются местами (Color1 игнорируется, Color2 подключается к выходу). Нода предназначена для отображения во viewport'е одной конструкции нодов, а в движке - другой.



Используется, как правило, для подключения карт нормалей. Нодовые материалы Blender'a не поддерживают тангенциальное пространство координат, в связи с чем единственный способ корректного отображения карт нормалей во viewport'e - их подключение внутри нод Material.

13.2.3 B4W_CLAMP

Осуществить операцию ограничения над входом. В результате, все элементы вектора на выходе получают значения от 0 до 1 включительно.

13.2.4 B4W_TIME

Осуществляет отсчет времени с момента старта движка в секундах. Может использоваться для анимации любых параметров в нодовых материалах.

13.2.5 B4W_VECTOR_VIEW

Осуществить преобразование вектора в пространство камеры. Преобразование необходимо, поскольку при работе в движке большинство векторов определены в мировой системе координат (например нормали, направления источников освещения и

т.д.). Преобразованный таким образом вектор нормали используется только для различных эффектов и не должен подключаться к входу ноды **Material** или **Extended Material**.

13.2.6 B4W_PARALLAX

Реализует смещение текстурных координат в соответствии с картой высот.

Входные параметры

UV Исходные текстурные координаты.

Height Map RGBA текстура с картой высот в альфа канале.

Scale Коэффициент смещения текстурных координат.

Steps Количество шагов при генерации смещенных текстурных координат. Чем больше данное значение, тем выше качество получаемой текстуры.

Lod Distance Максимальное расстояние от камеры, на котором виден эффект.

Выходные параметры

UV Измененные текстурные координаты, которые используются как вход для текстурных нод.

13.2.7 B4W_TRANSLUCENCY

Реализует эффект полупрозрачности (только по отношению к источникам света) для тонких объектов, таких как ткань, листва, бумага и др. Эффект состоит из двух частей: засвечивание обратной по отношению к источнику стороны объекта и появление светового пятна непосредственно в том месте, где должен был находиться источник.

Входные параметры

Color Одноканальная текстура, определяющая неоднородность материала, белый - максимальный эффект просвечивания, черный - его отсутствие. По умолчанию используется белый.

Backside Factor Коэффициент коррекции цвета материала на обратной от источника света стороне. Основан на визуальном эффекте большей насыщенности цвета при просвечивании.

- *Backside Factor < 1* - коррекция в сторону освещения
- *Backside Factor = 1* - без коррекции

- *Backside Factor* > 1 - коррекция в сторону затемнения

Значение по умолчанию: 1.

Spot Hardness Коэффициент размытия светового пятна. При увеличении размеры пятна уменьшаются, края становятся более резкими. Значение по умолчанию: 1000.

Spot Intensity Интенсивность светового пятна. При увеличении становится более ярким. Значение по умолчанию: 1.

Spot Diffuse Factor Коэффициент влияния диффузного цвета материала на цвет светового пятна.

- *Spot Diffuse Factor* = 0 - световое пятно имеет диффузный цвет
- *Spot Diffuse Factor* = 1 - световое пятно имеет белый цвет

Значение по умолчанию: 1.

Выходные параметры

Translucency Выход должен быть подключен ко входу **Translucency** ноды **Extended Material**.

13.2.8 B4W_REFRACTION

Реализует эффект преломления.

Входные параметры

Normal Карта нормалей, по которой происходит возмущение (сдвиг).

Refraction bump Коэффициент возмущения (сдвига) текстуры сцены позади объекта.

Значение по умолчанию: 0.001.

Выходные параметры

Color Текстура сцены позади объекта с внесённым возмущением.

Примечание: Для отображения эффекта необходимо включить опцию **Render refraction** на панели **Scene > Blend4Web**. Объект должен быть с типом прозрачности **Alpha Blend**.

См.также:

Прозрачность

13.2.9 B4W_LEVELS_OF_QUALITY

Устанавливает значение выходного цвета в зависимости от установленного профиля качества изображения.

См.также:

Профили качества изображения

Входные параметры

HIGH Значение, которое будет подано в выходной параметр **Color** при высоком и максимальном качестве изображения.

LOW Значение, которое будет подано в выходной параметр **Color** при низком качестве изображения.

Выходные параметры

Color Значение выходного цвета.

Примечание: Для отображения во виджете Blender'a используется входной параметр **HIGH**.

13.2.10 B4W_SMOOTHSTEP

Осуществить мягкую интерполяцию двух значений, исходя из первого значения.

Входные параметры

Value Значение, на основе которого осуществляется интерполяция.

Edge0 Первое значение для интерполяции.

Edge1 Второе значение для интерполяции.

Выходные параметры

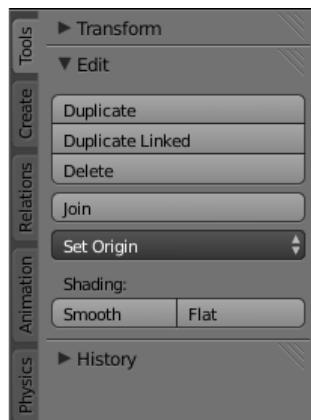
Value Сглаженное значение.

Примечание: Для корректной интерполяции входное значение "Value" должно лежать в диапазоне между "Edge0" и "Edge1".

Освещение, тени и фон

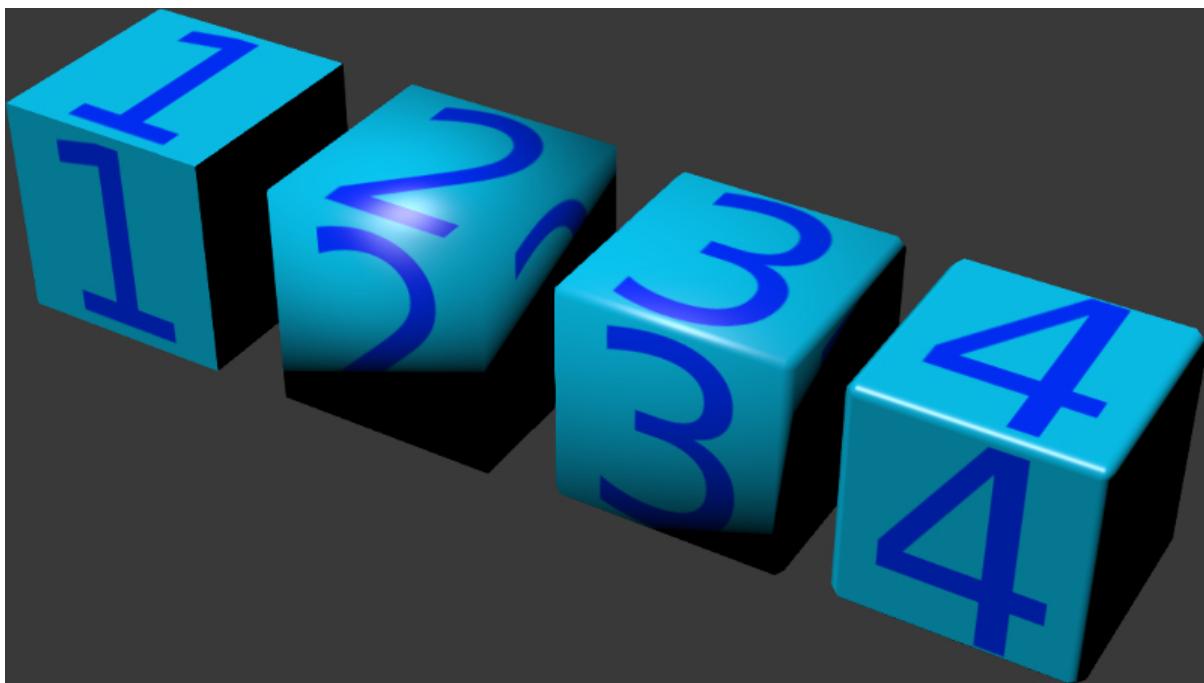
14.1 Типы затенения

Результат расчета освещения (затенения) зависит от направления векторов нормалей. Поддерживаются стандартные типы затенения, использующиеся в Blender, Shading: Flat (используются нормали граней) и Shading: Smooth (используются интерполированные нормали вершин), а также их комбинации.



Если желаемый результат недостижим стандартными средствами, можно воспользоваться [редактором нормалей](#).

Результат применения различных типов затенения и использования редактора нормалей:



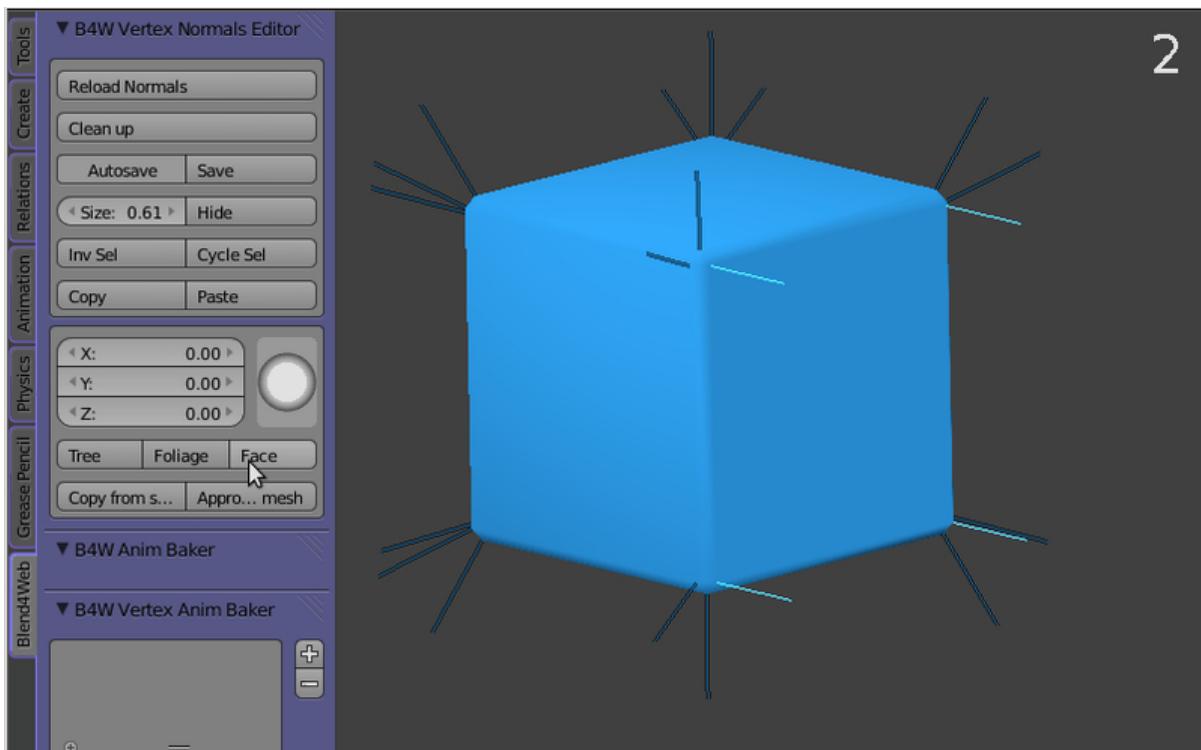
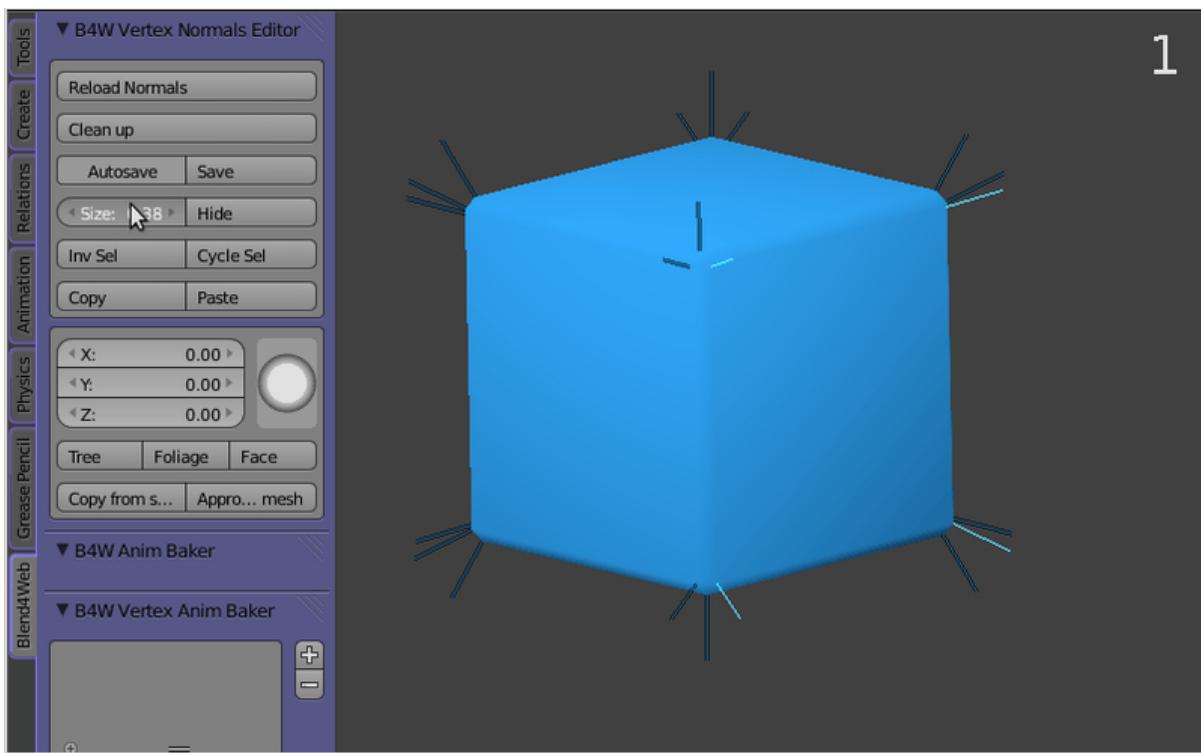
1. Flat Shading
2. Smooth Shading
3. Smooth Shading + фаска
4. Smooth Shading + фаска + редактирование нормалей

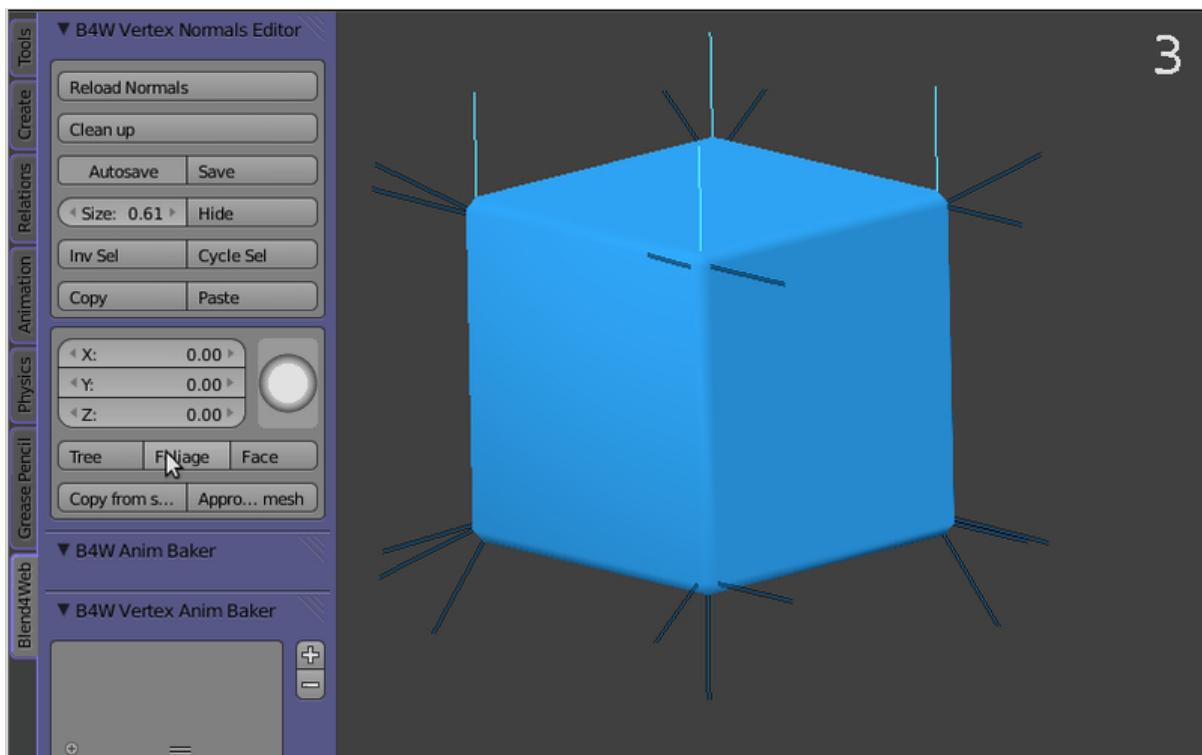
14.2 Редактор нормалей

С помощью редактора нормалей можно достичь результат, схожий в результате применения *карт нормалей*. При этом метод редактирования нормалей является более предпочтительным с точки зрения потребления вычислительных ресурсов и видео-памяти.

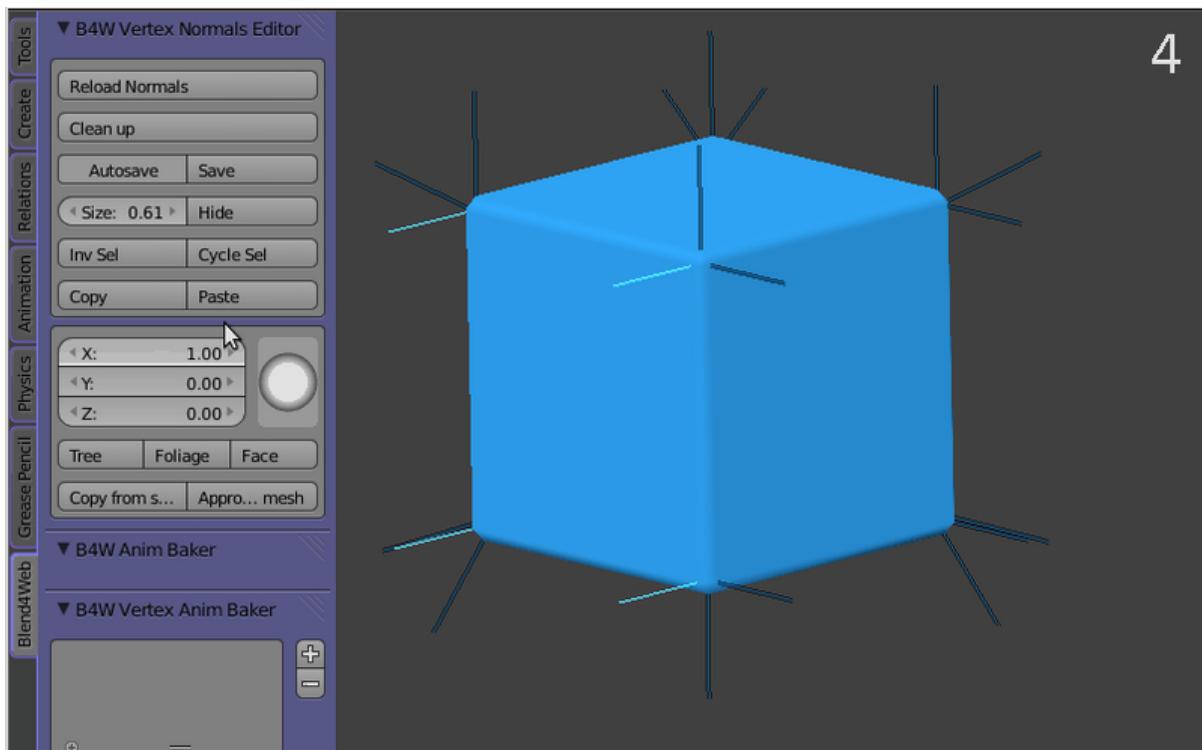
Интерфейс редактора нормалей располагается на панели инструментов Blend4Web > B4W Vertex Normals Editor. Перед началом работы редактора для объекта необходимо включить режим затенения **Shading: Smooth**.

Пример работы редактора нормалей:





3



4

1. Демонстрация бегунка Scale и кнопки Show/Hide
2. Демонстрация функции Face
3. Демонстрация функции Foliage
4. Демонстрация функции введения параметров вручную (XYZ)

Изменять нормали возможно только для выделенных вершин. Векторы выделенных вершин будут подкрашены ярко-голубым цветом, невыделенных - тусклым синим.

Чтобы их выделить, следует перейти в режим редактирования, выделить нужные вершины и вернуться в объектный режим.

После того, как вектор отредактирован, его направление необходимо сохранить, т.к. при переходе в режим редактирования и обратно нормали возвращаются в стандартное положение. Вектор можно задать несколькими способами: вручную, копированием с существующего объекта с отредактированными нормалами и используя несколько имеющихся опций.

Примечание: Если вы отредактировали группу нормалей и сохранили их, после выделения следующей группы необходимо нажать **Reload Normals**, чтобы уже отредактированные нормали вернулись в нужное положение, а потом уже переходить к следующей группе. В противном случае при повторном нажатии кнопки **Save** вы потеряете результаты предыдущего редактирования.

Reload normals Восстанавливает нормали в последнее сохраненное инструментом положение.

Clean Up Отменяет все произведенные инструментом действия, “сбрасывая” нормали в первоначальное состояние.

Save и **Autosave** Сохранение результатов редактирования нормалей. При активации **Autosave** результат каждого изменения сохраняется автоматически.

Size Длина отрезка, визуализирующего направление нормали.

Show/Hide Показать/скрыть нормали.

Inv Sel Invert selection. Инвертированное выделение. Аналог **CTRL+I** в режиме редактирования.

Cycle Sel Cycle selection. Циклическое выделение. Позволяет переходить к следующей по номеру вершине. Кнопка может быть полезной при повершинном редактировании.

Copy и **Paste** Скопировать направление вектора и назначить его другой вершине. Работает только с одной вершиной за операцию.

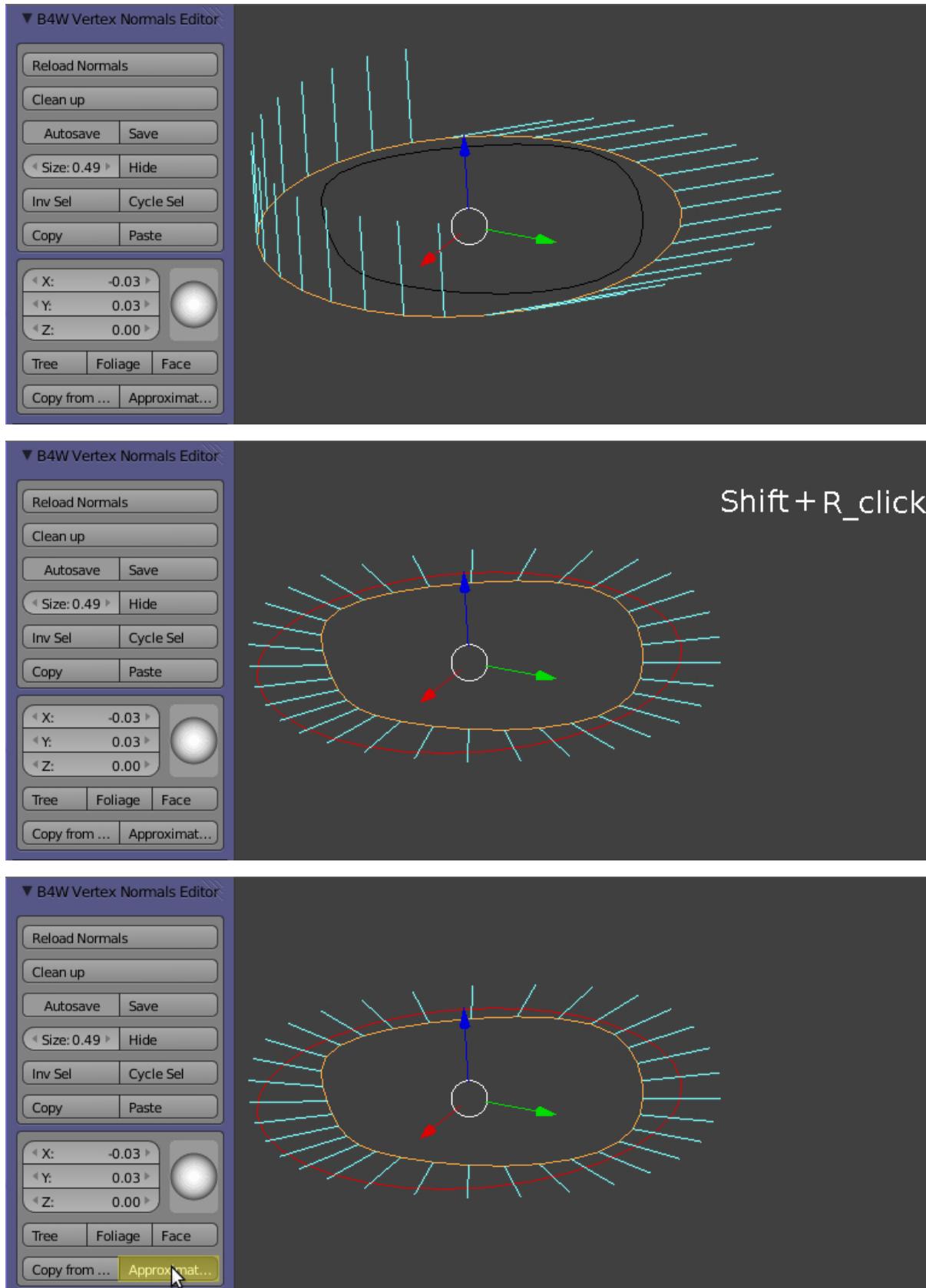
Шкала XYZ и сфера направления Инструменты для изменения направления векторов вручную.

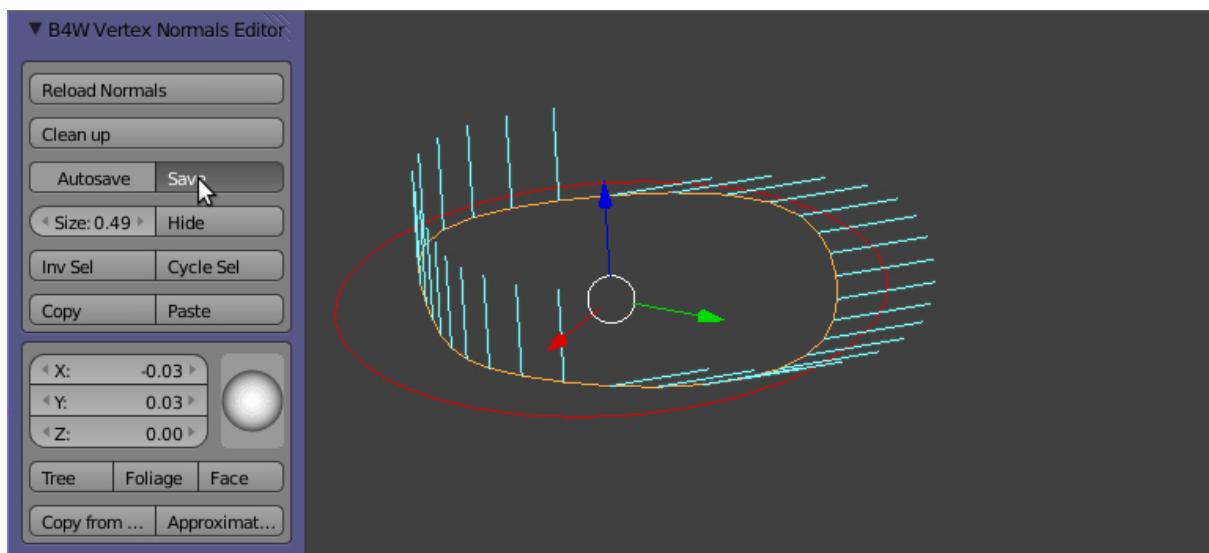
Tree, Foliage и **Face** Кнопка **Tree** задает всем выделенным нормалям направление от 3D-курсора, **Foliage** направляет нормали строго вверх (чаще всего используется для настройки частиц *травяного покрова* и *листвы деревьев*), **Face** присваивает векторам направление нормали выделенного полигона.

Copy from source mesh и **Approximate from source mesh** Кнопки копируют вектора нормалей с одного объекта на другой, при этом следует последовательно выделить исходный и целевой объекты. Разница между ними заключается в следующем. Первый способ требует точного соответствия этих объектов и копирует направление нормалей идеально точно. Вершины, не совпадающие с оригиналом, не затрагиваются. Второй способ работает с произвольной

геометрией и приводит к приближенному результату. На практике второй способ часто бывает удобнее первого и работает с приемлемой точностью.

Пример использования функции Approximate from source mesh:





В общем случае алгоритм редактирования нормалей следующий:

1. Переход в режим редактирования
2. Выделение нужных вершин
3. Переход в объектный режим
4. *Reload Normals*
5. Редактирование выделенных нормалей
6. *Save* (если *Autosave* не активна)

Необходимо учитывать, что привязка отредактированных нормалей к вершинам меша осуществляется по их номерам, которые могут измениться при редактировании геометрии меша. Это может привести к необходимости редактирования нормалей заново. Кроме того, результаты редактирования нормалей перекрываются некоторыми модификаторами, т.к. они меняют число или порядок вершин меша, поэтому желательно редактировать нормали после применения модификаторов. Модификатор **Armature** не влияет на работу инструмента.

Примечание: Если необходимо изменить меш с отредактированными нормалами, и при этом сохранить результаты редактирования, можно поступить следующим образом. Временно скопируйте этот объект, отредактируйте меш оригинала,бросьте на нем нормали кнопкой **Clean Up** и затем с копии объекта перенесите направления нормалей с помощью кнопок **Copy from source mesh** или **Approximate from source mesh**. Сохраните результаты копирования или аппроксимации.

14.3 Освещение от источников света

На сцене может быть несколько (но не менее одного) источников света разного типа.

14.3.1 Типы источников света

Поддерживаются источники света следующих типов:

Point Точечный. Свет распространяется из одной точки равномерно во все стороны, с постепенным затуханием.

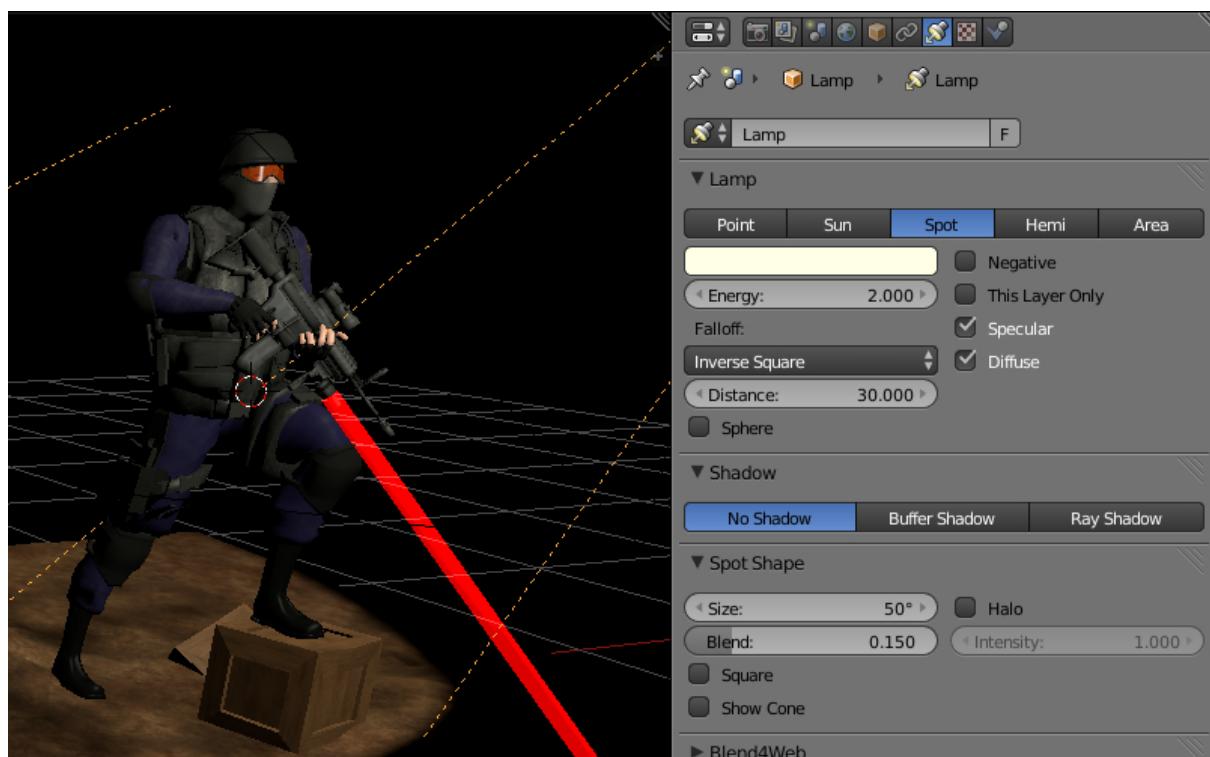
Sun “Солнце”. Свет распространяется из бесконечной плоскости прямолинейно в одном направлении, без затухания.

Spot Прожектор. Свет распространяется из одной точки, с ограничением угла распространения, с постепенным затуханием.

Hemi Полусфера. Свет распространяется из бесконечной полусферы, без затухания.

14.3.2 Настройка источников света

Производится во вкладке **Object Data** при выборе объекта-лампы.



Color Цветовая характеристика света. Значение по умолчанию (1.0, 1.0, 1.0) (белый).

Energy Интенсивность излучения. Значение по умолчанию 1.0.

Falloff Тип затухания. Значение экспортируется, но в движке всегда используется **Inverse Square** (обратный квадратичный). Применяется для источников света типа **Point** и **Spot**. Значение по умолчанию **Inverse Square**.

Distance Параметр затухания. Применяется для источников света типа **Point** и **Spot**. Значение по умолчанию 25.0.

Specular Создание отблеска на объектах. По умолчанию включено.

Diffuse Применение рассеянного освещения к объектам. По умолчанию включено.

Spot Shape > Size Угол конуса в градусах. Применяется для источников света типа **Spot**. Значение по умолчанию 45°.

Spot Shape > Blend Параметр смягчения края светового пятна. Применяется для источников света типа **Spot**. Значение по умолчанию 0.15.

Blend4Web > Do not export Не экспортовать. По умолчанию отключено.

Blend4Web > Generate shadows Источник света используется для расчета падающих теней. Применяется в случае наличия нескольких источников света. По умолчанию отключено.

Blend4Web > Dynamic intesity Источник света используется для расчета изменения времени суток. Применяется для источников света типа “Солнце”. По умолчанию отключено.

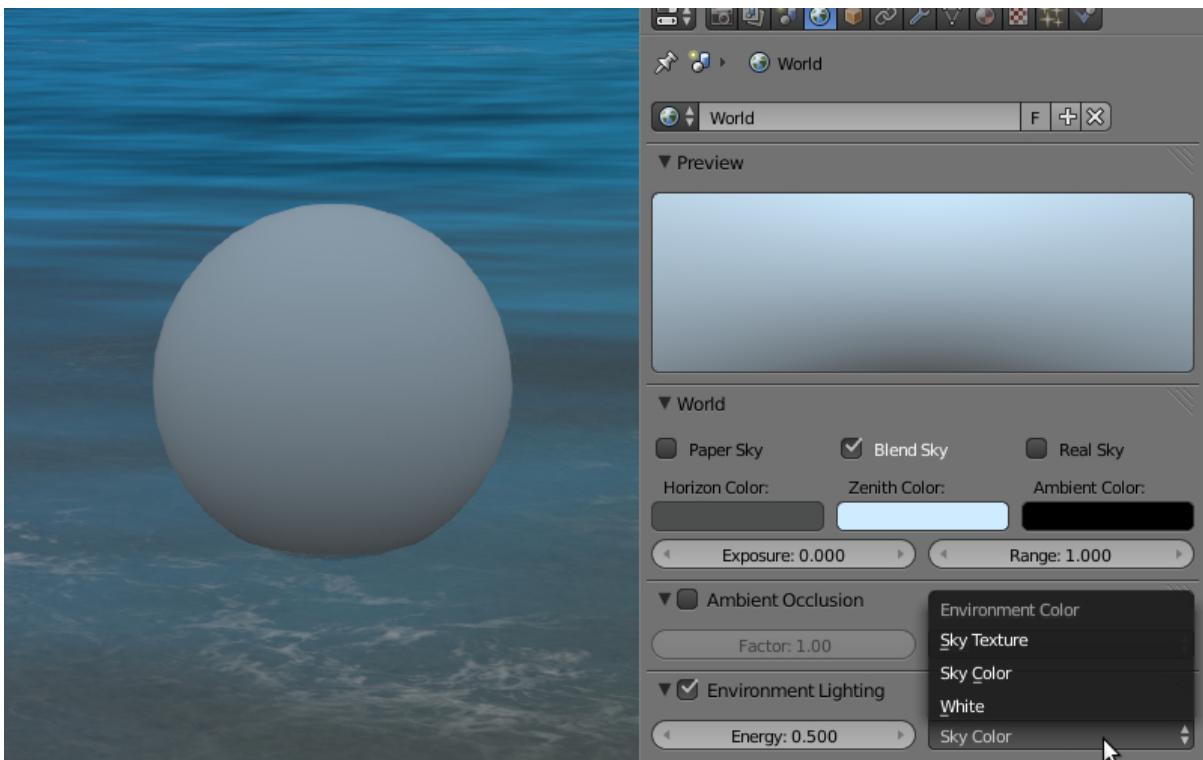
14.4 Освещение от окружающей среды

Движок поддерживает 3 метода симуляции рассеянного освещения от окружающих объектов.

1. “Плоское” освещение белым цветом.
2. Полусферическая модель освещения, в которой задается цвет горизонта и цвет зенита, в результате чего объекты заливаются градиентом между этими цветами в зависимости от направления нормалей.
3. Освещение с помощью *карты окружения* - т.н. методика image-based lighting.

14.4.1 Активация

Включить опцию **Environment Lighting** во вкладке **World**.



14.4.2 Настройка

World > Environment Lighting > Energy Интенсивность освещения от окружающей среды. Значение по умолчанию 1.0.

World > Environment Lighting > Environment Color Выбор метода симуляции рассеянного освещения: **White** - “плоское” освещение белым цветом, **Sky Color** - полусферическая модель, **Sky Texture** - освещение с помощью *карты окружения*. Значение по умолчанию **White**.

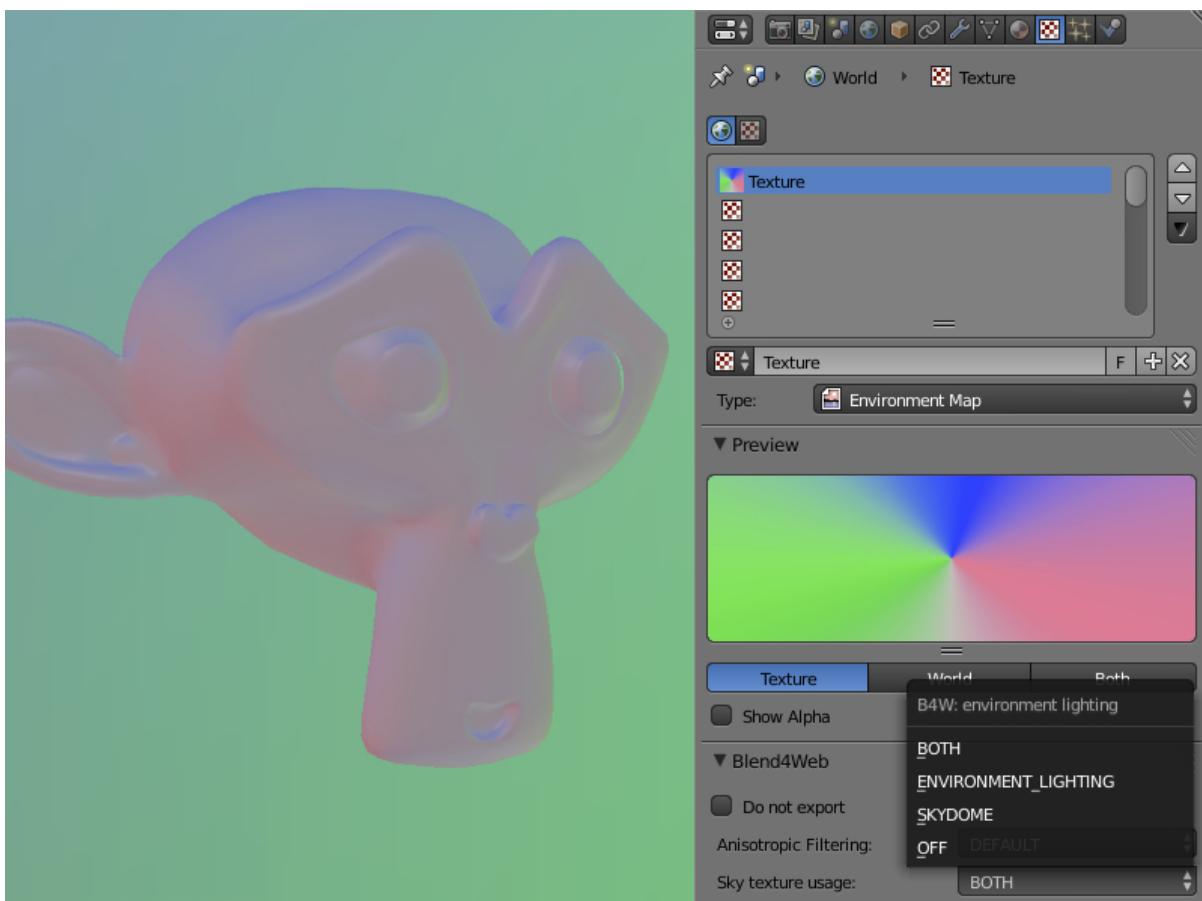
World > Horizon Color и World > Zenith Color Если выбрана полусферическая модель рассеянного освещения **Sky Color**, цвет горизонта и цвет зенита задаются цветоподборщиками **World > Horizon Color** (цвет горизонта) и **World > Zenith Color** (цвет зенита). При выборе цвета рекомендуется активировать опцию **World > Blend Sky**.

14.4.3 Использование карты окружения

Для того, чтобы использовать карту окружения в целях реализации освещения от окружающей среды, необходимо:

1. Включить опцию **Environment Lighting** во вкладке **World**.
2. Выбрать метод **Environment Lighting > Environment Color > Sky Texture**.
3. Перейти из вкладки **World** во вкладку **Texture**.
4. Создать *карту окружения*, загрузить в нее соответствующее изображение.

- Для карты окружения на панели Blend4Web для значения `Sky texture usage` выбрать `ENVIRONMENT_LIGHTING` или `BOTH` (опция `BOTH` активирует также использование этой текстуры в качестве *текстуры неба*).



14.5 Тени

Тени - один из важнейших элементов при рендеринге конечного изображения. Они предоставляют зрителю не только информацию о силуэте объектов, но и об их высоте и взаимном расположении, положении источника света и т.д.

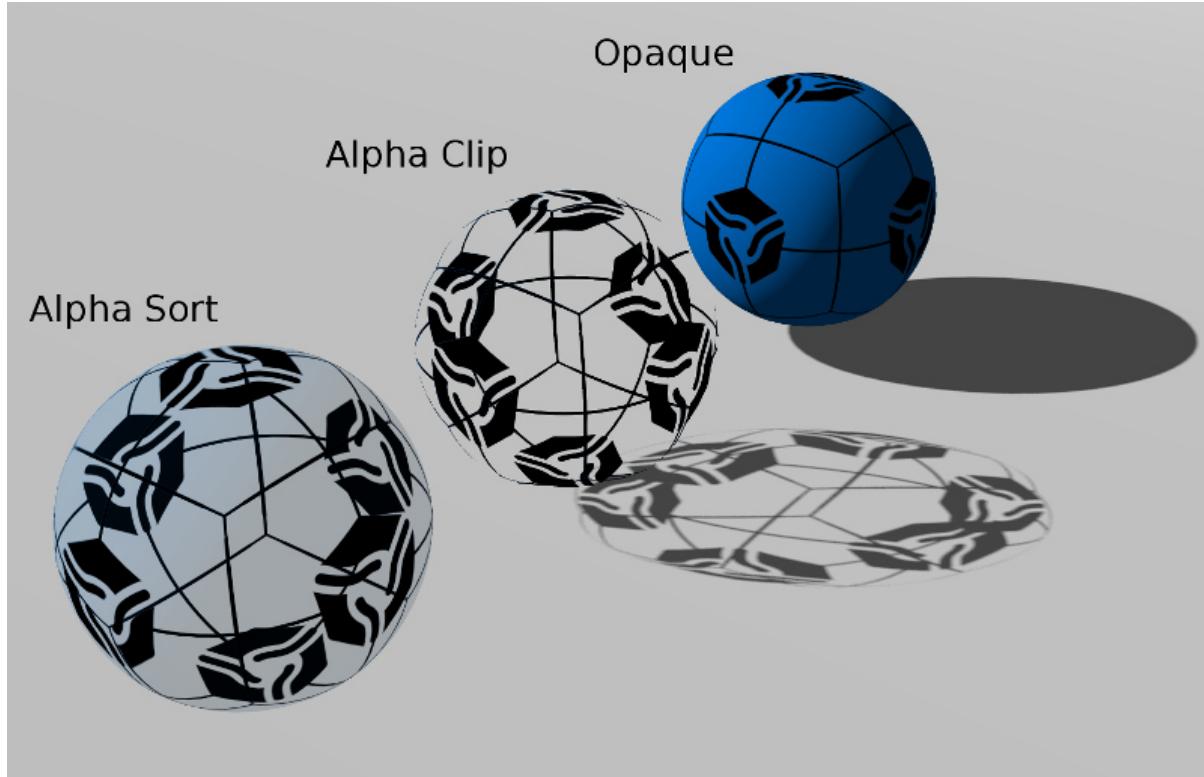
Платформой Blend4Web реализуются такие техники, как каскадные карты теней (CSM) и смягченные тени (PCF).

14.5.1 Активация

- На объектах, **отбрасывающих** тени, включить опцию `Blend4Web > Shadows: Cast` во вкладке `Object`.

2. На объектах, **получающих** тени, включить опцию Blend4Web > Shadows: Receive во вкладке Object.
3. Убедиться, что включена опция Blend4Web > Render shadows во вкладке Scene.

Примечание: Объекты, имеющие *прозрачные материалы с градиентом*, не отбрасывают теней.

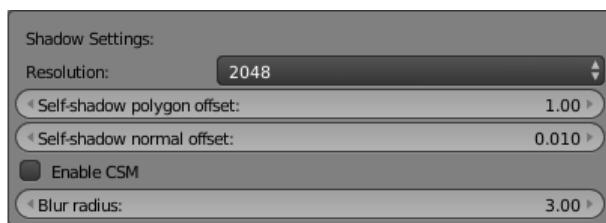


14.5.2 Настройка

Направление В случае наличия нескольких источников света рекомендуется указать, какой именно источник света будет использоваться для расчета падающих теней, включив опцию Blend4Web > Generate shadows во вкладке Object Data при выборе объекта-лампы.

Цвет Цвет тени определяется настройками *освещения от окружающей среды*.

Во вкладке World на панели Blend4Web > Shadow Settings находятся дополнительные настройки:



Resolution Разрешение используемой карты теней. Значение по умолчанию: 2048x2048px.

Self-shadow polygon offset Коэффициент смещения полигона в зависимости от ориентации к источнику света. Значение по умолчанию: 1.

Self-shadow normal offset Коэффициент смещения полигона по нормали. Значение по умолчанию: 0.01.

Последние две настройки служат для борьбы с артефактами самозатенения. Они проявляются на объектах, одновременно отбрасывающих и принимающих тени. Параметр **Self-shadow polygon offset** лучше справляется с артефактами во внутренних областях полигонов, а **Self-shadow normal offset** - в приграничных. Оба параметра приводят к искажению теней, поэтому рекомендуется держать их как можно меньшими.

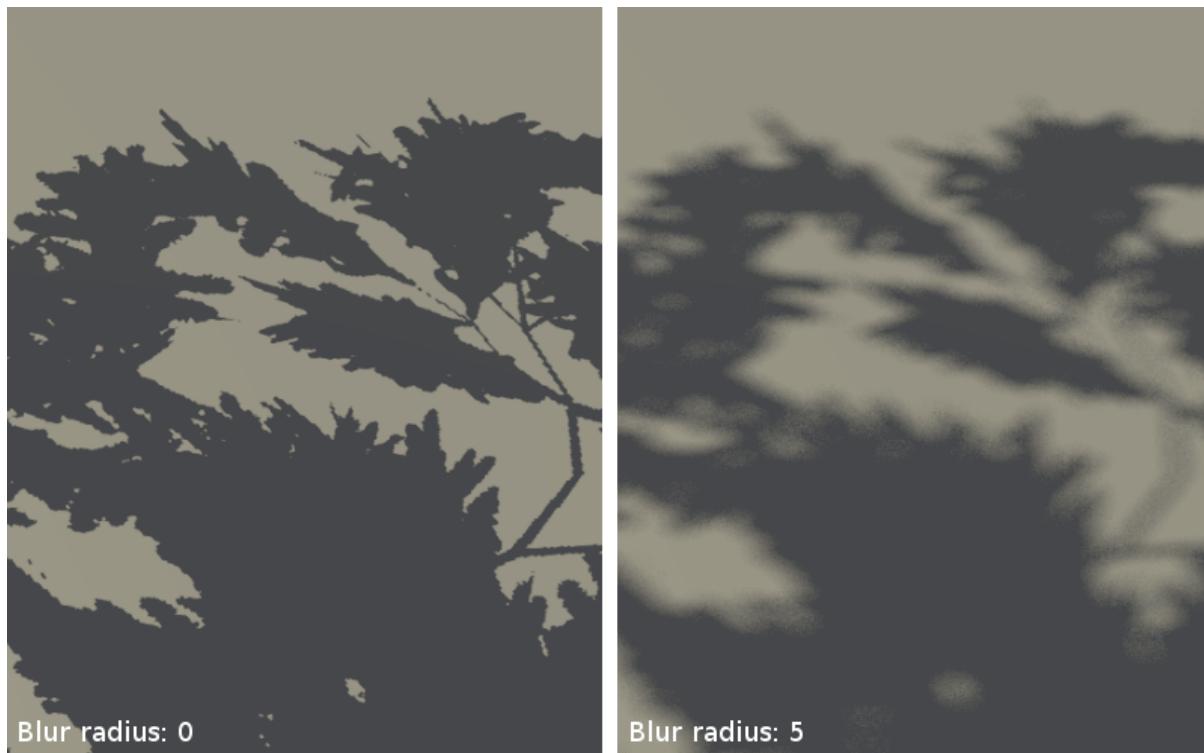


Enable CSM Включение каскадной модели теней; открывает доступ к расширенным настройкам. Отключено по умолчанию.

Эта опция позволяет выбрать один из следующих вариантов наложения теней:

- Стандартная модель, использующая одну оптимизированную карту теней, охватывающую всю сцену (**Enable CSM** отключена).
- Каскады теней (**Enable CSM** включена).

Blur radius Коэффициент размытия теней, позволяющий настроить смягченные тени. Значение по умолчанию: 3. Коэффициент 0 даст жесткие тени.



Смягченные тени могут повысить качество и реалистичность изображения. Они скрывают неизбежную при использовании основанных на изображениях техник зубчатость краев, особенно сильно проявляющуюся для карт теней низкого разрешения. Использование смягченных теней часто позволяет снизить разрешение без существенной потери качества.

Стандартная модель

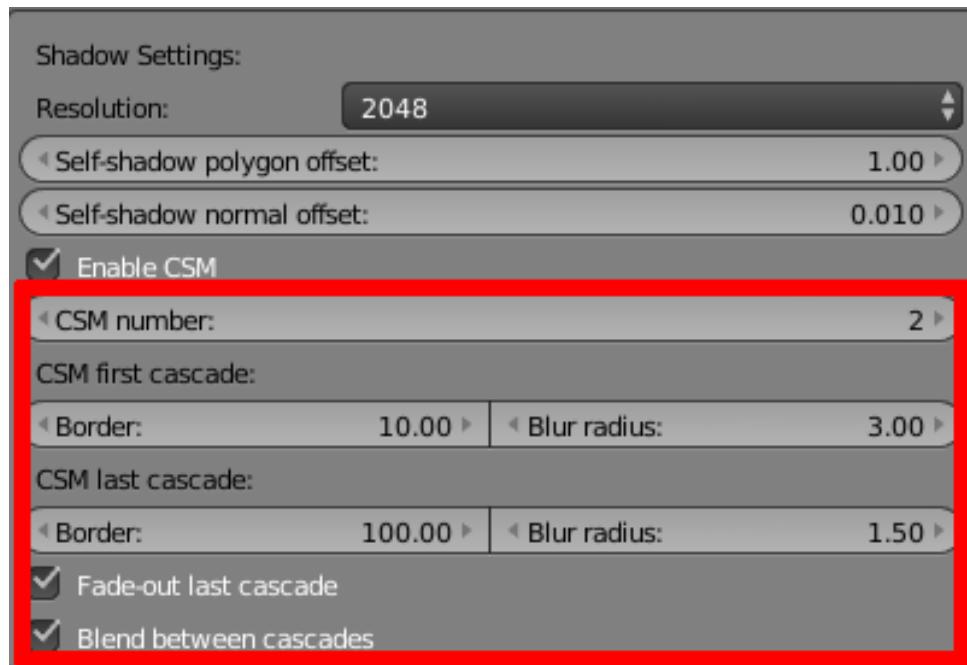
Этот вариант больше подходит для маленьких сцен, состоящих из небольшого числа объектов. Благодаря оптимизации на таких сценах можно добиться более высокого качества теней по сравнению с каскадной моделью. Данный вариант проще и быстрее настроить, а использование всего лишь одной карты теней положительно сказывается на производительности.

Каскады теней

Примечание: Данные настройки поддерживаются только для источников освещения типа Sun. Для других источников каскады автоматически отключаются.

Для обеспечения приемлемого качества теней и одновременно покрытия значительных пространств необходимо использовать несколько стадий генерации теней (каскадов). При этом вблизи наблюдателя располагается каскад с наилучшим качеством, вдали от наблюдателя — с наихудшим. Этот вариант больше подходит для сцен среднего и большого размера, например, игровых уровней.

При включении предоставляет расширенные настройки:



CSM number Количество каскадов теней. Поддерживается от 1 до 4 каскадов. Значение по умолчанию: 1.

CSM first cascade border Размер первого каскада. Значение по умолчанию: 10.0.

CSM last cascade border Размер последнего каскада. Значение по умолчанию: 100.0.

Размеры промежуточных каскадов интерполируются на основе последних двух параметров.

Примечание: При настройке следует помнить, что, увеличивая размер каскада, мы получаем на нем менее качественные тени. С другой стороны, уменьшение параметра **CSM first cascade border** приблизит к камере и сделает более заметными последние менее детальные каскады. Уменьшение параметра **CSM last cascade**

`border` приведет к исчезновению теней на более близком расстоянии от камеры. Однако, при использовании мягких теней качество в целом улучшится благодаря размытию на границах.



CSM first cascade blur radius Коэффициент размытия на первом каскаде. Значение по умолчанию: 3. Коэффициент 0 даст жесткие тени.

CSM last cascade blur radius Коэффициент размытия на последнем каскаде. Значение по умолчанию: 1.5. Коэффициент 0 даст жесткие тени.

Радиус размытия каждого промежуточного каскада интерполируется на основе этих параметров.

Примечание: Смягченные тени рекомендуется настраивать сначала на первом каскаде опцией `CSM first cascade blur radius`, а далее на всех остальных с помощью `CSM last cascade blur radius`. Часто на последнем каскаде может потребоваться размытие меньшее, нежели на первом. Это нужно для того, чтобы тени на последнем каскаде не стали слишком блеклыми из-за низкой детализации, к тому же это уменьшит нежелательные артефакты самозатенения.

Fade-out last cascade Плавное исчезновение последнего каскада. По умолчанию включено.

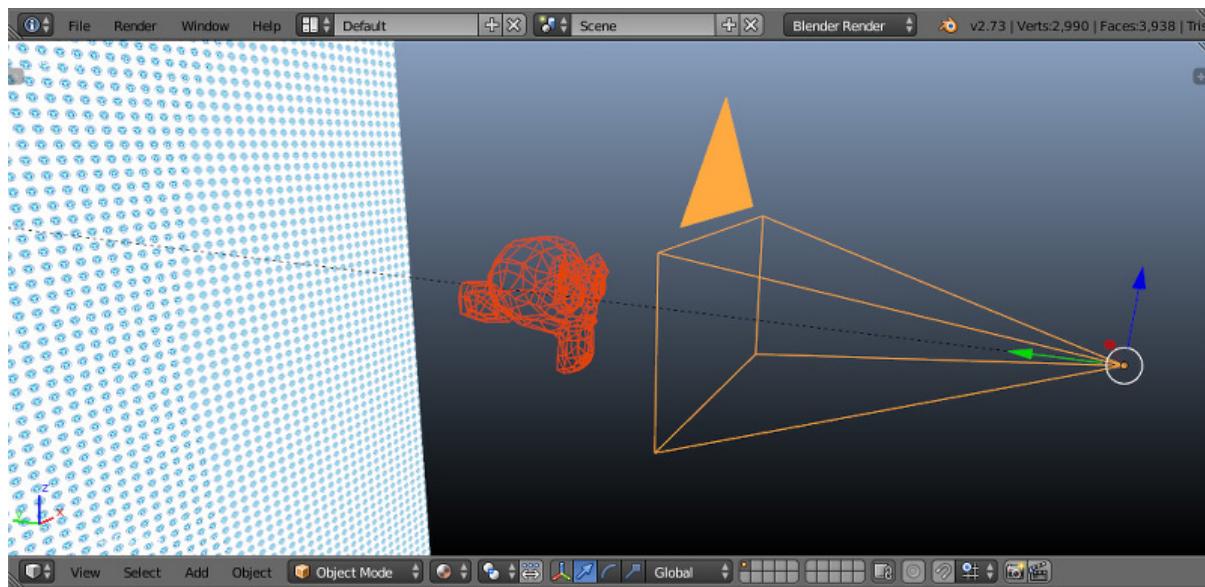
Blend between cascades Сглаживание границ между каскадами. По умолчанию включено.



14.6 Цвет фона

Цвет фона можно задать несколькими способами:

1. Установить параметры `Horizon Color` и `Zenith Color` на вкладке `World` в Blender, предварительно выставив опцию `Sky Settings > Render Sky`.
2. Поместить сцену внутрь модели (например, куба или сферы) с направленными внутрь нормалями, с материалом и optionalной текстурой.
3. Разместить перед камерой поверхность с материалом и optionalной текстурой, присоединить ее к камере связью родитель-потомок. При необходимости настроить расстояние до поверхности, переднюю и заднюю плоскости отсечения камеры.



4. Использовать *текстуру неба*.
5. Настроить динамически генерируемую *атмосферу*.
6. Установить параметр движка `background_color`, используя программный метод `config.set()`, предварительно отключив опцию *Sky Settings > Render Sky* на вкладке *World* в Blender. Установленное значение используется в качестве аргумента метода WebGL `clearColor()`. Для получения корректных результатов рекомендуется отключить прозрачность контекста WebGL (параметр `alpha`). Такая конфигурация используется по умолчанию в стандартном веб-плеере движка.

```

var m_cfg = b4w.require("config");
var m_main = b4w.require("main");

// gray
m_cfg.set("background_color", new Float32Array([0.224, 0.224, 0.224, 1.0]));
m_cfg.set("alpha", false);

m_main.init(...);

```

7. В качестве фона можно использовать любой HTML контент, находящийся позади элемента `canvas`, который используется для рендеринга. Для этого необходимо активировать прозрачность контекста WebGL (параметр `alpha`), предварительно отключив опцию *Sky Settings > Render Sky* на вкладке *World* в Blender. Для получения корректных результатов рекомендуется выставить полностью прозрачный черный цвет фона. Такая конфигурация используется по умолчанию в стандартном *просмотрщике сцен* SDK движка.

```

var m_cfg = b4w.require("config");
var m_main = b4w.require("main");

m_cfg.set("background_color", new Float32Array([0.0, 0.0, 0.0, 0.0]));
m_cfg.set("alpha", true);

```

`m_main.init(...);`

См.также:

Альфа-композитинг

Постпроцессинговые эффекты

15.1 Размытие при движении

Эффект размытия при движении (motion blur) служит целям увеличения реализма интерактивной сцены. Он проявляется при движении камеры или объектов в виде “смазывания” изображения.



15.1.1 Активация

Выставить опцию Enable Motion Blur на панели Scene > Blend4Web.

15.1.2 Дополнительные настройки

На панели World > Blend4Web > Motion blur settings:

Motion blur factor Степень проявления эффекта. Чем выше значение, тем сильнее эффект размытия. Значение по умолчанию 0.01.

Motion blur decay threshold Степень плавности размытия. Чем выше значение, тем более резким будет эффект. Значение по умолчанию 0.01.

15.2 Глубина резкости камеры

Эффект глубины резкости камеры (depth of field, DOF) акцентирует внимание зрителя на части сцены. Проявляется в размытии изображения ближе и дальше от фокуса камеры.



15.2.1 Активация

1. Выбрать активную камеру, перейти на панель ее настроек (Object Data).
2. Далее возможны два варианта:
 - На панели Depth of Field в меню Focus выбрать объект, на котором будет сфокусирована камера. В этом случае при удалении или приближении к этому объекту будет происходить соответствующая коррекция фокуса камеры.

- На панели Depth of Field установить ненулевое значение Distance (в метрах). В этом случае фокус камеры будет располагаться на заданном расстоянии от камеры и перемещаться вместе с ней.

15.2.2 Дополнительные настройки

На панели настроек активной камеры Object Data > Blend4Web:

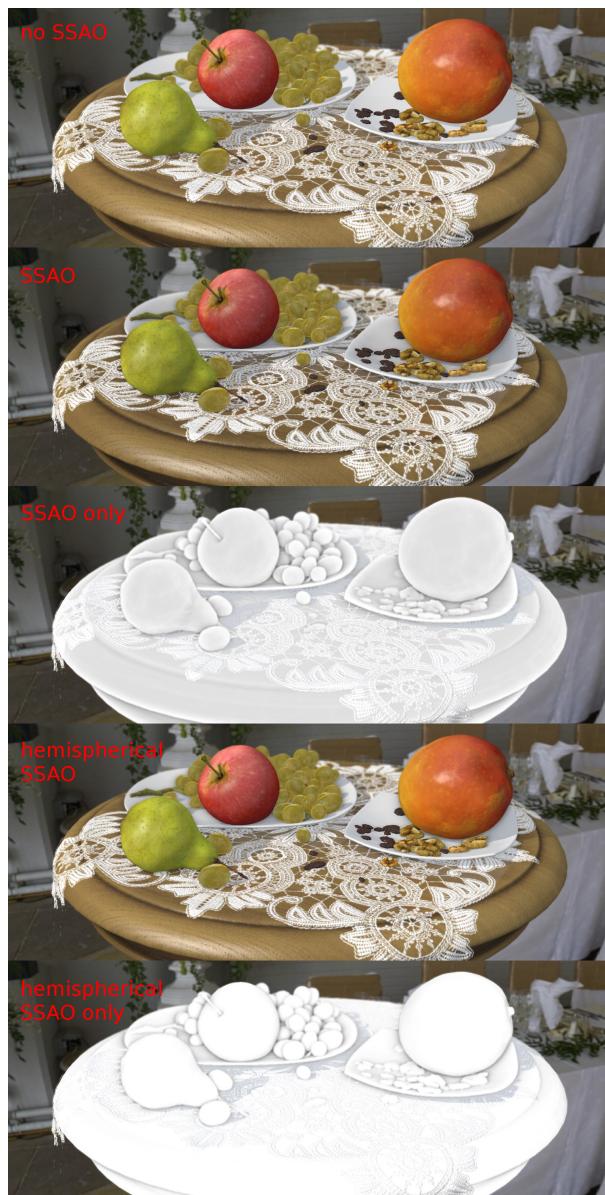
DOF front distance Расстояние от фокуса до ближней к камере плоскости, за которой происходит полное размытие (в метрах). Значение по умолчанию 1.0.

DOF rear distance Расстояние от фокуса до дальней от камеры плоскости, за которой происходит полное размытие (в метрах). Значение по умолчанию 1.0.

DOF power Степень размытия. Значение по умолчанию 3.0.

15.3 Взаимное затенение

Эффект взаимного затенения (screen-space ambient occlusion, SSAO) применяется с целью воспроизведения сложного переотражения света от объектов. Пространство между близкими объектами менее доступно для рассеянного света и поэтому затеняется сильнее.



15.3.1 Активация

Выставить опции Enable SSAO и Render Shadows на панели Scene > Blend4Web.

15.3.2 Дополнительные настройки

На панели “мира” World > Blend4Web > SSAO Settings:

Radius Increase Фактор умножения радиуса сферического сэмплинга при переходе от внутреннего кольца к внешнему. Значение по умолчанию 3.0.

Use Hemisphere Использовать для расчёта затенения полусферический сэмплинг вместо сферического. Помимо этого используется другой закон затенения.

Use Blur Depth Test Если активировано - используется размытие SSAO на основе буфера глубины. Иначе - размытие по квадарту 4x4 по соседним пикселям.

Blur Depth Test Discard Value Влияние разницы глубины сэмлов на их вес в размытии. Используется при активированном параметре Use Blur Depth Test. Значение по умолчанию 1.0.

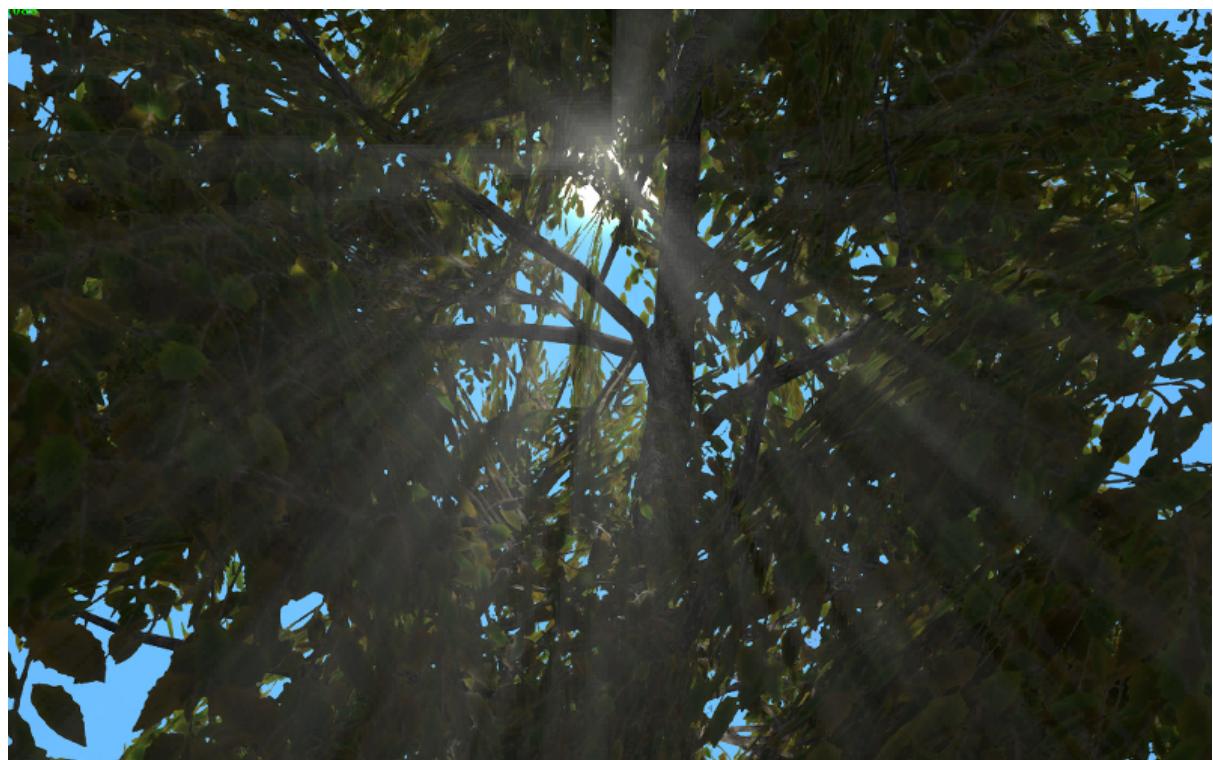
Influence Степень проявленности эффекта взаимного затенения. Значение по умолчанию 0.7.

Distance Factor Фактор уменьшения проявленности эффекта взаимного затенения с расстоянием. Значение по умолчанию 0.0 (т.е. уменьшения нет).

Samples Количество сэмплов (чем больше, тем лучше качество, но меньше производительность). Значение по умолчанию 16.

15.4 Сумеречные лучи

Эффект сумеречных лучей (crepuscular rays, “god rays”) симулирует известное природное явление - свечение освещенных областей воздуха.



15.4.1 Активация

Выставить опцию Enable God Rays на панели Scene > Blend4Web.

15.4.2 Дополнительные настройки

На панели “мира” World > Blend4Web > God Rays Settings:

God Rays Intensity Степень проявленности эффекта. Значение по умолчанию 0.7.

Maximum Ray Length Фактор длины лучей. Определяет шаг сэмплов радиального размытия. Значение по умолчанию 1.0.

Steps Per Pass Количество шагов на один сэмпл. Значение по умолчанию 10.0.

15.5 Эффект засветки ярких деталей

Эффект засветки (Bloom) проявляется при наличии на экране элементов с большой разницей в яркости. Вокруг ярких деталей создается светящийся ореол.



15.5.1 Активация

Выставить опцию Enable Bloom на панели Scene > Blend4Web.

15.5.2 Дополнительные настройки

На панели “мира” World > Blend4Web > Bloom Settings:

Key Интенсивность эффекта свечения.

Blur Степень размытия засветки.

Edge Luminance Граничное значение относительной яркости элемента, выше которого начинает проявляться эффект засветки.

15.6 Подсветка контура (outline glow)

В результате применения эффекта подсветки контура вокруг объекта появляется светящийся ореол произвольного цвета.



15.6.1 Активация

Эффект подсветки контура активируется программно через API. Может быть реализован как эффект постоянного свечения, так и затухающего, пульсирующего и любой другой модели. Для включения возможности подсветки объектов необходимо убедиться, что в панели Scene > Blend4Web поле **Enable Object Outlining** установлено в состояние YES или AUTO.

15.6.2 Дополнительные настройки

На панели Object > Blend4Web:

Enable Outlining Разрешить использование эффекта подсветки контура на конкретном объекте.

Outline Duration Длительность анимации подсветки, сек. Значение по умолчанию 1.

Outline Period Период повторения анимации подсветки, сек. Значение по умолчанию 1.

Outline Relapses Количество итераций анимации подсветки. В случае 0 анимация будет повторяться бесконечно. Значение по умолчанию 0.

Outline on Select Активация анимации подсветки при выделении объекта. Для данного случая необходимо выставить опцию **Selectable**. При необходимости реализации собственной модели подсветки объекта следует отключить эту опцию во избежание конфликта.

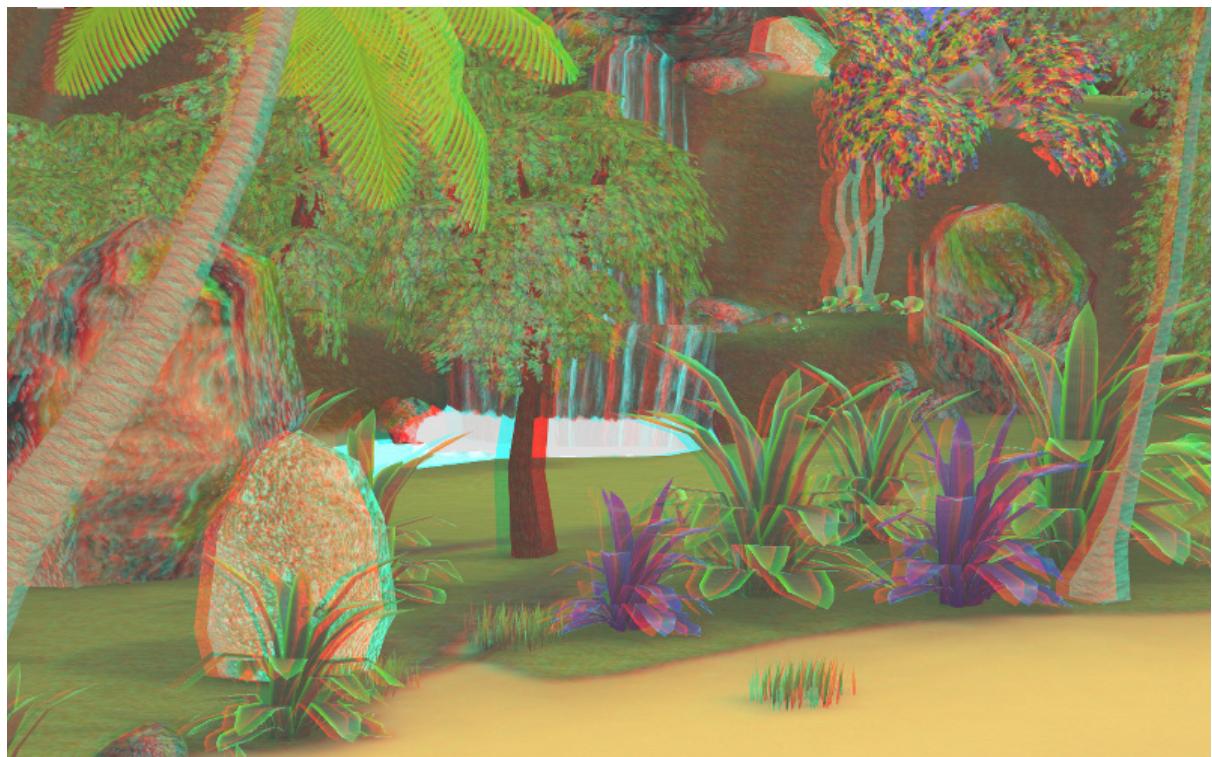
На панели World > Blend4Web:

Objects Outline Color Общий цвет эффекта для всех объектов. Значение по умолчанию (1,1,1), т.е. белый.

Outline Factor Толщина и яркость ореола, окружающего объект. Падает с уменьшением параметра. Значение по умолчанию 1.

При управлении через API настройки на панели World > Blend4Web воспринимаются как настройки по умолчанию.

15.7 Аналиф стереоизображение



15.7.1 Активация

Режим стереоизображения предназначен для просмотра контента в специальных очках и активируется приложением.

15.7.2 Дополнительные настройки

Нет.

15.8 Коррекция цвета



15.8.1 Активация

Выставить опцию Enable Color Correction на панели Scene > Blend4Web.

15.8.2 Дополнительные настройки

На панели “мира” World > Blend4Web > Color Correction Settings:

Brightness Яркость. Значение по умолчанию 0.0.

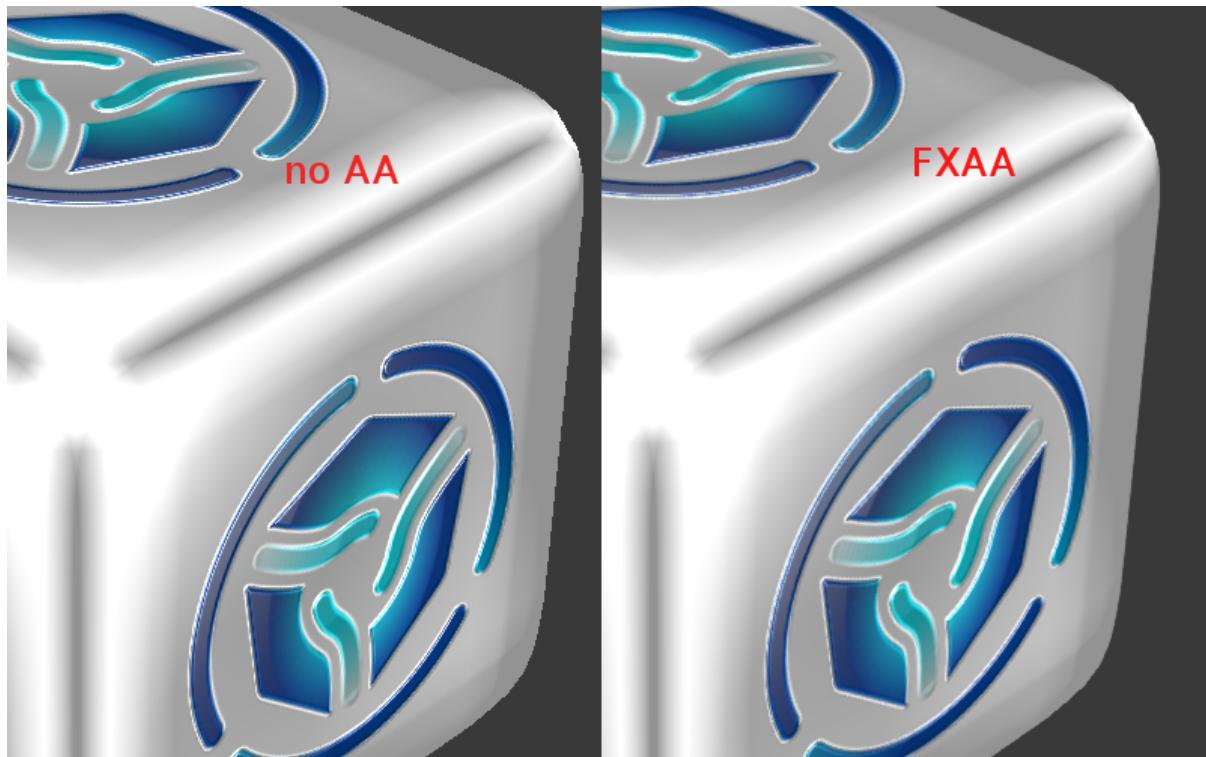
Contrast Контрастность. Значение по умолчанию 0.0.

Exposure Экспозиция. Значение по умолчанию 1.0.

Saturation Насыщенность. Значение по умолчанию 1.0.

15.9 Сглаживание

Сглаживание (anti-aliasing) необходимо для уменьшения влияния нежелательных артефактов рендеринга (“зубчатости”).



15.9.1 Активация

Выставить опцию Enable Antialiasing на панели Scene > Blend4Web.

15.9.2 Дополнительные настройки

Метод сглаживания назначается одновременно с выбором профиля работы движка.

- *низкое качество* - антиалиасинг отключен
- *высокое качество* - метод антиалиасинга FXAA (Fast Approximate Anti-Aliasing), предложенный Nvidia
- *максимальное качество* - метод антиалиасинга SMAA (Enhanced Subpixel Morphological Anti-Aliasing), предложенный Crytek

Система частиц. Флюиды

Система частиц предназначена для визуализации явлений, обусловленных движением множественных малых объектов, таких как дым, огонь, брызги воды и др.



Необходимым элементом системы частиц является эмиттер - объект, определяющий местоположение и направление исходящего потока частиц.

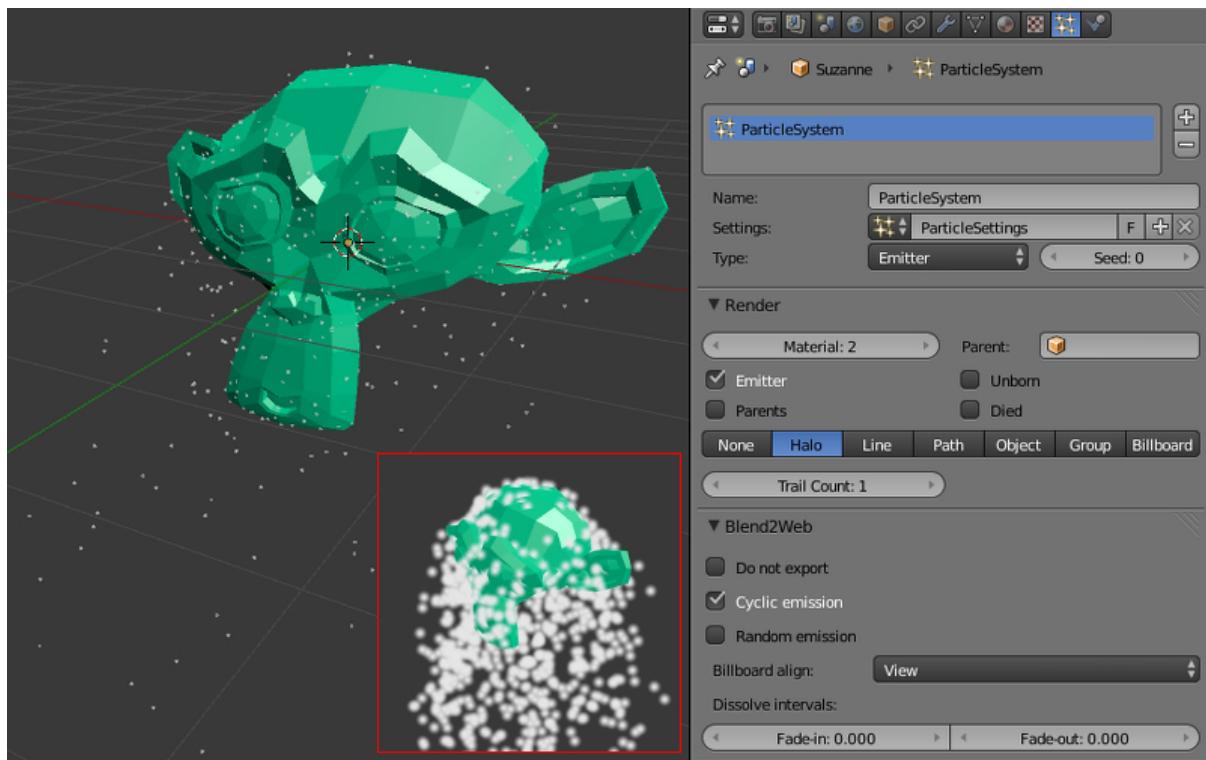
16.1 Использование

16.1.1 Необходимые этапы

1. Добавить на сцену меш - эмиттер.
2. Создать на эмиттере материал для частиц, например типа *Halo*. Поддерживается также материал типа *Surface* с обязательной диффузной текстурой.
3. Добавить на эмиттере систему частиц.
4. Инициализировать воспроизведение в движке. Возможны два варианта:
 - “циклическое испускание” - для системы частиц выставить опцию *Blend4Web > Cyclic emission*.
 - “нециклическая анимация” - для эмиттера выставить опцию *Blend4Web > Animation > Use default*.

16.1.2 Рекомендуемые дополнительные настройки

1. Для материала частиц выставить тип прозрачности *Add*.
2. Если отображение эмиттера на сцене не требуется, отключить опцию *Particles > Render > Emitter*.
3. Если отображение эмиттера на сцене необходимо, для него можно использовать дополнительные материалы. В этом случае в настройках системы частиц нужно выбрать материал частиц *Particles > Render > Material*.
4. В случае использования для частиц материала типа *Surface*, к материалу необходимо подключить диффузную текстуру (обычно с альфа-каналом). В меню *Mapping > Coordinates* выбрать *UV*. Убедиться, что меш эмиттера имеет развертку.



16.2 Настройка

Параметры системы частиц настраиваются во вкладке **Particles**. Поддерживается несколько систем частиц на одном эмиттере.

16.2.1 Общие настройки

Name Название системы частиц. Значение по умолчанию “ParticleSystem”.

Settings Ссылка на блок данных с настройками системы частиц. Блоки данных с настройками могут быть общими для разных систем частиц.

Type Тип системы частиц: **Emitter** или **Hair**. Системы частиц типа **Hair** используются для создания множественных копий (инстансинга) объектов. Значение по умолчанию **Emitter**.

Seed Индекс в таблице случайных чисел, используемых для генерации системы частиц. Значение по умолчанию 0.

16.2.2 Настройки испускания

Emission > Number Количество частиц. Значение по умолчанию 1000.

Emission > Start Первый кадр, после которого начинается испускание частиц. Значение по умолчанию 1.0.

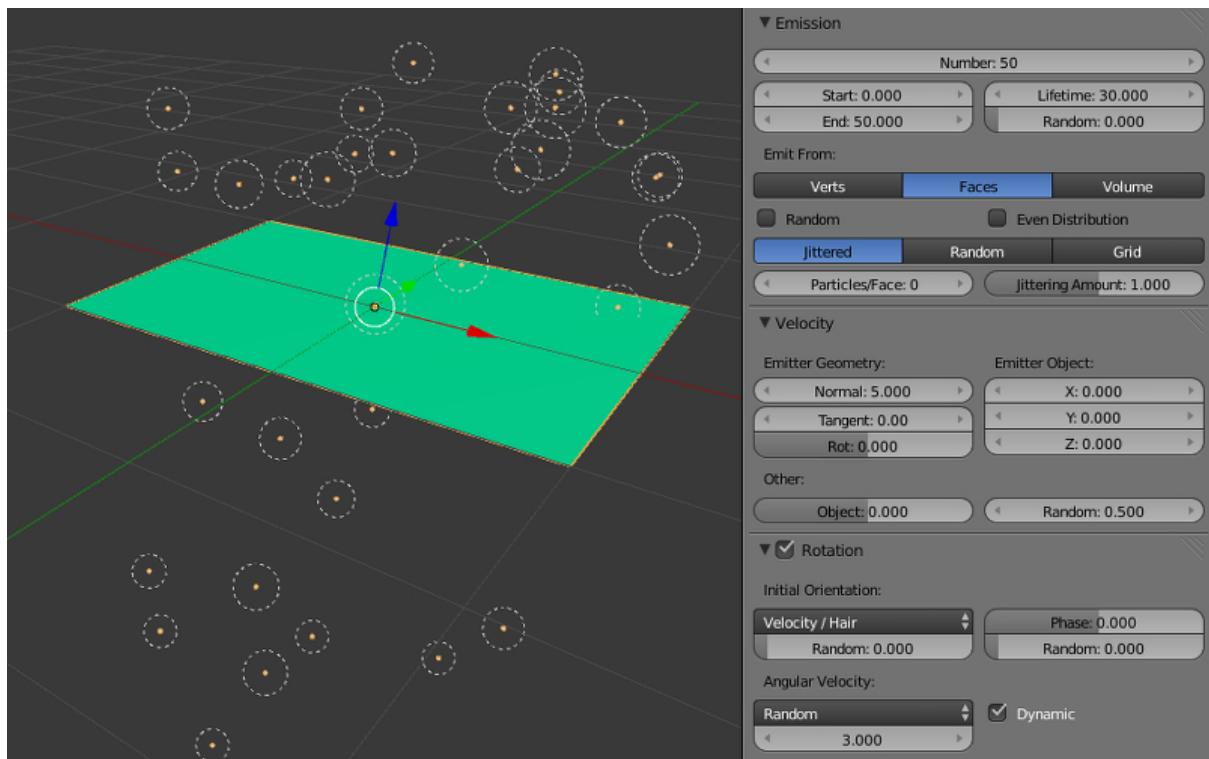
Emission > End Последний кадр, после которого прекращается испускание частиц. Значение по умолчанию 200.0.

Emission > Lifetime Время жизни частиц в кадрах. Значение по умолчанию 50.0.

Emission > Lifetime > Random Фактор случайности для времени жизни. Значение по умолчанию 0.0.

Emission > Emit From Источник испускания. Поддерживаются вершины Verts, грани Faces. Значение по умолчанию Faces.

Emission > Emit From > Distribution Настройки распределения испускания: Jittered, Random, Grid. Игнорируются движком. Всегда используется случайное распределение (Random). Значение по умолчанию Jittered.



16.2.3 Настройки направления

Поддерживаются только:

Velocity > Emitter Geometry > Normal Фактор влияния на испускание вдоль нормалей меша эмиттера. Значение по умолчанию 1.0.

Velocity > Other > Random Фактор случайности для направления испускания. Значение по умолчанию 0.0.

16.2.4 Настройки вращения

Поддерживаются только:

Rotation > Angular Velocity > Mode Режим собственного вращения биллбордов частиц. Поддерживаются *Velocity* (постоянная скорость вращения), *Random* (случайное вращение), *None* (нет вращения). Значение по умолчанию *Velocity*.

Rotation > Angular Velocity > Factor Фактор скорости собственного вращения биллбордов частиц. Значение по умолчанию 0.0.

16.2.5 Настройки физики

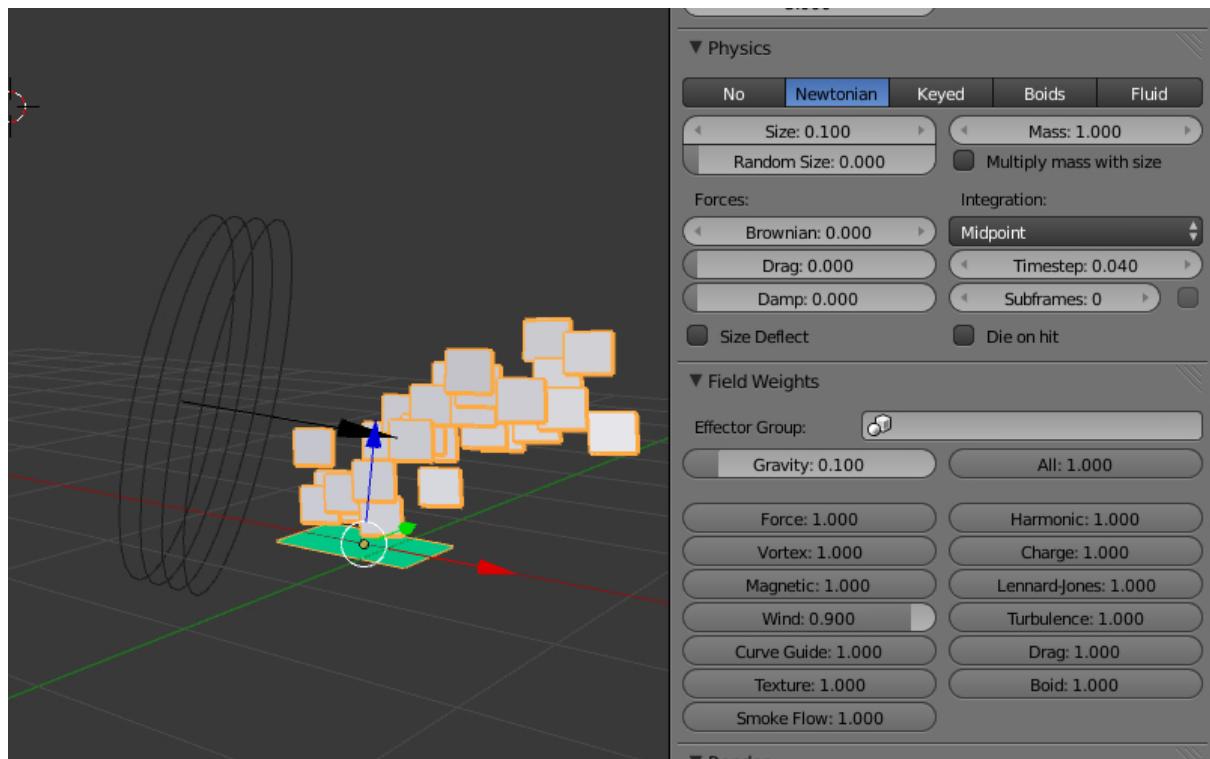
Поддерживаются только:

Physics > Type Тип расчетов физики: *No*, *Newtonian*, *Keyed*, *Boids*, *Fluid*. Игнорируется движком. Всегда используется физика Ньютона (*Newtonian*). Значение по умолчанию *Newtonian*.

Physics > Size Размер частиц. Значение по умолчанию 0.05.

Physics > Mass Масса частиц. Влияет на взаимодействие с силовыми полями (в частности, с ветром). Значение по умолчанию 1.0.

Physics > Forces > Brownian Экспортируется, но не используется движком.



16.2.6 Настройки отображения

Поддерживаются только:

Render > Material Меню выбора материала частиц. Используется в случае использования эмиттером нескольких материалов. Значение по умолчанию

Default Material.

Render > Emitter Опция включения отображения эмиттера на сцене. По умолчанию включено.

Render > Type Режим отображения частиц: **None**, **Halo**, **Line**, **Path**, **Object**, **Group**, **Billboard**. Движком различаются режимы **Object** и **Group**, использующиеся для инстансинга объектов и групп объектов, соответственно. Другие режимы игнорируются. Для удобства отображения биллбордов рекомендуется включать режим **Billboard**. Значение по умолчанию **Halo**.

16.2.7 Настройки влияния силовых полей

Поддерживаются только:

Field Weights > Gravity Фактор влияния гравитационного поля (земное притяжение). Значение по умолчанию 1.0.

Field Weights > Wind Фактор влияния ветра. Необходимо присутствие объекта силового поля (добавляется **Add > Force Field**) типа **Wind** (ветер). На систему частиц оказывают также настройки направления и силы ветра. Значение по умолчанию 1.0.

16.2.8 Специальные настройки движка

Blend4Web > Do not export Не поддерживается.

Blend4Web > Cyclic emission Опция включает циклический режим испускания. Применяется для постоянных эффектов (дым, горение, брызги). Рекомендуется выставить нулевое значение **Emission > Start**. По умолчанию выключено.

Blend4Web > Random emission Опция устанавливает случайный характер времени испускания частиц. По умолчанию выключено.

Blend4Web > Billboard align Способ ориентирования биллбордов: **View** - поворачивать к камере, **XY plane**, **YZ plane**, **ZX plane** - ориентировать в соответствующей плоскости (в мировой системе координат Blender'a). Значение по умолчанию **View**.

Blend4Web > Dissolve intervals > Fade-in и Fade-out Начальный и конечный интервалы (в кадрах) для постепенного увеличения и уменьшения прозрачности частиц.

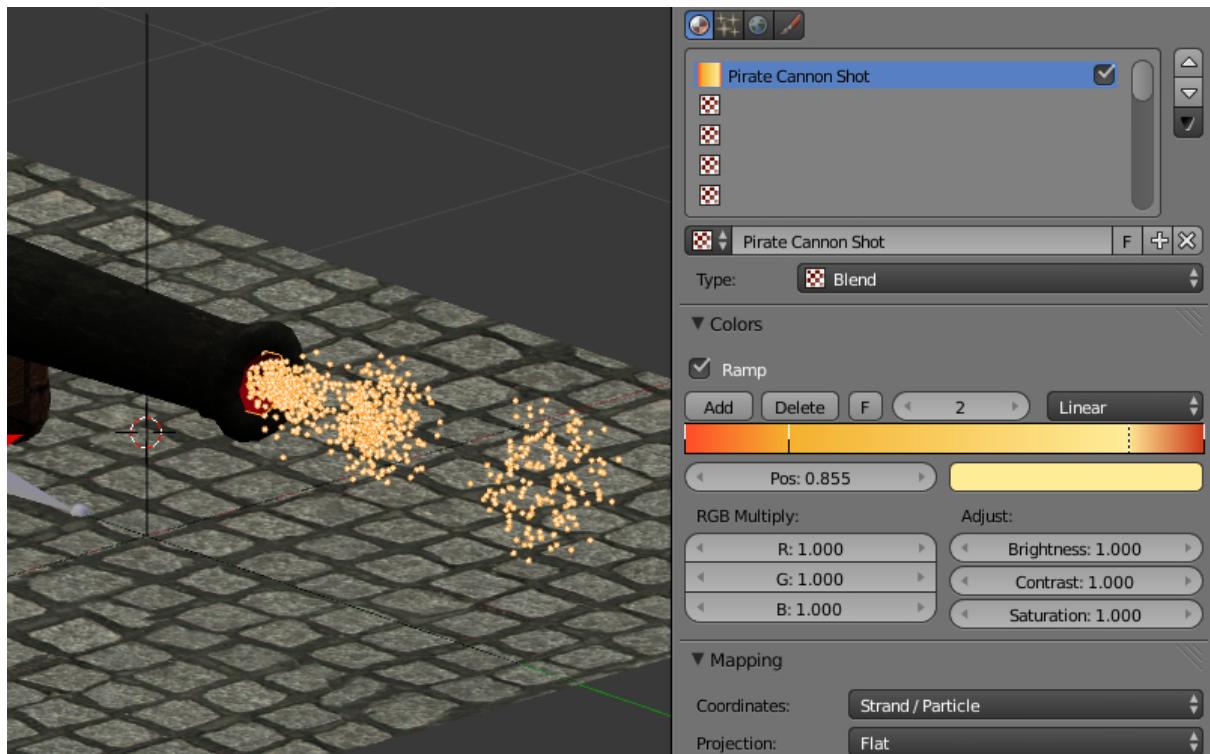
Blend4Web > Coordinate system Система координат испускаемых частиц: **Local** - использовать локальную систему координат эмиттера, **World** - использовать мировую систему координат.

16.3 Текстуры в системах частиц

16.3.1 Текстуры материала частиц

В материалах частиц типа **Surface** **необходимо** наличие диффузной текстуры (обычно с альфа-каналом). В меню **Mapping > Coordinates** выбрать **UV**. Убедиться, что меш эмиттера имеет развертку.

В материалах частиц типа **Halo** **возможно** использование текстуры типа **Blend** с линейным (**Linear**) градиентом. В меню **Mapping > Coordinates** выбрать **Strand / Particle**. На текстуре необходимо включить использование рампы (**Ramp**). Допускается использование до 4 контрольных точек градиента.

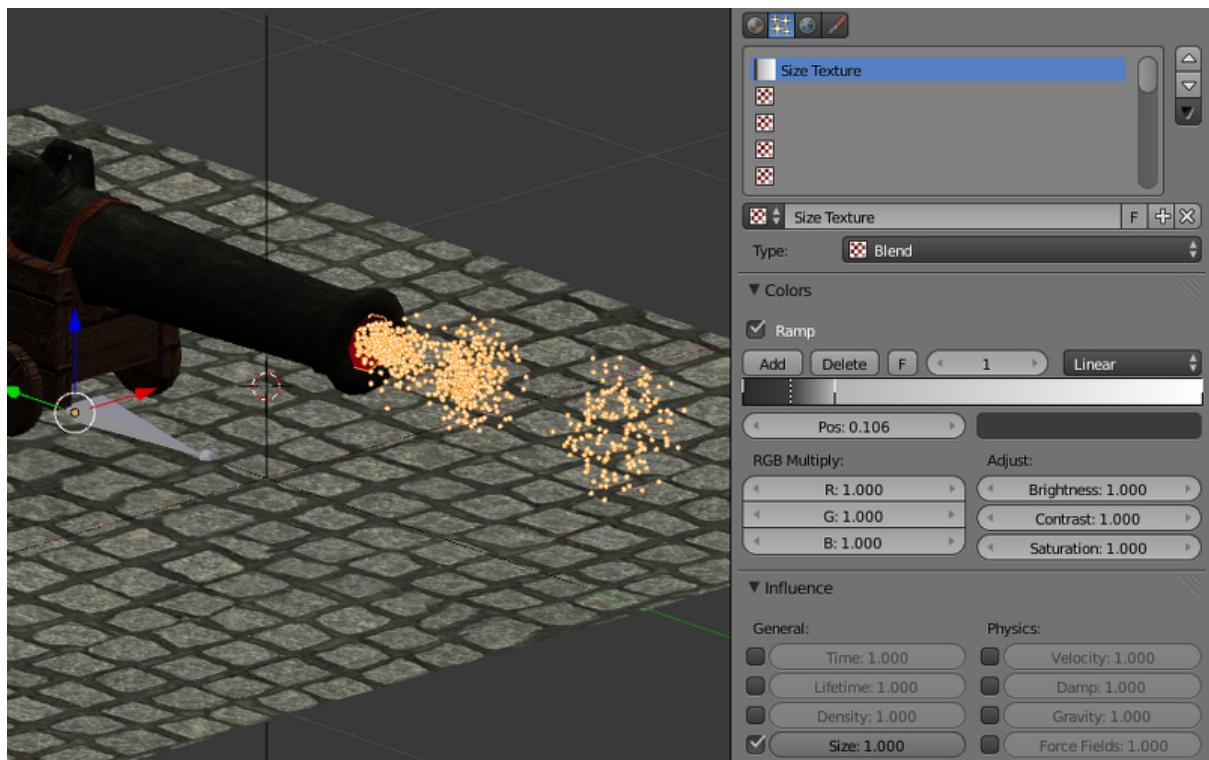


16.3.2 Текстуры системы частиц

Для настройки поведения системы частиц могут быть использованы текстуры. В отличие от текстур, используемых материалами частиц, такие текстуры относятся к блоку данных (datablock) системы частиц, а не к блоку данных материала. Чтобы создать текстуру системы частиц, необходимо из вкладки **Particles** перейти во вкладку **Textures**, после чего нажать **New**.

Поддерживаются только текстуры типа **Blend** с линейным (**Linear**) градиентом. На текстуре необходимо включить использование рампы (**Ramp**). Допускается использование до 4 контрольных точек градиента.

На панели **Influence** необходимо выбрать параметр, на который воздействует текстура. В настоящий момент поддерживается только **Size** (размер).



Результат применения текстур градиента для материала частиц и для системы частиц:



[Ссылка на модель](#)

Система частиц. Инстансинг

Система частиц может использоваться для создания множественных копий объектов (инстансинга). Кроме сокращения времени на создание сцены, инстансинг позволяет сэкономить на времени загрузки и сократить потребление памяти сценой, нежели при использовании отдельных объектов.



Использование системы частиц для инстансинга имеет ограничения:

- Невозможно осуществлять движение и анимацию объектов внутри системы частиц.
- Не допускается создавать иерархические сущности среди объектов внутри си-

стемы частиц за исключением дуплицирования групп.

- Не допускается инстансинг отличных от объектов-мешей сущностей.

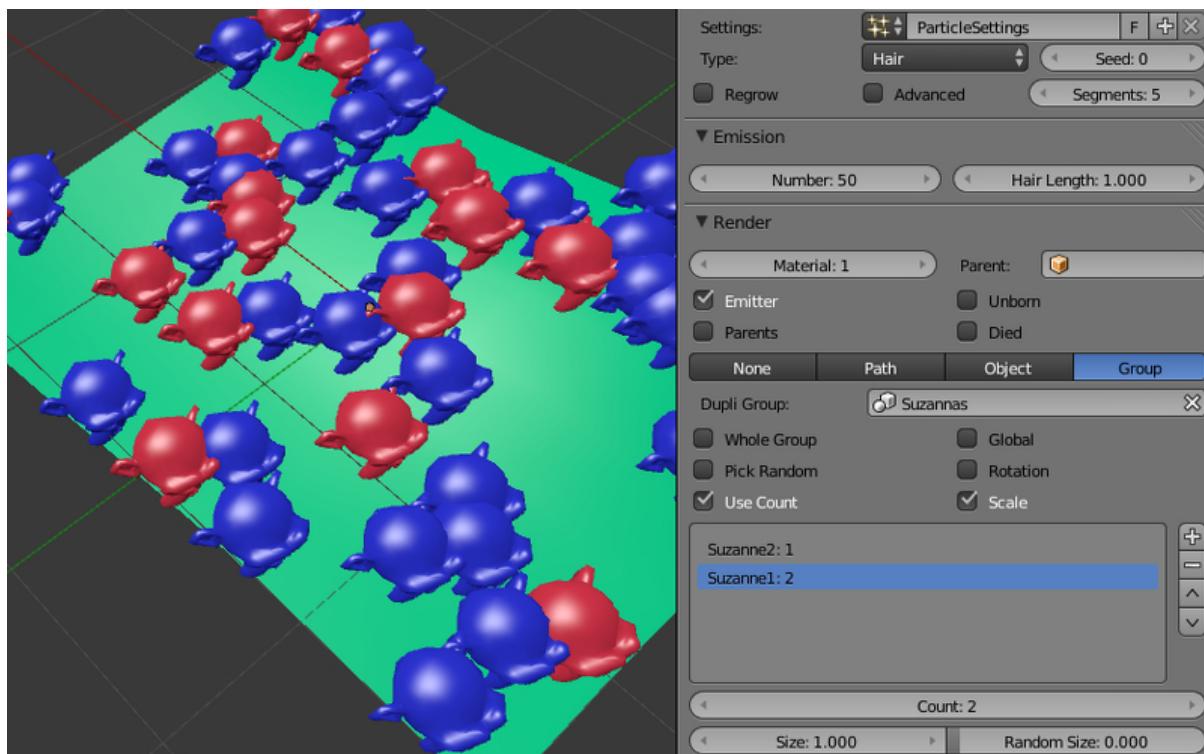
17.1 Настройки системы частиц

Активация

1. На эмиттере создать систему частиц типа Hair.
2. В панели Render выбрать тип отображения Object (или Group).
3. В поле Dupli Object (или Dupli Group) выбрать объект (или группу объектов) для инстансинга. Поддерживаются как локальные, так и подключенные по ссылке объекты (или группы).

Рекомендуемые дополнительные настройки

1. Для корректного отображения размера установить значение 1.0 для параметров Emission > Hair Length и Render > Size.
2. Для установки корректной ориентации временно включить опцию Advanced, активировать панель Rotation и в меню Initial Orientation выбрать None. Отключить опцию Advanced. Также рекомендуется включить опцию Render > Rotation.



Настройка

Render > Use Count

Опция доступна для групп объектов-частиц. При включении появляется интерфейс установки относительного количества входящих в группу объектов. Движок не воспроизводит точное местонахождение объектов заданных типов.

Blend4Web > Random location and size

Опция устанавливает случайный характер расположения и размеров объектов. Если опция включена, движок генерирует случайные координаты и размер (в пределах $\pm 25\%$) объектов-частиц. Если опция выключена, производится экспорт и использование текущих координат и размеров объектов-частиц. По умолчанию включено.

Blend4Web > Initial random rotation

Опция устанавливает случайный характер вращения объектов относительно оси определяемой параметром **Rotation type**. Если опция включена, движок генерирует случайные углы вращения объектов-частиц. Если опция выключена, устанавливается нулевой угол вращения. По умолчанию включено.

Blend4Web > Rotation type

Ось случайного поворота объекта (опция доступна при включении *Blend4Web > Initial random rotation*). Возможны 2 варианта:

- **Z axis** - случайный поворот будет осуществлен относительно вертикальной оси Z
- **Random axis** - случайный поворот будет осуществлен относительно случайной оси

Значение по умолчанию **Z axis**.

Blend4Web > Rotation strength

Коэффициент, определяющий диапазон случайных углов поворота, отсчитываемых от направления на камеру (опция доступна при включении *Blend4Web > Initial random rotation*). Например:

- **Rotation strength = 1** - углы будут лежать в пределах $[-\pi, \pi]$
- **Rotation strength = 0.5** - углы будут лежать в пределах $[-0.5 \cdot \pi, 0.5 \cdot \pi]$
- **Rotation strength = 0.1** - углы будут лежать в пределах $[-0.1 \cdot \pi, 0.1 \cdot \pi]$

Значение по умолчанию 1.

Blend4Web > Billboard

Включение биллбординга для частиц. По умолчанию выключено.

Blend4Web > Billboard type

Тип биллбординга (опция доступна при включении Blend4Web > Billboard). Доступны 3 типа:

- **Basic** - простой односторонний биллбординг: частицы всегда будут повернуты лицевой стороной
- **Random** - случайный двусторонний биллбординг: частицы чаще всего будут повернуты лицевой, либо обратной стороной, реже - боком; присутствует небольшой случайный поворот; модель создана специально для инстансинга травы
- **Jittered** - односторонний биллбординг с колебанием частиц в плоскости, обращенной к наблюдателю; модель создана специально для инстансинга листвы деревьев

Значение по умолчанию **Basic**.

Blend4Web > Jitter amplitude

Коэффициент амплитуды колебаний частиц (опция доступна при выборе типа **Jittered** в Blend4Web > **Billboard type**). При увеличении параметра амплитуда растет. Значение по умолчанию 0.

Blend4Web > Jitter frequency

Частота колебаний частиц, Гц (опция доступна при выборе типа **Jittered** в Blend4Web > **Billboard type**). Значение по умолчанию 0.

Blend4Web > Billboard geometry

Тип вращения биллбордов (опция доступна при включении Blend4Web > Billboard). Доступны 2 типа:

- **Spherical** - сферический биллбординг, полная ориентация частиц по отношению к наблюдателю, вращение ничем не ограничено
- **Cylindrical** - цилиндрический биллбординг, вращение частиц только относительно оси Z

Значение по умолчанию **Spherical**.

Blend4Web > Dynamic Grass

Опция включает режим динамического рендеринга травяного покрова. По умолчанию отключено.

Blend4Web > Wind bending

Наследование частицами настроек Wind bending:

- **Parent** - наследование с эмиттера
- **Instance** - наследование с объекта самой частицы

Значение по умолчанию **Parent**.

Blend4Web > Shadows

Наследование частицами настроек теней:

- **Parent** - наследование с эмиттера
- **Instance** - наследование с объекта самой частицы

Значение по умолчанию **Parent**.

Blend4Web > Reflection

Наследование частицами настроек отражений:

- **Parent** - наследование с эмиттера
- **Instance** - наследование с объекта самой частицы

Значение по умолчанию **Parent**.

Blend4Web > Vertex color

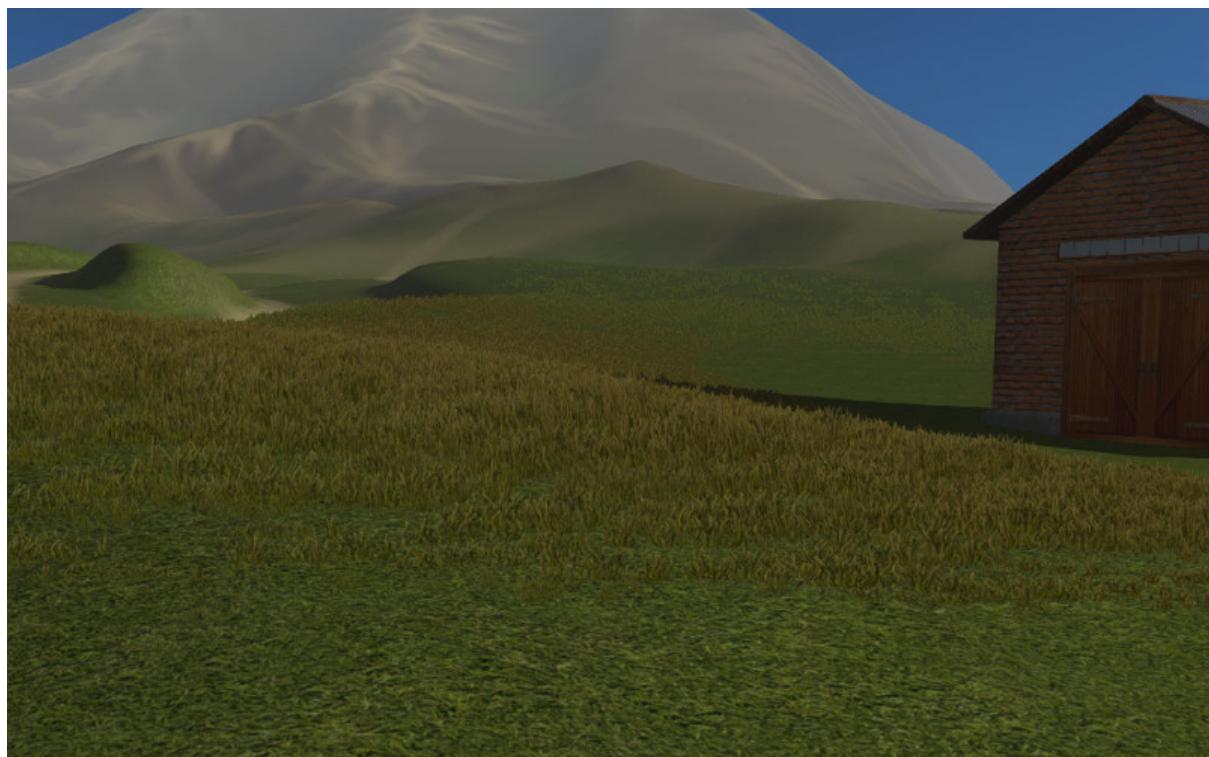
Наследование частицами вертексного цвета с эмиттера. Содержит 2 поля:

- **from** - имя существующего у эмиттера вертексного цвета
- **to** - имя существующего у частицы вертексного цвета

По умолчанию наследования не происходит.

17.2 Травяной покров

Инстансинг объектов может использоваться для визуализации травяного покрова на обширных площадях. При этом происходит отрисовка травы вблизи камеры по мере ее движения по ландшафту.



Активация

1. На отдельном объекте-плоскости создать систему частиц для инстансинга объектов. Включить опцию Blend4Web > Dynamic Grass.
2. На предполагаемом материале ландшафта включить опцию Blend4Web > Terrain dynamic grass.

Настройка

Рекомендуется создать несколько плоскостей (например, 3) с размерами, соответствующими желаемому размеру каскада травяного покрова (например, 100, 150 и 250 м).

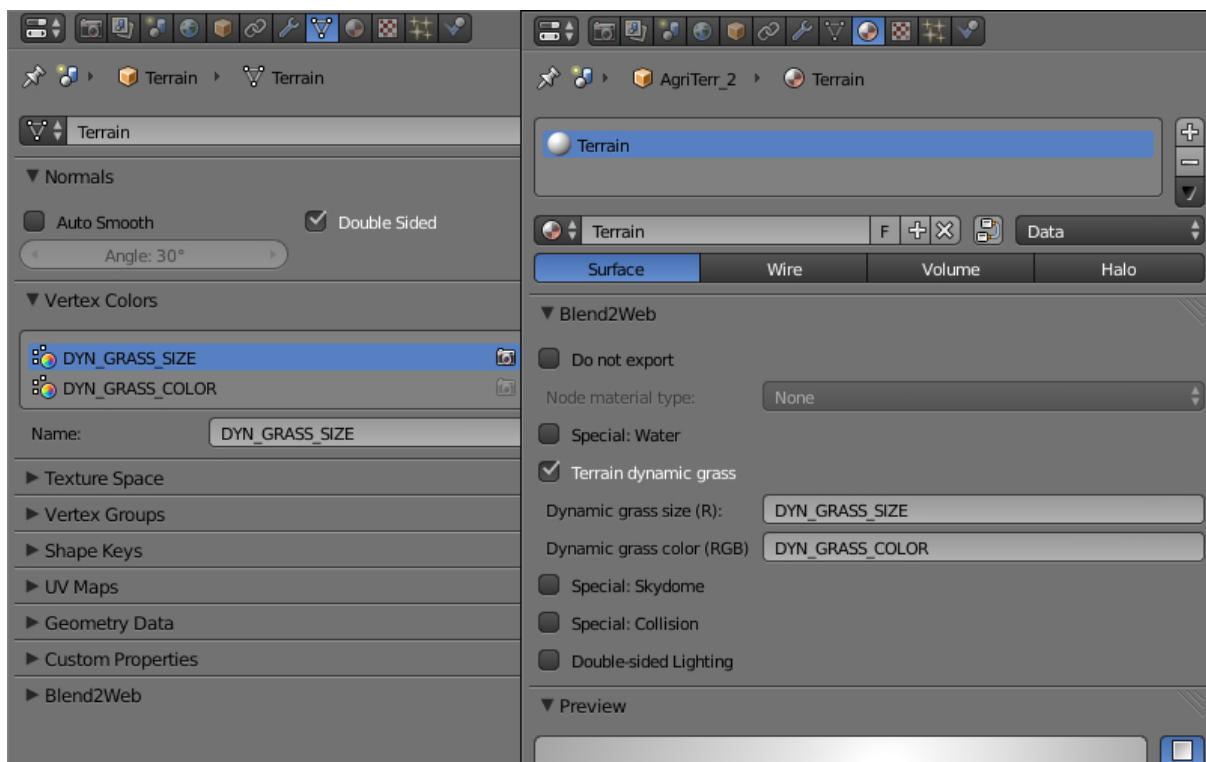
На **материале** ландшафта при включении опции Blend4Web > Terrain dynamic grass становятся активными текстовые поля:

Dynamic grass size (R) Название слоя вертекского цвета меша ландшафта, предназначенног для модифицирования размера травяного покрова. Размер (“высота”) травяного покрова задается оттенками серого - чем светлее, тем больше.

Dynamic grass color (RGB) Название слоя вертекского цвета меша ландшафта, предназначенного для подкраски травяного покрова. Вертекский цвет умножается на цвет материала травы. Параметр Influence > Blend диффузной текстуры материала травы должен иметь значение Multiply.

Слои вертекского цвета с такими названиями должны существовать в меше ландшафта.

Рекомендуется также отключить отображение эмиттера (опция Render > Emitter).



17.3 Листва деревьев

Инстансинг хорошо подходит для отображения листвы на деревьях, и позволяет добиться более высокого уровня детализации.



Активация

Осуществляется как описано выше в разделе **Настройки системы частиц -> Активация**. Здесь соответственно эмиттером будет выступать дерево, а частицами - ветки, листья и т.д.

Для эмиттера дополнительно можно сделать следующее:

- создать вертекскую группу, включающую вершины, на которых будут располагаться частицы
- создать слой вертекского цвета для настройки Wind Bending дерева и листвы
- создать слой вертекского цвета для наследования его частицами (можно использовать, например, для подкраски частиц)

Настройка

1. Настройки случного поворота

Если включена опция Blend4Web > Initial random rotation, то рекомендуется выставить вертикальную ось случного поворота - Z axis (опция Blend4Web > Rotation type). Опция Blend4Web > Rotation strength - на свое усмотрение.

2. Настройки биллбординга

Рекомендуется включить биллбординг, выставить тип Jittered (опция Blend4Web > Billboard type) и сделать его сферическим - Spherical (опция Blend4Web > Billboard geometry). Настройки Blend4Web > Jitter amplitude и Blend4Web > Jitter frequency - на свое усмотрение.

3. Настройки расположения частиц

Рекомендуется выставить опцию Emission > Emit From в значение Verts, а в Vertex Group > Density выбрать вертекскую группу эмиттера с вершинами для расположения частиц. Также нужно отключить опцию Blend4Web > Random location and size.

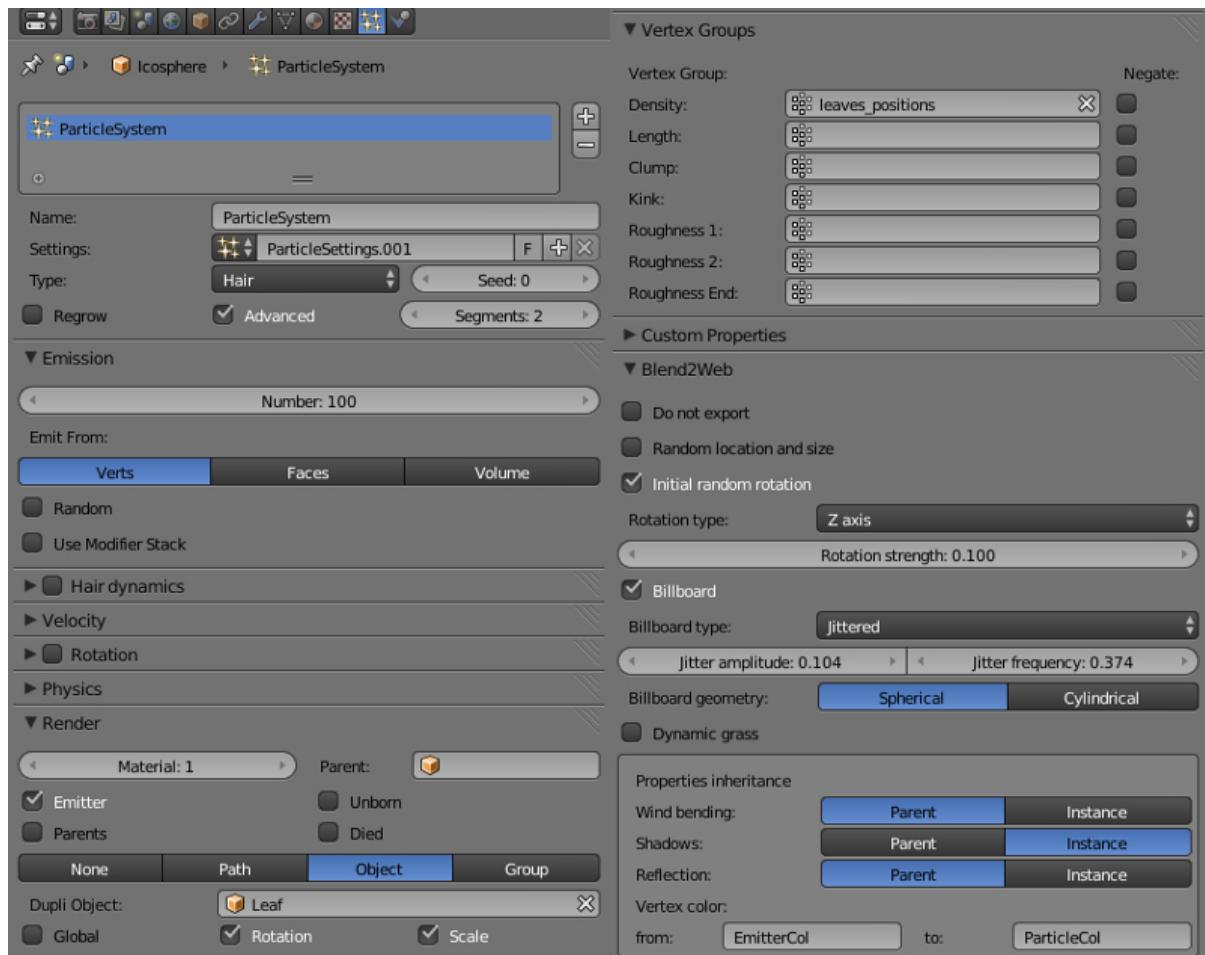
4. Настройки Wind Bending

Рекомендуется включить наследование настроек из эмиттера - выставить Parent в опции Blend4Web > Wind bending. Затем у эмиттера в панели Object выбрать опцию Blend4Web > Wind bending и настроить параметры бендинга. Для дерева достаточно указать параметры Blend4Web > Main Bending > Angle и Blend4Web > Main Bending > Frequency, а также вертекский цвет для бендинга - Blend4Web > Main Bending > Main stiffness.

5. Настройки наследования вертекского цвета

Для наследования частицами вертекского цвета эмиттера нужно указать имя цвета эмиттера и имя цвета частицы соответственно в полях Blend4Web > Vertex Color > from и Blend4Web > Vertex Color > to. При наследовании цвет ближайшей к частице вершины эмиттера из from будет скопирован и размножен в цвет to частицы.

Полученный таким образом вертексный цвет с именем Blend4Web > Vertex Color > to можно будет использовать в нодовом материале частицы для ее подкрашивания либо каких-то других эффектов.



Анимация

В общем случае, к анимации относятся изменения параметров объектов во времени. Движком поддерживаются следующие типы анимации:

- Объектная анимация. Заключается в перемещении объекта в пространстве как единого целого.
- Скелетная анимация, то есть деформация геометрии объекта с помощью системы костей (скиннинг). Сюда же относится анимация костей в арматурном объекте с целью прикрепления объектов к костям.
- Вертексная анимация. Заключается в покадровой записи деформаций объекта с их последующим воспроизведением.
- Параметризация источников звука. Изменяемые параметры: громкость (`Volume`) и высота звука (`Pitch`).
- Анимация выходного значения ноды `Value` в нодовом материале.
- Процедурная анимация в виде колебаний объекта под действием ветра. Описано *отдельно*.
- Эмиссия частиц из источника. Описано в *соответствующем разделе*.

18.1 Управление анимацией

Управление анимацией в движке осуществляется одним из двух способов:

1. Автоматически, с помощью указания свойств `Animation: Use default` и `Animation: Behavior` в свойствах объекта. В данном случае будет осуществлён поиск доступного метода и в случае положительного результата, объект анимируется с момента загрузки сцены. В случае скелетной анимации, по умолчанию воспроизводится актор, назначенный на объекте в окне `Action Editor`.
2. Программно, используя функции модуля движка `animation`.

Для отладки анимации имеет смысл использовать интерфейс `Animation` программы-просмотрщика, рассмотренный в *соответствующем разделе*.

18.2 Объектная анимация

Изменяемые параметры: координаты центра (**Location**), поворот (**Rotation**) и масштабирование (**Scale**).



Осуществляется с помощью добавления ключей анимации для движения объекта в программе Blender и их последующего воспроизведения в движке.

Поддерживаются следующие типы ключей:

- *Location*
- *Rotation* – необходимо осуществлять в режиме Quaternion(WXYZ) либо XYZ Euler.
- *Scale* – для получения корректных результатов, фактор масштабирования должен быть одинаковым вдоль любых из осей.
- *LocRot* – комбинация *Location* и *Rotation*.
- *LocScale* – комбинация *Location* и *Scale*.
- *LocRotScale* – комбинация *Location*, *Rotation* и *Scale*.
- *RotScale* – комбинация *Rotation* и *Scale*.

В случае анимации объекта-меша, необходимо назначение свойства **Force Dynamic Object** на вкладке свойств объекта.

18.3 Скиннинг и скелетная анимация



Для осуществления скелетной анимации, кроме деформируемого объекта-меша требуется объект-арматура. Осуществляется в четыре этапа:

1. Создание скелета объекта в арматурном объекте.
2. Назначение вертексных групп в объекте-меше и их привязка к костям. Может быть осуществлено, например, методом “раскраски” весов (weight painting).

3. Анимация костей в арматурном объекте. Используются те же ключи, что и в случае объектной анимации.
4. В случае нетривиальных видов скелетной анимации, включающих инверсную кинематику, требуется стадия запекания анимационных акторов (блок Action в Blender). Запекание производится с помощью интерфейса B4W Animation Bake, расположенного на панели инструментов Blend4Web:



Запекание производится при выделенном арматурном объекте. Элементы интерфейса B4W Animation Bake:

- *Clean keyframes* – произвести оптимизацию ключей анимации после запекания. В случае получения некорректных результатов, рекомендуется отключить опцию.
- окно со списком запекаемых акторов – запекать только те акторы, которые указаны в списке, иначе запекать все возможные акторы.
- *Name* – имя текущего актора из списка запекаемых акторов.
- *Bake* – произвести запекание. В случае успешного окончания процесса, на сцене появляются акторы с именами вида *ИМЯ_B4W_BAKED*. Данные акторы будут автоматически назначены на арматурном объекте и воспроизведены в движке. Стоит отметить, что работа подобных акторов в Blender не гарантируется, хотя в ряде случаев может помочь интерфейс *Cons Mute/Cons Unmute*.
- *Cons Mute/Cons Unmute* – отключить/активировать ограничители, установленные на костях. Инструмент может быть использован для тестирования за-

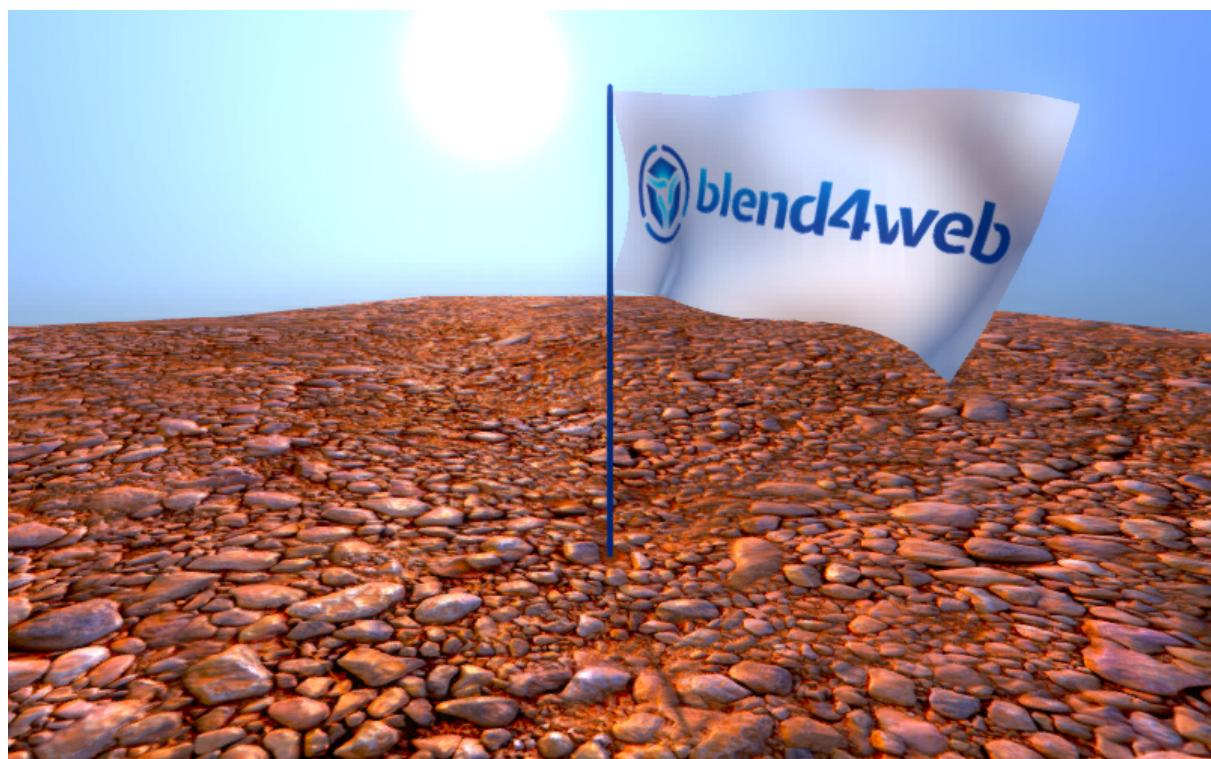
печёных акторов.

Примечание: Движок поддерживает не более 4-х вертексных групп на каждом из вертексов, эти группы отбираются по величине влияния или “веса” вертекса. В процессе загрузки исходного файла со сценой “веса” вертексов проходят через процедуру нормализации, т.е. их сумма приводится в единице.

Для удаления вертексных групп, которые не используются арматурой, можно воспользоваться кнопкой **Remove Unused VGroups**, которая расположена в интерфейсе **B4W Vertex Group**.



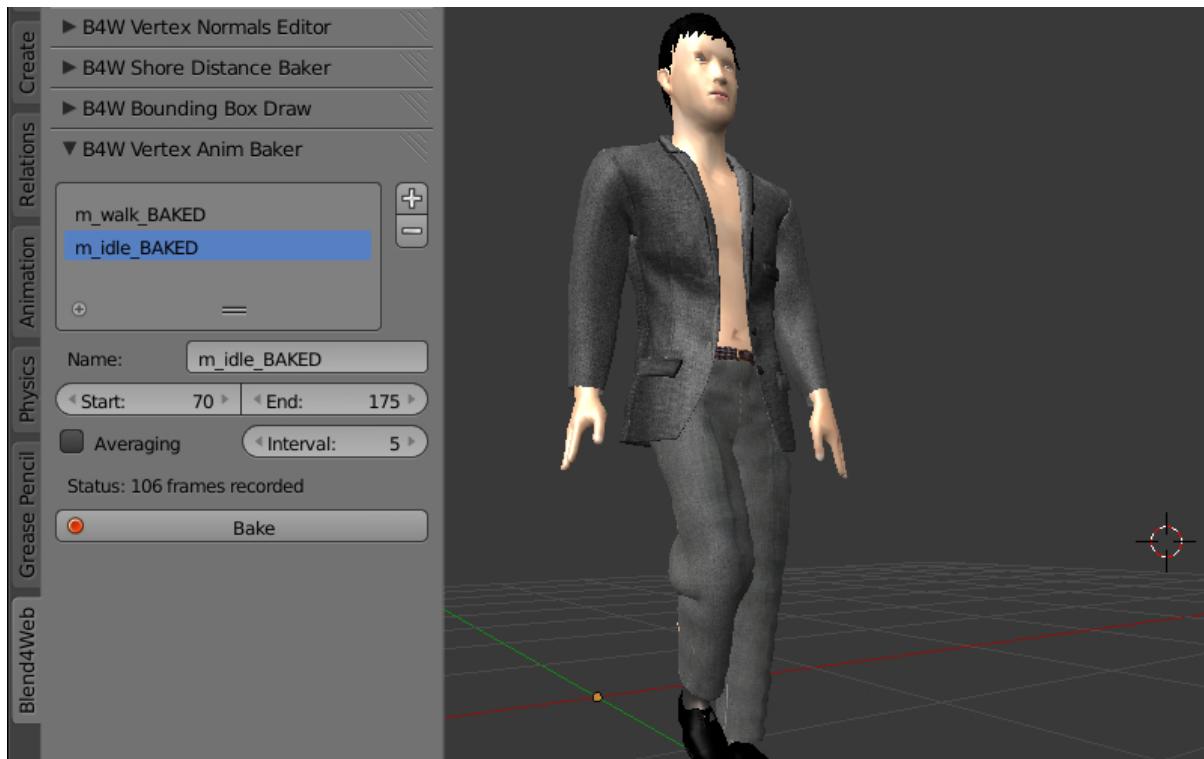
18.4 Вертексная анимация



Позволяет записать любые изменения геометрии объекта-меша. Необходимо учитывать, что каждый кадр вертексной анимации эквивалентен мешу. Не рекомендуется создание длинной анимации для высокополигонального меша, поскольку это может

привести к существенному возрастанию размера исходного и экспортируемого файлов, а также замедлить работу движка.

Для запекания вертекской анимации предусмотрен инструмент **B4W Vertex Anim Baker**, расположенный на панели инструментов Blend4Web.



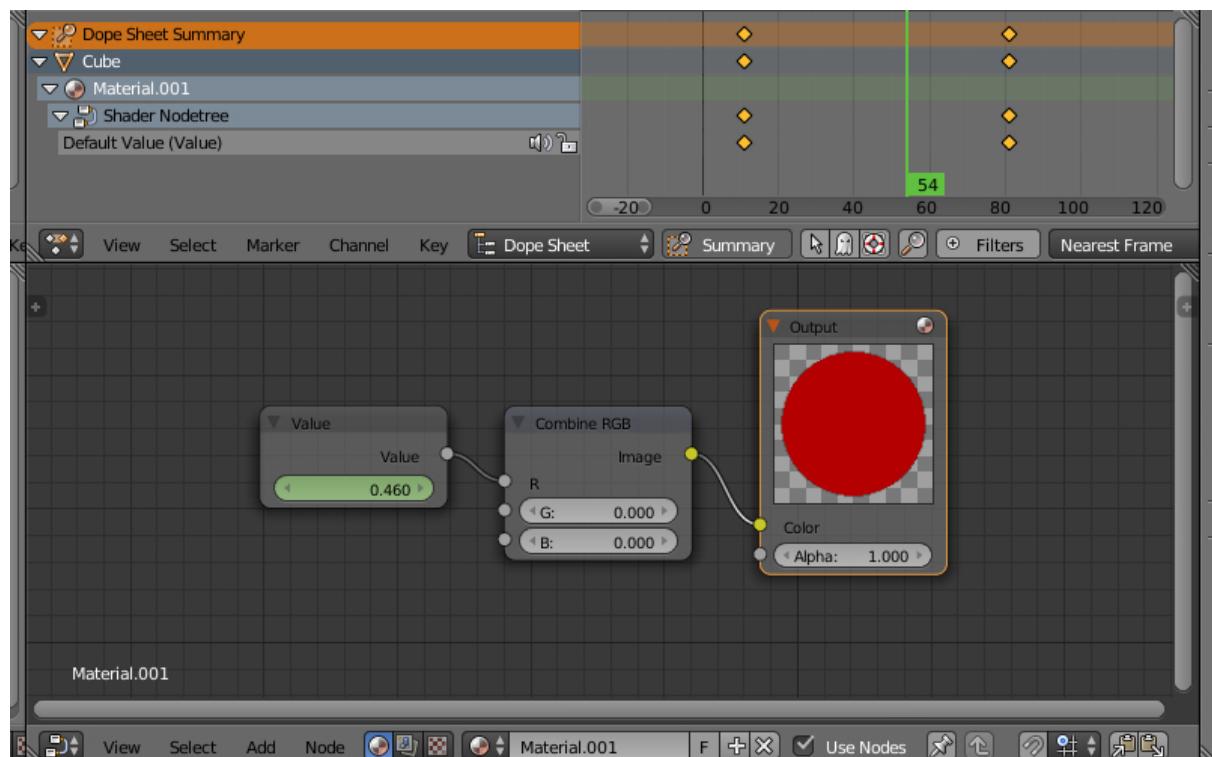
18.5 Параметризация источников звука

На объектах-спикерах дополнительно поддерживаются следующие типы анимационных ключей:

- *Volume* – громкость звука источника.
- *Pitch* – высота звука источника.

Параметризация источников звука по своей сути повторяет объектную анимацию.

18.6 Анимация ноды Value



Работает аналогично [объектной анимации](#). Может быть также использована для создания треков в [редакторе нелинейной анимации](#). Поддерживается несколько анимированных нод **Value** в одном материале.

См.также:

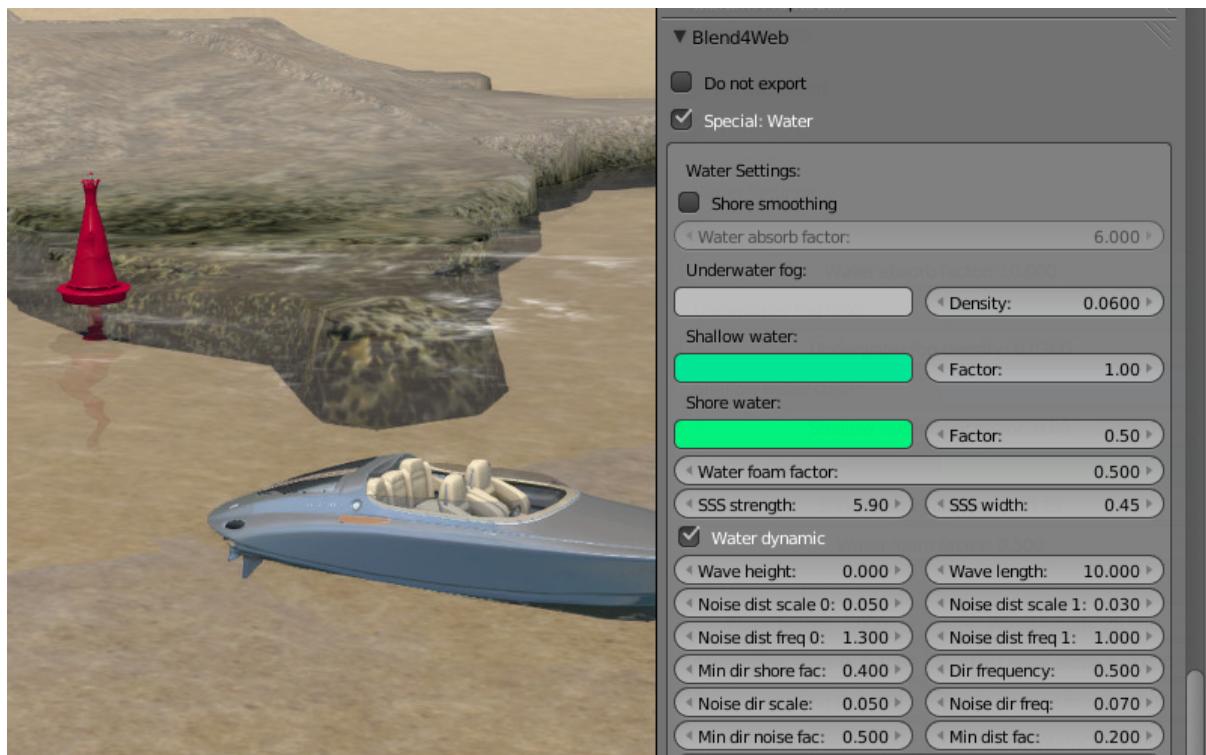
[Нодовые материалы](#)

Рендеринг наружных сцен

19.1 Вода

19.1.1 Активация

Для предполагаемого материала воды включить опцию Blend4Web > Special: Water во вкладке Material.



19.1.2 Базовые настройки

Прозрачность Рекомендуется включить прозрачность с градиентом Game Settings > Alpha Blend и настроить значение Transparency > Alpha.

Параметры освещения Параметры освещения материала воды настраиваются как описано в разделе [Параметры освещения](#).

19.1.3 Динамика волн

Симуляция волн осуществляется картами нормалей с анимированными развертками (в количестве от 0 до 4). Для текстур - карт нормалей используется только одно общее изображение, текстуры различаются параметрами **Mapping > Size**. Меш для воды должен иметь текстурную развертку.



19.1.4 Смачивание поверхностей

Осуществляется автоматически. Для включения эффекта на соответствующих материалах выставляется флаг **Wettable**.

19.1.5 Отражение и эффект Френеля

Для материала воды поддерживается как статическое, так и динамическое зеркальное отражение, с эффектом Френеля. См. раздел *Зеркальное отражение*.



19.1.6 Сглаживание береговой линии

Blend4Web > Water Settings > Shore Smoothing Включить сглаживание.

Blend4Web > Water Settings > Water Absorb Factor Коэффициент поглощения света водой. Чем он выше, тем прозрачнее вода.

19.1.7 Градиент цвета

Для создания цветного градиента на материале воды должна быть наложена текстура с включенной опцией *Blend4Web > Shore Distance Map*, генерируемая с помощью скрипта для *запекания параметров береговой линии*.

Blend4Web > Water Settings > Shallow Water Color Цвет воды на мелководье.

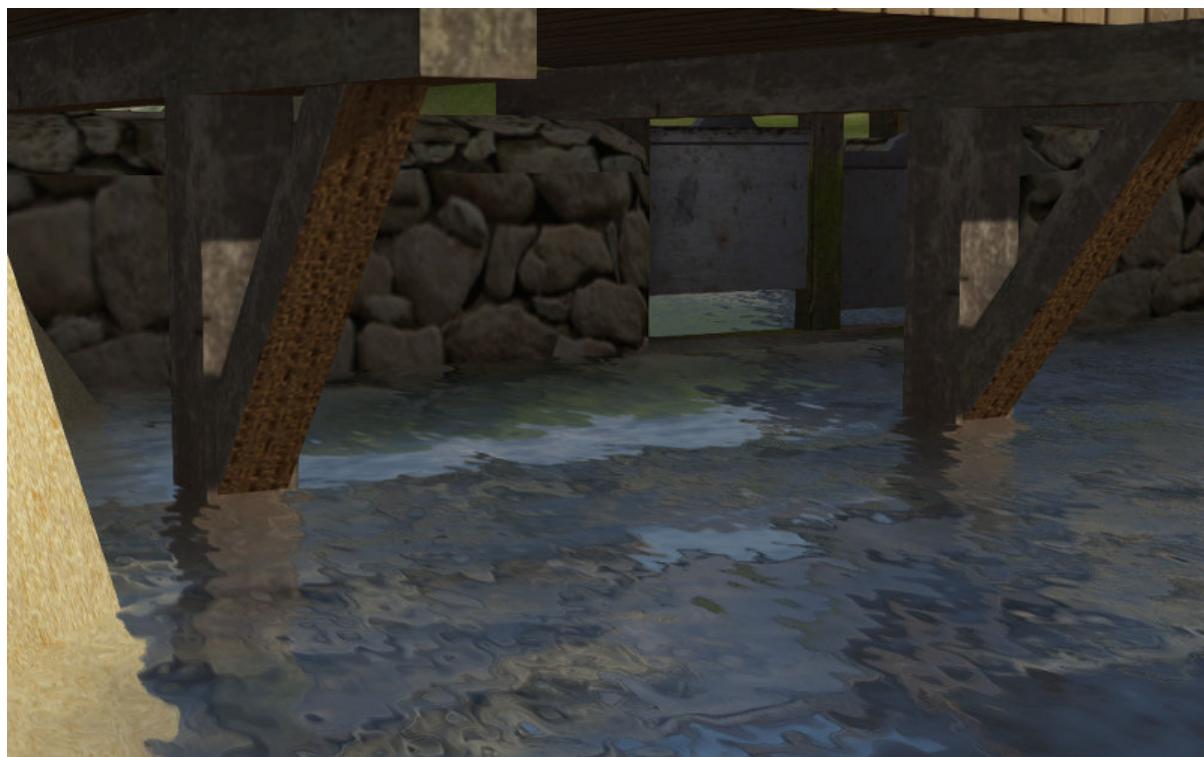
Blend4Web > Water Settings > Shallow Water Color Factor Коэффициент примешивания цвета воды на мелководье.

Blend4Web > Water Settings > Shore Water Color Цвет воды непосредственно у береговой линии.

Blend4Web > Water Settings > Shore Water Color Factor Коэффициент примешивания цвета воды на береговой линии.

19.1.8 Преломление

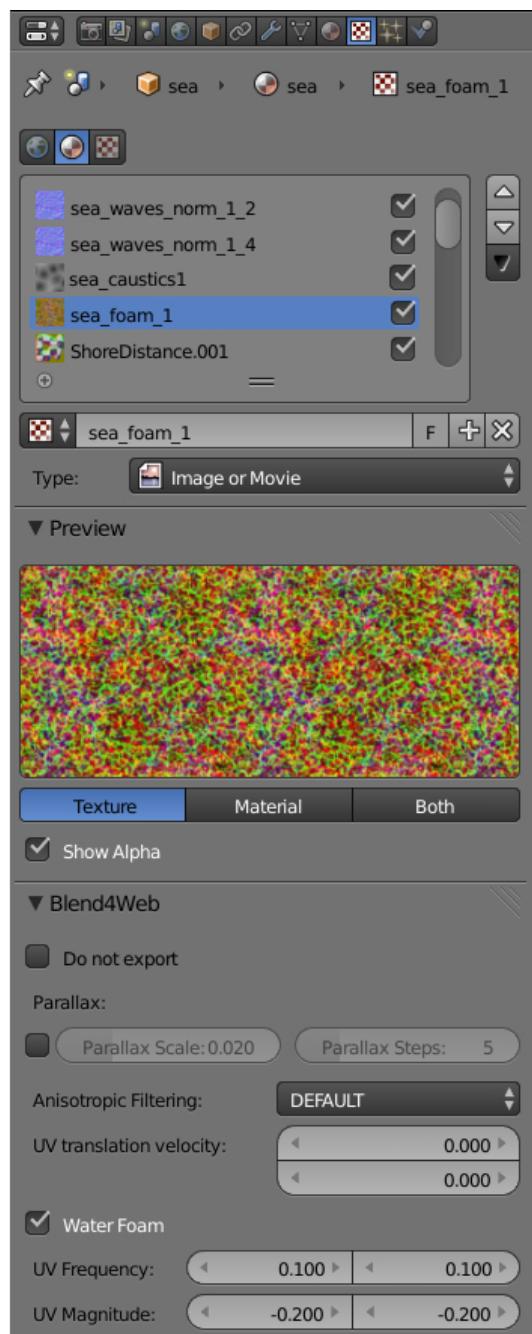
Во вкладке Scene включить опцию Blend4Web > Render Refractions.



19.1.9 Пена

Активация

Для создания пены необходимо добавить в текстурные слоты материала воды две диффузные текстуры. Для текстур необходимо выставить опцию Blend4Web > Water Foam.



Настройка текстур

Influence > Color Фактор влияния цвета текстуры. Значение по умолчанию 1.0.

Blend4Web > UV Frequency Частота колебаний анимированной развертки. Значение по умолчанию (1.0, 1.0).

Blend4Web > UV Magnitude Амплитуда колебаний анимированной развертки. Значение по умолчанию (1.0, 1.0).

Настройка материала

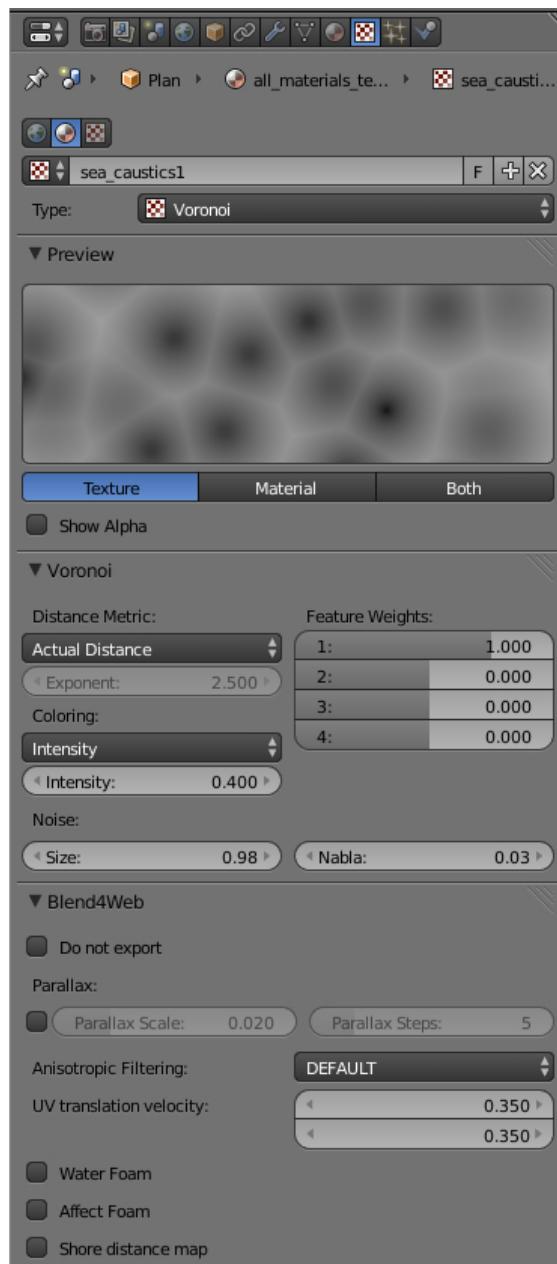
Blend4Web > Water Settings > Water foam factor Фактор общего влияния пены. Значение по умолчанию 0.5.

19.1.10 Каустика и хроматическая аберрация

Для создания каустики необходимо добавить в текстурные слоты материала воды одну текстуру типа Voronoi.



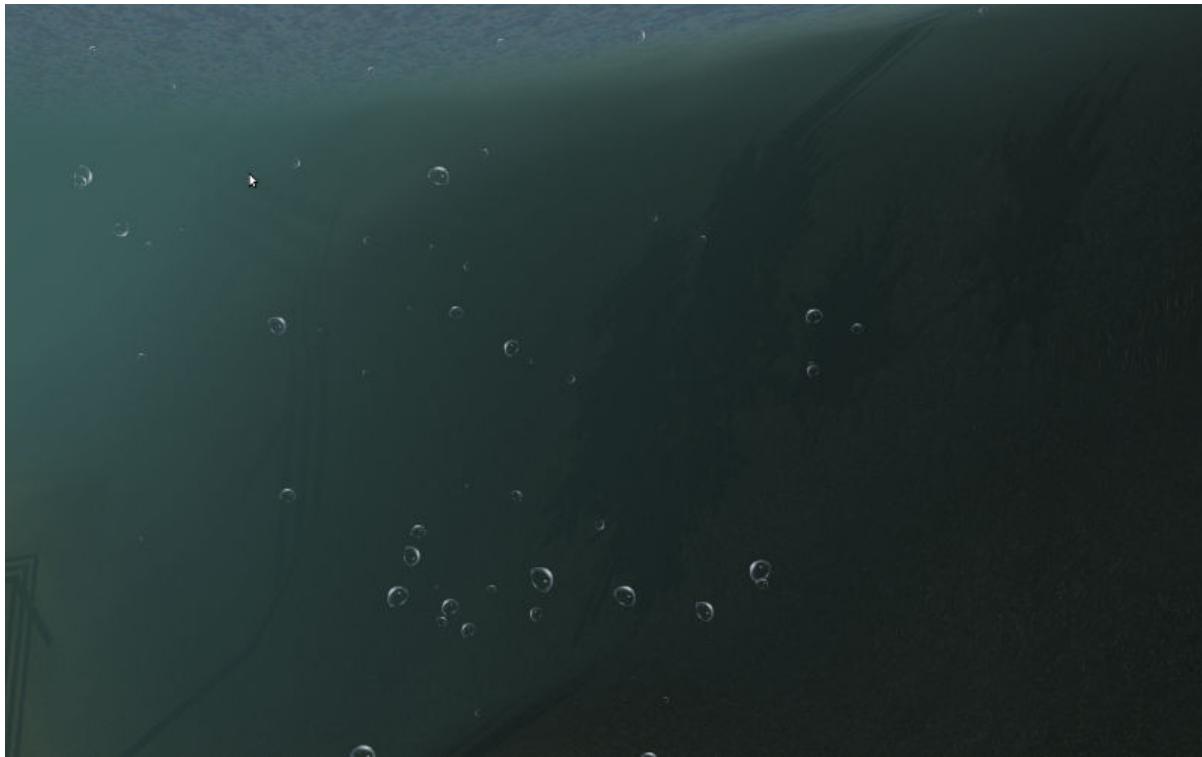
Настройка



Voronoi > Coloring: Intensity Фактор влияния каустики. Значение по умолчанию 1.0.

Voronoi > Noise: Size Размер ячеек процедурной текстуры. Значение по умолчанию 0.25.

19.1.11 Подводная среда



Настройки видимости (“туман”)

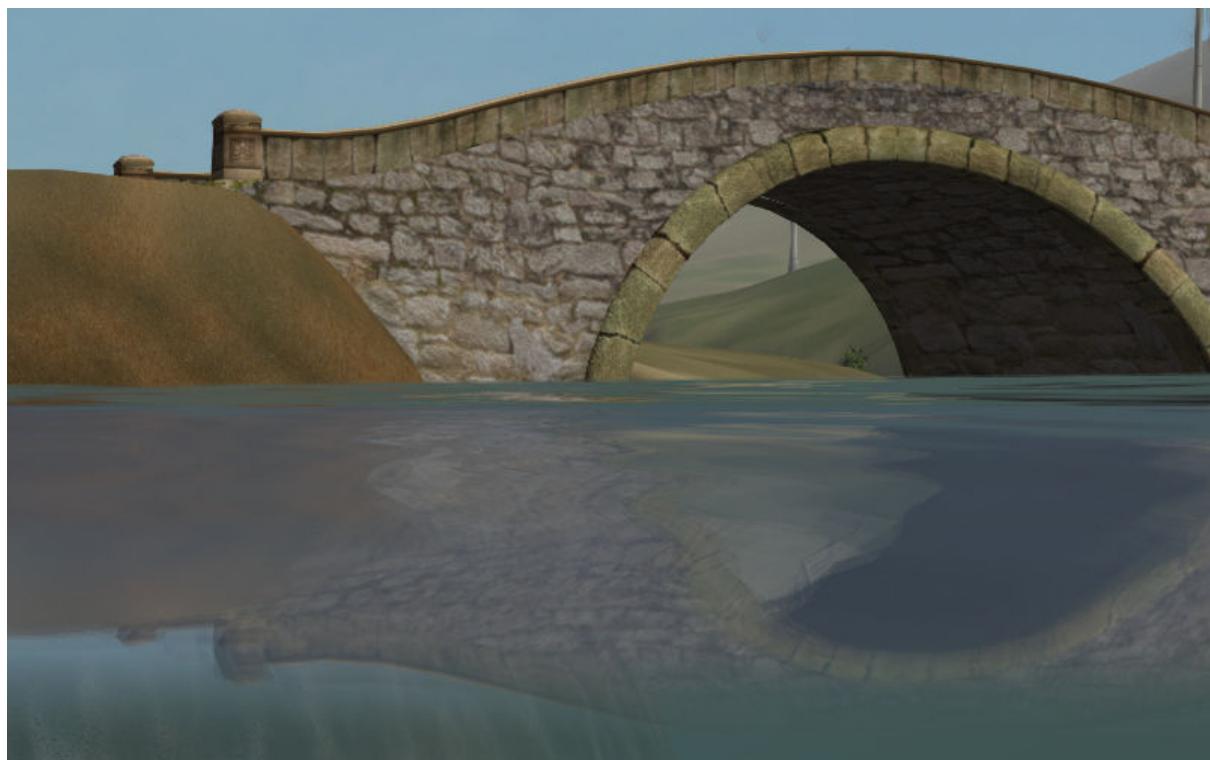
Blend4Web > Water Settings > Underwater Fog Density Экспоненциальный фактор, влияющий на плотность и расстояние. Значение по умолчанию 0.06.

Blend4Web > Water Settings > Underwater Fog Color Цвет тумана. Значение по умолчанию (0.5, 0.5, 0.5) (серый).

Применяются также настройки *сумеречных лучей*.

19.1.12 Граница сред

Выключить опцию Game Settings > Backface Culling.

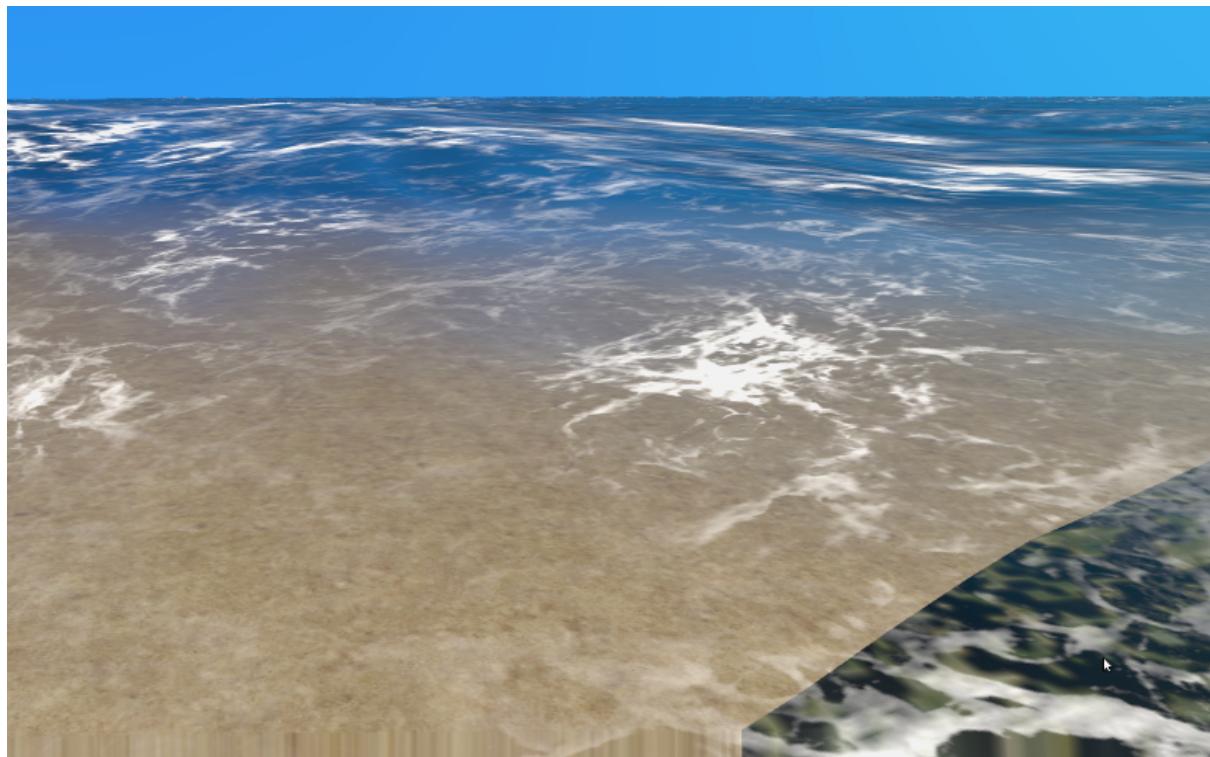


19.1.13 Объемные волны

Активация

Blend4Web > Water Settings > Water Dynamic

Включить объемные волны.



Настройка

Blend4Web > Water Settings > Wave Height Высота волн. Значение по умолчанию 0.0.

Blend4Web > Water Settings > Wave Length Длина волн. Значение по умолчанию 10.0.

Blend4Web > Water Settings > Dist Noise Scale 0 Размер первого компонента волн, удаленных от берега.

Blend4Web > Water Settings > Dist Noise Scale 1 Размер второго компонента волн, удаленных от берега.

Blend4Web > Water Settings > Dist Noise Freq 0 Частота первого компонента волн, удаленных от берега.

Blend4Web > Water Settings > Dist Noise Freq 1 Частота второго компонента волн, удаленных от берега.

Blend4Web > Water Settings > Dir Min Shore Fac Минимальный коэффициент уменьшения высоты прибрежных волн.

Blend4Web > Water Settings > Dir Frequency Частота накатывания прибрежных волн.

Blend4Web > Water Settings > Dir Noise Scale Размер шума на прибрежных волнах.

Blend4Web > Water Settings > Dir Noise Freq Частота шума на прибрежных волнах.

Blend4Web > Water Settings > Dir Min Noise Fac Минимальное значение шума на прибрежных волнах.

Blend4Web > Water Settings > Dist Min Fac Минимальный коэффициент примешивания волн, удаленных от берега.

Blend4Web > Water Settings > Waves Horizontal Factor Коэффициент смещения прибрежных волн в направлении к берегу.

19.1.14 Настройки генерируемой поверхности

Blend4Web > Water Settings > Generate Mesh Включить генерируемую поверхность.

Blend4Web > Water Settings > Number of Cascades Количество каскадов в генерируемой поверхности.

Blend4Web > Water Settings > Detailed Distance Максимальное расстояние от камеры до края последнего каскада.

Создание текстуры с параметрами береговой линии

На панели инструментов (горячая клавиша “Т”) во вкладке Blend4Web открыть панель B4W Shore Distance Baker. Выставить настройки максимального расстояния до берега **Maximum Distance** и размера получаемой текстуры **Texture Size**. Выбрать сначала объект (или несколько объектов) ландшафта, затем объект воды. Нажать кнопку **Bake Shore Distance**.

В зависимости от размера текстуры и количества вершин в обрабатываемых мешах время выполнения скрипта варьируется от долей секунды до нескольких минут. Убедиться, что в меше воды создана текстура с названием **ShoreDistance**.

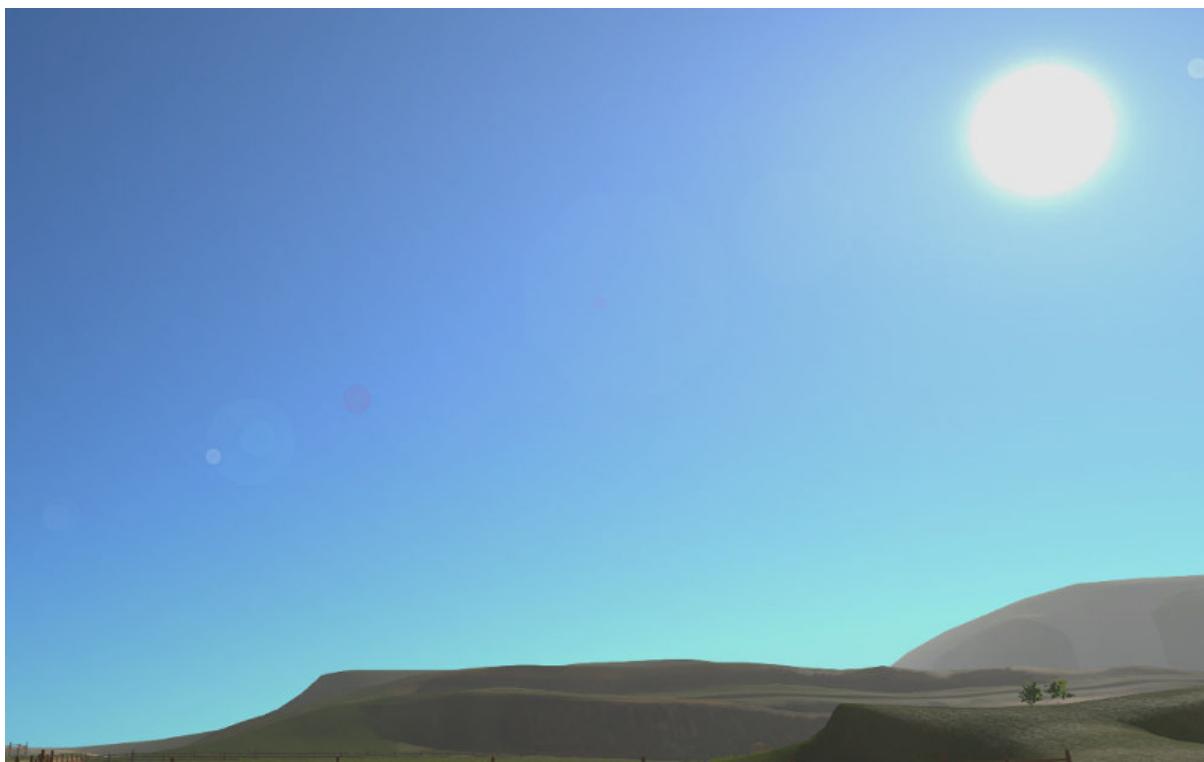
При вызове скрипта в материале воды сохраняются некоторые системные свойства. Поэтому, после его работы обязательно нужно сохранять сцену.

19.2 Атмосфера

19.2.1 Рассеивание

Во вкладке **World** выставить опцию **Sky Settings > Procedural Skydome**, предварительно выставив опцию **Sky Settings > Render Sky**. Если одновременно с этим используется статическая *текстура неба*, она будет заменена.

Примечание: Кроме того, процедурная текстура неба может быть использована для имитации рассеянного *освещения от окружающей среды*, по аналогии со статической *текстурой неба*. Для этого необходимо выставить опции **Sky Settings > Use as Environment Lighting** и **Environment Lighting > Sky Texture**. Если текстура мира для рассеянного освещения уже существует, она будет заменена.



Движком поддерживаются следующие настройки:

Sky Settings > Sky Color Базовый цвет неба. Значение по умолчанию (0.087, 0.255, 0.6) (голубой).

Sky Settings > Rayleigh Brightness Яркость рэлеевского рассеяния (на малых частицах). Значение по умолчанию 3.3.

Sky Settings > Mie Brightness Яркость рассеяния Ми (на крупных частицах). Значение по умолчанию 0.1.

Sky Settings > Spot Brightness Яркость пятна солнца. Значение по умолчанию 20.0.

Sky Settings > Scatter Strength Фактор рассеяния света. Значение по умолчанию 0.2.

Sky Settings > Rayleigh Strength Фактор рэлеевского рассеяния. Значение по умолчанию 0.2.

Sky Settings > Mie Strength Фактор рассеяния Ми. Значение по умолчанию 0.006.

Sky Settings > Rayleigh Collection Power Степенной коэффицент рэлеевского рассеяния. Значение по умолчанию 0.35.

Sky Settings > Mie Collection Power Степенной коэффицент рассеяния Ми. Значение по умолчанию 0.5.

Sky Settings > Mie Distribution Распределение рассеяния Ми. Значение по умолчанию 0.4.

19.2.2 Туман

Настраивается во вкладке **World**.

Blend4Web > Fog Settings > Fog Density Экспоненциальный фактор, влияющий на плотность и расстояние. Значение по умолчанию 0.0.

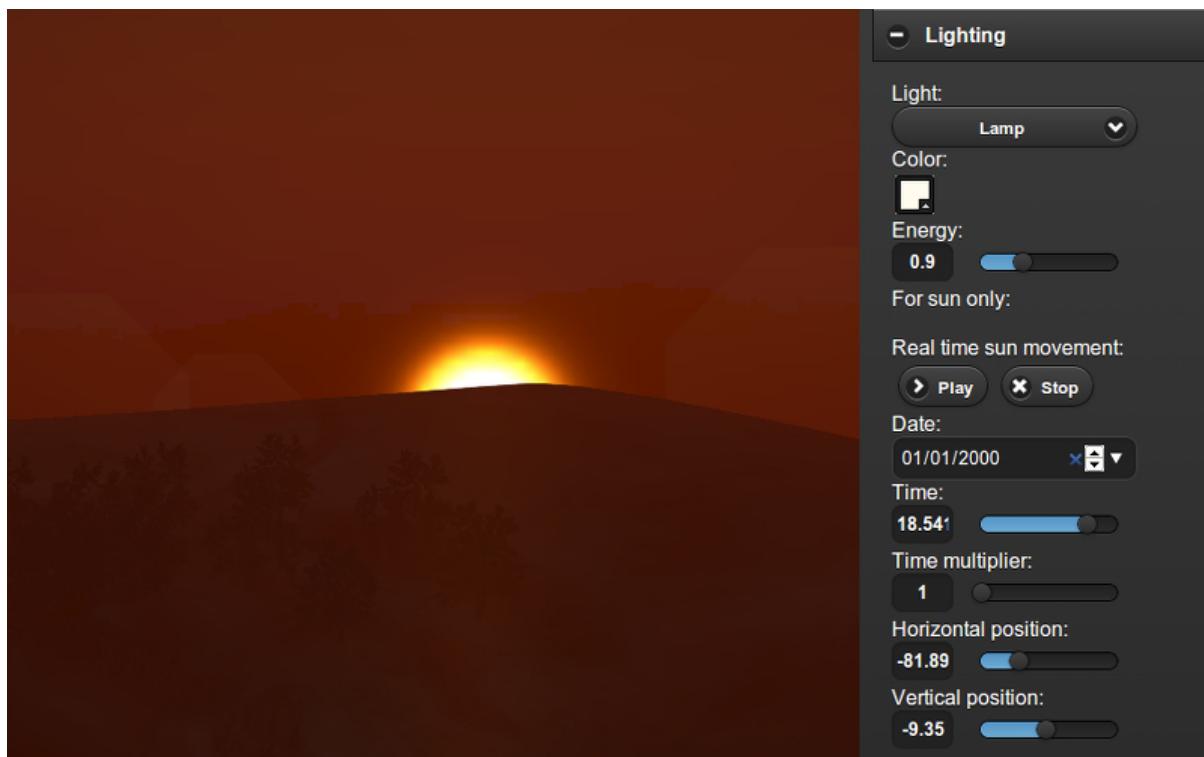
Blend4Web > Fog Settings > Fog Color Цвет тумана. Значение по умолчанию (0.5, 0.5, 0.5) (серый).

При использовании динамического неба цвет тумана определяется цветом неба.

19.2.3 Время суток

Для лампы необходимо выставить опцию **Blend4Web > Dynamic Intensity**.

Время суток устанавливается приложениями с использованием соответствующего API. В частности, время суток может устанавливаться в интерфейсе *Lighting просмотрщика сцен*.



19.2.4 Звезды

Настраиваются как описано в разделе *Материалы гало (Halo)*.



19.3 Ветер

Сила и направление ветра оказывает воздействие на

- анимацию травы и крон деревьев
- динамику систем частиц,
- частоту колебаний волн воды (в настоящий момент влияет только сила)

19.3.1 Активация

Добавить на сцену объект - силовое поле типа Wind.

19.3.2 Настройка

Направление Направление задается посредством вращения объекта - силового поля.

Force Fields > Strength Сила ветра. Располагается во вкладке Physics. Значение по умолчанию 1.0.

19.3.3 Анимация травы и крон деревьев

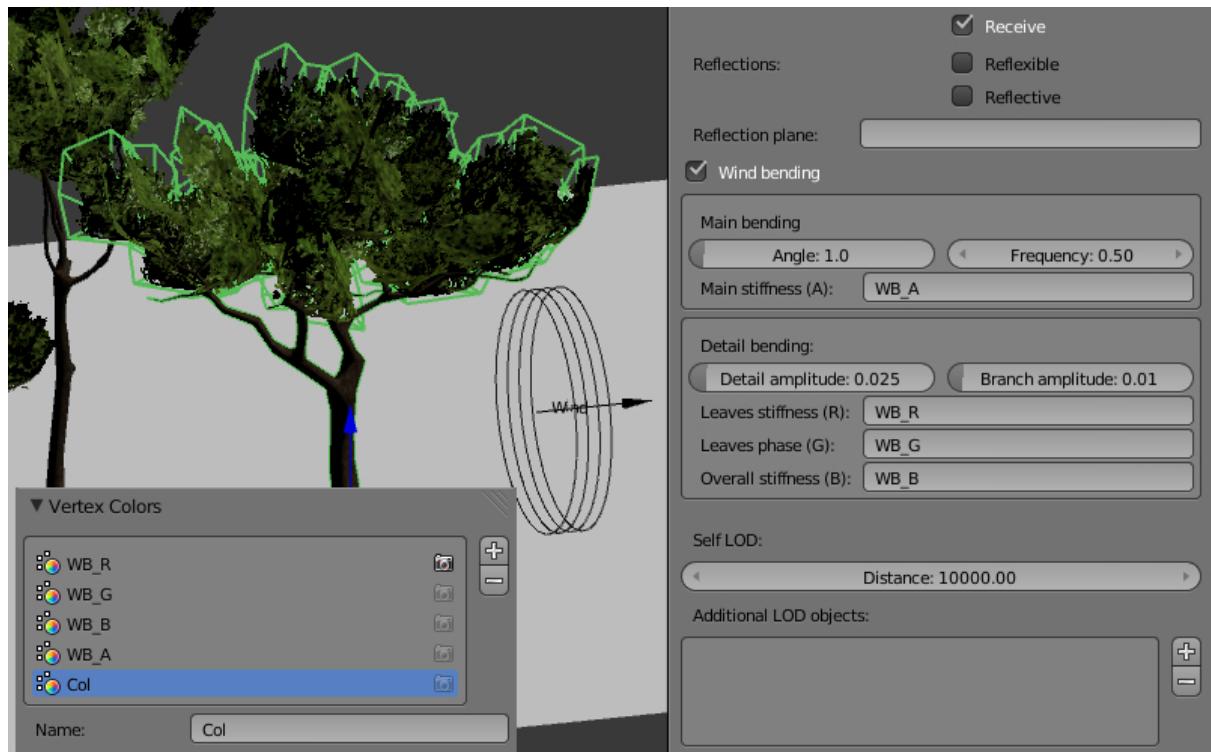
Подготовка ресурсов для рендеринга травы описана в разделе [Травяной покров](#).

Активация

На объекте травы или дерева включить опцию Blend4Web > Wind Bending.

Настройка

Интерфейс для настроек появляется после активации опции Blend4Web > Wind Bending.



Main bending > Angle Амплитуда угла “основного” отклонения под действием ветра (в градусах). Значение по умолчанию 10.0.

Main bending > Frequency Частота “основного” отклонения под действием ветра. Значение по умолчанию 0.25.

Main bending > Main Stiffness (A) Текстовое поле для названия слоя вертекстного цвета, содержащего информацию о жесткости “основного” отклонения. Может быть оставлено пустым.

Detail bending > Detail Amplitude Амплитуда угла “детализированного” отклонения под действием ветра (в градусах). Значение по умолчанию 0.1.

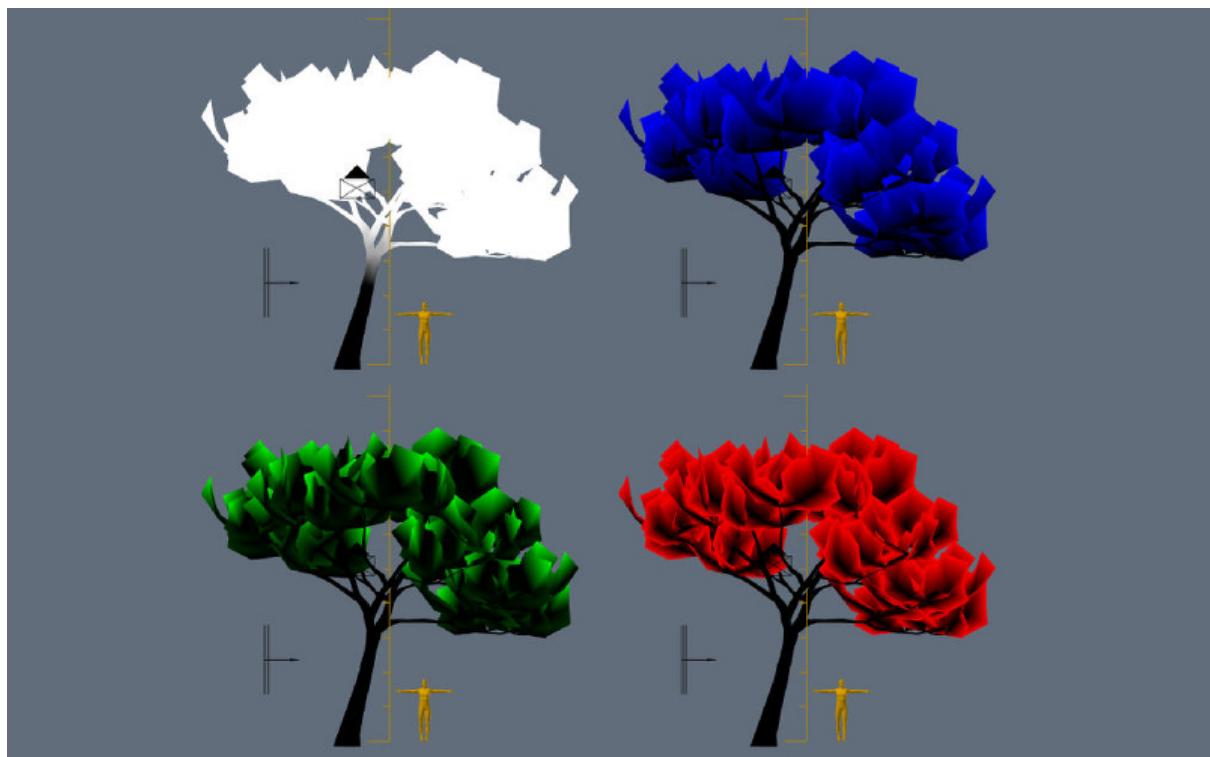
Detail bending > Branch Amplitude Амплитуда угла отклонения ветвей под действием ветра (в градусах). Значение по умолчанию 0.3.

Detail bending > Leaves Stiffness (R) Текстовое поле для названия слоя вертексного цвета, содержащего информацию о жесткости листвы. Может быть оставлено пустым.

Detail bending > Leaves Phase (G) Текстовое поле для названия слоя вертексного цвета, содержащего информацию о фазе отклонения листвы. Может быть оставлено пустым.

Detail bending > Overall Stiffness (B) Текстовое поле для названия слоя вертексного цвета, содержащего информацию об общей жесткости листвы. Может быть оставлено пустым.

Слои вертексных цветов с указанными в настройках названиями должны существовать в меше.

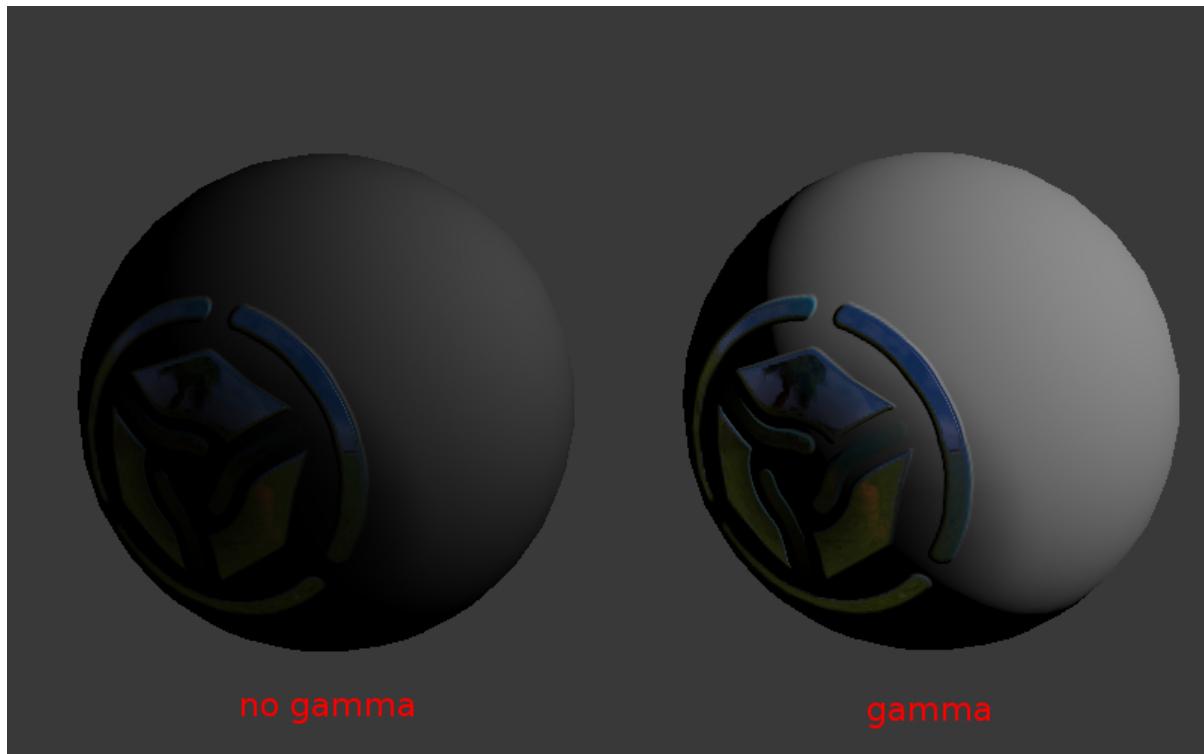


Гамма-коррекция и альфа-композитинг

20.1 Общее описание

Сущность гамма-коррекции заключается в упаковке яркости канала изображения в 8 битах информации.

Графические редакторы обычно работают в нелинейном цветовом пространстве, где тёмные компоненты кодируются большим числом битов чем светлые. Это означает, что значению 0.5 от реальной интенсивности света (физической величины, называемой освещённость) будет соответствовать большее значение, содержащееся в каналах RGB (в самом простом случае $0.5 ^ (1/2.2) = 0.73$).



Изображения всегда сохраняются в нелинейном пространстве, в противном случае 8 бит информации не достаточно для кодирования интенсивности света, что приведёт к тому, что тёмные тона будут отображаться некорректно.

Веб-браузеры работают в нелинейном пространстве.

Blender при настройке сцены **Color Management > Display Device > sRGB** работает в линейном пространстве. Значения цветов материалов и настройки источников света соответствует физическим величинам. При работе с текстурами, за исключением карт нормалей необходимо выставить настройку изображения **Image > Input Color Space > sRGB**. В этом случае при рендеринге будет производится автоматическая распаковка изображения: **sRGB->Linear**.

Движки и рендереры работают в линейном пространстве, поскольку оно может адекватно представлять поведение света в реальном мире. Например, освещённость от двух одинаковых ламп будет ровно в два раза превышать освещённость только от одной.

Примеры величин освещённости:

Описание	Освещённость, лк
Летом в полдень	17 000
Зимой в полдень	5 000
В пасмурный день	1 000
В светлой комнате	100
Ночью в полнолуние	0.2
В безлунную ночь	0.001

20.2 Человеческое зрение и мониторы

Человеческое восприятие света нелинейно (человек лучше различает градации тусклого света чем яркого), однако свет, поступающий в глаз, по-прежнему должен подчиняться физическим законам (см. пример с лампочками).

Мониторы с электронно-лучевыми трубками (ЭЛТ) имеют нелинейную характеристику яркости от приложенного к их входу электрического напряжения, которое, в свою очередь, определяется значением канала цветности в видеопамяти. Подобную же характеристику копируют жидкокристаллические мониторы. Тем не менее свет, излучаемый такими мониторами, по-прежнему должен подчиняться физическим законам. Например, в идеальном случае при добавлении второго источника света на сцену в виртуальном мире, яркость пикселей на экране монитора должна увеличиваться в два раза.

Таким образом, особенности восприятия человеческого глаза и технические характеристики мониторов имеют вторичное значение по отношению к гамма-коррекции.

20.3 Гамма

Используется в следующей упрощенной формуле:

$$V_{\text{out}} = V_{\text{in}}^{\gamma}$$

$\gamma < 1$ - упаковывающая гамма, $\gamma > 1$ - распаковывающая гамма. В наиболее простом случае используются значения 1/2.2 и 2.2 соответственно. Далее вместо термина “гамма-коррекция” будут использованы термины “упаковка” (Linear -> sRGB) и “распаковка” (sRGB -> Linear).

20.4 Коррекция в нодовых материалах

20.4.1 Ноды для окраски

При использовании текстур и вертексы цветов для окраски (не в качестве масок), необходима распаковка (sRGB -> Linear). Нода текстуры реализует распаковку автоматически. Для вертекского цвета распаковку необходимо осуществлять явно, с помощью специальной ноды *SRGB_TO_LINEAR*.

Отметим, что альфа-канал ноды текстуры коррекции не подвергается, его значения находятся в линейном пространстве.

20.4.2 Ноды для масок

Текстуры и вертексы цвета могут использоваться в качестве масок, т.е. для смешения цветов или других математических операций. В таком случае в преобразованиях нет необходимости.

В случае текстуры, тем не менее, имеется нюанс: нода текстуры реализует распаковку автоматически. Это приводит к необходимости дополнительного преобразования обратно в нелинейное пространство, для чего используется нода *LINEAR_TO_SRGB*.

20.4.3 Карты нормалей

При использовании карт нормалей никакие преобразования не производятся.

20.4.4 Сводная таблица коррекции в нодовых материалах

Случай использования	Коррекция
Текстура для окраски	реализуется автоматически в node текстуры (альфа-канал коррекции не подвергается)
Текстура для маски	LINEAR_TO_SRGB
Вертексный цвет для окраски	SRGB_TO_LINEAR
Вертексный цвет для маски	не требуется
Карта нормалей	не требуется

20.5 Альфа-композитинг

20.5.1 Общие сведения

Физически корректный альфа-композитинг осуществляется по формуле [источник]:

$$C_o = C_a \alpha_a + C_b \alpha_b (1 - \alpha_a).$$

Формула отличается от классической операции смешивания (mix, выпуклая комбинация) наличием множителя α_b во втором слагаемом. Таким образом, для осуществления альфа-композитинга должно быть известно не только значение α_a пикселя-источника, но и значение α_b пикселя, поверх которого осуществляется рендеринг.

В случае предварительного умножения значений α на цветовые каналы (т.н. premultiplied alpha) формула принимает вид:

$$C_o = C_a + C_b (1 - \alpha_a).$$

Последняя формула используется также для расчёта результирующего значения α_o :

$$\alpha_o = \alpha_a + \alpha_b (1 - \alpha_a).$$

Предварительное умножение цветовых каналов на значения α позволяет сэкономить две операции умножения. Еще более существенным является тот факт, что полученная формула может использоваться многократно, без необходимости деления цвета C_o на значение α_o на каждой последующей итерации.

20.5.2 Реализация

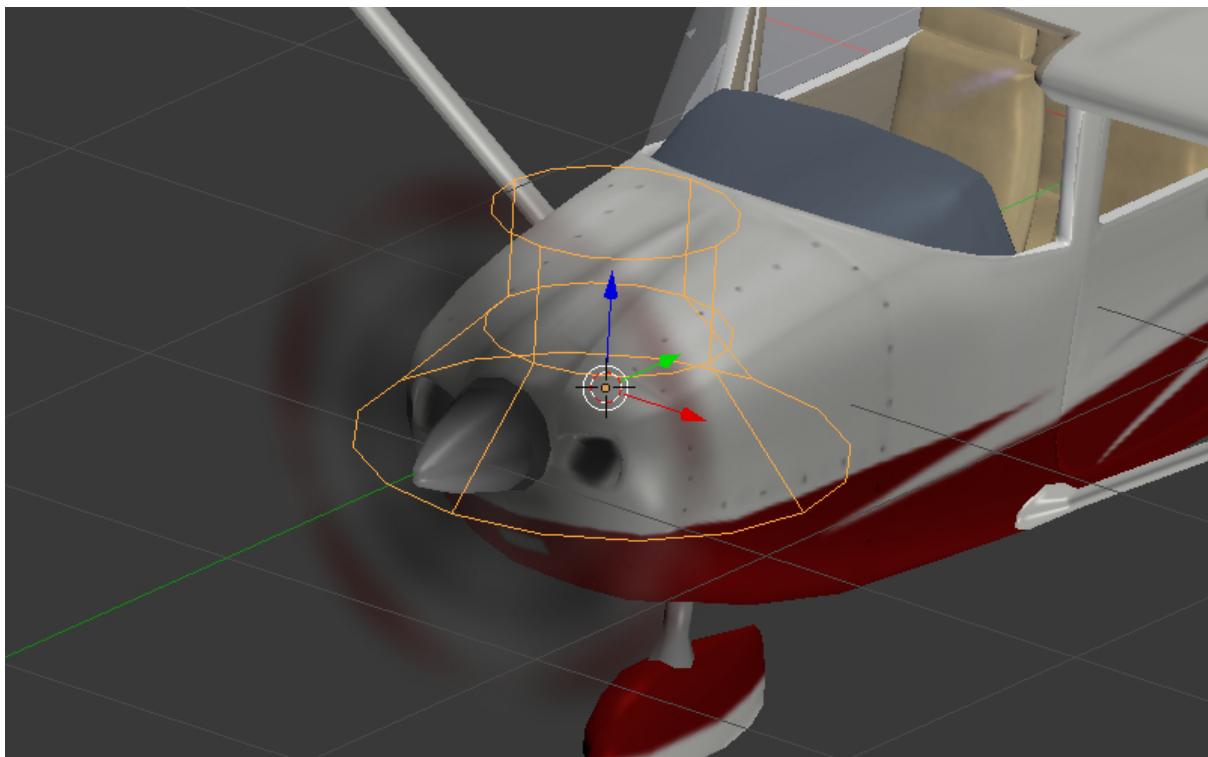
Функция смешивания в движке Blend4Web имеет вид:

```
gl.blendFunc(gl.ONE, gl.ONE_MINUS_SRC_ALPHA);
```

Инициализация контекста WebGL производится с параметром *premultipliedAlpha = true* (что является значением по умолчанию). Кроме того, на выходе шейдеров производится умножение всех каналов цветности на значение α .

Звуковая подсистема

Создание звуковых источников осуществляется в Blender'e. Используется стандартный объект **Speaker**.



21.1 Настройка звуковых источников

Настройки спикера выставляются в панели **Properties** на вкладке **Object Data**.

Движком поддерживаются все стандартные параметры. Специфичные для движка настройки располагаются на панели **Blend4Web**.

Speaker behavior Поведение звукового источника.

Positional — высококачественный звук, допускающий позиционирование и имеющий направленность (конусность). Для рендеринга используется Web Audio API. Воспроизведение подобных звуков обладает наименьшей производительностью, поэтому их использовать целесообразно только для коротких сэмплов.

Background sound — высококачественный всенаправленный звук без возможности позиционирования в пространстве. Для рендеринга используется Web Audio API. Более производителен, однако нецелесообразен для музыки.

Background music — используется для воспроизведения музыки. Максимальная производительность вследствие использования тега `Audio`, минимальная гибкость.

Disable doppler Игнорировать смещение частоты источника при его перемещении.

Cyclic play Зацикливать воспроизведение звука.

Delay Задержка в секундах перед началом проигрывания звука.

Random delay Дополнительная рандомизация задержки, результирующее значение определяется по формуле $Delay_{result} = Delay + Delay_{random} * Random_{[0-1]}$.

Random volume Дополнительная рандомизация громкости. Результирующее значение определяется аналогично задержке.

Random pitch Дополнительная рандомизация скорости проигрывания звука. Результирующее значение определяется аналогично задержке.

Fade-in Интервал плавного включения звука.

Fade-out Интервал плавного выключения звука.

Loop Зацикливать воспроизведение звука. Отличается от *Cyclic play*, тем, что способен обеспечить нулевую задержку при повторении. Опция доступна только для звуковых источников с поведением **Positional** или **Background sound**.

Loop count Не реализовано

Random loop count Не реализовано

Playlist ID Не реализовано



21.2 Обработка и кодирование

21.2.1 Поддерживаемые форматы (контейнеры):

- ogg, кодек Vorbis (Chrome, Firefox)
- mp3 (Chrome, Safari)
- mp4, кодек AAC (Chrome, Safari)

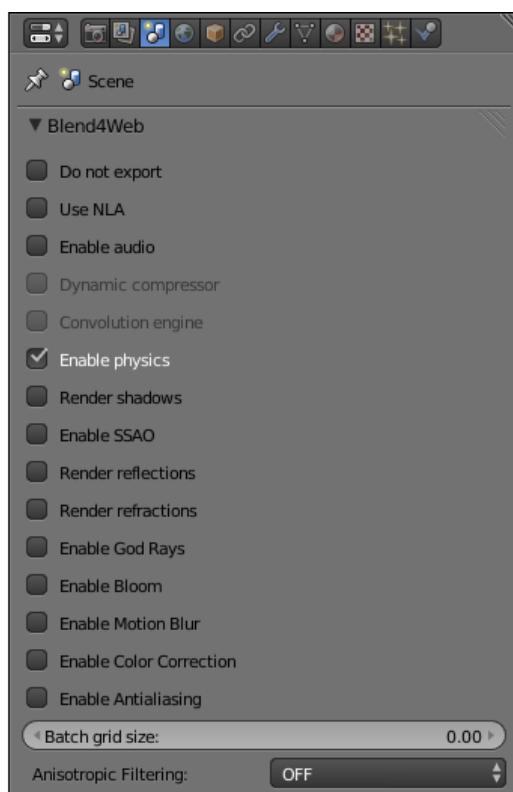
Рекомендуется использовать *Ogg*, который является открытым стандартом, поддерживается многими браузерами, обеспечивает хорошее качество звука. Оптимальным с точки зрения качества и совместимости является формат 48кГц/16бит. Одноканальный звук (моно) используется для хранения коротких сэмплов, двухканальный звук (стерео) - для музыкального сопровождения.

Конвертация ресурсов в различные форматы описывается в *соответствующем разделе*.

Физика

22.1 Подготовка к использованию

Для задействования физики на сцене необходимо установить флаг Enable Physics вкладке сцены в Blender'e.

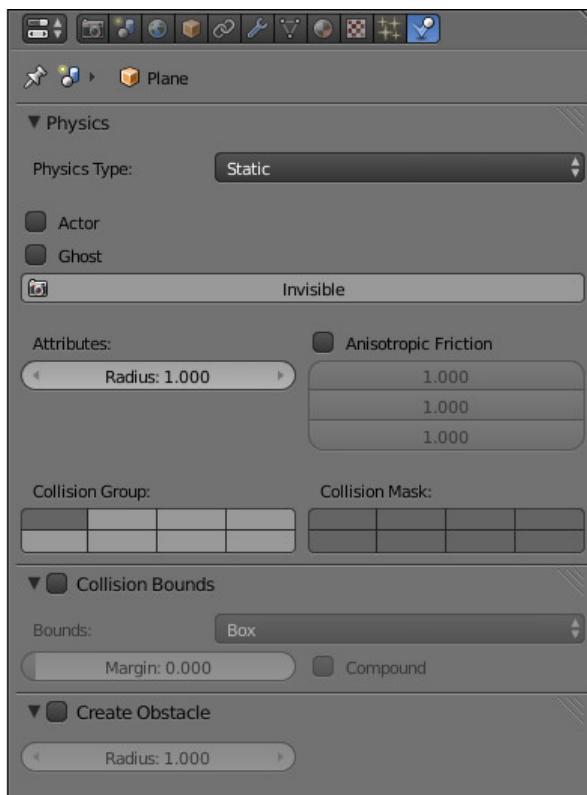


Настройка физических параметров производится в режиме Blender Game.

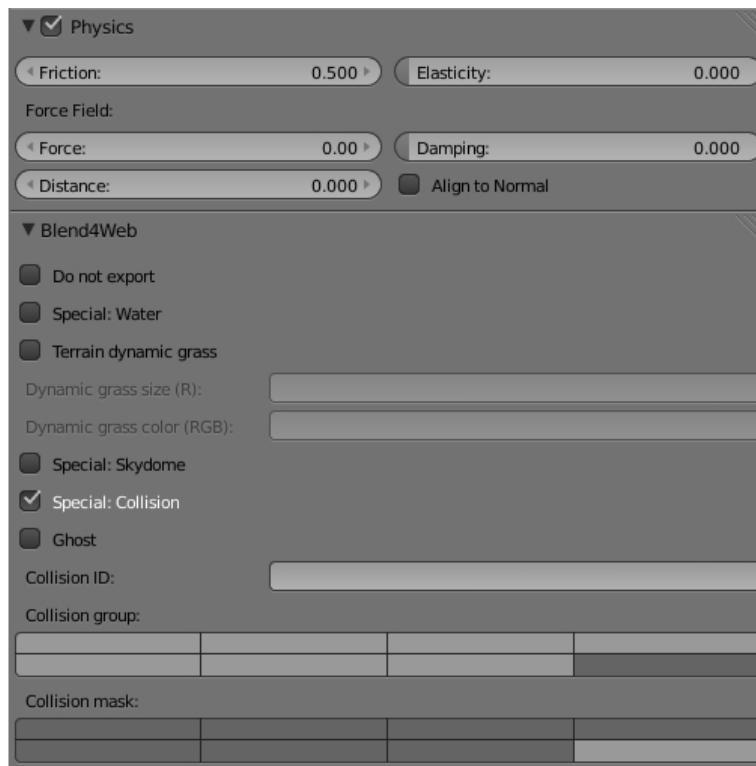


22.2 Статический тип физики

Может использоваться как ограничитель движения других объектов, например, для определения столкновений с ландшафтом, стенами и т.д. В настройках физики такого объекта для опции Physics Type должно быть выбрано значение Static (значение по умолчанию).

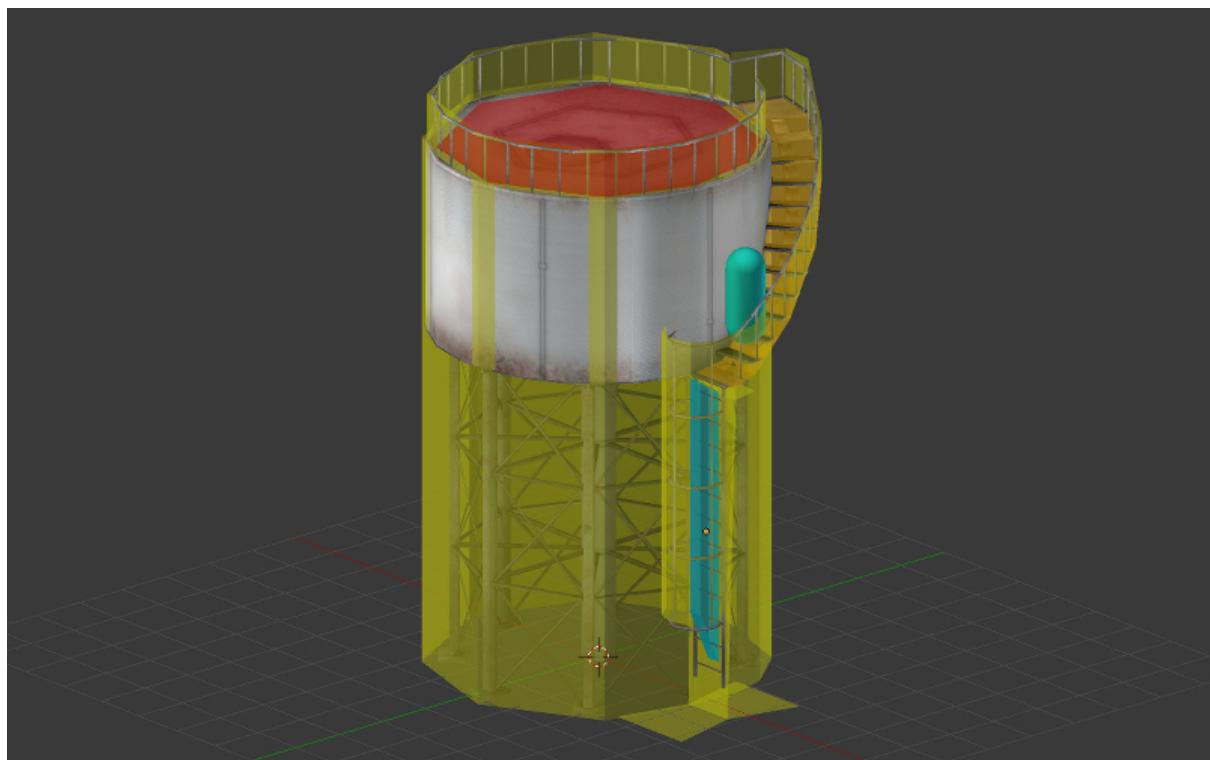


Меш может быть покрыт одним или несколькими физическими материалами. Во вкладке Material должна быть включена опция Blend4Web > Special: Collision. Также во вкладке Material на панели Physics (в режиме Blender Game) располагаются физические настройки материала. Поддерживаются следующие физические настройки материала: трение (Friction), упругость (Elasticity).



Поле **Collision ID** предназначено для определения столкновения со специфическим материалом, и может быть оставлено пустым. Пример использования **Collision ID** - определение нахождения игрового персонажа на разных типах покрытия ландшафта - трава, песок, деревянное покрытие и т.д.

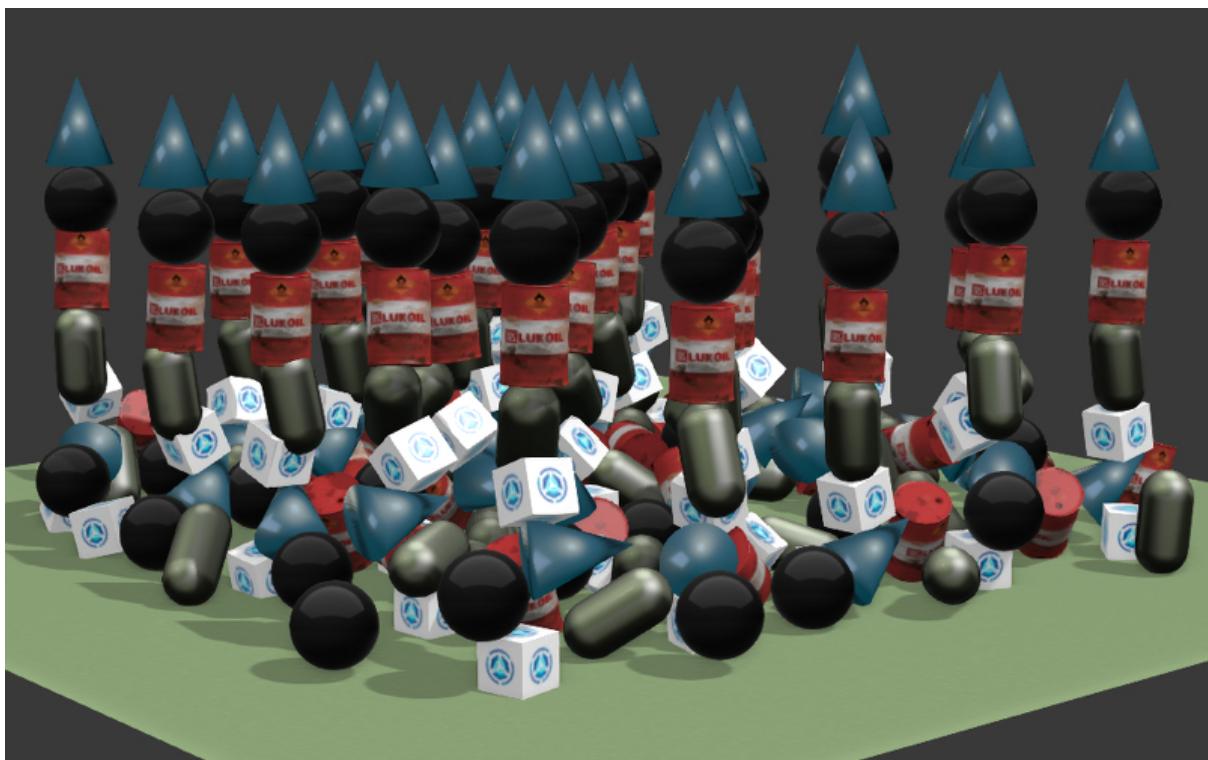
Опция **Ghost** исключает материал из физических взаимодействий, но сообщает приложению о контакте с ним. Пример - определение, что игровой персонаж находится на вертикальной лестнице.



Поле **Collision Group** отвечает за физическую группу, к которой относится материал. Поле **Collision Mask** определяет все физические группы, с которыми будет взаимодействовать данный материал.

22.3 Динамический тип физики

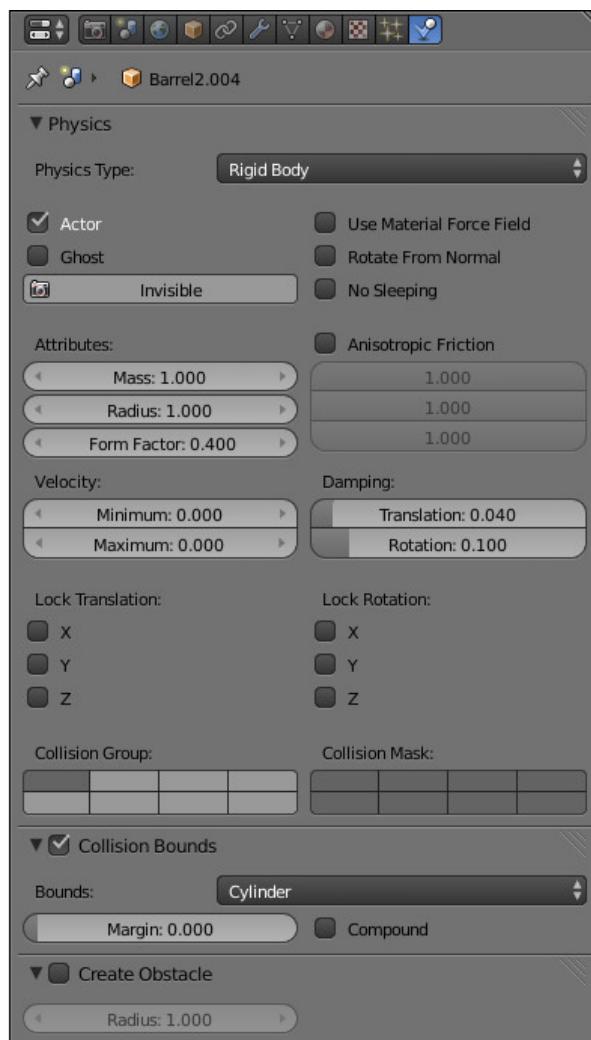
Предназначен для симуляции движения жесткого тела.



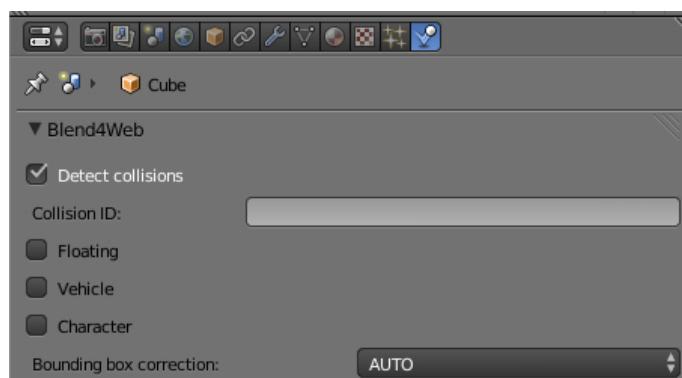
В настройках физики такого объекта для опции **Physics Type** может быть выбрано значение **Rigid Body** (с вращениями) или **Dynamic** (без вращений). В настройках **Collision Bounds** может быть выбран тип коллайдера, поддерживаются: **Box**, **Capsule**, **Sphere**, **Cylinder**, **Cone**. Другие поддерживаемые настройки: масса (**Mass**), демпфирование (**Damping**) - для перемещения (**Translation**) и вращения (**Rotation**).

Поле **Collision Group** отвечает за физическую группу, к которой относится объект.

Поле **Collision Mask** определяет все физические группы, с которыми будет взаимодействовать данный объект.



В настройках панели физики объекта должен быть установлен флаг **Detect Collisions**. Поле **Collision ID** предназначено для определения столкновения со специфическим объектом (например, прикрепленный к камере объект для определения близости FPS персонажа к предметам), и может быть оставлено пустым.

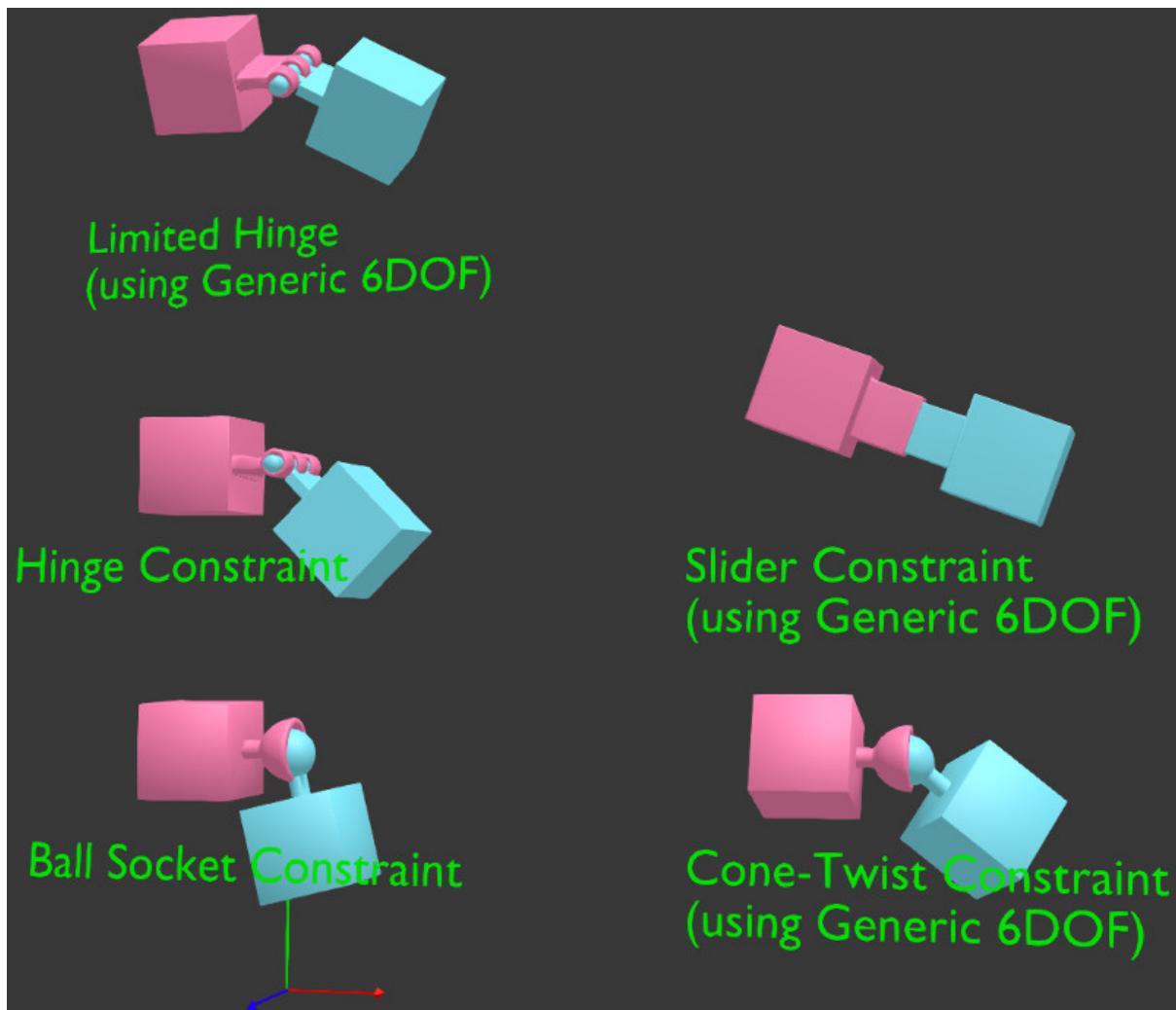


Для материала такого объекта поддерживаются: трение (Friction), упругость (Elasticity). В случае использования на одном меше нескольких материалов физические настройки считаются с первого из них.

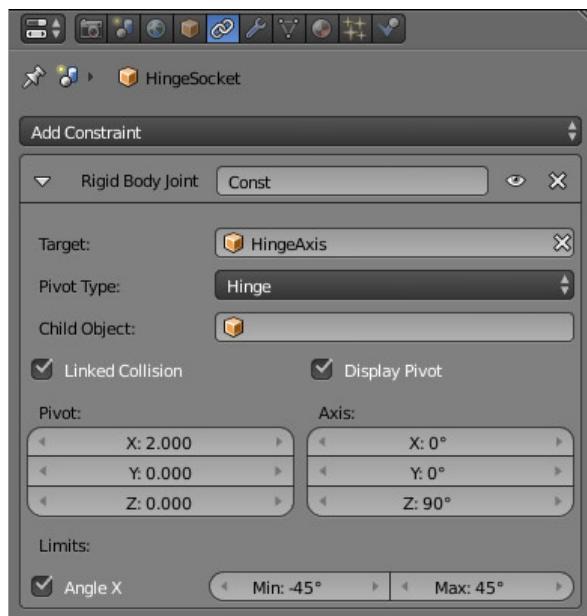
Для объекта-камеры должна использоваться настройка Physics Type = Dynamic, должен быть установлен флаг Detect Collisions.

22.4 Ограничители (Constraints)

Физические ограничители используются для уменьшения числа степеней свободы объектов.

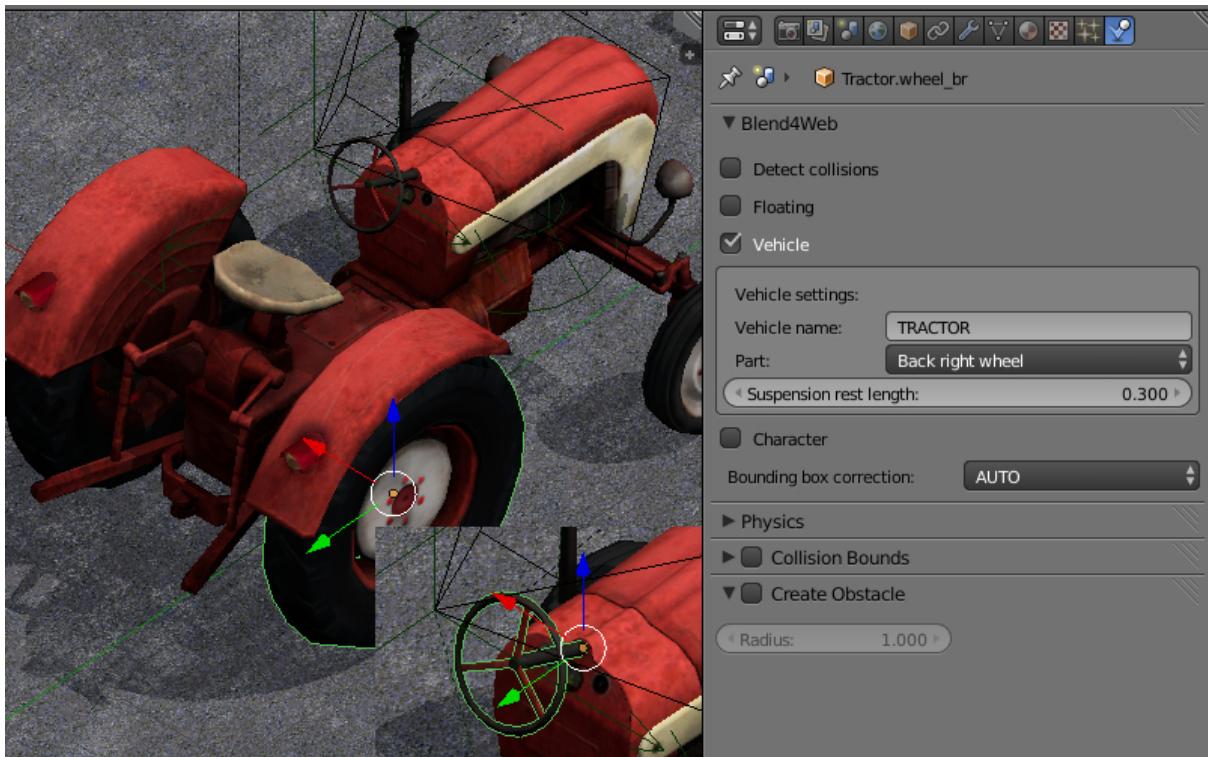


Установка физического ограничителя (**Rigid Body Joint**) на объект происходит в панели **Object Constraints**. Поддерживаемые типы (**Pivot Type**): **Ball**, **Hinge**, **Cone**, **Twist**, **Generic 6 DoF**. Физический ограничитель можно установить на один из двух взаимодействующих объектов, при этом другой выступает в качестве цели (**Target**). Оба объекта могут быть со статическим и/или динамическим типом физики. В ограничителях (кроме **Ball**) могут настраиваться пределы перемещения и вращения.



22.5 Колесные транспортные средства

Модель транспортного средства (ТС) должна состоять из 6 отдельных объектов - шасси, 4 колеса, рулевое колесо. Центр меша шасси должен соответствовать центру масс. Центры мешей колес и рулевого колеса должны располагаться на осях вращения. Рулевое колесо должно быть ориентировано в локальной системе координат: X - ось вращения, Y - вправо, Z - вверх. Объекты могут иметь любые названия.



На всех 6 объектах нужно выставить **Vehicle Part**, указать один и тот же идентификатор в поле **Vehicle Name**, выбрать соответствующий тип объекта - **Chassis**, **Steering Wheel**, **Back Right Wheel** и т.д. Для колес имеется также настройка компенсирующего хода подвески **Suspension Rest Length**.

Для шасси необходимо указать реалистичную массу (т.к. значение по умолчанию 1 кг). Для этого перейти в настройки физики, для опции **Physics Type** выбрать значение **Rigid Body**, и выставить нужное значение (например, 1000 кг) в поле **Mass**.

22.5.1 Параметры настройки для шасси

Vehicle Settings > Force Max Максимальная движущая сила транспортного средства.

Vehicle Settings > Brake Max Максимальный коэффициент торможения.

Vehicle Settings > Suspension Compression Коэффициент демпфирования при растяжении подвески.

Vehicle Settings > Suspension Stiffness Коэффициент жесткости подвески.

Vehicle Settings > Suspension Damping Коэффициент амортизации подвески.

Vehicle Settings > Wheel Friction Константа трения колес о поверхность. Для реалистичных Т.С. должен быть в районе 0.8. Но может быть значительно

увеличен, для улучшения управляемости (1000 и более)

Vehicle Settings > Roll Influence Снижает вращающий момент от колес, уменьшая вероятность переворота транспортного средства (0 - нет вращающего момента, 1 - реальное физическое поведение).

Vehicle Settings > Max Suspension Travel Cm Максимальный ход подвески в сантиметрах.

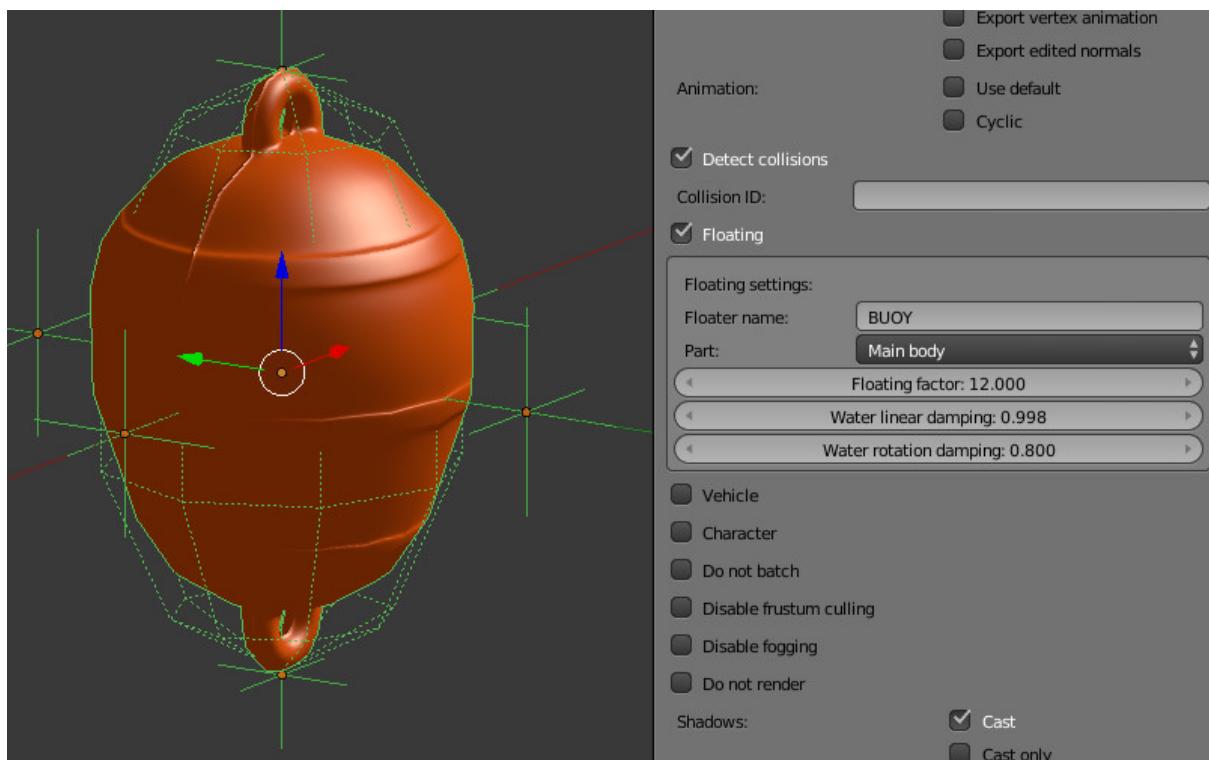
Для рулевого колеса (**Steering Wheel**) необходимо указать максимальный угол поворота (**Steering Max**) и передаточное отношение угла поворота руля к передним колесам (**Steering Ratio**). Максимальное значение угла поворота указывается в оборотах. Один оборот равен 360 градусам. Таким образом, поставив **Steering Max** равным единице, а **Steering Ratio** равным 10, максимальный поворот руля получится равным 360 градусам, а максимальный поворот передних колес 36 градусов.

На этом этапе можно произвести экспорт и загрузить сцену в движок. Рекомендуется создать дорожную поверхность с физическим материалом. В просмотрщике нажать клавишу **Q** для выбора контролируемого объекта, и выбрать шасси. Использовать **W**, **A**, **S**, **D** для управления.

Дополнительно можно настроить демпфирование **Damping** перемещения (**Translation**) и вращения (**Rotation**). Свойство влияет на скорость перемещения и инерционность ТС.

Настройка трения и эластичности физического материала дорожного покрытия не влияют на поведение ТС.

22.6 Плавающие объекты



Для того, чтобы объект мог плавать на поверхности воды (объекта с материалом Special Water), необходимо выставить свойство Floating. Существует два типа частей плавающего объекта: Main Body - непосредственно сам плавающий объект и Bob - вспомогательный объект-поплавок, на который будет действовать выталкивающая из воды сила. Плавающий объект может иметь неограниченное количество объектов типа Bob. В качестве поплавков могут использоваться как меши, так и объекты типа Empty.

Всем объектам, входящим в состав одного плавающего объекта необходимо выставить одинаковое имя в поле Floater Name

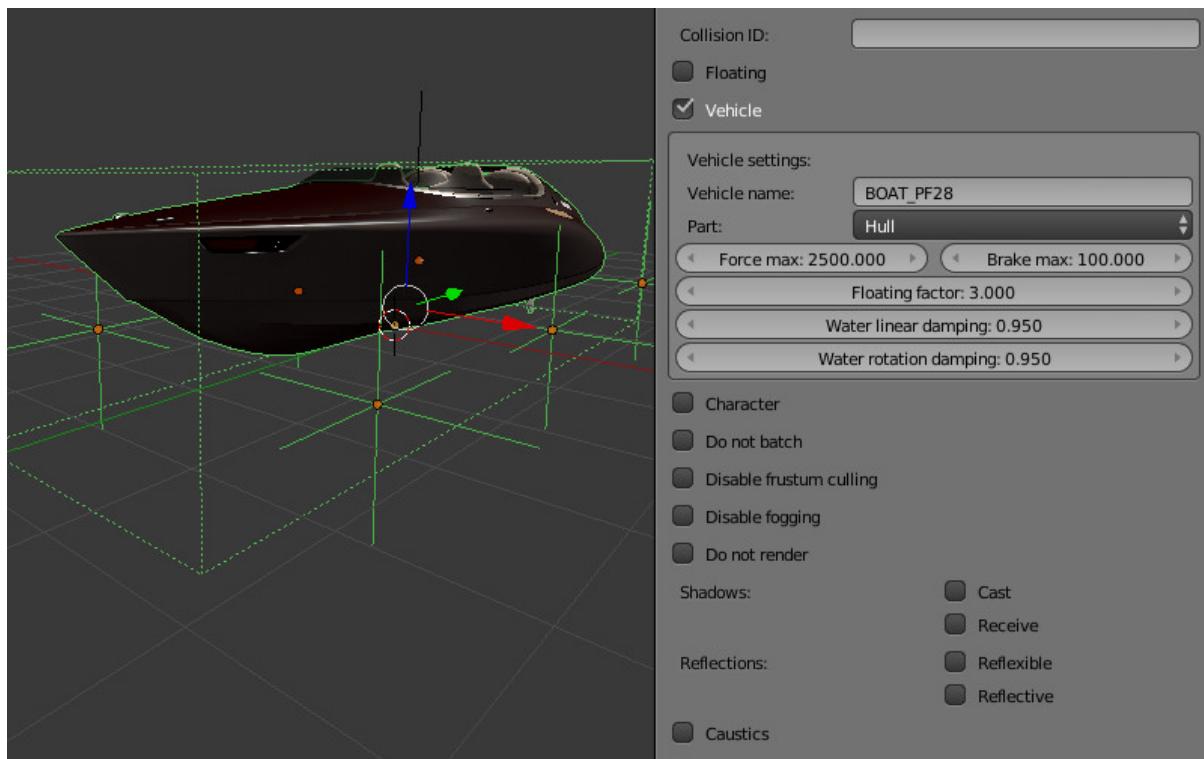
22.6.1 Параметры настройки плавающего объекта

Floating Settings > Floating Factor Коэффициент выталкивания объекта из воды.

Floating Settings > Water Linear Damping Демпфирование линейной скорости при нахождении объекта на поверхности воды (или под водой). Когда объект находится вне воды, используется значение из настроек физики.

Floating Settings > Water Rotation Damping Демпфирование вращения при нахождении объекта на поверхности воды (или под водой). Когда объект находится вне воды, используется значение из настроек физики.

22.7 Плавающие транспортные средства



Плавающие транспортные средства используют часть параметров из настроек **Vehicle Settings** и все настройки аналогичные **Floating Settings**. На основном объекте необходимо выставить **Vehicle Part**, типа **Hull**. Так же как и плавающий объект плавающее транспортное средство требует наличия вспомогательных объектов типа **Bob**.

22.7.1 Параметры настройки плавающего транспортного средства

Vehicle Settings > Force Max Максимальная движущая сила транспортного средства.

Vehicle Settings > Brake Max Максимальный коэффициент торможения.

Floating Settings > Floating Factor Коэффициент выталкивания объекта из воды.

Floating Settings > Water Linear Damping Демпфирование линейной скорости при нахождении объекта на поверхности воды (или под водой). Когда объект находится вне воды, используется значение из настроек физики.

Floating Settings > Water Rotation Damping Демпфирование вращения при нахождении объекта на поверхности воды (или под водой). Когда объект находится вне воды, используется значение из настроек физики.

22.8 Особенности использования в приложениях

Физическая подсистема реализована в модуле **uranium.js** и загружается отдельно от основного кода движка. Модуль **uranium.js** представляет собой модификацию физического движка **Bullet**, портированную для работы в браузерах. Быстрое подключение физической подсистемы можно осуществить, разместив файлы **uranium.js** и **uranium.js.mem** в той же директории, где расположен исходный код движка, используемого в приложении.

В противном случае необходимо указать путь к модулю, используя конструкцию вида:

```
m_config.set("physics_uranium_path", ".../uranium.js");
```

Примечание: При разработке приложения *в составе SDK* путь к физическому движку определяется автоматически.

Если использование физики не требуется, рекомендуется отключить флаг **Enable Physics** во вкладке сцены в Blender'e. Также можно принудительно отключить загрузку модуля **uranium.js**, если до начала инициализации движка вызывать следующий метод:

```
m_config.set("physics_enabled", false);
```

Нелинейная анимация

23.1 Редактор NLA

Редактор нелинейной анимации, имеющийся в программе Blender, позволяет в удобной форме задавать поведение сцены. С его помощью можно реализовать несложные сценарии. Таким образом исключается необходимость программирования простых сцен и приложений.



Движком поддерживается управление следующими сущностями:

- Любая анимация, параметры которой могут быть представлены с помощью *действий* (Action)

- Воспроизведение звуков
- Эмиссия частиц (в виде привязки к глобальной шкале времени)



23.1.1 Использование

1. Во вкладке **Scene** выбрать опцию **Use NLA**.
2. В редакторе **NLA Editor** задать необходимую конфигурацию поведения сцены.
3. На панели **Timeline** выбрать интервал времени анимации.

23.1.2 Дополнительные настройки

Настройка сцены *Blend4Web > Cyclic NLA* позволяет активировать режим циклической NLA-анимации.

23.1.3 Ограничения

- Вертекальная анимация не поддерживается.
- Масштабирование и зацикливание отдельных полос анимации (Strip) не поддерживается.
- Одновременное проигрывание различных видов анимации для одного объекта не поддерживается.

23.2 Визуальное программирование (NLA Script)

Осуществляется с помощью добавления слотов логики (NLA Script Slot) на вкладке настроек сцены Blender. С их помощью можно существенно расширить функциональность сцены, не прибегая при этом к программированию.

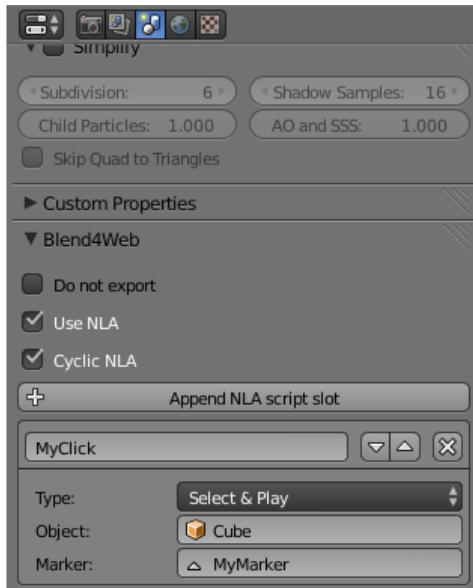


Слоты являются логическими блоками, которые выполняются с первого по последний последовательно, за исключением случаев, когда в слоте явно обозначен переход (*Jump* или *Conditional Jump*). При достижении конца списка, выполнение логики останавливается, либо, если на сцене активирована настройка *Blend4Web > Cyclic NLA*, начинается сначала.

Примечание: Использование настройки *Blend4Web > Cyclic NLA* совместно с NLA Script не приводит к автоматическому зацикливанию анимации.

Для реализации сложной логики предусмотрены числовые переменные, называемые регистрами. Каждый из 8 регистров может хранить в себе одно числовое значение. Регистры могут быть использованы для хранения какого-либо состояния сцены (например, это может быть счётчик проигрываемых анимаций, количество жизней, оставшихся у персонажа итд.).

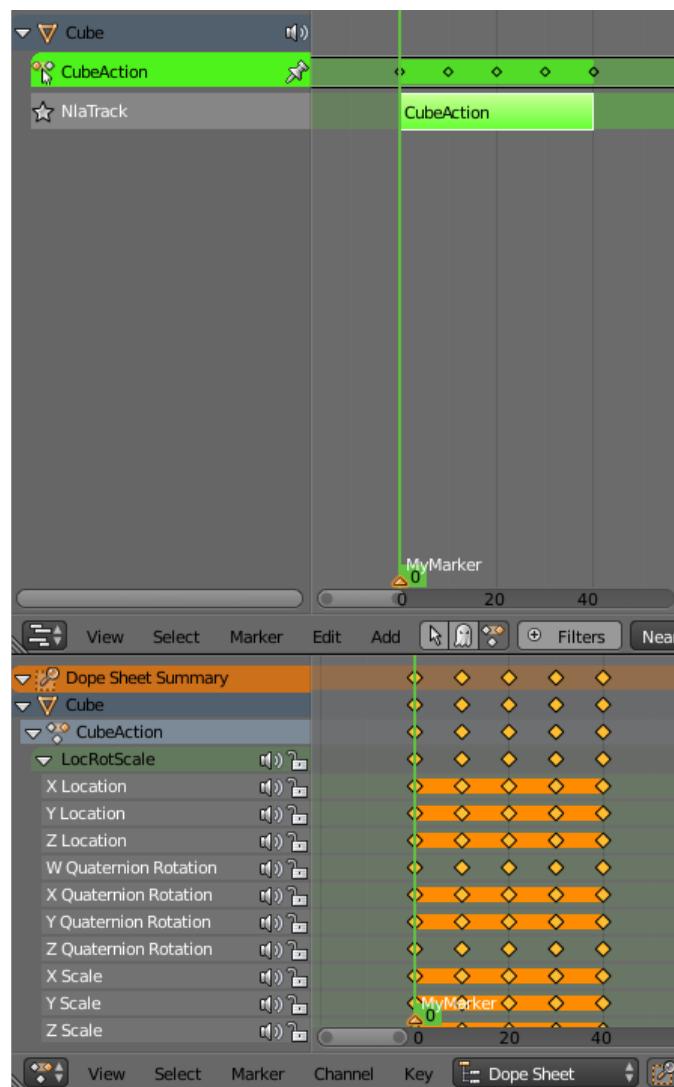
Пример использования NLA Script:



Ниже рассмотрены все возможные варианты слотов.

23.2.1 Play

Проигрывать участок NLA, начиная с кадра, на который указывает маркер. Анимация воспроизводится до следующего маркера, либо до конца шкалы времени сцены, после чего управление переходит к следующему слоту.



23.2.2 Select & Play

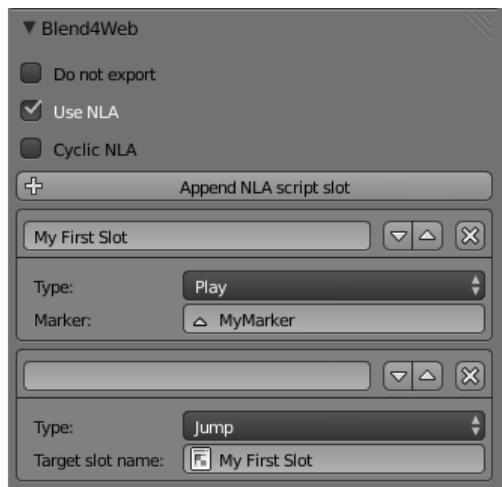
Ожидать, когда пользователь выберет объект (с помощью клика мышью в версии движка для десктопа, либо нажатия на экране в мобильной версии). Если выбран объект, указанный в параметрах слота, начать играть анимацию аналогично слоту **Play**, если же был выбран любой другой объект - немедленно передать управление следующему слоту.

Последняя функциональность используется для выбора одного из нескольких объектов, причём **Select & Play** удобно размещать в стеке один за другим. В данном случае результат выбора пользователя может быть подхвачен одним из слотов из данной последовательности, поскольку переход без анимации производится мгновенно.

Примечание: Чтобы пользователь имел возможность выбрать объект, в его настройках необходимо активировать опцию **Selectable**, как в случае реализации *эффекта свечения силуэта*.

23.2.3 Jump

Перейти к указанному слоту.



23.2.4 Select & Jump

Аналогично слоту **Select & Play**, за исключением того, что вместо анимации осуществляется переход. Указанная функция позволяет реализовать более сложную логику, поскольку появляется возможность распознавания результата выбора пользователя (выделение текущего объекта приводит к переходу по имени слота, который не обязательно является следующим в списке).

23.2.5 Conditional Jump

Перейти к указанному слоту в случае выполнения выбранного условия. В качестве параметров условия (операндов) могут выступать также регистры, которые активируются с помощью соответствующих переключателей.

23.2.6 Register Store

Записать числовое значение в регистр.

23.2.7 Math Operation

Выполнить математическую операцию и сохранить результат в регистр. Любой из параметров (операндов) может быть либо числовым значением, либо регистром.

23.2.8 Show Object и Hide Object

Используются для скрытия и отображения объектов.

23.2.9 Page Redirect

Служит для перенаправления на другие веб-страницы.

23.2.10 Page Param

Позволяет сохранить произвольный параметр веб-страницы в выбранном числовом регистре.

23.2.11 Noop

Сокращение от “No Operation”. Означает, что при обработке слота данного типа, никаких действий произведено не будет. Слот удобно использовать совместно с типом Jump или Conditional Jump.

Разработчикам приложений

24.1 Hello world!

Простейшее приложение на основе Blend4Web может иметь вид:

```
<!DOCTYPE html>
<html>
<head>
<script src="b4w.min.js"></script>
<script>
function hello() {
    var m_version = b4w.require("version");
    document.body.innerHTML = "Hello, Blend4Web " + m_version.version() + "!";
}
</script>
</head>

<body onload="hello()"></body>

</html>
```

Приложение выводит сообщение и текущую версию движка в окне браузера. Рассмотрим представленный пример. Базовый код движка (без дополнений) подключается с помощью тега `<script src="...>`. Далее, приложение ожидает окончания загрузки страницы и выводит сообщение в окне браузера. В данном примере используется единственный модуль `version`, в котором находится одноимённая функция `version()`. Подробную информацию о предназначении модулей и функций движка можно найти в [документации по API](#).

Файл `b4w.min.js` со скомпилированным кодом движка необходимо скопировать из директории SDK `deploy/apps/common` и разместить в той же директории, что и представленный HTML-файл.

24.2 Загрузка сцены в приложение

Для того, чтобы загрузить трёхмерную сцену, требуется выполнить следующую последовательность действий:

1. Разместить на странице элемент <canvas>, на котором будет производиться рендеринг.
2. После загрузки страницы, для инициализации контекста WebGL, вызвать функцию `m_main.init()` с идентификатором созданного элемента.
3. Вызвать функцию `m_data.load()` для загрузки трёхмерной сцены.

```
<!DOCTYPE html>
<html>
<head>
<script src="b4w.min.js"></script>
<script>
function hello() {
    var m_main = b4w.require("main");
    var m_data = b4w.require("data");

    var canvas_elem = document.getElementById("canvas_id");
    m_main.init(canvas_elem);
    m_data.load("some_scene.json");
}
</script>
</head>

<body onload="hello()"><canvas id="canvas_id"></canvas></body>

</html>
```

Следует отметить, что реальное приложение должно включать в себя проверку ошибок, настройку движка перед инициализацией, а также базовую систему взаимодействия с пользователем.

24.3 Быстрое создание приложений

Поскольку создание приложения с нуля может быть достаточно сложной операцией, особенно для начинающих пользователей, в движке существует специальное дополнение `app`:

```
<!DOCTYPE html>
<html>
<head>
<script src="b4w.full.min.js"></script>
<script>

var m_app = b4w.require("app");
```

```

var m_data = b4w.require("data");

m_app.init({
    canvas_container_id: "container_id",
    callback: load_cb
});

function load_cb()
    m_data.load("some_scene.json", loaded_cb);
}

function loaded_cb() {
    m_app.enable_controls();
    m_app.enable_camera_controls();
}

</script>
</head>

<body>
    <div id="container_id" style="width: 350px; height: 200px;"></div>
</body>

</html>

```

В данном случае модуль `app` создаст элемент `<canvas>` внутри контейнера с указанным идентификатором `container_id`, осуществит инициализацию движка при загрузке страницы и сообщит о её окончании с помощью обработчика `load_cb`.

Далее загружается сцена `some_scene.json`, аналогично предыдущему примеру, с тем лишь отличием, что по окончании загрузки этой сцены инициализируется подсистема управления и активируются средства для перемещения камеры с помощью клавиатуры и мыши (сенсорного экрана для мобильных устройств).

При использовании модуля `app` необходимо явно задавать размеры контейнерного элемента, поскольку в противном случае создаваемый элемент `<canvas>` будет иметь нулевые размеры.

24.4 Разработка приложений в составе SDK

В составе SDK присутствует скрипт `apps_dev/project.py`, предназначенный для сборки приложений.

Пример:

```
./project.py -a my_app_path -o advanced -b copy -v 15.02 -s my_resources_path
```

- Опция `a`" указывает на имя директории, в которой находится приложение.
- Опция `o`" (необязательная) используется для указания метода оптимиза-

ции js-файлов. Доступные варианты: **whitespace**, **simple** (по умолчанию) и **advanced**.

- Опция **b"** (необязательная) указывает из какой директории брать движок. Доступные варианты: **link** (по умолчанию) - движок подключается из директории **deploy/apps/common/**, **copy** - движок копируется в директорию с собранным приложением, **combine** - движок объединяется с минимизированным скриптом в корне приложения, **compile** - необходимые модули движка компилируются вместе со скриптами приложения.
- Опция **d"** (необязательная) указывает в какой директории размещать скомпилированное приложение.
- Опция **s"** (необязательная) указывает из какой директории брать ресурсы для сцены.
- Опция **r"** (необязательная) указывает в какой директории будут располагаться ресурсы для сцены относительно корня приложения.
- Опция **v"** (необязательная) добавляет к скриптам и стилям версию. Это делается для того, чтобы браузер не брал из кэша старые скрипты и стили.
- Опция **j"** (необязательная) добавляет в исключения скрипты, которые не будут скомпилированы.
- Опция **c"** (необязательная) добавляет в исключения стили, которые не будут скомпилированы.

Структура приложения:

- В корне директории должен находиться единственный html-файл.
- Скрипты и стили могут находиться как в корне приложения, так и во вложенных папках.

Описание работы сборщика приложений:

- Приложение полностью копируется в директорию **deploy/apps/app_name/**.
- Если указана опция '**-d**', приложение копируется в соответствующую директорию.
- Скрипты и стили компилируются относительно родительской директории.

24.5 Примеры кода

В составе SDK присутствует приложение Code Snippets, демонстрирующее примеры использования функционала движка.

На данный момент приложение включает в себя следующие примеры:

- Canvas Texture - пример работы с canvas-текстурой
- Camera Animation - создание процедурной анимации камеры

- Custom Anchors - процедурное создание аннотаций
- Dynamic Geometry - процедурное изменение геометрии
- Gyro (Mobile Only) - пример работы с гироскопом мобильных устройств
- Instancing - копирование объектов сцены
- Material API - изменение свойств материалов и замена материалов объекта
- Morphing - использование ключей деформации объекта

Приложение Code Snippets доступно по пути `SDK/apps_dev/code_snippets/code_snippets_dev.html`.
Также оно доступно из файла `index.html` в корне SDK.

24.6 Конвертация ресурсов

Существующие браузеры не полностью поддерживают основные форматы медиаданных, поэтому для создания кроссбраузерных приложений, а также с целью оптимизации, необходимо использовать конвертер ресурсов. Преобразование происходит по схеме:

для аудио (`convert_media`):

- ogg -> mp4
- mp3 -> ogg
- mp4 -> ogg

Рекомендуется использовать в качестве базового формата *Ogg*, в этом случае для обеспечения кросс-браузерной совместимости потребуется только преобразование из *ogg* в *mp4*. Пример файла на входе: `file_name.ogg`, пример файла на выходе: `file_name.altconv.mp4`.

для видео (`convert_media`):

- webm -> m4v
- m4v -> webm
- ogv -> webm
- webm -> seq
- m4v -> seq
- ogv -> seq

Рекомендуется использовать в качестве базового формата *WebM*, в этом случае для обеспечения кросс-браузерной совместимости потребуется только преобразование из *webm* в *m4v* (из *webm* в *seq* для iPhone). Пример файла на входе: `file_name.webm`, пример файла на выходе: `file_name.altconv.m4v`.

для изображений (`convert_dds`):

- png -> dds

- jpg -> dds

Пример файла на входе: `file_name.jpg`, пример файла на выходе: `file_name.jpg.dds`.

В целях оптимизации работы приложения существует возможность использования *min50* (уменьшенных вдвое) и *DDS* текстур. Для этого при инициализации приложения необходимо передать следующие параметры:

```
exports.init = function() {
    m_app.init({
        // ...
        assets_dds_available: true,
        assets_min50_available: true,
        // ...
    });
    // ...
}
```

Файл формата `.seq` представляет собой раскадрированное видео. Применяется на IE 11 и iPhone, поскольку на них возникают трудности при использовании видео стандартного формата в качестве текстуры. Использование dds-формата для изображений является более оптимальным по сравнению с другими форматами.

В состав дистрибутива включен Python скрипт (`scripts/converter.py`) для конвертации исходных файлов в другие форматы с целью расширения спектра поддерживаемых платформ, а также для уменьшения размера ресурсов. Для конвертации необходимо выполнить команду:

```
> ./converter.py [-d file_path] resize_textures | convert_dds | convert_media
```

С помощью опции `-d` можно указать путь к директории, в которой будет производится конвертация.

При необходимости исключить некоторую директорию при конвертации, достаточно разместить в ней файл с именем `.b4w_no_conv`. На конвертацию во вложенных директориях это не повлияет.

Аргумент `resize_textures` используется для изменения размера текстур в режиме **LOW**.

Движком могут использоваться файлы, созданные пользователем вручную и имеющие следующие наименования: `file_name.altconv.m4v`, `file_name.altconv.mp3` и т.д. Такие файлы необходимо размещать в одной директории с медиафайлом, используемым в Blender'e.

Вы также можете использовать бесплатную кроссплатформенную программу [Miro Video Converter](#) для конвертации медиаданных.

24.7 Система модулей

Несмотря на то, что движок предоставляет прикладному программисту API в объеме десятков модулей, в процессе работы он занимает в глобальном пространстве имен единственный объект `b4w`. При необходимости обращения к модулю, последний импортируется с помощью вызова функции `b4w.require`.

Допустима регистрация сторонних модулей, если их имена не пересекаются с имеющимися. Регистрация происходит посредством вызова `b4w.register`. Проверка наличия модуля с некоторым именем может быть осуществлена с помощью `b4w.module_check`.

Пример:

```
// check if module exists
if (b4w.module_check("my_module"))
    throw "Failed to register module: my_module";

// register my_module
b4w.register("my_module", function(exports, require) {

    // import module "version"
    var m_version = require("version");

    // export print_build_date() from module "my_module"
    exports.print_build_date = function() {
        // exec function date() from module "version"
        console.log("Engine build date: " + m_version.date());
    }
});

// import module "my_module"
var m_my_module = b4w.require("my_module");

// exec function print_build_date() from module "my_module"
m_my_module.print_build_date();
```

24.8 Событийная модель

Событийная модель предоставляет унифицированный интерфейс для описания изменения состояний 3D сцены, упрощая обработку событий физики и действий пользователя.

24.8.1 Сенсоры

Основным блоком событийной модели является сенсор (`sensor`). Сенсор является программной сущностью, и может быть только активным (1, единица) или неактив-

ным (0, ноль). Некоторые сенсоры несут полезную нагрузку (payload). Например, сенсор трассировки лучей (Ray Sensor) предоставляет относительную длину луча пересечения.

Управление сенсорами не доступно пользователю в виде открытого API. Вместо этого каждый сенсор должен присутствовать в одном или нескольких множествах (sensor manifold). Множество является логическим контейнером, ассоциированным с объектом на сцене. Оно генерирует ответ на определенный набор событий сенсоров в виде вызова функции-обработчика. Для определения множества необходимо иметь следующую информацию (см. также описание функции `controls.create_sensor_manifold()` в документации по API):

- Объект-носитель множества (например, бросаемый объект).
- Уникальный идентификатор множества (например, "IMPACT").
- Тип вызова функции-обработчика (варианты: `CT_CONTINUOUS` - непрерывный, `CT_LEVEL` - уровень, `CT_SHOT` - одномоментный, `CT_TRIGGER` - переключающий).
- Массив сенсоров.
- Логическая функция, определяющая при какой комбинации состояний сенсоров вызывается функция-обработчик.
- Функция-обработчик.
- Необязательный параметр, который может быть передан в функцию-обработчик.

24.8.2 Пример

Поставлена задача озвучить удар бросаемого камня так, чтобы при ударе о различные среды (например, земля и стена) выводился характерный звук. На сцене в Blender'е имеются ограничивающие меши с физическими материалами, их идентификаторы "TERRAIN" и "WALL". На сцене также присутствует бросаемый физический объект с названием "Stone".

Определим по одному сенсору соударения (Collision Sensor) для каждой среды, по типу издаваемого звука.

```
// import the modules
var m_scenes = b4w.require("scenes");
var m_controls = b4w.require("controls");

// get the object being thrown
var stone = m_scenes.get_object_by_name("Stone");

// create the sensors
var sensor_impact_terrain = m_controls.create_collision_sensor(stone, "TERRAIN");
var sensor_impact_wall     = m_controls.create_collision_sensor(stone, "WALL");
```

Добавим сенсоры в массив. В качестве логической функции используем логическое ИЛИ. В обработчике напишем код для воспроизведения звука. Создадим множество сенсоров с идентификатором “IMPACT” и типом CT_SHOT (одномоментный).

```
// array of the sensors
var impact_sens_array = [sensor_impact_terrain, sensor_impact_wall];

// manifold logic function
var impact_sens_logic = function(s) {return (s[0] || s[1])};

// callback
var impact_cb = function(obj, manifold_id, pulse) {

    // NOTE: it's possible to play both sounds simultaneously

    if (m_controls.get_sensor_value(obj, manifold_id, 0) == 1) {
        // ...
        console.log("play the terrain impact sound");
    }

    if (m_controls.get_sensor_value(obj, manifold_id, 1) == 1) {
        // ...
        console.log("play the wall impact sound");
    }
}

// create the manifold
m_controls.create_sensor_manifold(stone, "IMPACT", m_ctl.CT_SHOT,
    impact_sens_array, impact_sens_logic, impact_cb);
```

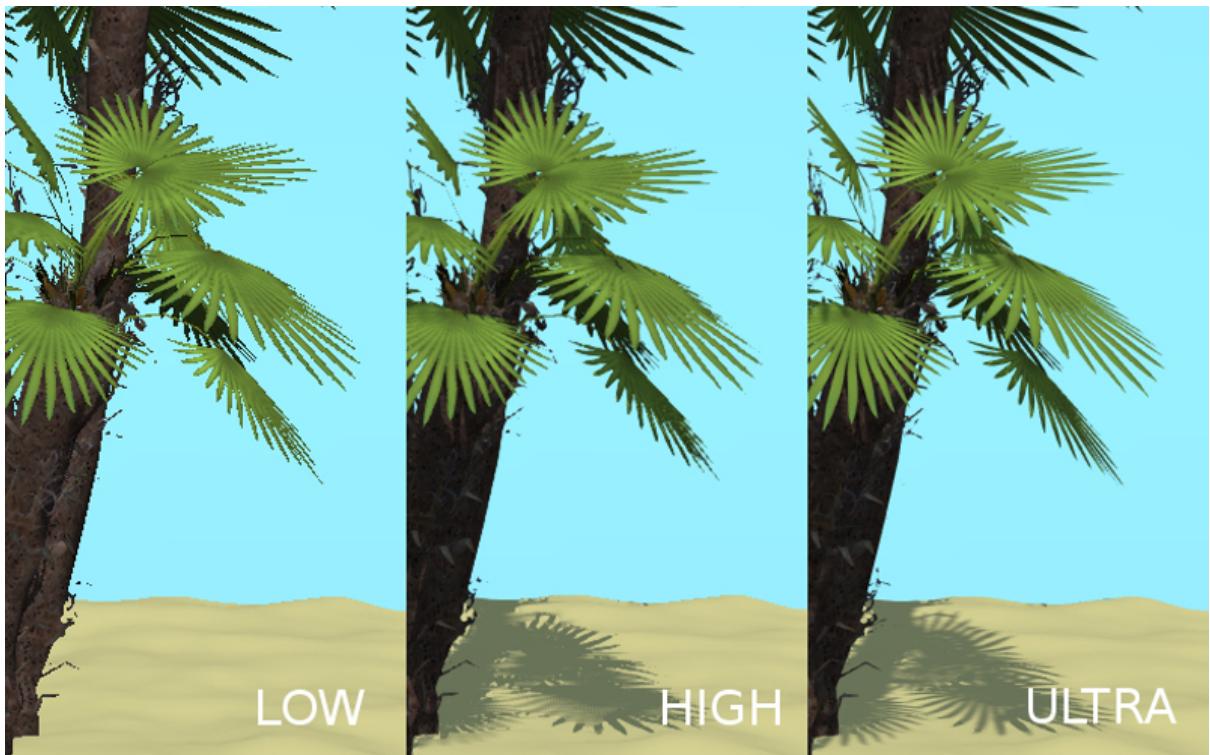
При столкновении объекта “Stone” с любым из физических материалов “TERRAIN” или “WALL” происходит вызов функции-обработчика. Внутри этой функции получим значения обоих сенсоров по их индексу в массиве сенсоров (0 - “TERRAIN”, 1 - “WALL”). Значение сенсора = 1 (активный) означает, что произошло столкновение с соответствующим физическим материалом. В результате воспроизводится соответствующий звук (код не показан).

24.9 Профили качества изображения

Для поддержки различных по функциональности платформ в движке реализовано несколько профилей качества изображения:

- *низкое качество* (P_LOW) - отключен ряд функций (тени, динамическое отражение, постпроцессинг), размер текстур для сборочной версии уменьшен вдвое, антиалиасинг отключен
- *высокое качество* (P_HIGH) - используются все запрошенные сценой функции, метод антиалиасинга FXAA

- *максимальное качество* (`P_ULTRA`) - вдвое увеличено разрешение рендеринга, увеличено разрешение карт теней, метод антиалиасинга SMAA



Переключение профилей качества осуществляется программно, до инициализации контекста WebGL. Профиль по умолчанию `P_HIGH`.

```
var m_cfg = b4w.require("config");
var m_main = b4w.require("main");

m_cfg.set("quality", m_cfg.P_LOW);
m_main.init(...);
```

Разработчики приложений могут также установить параметр `quality` при инициализации движка с использованием дополнения `app.js`:

```
var m_cfg = b4w.require("config");
var m_app = b4w.require("app");

m_app.init({
    canvas_container_id: "body_id",
    quality: m_cfg.P_HIGH
});
```

24.10 Файловая структура SDK

apps_dev исходный код приложений (не все приложения доступны в бесплатной версии)

Makefile файл для сборки всех приложений из состава SDK

project.py скрипт для разработчиков приложений

viewer исходные файлы приложения для просмотра сцен Viewer

assets.json метаданные с информацией о сценах, загружаемых просмотрщиком сцен Viewer

csrc исходный код бинарной части экспортера движка и других утилит на языке C

doc_src исходный код настоящего руководства пользователя на языке разметки reST

blender исходные файлы сцен в формате Blender (не все сцены доступны в бесплатной версии)

blender_scripts экспортер и вспомогательные скрипты для Blender'a

deploy директория с ресурсами для размещения на сервере (исходные файлы сцен, скомпилированные приложения и документация)

api_doc документация API движка для разработчиков (собирается автоматически, на основе исходного кода движка)

apps 3D-приложения, предназначенные для развертывания, директория дублирует *apps_dev*

common Файлы скомпилированного движка. Используются приложениями из состава SDK (отсюда и название).

assets загружаемые ресурсы: сцены, текстуры, звуковые файлы

doc настоящее руководство пользователя в формате HTML, собирается автоматически из *doc_src*

globals_detect вспомогательный код для определения глобальных переменных

tutorials исходные файлы обучающих примеров

index.html и **index_assets** файлы главной веб-страницы SDK

license файлы с текстами лицензионных соглашений

Makefile файл сборки для компиляции движка, приложений, документации, развертывания на удаленном сервере (недоступен в бесплатной версии)

README.rst файл README

scripts дополнительные скрипты

chrome_debug.sh скрипт, запускающий браузер Chrome в режиме отладки

compile_b4w.py скрипт для сборки кода движка и приложений

converter.py скрипт, осуществляющий: уменьшение разрешения текстур вдвое, компрессию текстур в формат DDS, конвертацию звуковых файлов в форматы mp4 и ogg

custom_json_encoder.py форк Python-модуля json, сортирует ключи по алфавиту в обратном порядке

gen_glmatrix.sh скрипт для генерации математического модуля на основе исходных файлов из репозитория glMatrix 2

graph.sh генератор текущего графа сцены в формате svg, используется для отладки рендеринга

memory.sh скрипт для проверки обычной (RAM) и видео-памяти (VRAM)

plot.sh построитель графиков отладочной информации

reexporter.py скрипт для автоматического переэкспорта всех сцен из состава SDK

remove_alpha_channel.sh скрипт для удаления альфа-канала изображения

report_unused_resources.py скрипт для проверки и сообщения о неиспользуемых ресурсах (изображения и звуки, на которые ссылаются экспонируемые файлы)

screencast.sh скрипт для записи видео с экрана

shader_analyzer.py скрипт, запускающий локальный веб-сервер, который осуществляет подсчет сложности шейдеров

shaders GLSL-шейдеры движка

src основной исходный код ядра движка

addons исходный код дополнений движка

ext исходный код внешних объявлений, формирующих API движка

libs исходный код библиотек

tools Различные инструменты для сборки движка и приложений

closure-compiler компилятор Google Closure, файлы исключений к нему, генераторы файлов исключений

glsl

compiler компилятор GLSL-шейдеров движка

pegjs грамматики парсер-генератора PEG.js для реализации препроцессора GLSL, а также скрипт для генерации модулей парсеров из этих грамматик

yuicompressor утилита для сжатия файлов CSS

uranium исходный код и скрипты сборки физического движка Uranium (форк Bullet)

VERSION содержит текущую версию движка

24.11 Текстура типа Canvas

Для работы с подобными текстурами используется модуль `textures`. Пример работы с ним описан ниже.

```
var m_tex = require("textures");
...
var ctx = m_tex.get_canvas_texture_context("canvas_id");
...
// operations with canvas context
...
m_tex.update_canvas_texture_context("canvas_id");
```

Для получения контекста используется функция `get_canvas_texture_context()`, куда передается идентификатор “`canvas_id`”, задаваемый пользователем в Blender. После преобразований с контекстом вызывается функция `update_canvas_texture_context()`, которая осуществляет визуализацию изменений элемента с идентификатором “`canvas_id`”.

24.12 Настройка браузера для загрузки локальных ресурсов

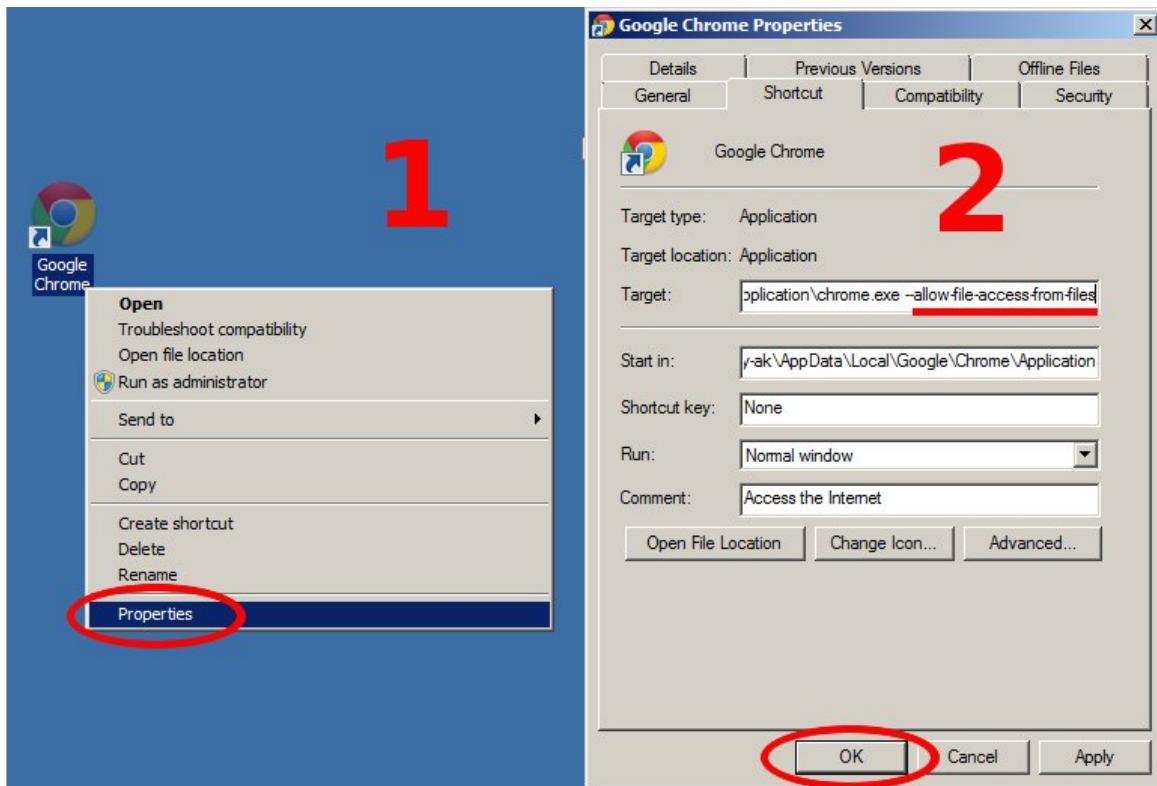
Начиная с версии 15.02, в состав Blend4Web SDK входит *сервер разработки*, решający проблему загрузки локальных ресурсов. Тем не менее, приведенные ниже инструкции могут быть полезны разработчикам.

Рендерер движка является Web-приложением, и его работа происходит при просмотре HTML-файла в браузере. После инициализации происходит загрузка ресурсов (сцен, текстур), которая подчиняется *правилу ограничения домена*, запрещающему, в частности, загрузку из локальной директории. Простым способом обхода этого ограничения может быть настройка браузера для загрузки локальных ресурсов. Другой способ - *использование локального web-сервера*.

Настройка браузера для загрузки локальных ресурсов является простым и универсальным методом обойти ограничения безопасности. Однако использовать настроенный подобным образом браузер рекомендуется только для просмотра локального контента.

Chrome на Windows:

Правой кнопкой мыши нажать на ярлыке на рабочем столе, выбрать **Свойства** (**Properties**), после чего в поле для пути к исполняемому файлу добавить после пробела `--allow-file-access-from-files`. Нажать **OK**.



Для удобства можно предварительно создать копию ярлыка и изменить ее для локального просмотра, оставив оригинальную версию ярлыка для запуска браузера в обычном режиме.

Chrome на OS X:

Открыть Терминал и запустить браузер с параметром:

```
> /Applications/GoogleChrome.app/Contents/MacOS/GoogleChrome
```

```
> --allow-file-access-from-files
```

Chrome/Chromium на Linux:

Запустить браузер с параметром:

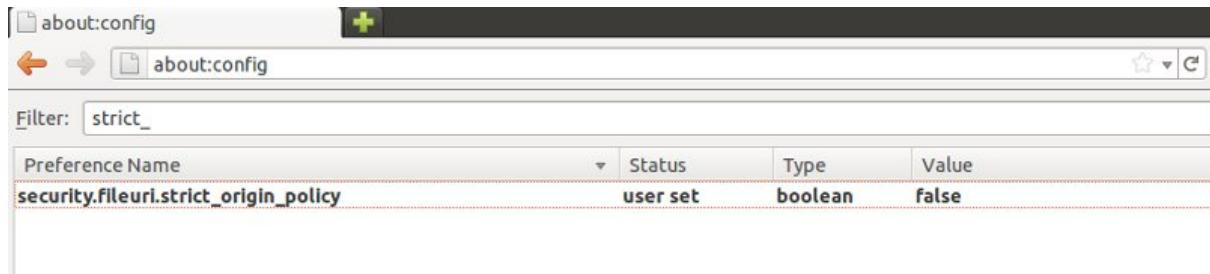
```
> google-chrome --allow-file-access-from-files
```

ИЛИ:

```
> chromium-browser --allow-file-access-from-files
```

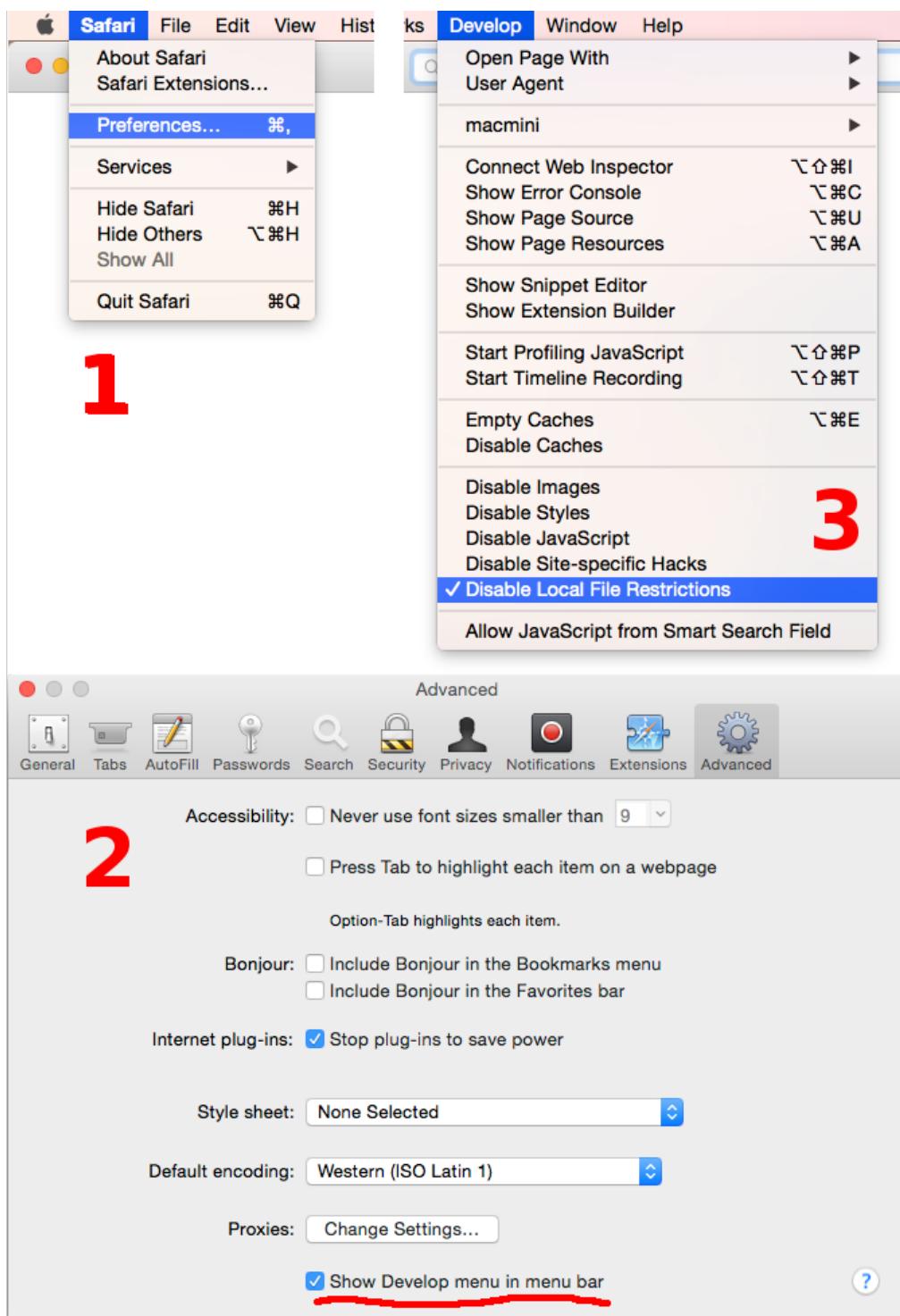
Firefox на Windows/Linux/OS X:

Ввести **about:config** в адресную строку браузера, найти параметр **security.fileuri.strict_origin_policy** и переключить его двойным щелчком мыши из **true** в **false**.



Safari/OS X:

Включить в настройках отображение меню “Develop”, затем активировать опцию “Disable Local File Restrictions”.



24.13 Запуск локального сервера

При желании вместо имеющегося в составе SDK *сервера разработки* можно использовать собственный web-сервер. Наиболее простым решением является использование простого веб-сервера из стандартной библиотеки *Python*.

На Windows:

1. Загрузить и инсталлировать последнюю версию Python с [официального сайта](#). В процессе установки выберите опцию добавления пути к исполняемому файлу (*Add python.exe to Path*).
2. Запустить командную строку (Command Prompt).
3. Выполнить команду в корневой директории SDK:

```
> python -m http.server
```

На Linux/OS X:

1. В случае отсутствия в дистрибутиве, загрузить и инсталлировать последнюю версию Python с [официального сайта](#).
2. Запустить командную строку (Terminal).
3. Выполнить команду в корневой директории SDK:

```
> python3 -m http.server
```

После загрузки сервера, в любом браузере открыть корневую страницу с приложениями из состава SDK, расположенную по адресу <http://localhost:8000>.

В случае необходимости, можно указать порт дополнительным параметром:

```
> python3 -m http.server 8080
```

Разработчикам движка

25.1 Стиль оформления кода

В движке применяется структурное программирование. Код организуется в модули. Подходы ООП не используются, классы не определяются, наследование не осуществляется и т.п.

Используется K&R стиль, за исключением того, что открывающая скобка для составного оператора ставится на той же строке, например:

```
function foo_bar() {  
    // ...  
}  
  
if (a > b) {  
    // ...  
}
```

Для выравнивания используются 4 пробела (табуляция запрещена).

25.1.1 Примеры

В именах переменных и функций используется знак подчеркивания:

```
var foo_bar = 123; // correct  
var fooBar = 123; // wrong
```

Все глобальные переменные начинаются со знака подчеркивания:

```
var _foo_bar = null;
```

Константы пишутся прописными буквами и никогда не начинаются со знака подчеркивания:

```
var FOO_BAR = 100;
```

Для внешних API названия методов и свойств задаются через точку. Поля, требующие защиты от обфускации, помещаются в специальный тэг `@cc_externs`:

```
exports.FOO_BAR = 123;

exports.foo_bar = function() {

}

/**
 * Set properties.
 * @method module:properties.set_props
 * @param {Object} foo Foo object
 * @cc_externs props_1 props_2
 * @cc_externs props_3 props_4
 */
exports.set_props = function(foo) {

    var bar_1 = foo.props_1;
    var bar_2 = foo.props_2;
    var bar_3 = foo.props_3;
    var bar_4 = foo.props_4;

    ...
}
```

Комментарии только на английском языке. Стиль комментирования - JSDoc.

25.2 Сборка движка

Перед сборкой необходимо убедиться, что в системе присутствуют все необходимые зависимости, для чего следует свериться с [таблицей](#).

Для компиляции движка и входящих в SDK приложений достаточно выполнить команду из корневой директории SDK:

```
make compile
```

Полная сборка, включающая конвертацию ресурсов (текстур, звуков и видео), компиляцию и подготовку документации вызывается командой:

```
make build
```

Сборка архивов с дистрибутивами:

```
make dist
```

Все вышеперечисленные операции могут быть выполнены одной командой:

```
make all
```

25.3 Сборка аддона

Бинарные сборки аддона Blend4Web подготовлены для следующих платформ: Linux x32/64, OS X x64, Windows x32/64. В то же время пользователи имеют возможность произвести сборку самостоятельно.

Для этого необходимо наличие Python 3.x (желательно, чтобы версия была эквивалентна используемой в Blender) и компилятора языка C (в Linux достаточно установить пакеты python3-dev и build-essential).

Пути относительно корневой директории SDK:

- скрипт сборки: `csrc/b4w_bin/build.py`
- аддон Blend4Web: `blender_scripts/addons/blend4web/`

Запуск сборки осуществляется следующим образом:

```
python3 ./csrc/b4w_bin/build.py
```

Результатом сборки будет бинарный файл с именем:

`b4w_bin_[ПЛАТФОРМА]_[АРХИТЕКТУРА].[СТАНДАРТНОЕ_РАСШИРЕНИЕ]`,

размещенный в каталоге с аддоном. Пример: `b4w_bin_Linux_64.so`. После этого аддон станет готовым к использованию на данной платформе.

25.4 Зависимости

Для ведения эффективной разработки движка и приложений, необходим ряд сторонних программ (зависимостей). Большинство этих зависимостей находится в составе современных дистрибутивов GNU/Linux, таких как Ubuntu. В других Unix-подобных системах (Apple OS X, FreeBSD) их установка из исходных кодов или иных источников не представляет существенных проблем.

В таблице ниже перечислены все зависимости, в порядке убывания важности для разработки.

Название	Пакет в дистрибутиве Ubuntu 14.04	Назначение
Bash	в составе по умолчанию	интерпретатор скриптов
Python 3	в составе по умолчанию	интерпретатор скриптов
NodeJS	nodejs	компиляция шейдеров
Java	default-jre	компиляция и обfuscация модулей движка
Emscripten	из исходных текстов EMSDK	сборка Uranium
ImageMagick, GraphicsMagick	imagemagick, graphicsmagick	конвертация ресурсов
NVIDIA Texture Tools	libnvtt-bin	конвертация ресурсов
NVIDIA Cg Toolkit	nvidia-cg-toolkit	отладка шейдеров
Libav	libav-tools	конвертация ресурсов
Gnuplot	gnuplot	отладка
Graphviz	graphviz	отладка
xsel	xsel	отладка
Sphinx	sphinx-doc	сборка документации (HTML-версия)
sphinx-intl	устанавливается с помощью PIP	сборка документации (перевод)
TeX Live	texlive, texlive-latex-extra texlive-lang-cyrillic	сборка документации (PDF-версия)
JSDoc 3	из исходных текстов JSDoc	сборка документации (документация на API)
PEG.js	из исходных текстов PEG.js	препроцессинг шейдеров

25.5 Названия функций и переменных

Рекомендуется при создании новых функций и переменных использовать следующие префиксы и суффиксы.

init_ создание абстрактного объекта

create_ создание конкретного объекта

update_ обновить состояние имеющегося объекта

attach_ / detach_ добавить/удалить временное свойство к объекту

append_ / remove_ добавить/удалить временное свойство к уже существующим подобного рода

insert_ / pop_ добавить/удалить элемент массива (доступ по индексу места)

apply_ / clear_ операция с флагом, бинарной величиной или произвольным параметром

set_ / get_ установить/получить значение свойства/переменной

- *tmp* глобальная переменная - кеш в виде простого объекта (массив, вектор)
- *cache* глобальная переменная - кеш в виде сложного объекта

25.6 Отладка

Отладка движка производится с помощью методов модуля `debug.js`.

Структура текущего рендер-графа может быть сохранена в формате DOT с помощью вызова `b4w.debug.scenegraph_to_dot()`, например, в консоли браузера. После вызова данного метода содержимое консоли сохранить в файл с расширением `.gv`. Чтобы получить граф в графическом виде, необходим набор утилит `graphviz`. Преобразование в формат SVG выполняется с помощью вызова:

```
> dot -Tsvg graph.gv -o graph.svg
```

где `graph.gv` имя файла с сохранённым графиком.

25.7 Компиляция шейдеров

Используемые в движке шейдеры подвергаются обработке компилятором. Компилятор выполняет 3 основных процедуры:

- валидацию кода шейдеров,
- обфускацию кода шейдеров,
- оптимизацию кода шейдеров.

Для запуска компиляции требуется выполнить одну из команд в корневой директории SDK:

```
> make compile_shaders  
> make verify_shaders
```

- **make compile_shaders** - проверка, обфускация, оптимизация и экспорт скомпилированных шейдеров,
- **make verify_shaders** - проверка, обфускация и оптимизация.

В процессе компиляции сначала осуществляется синтаксический анализ (парсинг) текста шейдера. Соответствующий парсер создается автоматически на основе грамматики с помощью генератора `PEG.js`. Далее по данным парсинга производится валидация, обфускация и оптимизация шейдеров, после чего шейдеры экспортируются в виде абстрактного синтаксического дерева (Abstract Syntax Tree, AST) для непосредственной загрузки движком.

Расположение основных файлов в репозитории:

- исходная грамматика - `glsl_utils/pegjs/glsl_parser.pegjs`
- скрипт генерации парсера - `glsl_utils/pegjs/gen_nodejs.sh`

- парсер - glsl_utils/compiler/glsl_parser.js

25.7.1 Валидация

Компилятор шейдеров выполняет следующие процедуры, связанные с проверкой кода:

- вывод сообщений о неиспользуемых переменных и функциях (dead code),
- проверка синтаксиса шейдеров,
- проверка шейдеров на соответствие import/export-механизма,
- удаление лишних пробелов, переводов строк и повторяющихся символов ”,”.

25.7.2 Обфускация

Обфускация служит для сокращения объема и затруднения понимания GLSL-кода. На данный момент в нем реализована следующая процедура:

- замена пользовательских идентификаторов более короткими односимвольными, двухсимвольными и т.д. именами (с поддержкой import/export-механизма).

25.7.3 Оптимизация

Оптимизация заключается в выполнении следующих процедур:

- удаление фигурных скобок, которые не несут функциональной нагрузки, но порождают новые области видимости (даный функционал полезен при обработке директив node/lamp),
- внутрифункциональная оптимизация, связанная с использованием малого числа буферных локальных переменных взамен локальных переменных, заданных программистом.

Примером удаления бесполезных фигурных скобок может служить замена кода

```
void function(){
    int a;
{
    a = 1;
}
}
```

следующим кодом

```
void function(){
    int a;
    a = 1;
}
```

Использование малого числа буферных локальных переменных заключается в том, что они повторно используются в разных контекстах. Например, следующий код

```
int function(){
    int a = 1;
    int b = a + 3;
    return b;
}
```

будет заменен на

```
int function(){
    int _int_tmp0 = 1;
    _int_tmp0 = _int_tmp0 + 3;
    return _int_tmp0;
}
```

Примечание: Не производится оптимизация локальных переменных структур и переменных массивов.

25.7.4 Директивы import/export

В целях упорядочивания, структурирования и повышения удобочитаемости кода шейдеров в include-файлах используются директивы import и export. Они указываются в начале файла и должны выглядеть примерно следующим образом:

```
#import u_frame_factor u_quatsb u_quatsa u_transb u_transa a_influence
#import qrot

#export skin
```

Директива #import определяет набор идентификаторов, которые объявлены вне этого include-файла, но доступны для использования в нем. Имеется ограничение: такие идентификаторы должны быть обязательно объявлены где-либо выше места подключения include-файла.

Директива #export определяет набор идентификаторов, доступных для использования вне данного файла. Такие идентификаторы должны быть обязательно объявлены в этом файле.

Таким образом, шейдер, использующий include-файл, обязан до места подключения содержать объявления, необходимые для импорта, а после него может использовать экспортимые идентификаторы.

Идентификаторами могут быть как имена переменных, так и имена функций. По умолчанию при отсутствии директив import/export считается, что include-файл не использует внешние объявления и не предоставляет пользование внутренними.

25.7.5 Рекомендации и ограничения

В связи с наличием препроцессинга, необходимостью совместной обработки нескольких шейдеров и include-файлов, а также особенностями реализации компилятора гарантировать работоспособность полученного на выходе кода можно только при соблюдении ряда правил или ограничений на текст исходных шейдеров:

1. Обязательное использование специальной директивы `#var` для описания констант, определяемых движком в момент запуска. Например:

```
#var AU_QUALIFIER uniform
AU_QUALIFIER float a;
```

Синтаксис здесь схож с директивой `#define`. Смысл директивы `#var` в том, чтобы определяемое ею значение позволило распарсить исходный шейдер. Что это будет конкретно (например, ‘uniform’ или ‘attribute’ в примере выше), не важно, т.к. на этом этапе оно все равно неизвестно. Однако, желательно указывать более-менее подходящее описание, а не что-то совершенно произвольное.

Примечание: Для констант, используемых не в коде шейдера, а в выражениях препроцессинга, директива `#var` не обязательна.

2. Использование при необходимости директив `import/export`.
3. Не следует перегружать встроенные функции, только пользовательские.
4. Не следует объявлять переменные с именем одной из встроенных функций, либо `main` (даже если это не приводит к ошибке).
5. Нельзя использовать директивы `#var` и `#define` для замены отдельных символов в таких операторах, как: “`++`”, “`-`”, “`*=`”, “`/=`”, “`+=`”, “`-=`”, “`==`”, “`<=`”, “`>=`”, “`!=`”, “`&&`”, “`||`”, “`^^`”.

Например:

```
#var EQUAL =
...
a *EQUAL b;
...
```

6. Использование директивы `#include`, не должно приводить к неоднозначности при обfuscации содержимого include-файла. Это может произойти в том случае, когда один и тот же файл включается в несколько разных шейдеров, и в каком-то из них могут повлиять определенные выше директивы, вроде `#var` или `#define`. Так же не стоит использовать в include-файле необъявленные функции и переменные.
7. Использование вложенных `include`'ов или множественного включения одного и того же `include`'а в один и тот же шейдер не поддерживается.
8. К неработоспособности шейдера может привести нетривиальное использование препроцессинга, например, создающее невалидный GLSL-код:

```
#if TYPE
void function1() {
#else
void function1(int i) {
#endif
...
}
```

9. Не следует объявлять переменные с именами вида `node_[NODE_NAME]_var_[IN_OUT_NODE]`, где `NODE_NAME` — название некоторой ноды, `IN_OUT_NODE` — название одного из входов или выходов ноды.
10. Не разрешается множественное использование одной и той же директивы `#nodes_main`, `#nodes_global` или `#lamps_main` в одном шейдере.

25.7.6 Поддержка расширений WebGL

Работа обфускатора может зависеть от используемых WebGL-расширений, если они каким-либо образом влияют на шейдерный язык. На данный момент поддерживаются следующие расширения:

- `OES_standard_derivatives`

25.7.7 Ошибки компилятора

В случае ошибки компилятор выведет соответствующее сообщение в консоли.

Перечень возможных ошибок:

Сообщение об ошибке	Причина
Error! Ambiguous obfuscation in include file ‘FILE_NAME’. Error! Bad preprocessing collision while obfuscation identifier: ‘NAME’. Varying/uniform or varying/attribute qualifiers combination. File: ‘FILE_NAME’.	Ошибка! Неоднозначная обфускация include-файла FILE_NAME. Ошибка в файле FILE_NAME. Невозможность обфускации переменной с именем NAME из-за переопределения при препроцессинге. Переопределение одной и той же переменной с разными квалификаторами. Недопустимые комбинации: varying/uniform, varying/attribute.
Error! Extension NAME is unsupported in obfuscator. File: ‘FILE_NAME’.	Ошибка! WebGL-расширение с именем NAME, использованное в файле FILE_NAME, не поддерживается обфускатором.
Error! Include ‘FILE_NAME’ not found.	Ошибка! При подключении не найден include-файл FILE_NAME.
Error! Undeclared TYPE: ‘NAME’. File: ‘FILE_NAME’.	Ошибка в файле FILE_NAME. Необъявленный идентификатор типа TYPE (переменная, функция, структура, ...) с именем NAME.
Error! Undeclared TYPE: ‘NAME’. Importing data missed. File: ‘FILE_NAME’.	Ошибка! Необъявленный идентификатор типа TYPE (переменная, функция, структура, ...) с именем NAME. Отсутствует объявление идентификатора, требуемого в include-файле FILE_NAME согласно директиве #import.
Error! Undeclared TYPE: ‘NAME’. Possibly exporting needed in include file ‘INCLUDE_NAME’. File: ‘FILE_NAME’.	Ошибка в файле FILE_NAME. Необъявленный идентификатор типа TYPE (переменная, функция, структура, ...) с именем NAME. Возможно требуется разрешить его экспорт в include-файле INCLUDE_NAME.
Error! Undeclared TYPE: ‘NAME’. Possibly importing needed. File: ‘FILE_NAME’.	Ошибка! Необъявленный идентификатор типа TYPE (переменная, функция, структура, ...) с именем NAME. Возможно требуется указать его как импортируемый в include-файле FILE_NAME.
Error! Unused export token ‘NAME’ in include file ‘FILE_NAME’.	Ошибка! В include-файле FILE_NAME разрешен для экспорта необъявленный идентификатор с именем NAME.

<p>Error! Using reserved word in TYPE 'NAME'. File: 'FILE_NAME'.</p>	<p>Ошибка в файле FILE_NAME. Использование зарезервированного слова при объявлении идентификатора типа TYPE (переменная, функция, структура, ...) с именем NAME.</p>
<p>Error! 'all' extension cannot have BEHAVIOR_TYPE behavior. File: 'FILE_NAME'.</p>	<p>Ошибка! Директива #extension, указанная для всех (all) WebGL-расширений в файле FILE_NAME, не поддерживает поведение BEHAVIOR_TYPE.</p>
<p>Syntax Error. ERROR_MESSAGE. File: FILE_NAME, line: LINE_NUMBER, column: COL_NUMBER.</p>	<p>Ошибка синтаксиса в строке LINE_NUMBER, столбце COL_NUMBER при парсинге шейдера FILE_NAME. Исходное описание ошибки приведено в ERROR_MESSAGE. В сообщении прилагается листинг кода в окрестности соответствующей строки (следует учитывать особенность pegjs-парсеров, указывающих чуть далее места, вызвавшего ошибку). В файле FILE_NAME объявлена функция NAME, которая нигде не используется.</p>
<p>Warning! Function 'NAME' is declared in [include]file FILE_NAME, but never used. Warning! Include file 'FILE_NAME' not used in any shader, would be omitted! Warning! Unused import token 'NAME' in include file 'FILE_NAME'. Warning! Variable 'NAME' is declared in include file FILE_NAME, but never used.</p>	<p>Include-файл FILE_NAME не используется ни в одном из шейдеров, поэтому будет исключен из закомпилированной версии. Идентификатор с именем NAME импортируется в include-файле FILE_NAME, но нигде не используется. В файле FILE_NAME объявлена переменная NAME, которая нигде не используется.</p>

Работа в команде с использованием Git

26.1 Общие сведения

Для организации командной работы над проектом может быть использована система контроля версий файлов Git. Использование Git имеет ряд преимуществ перед другими способами организации совместной работы:

- сохранение полной истории изменений файлов с возможностью возврата к предыдущим версиям
- синхронизация изменений между пользователями и автоматическое слияние изменений
- возможность работы с бинарными файлами большого объёма

Git - распределенная система, и каждый разработчик или дизайнер имеет собственный локальный репозиторий (хранилище). Синхронизация между локальными репозиториями может осуществляться через центральное “общее” хранилище, которое можно разместить на специально выделенной для этой цели машине (сервере). К серверу может быть организован доступ по протоколу SSH.

Хотя для Git существует множество графических утилит, упрощающих работу начинающих пользователей, здесь мы рассмотрим работу со штатной консольной утилитой, вызываемой командой `git`.

26.2 Типичный рабочий процесс

1. В ходе работы в локальных репозиториях создаются, изменяются или удаляются файлы.
2. По завершении некоторого логического этапа работы возникает необходимость фиксации изменений (коммит) и/или синхронизации с коллегами.
3. Проводится подготовка файлов к коммиту - учет измененных, новых и удаленных файлов, а также сброс изменений.
4. Осуществляется коммит.

5. Локальные изменения загружаются в общее хранилище и становятся доступными для коллег.

Далее описывается ограниченный набор команд Git, рекомендуемых к использованию при создании приложений и графических ресурсов.

Перед выполнением команд необходимо перейти в репозиторий, например:

```
> cd ~/blend4web
```

26.3 Индивидуальные настройки

Новый пользователь может установить имя и почтовый адрес командами:

```
> git config --global user.name "Ivan Petrov"  
> git config --global user.email ipetrov@blend4web.com
```

Установленные данные будут использоваться в логе изменений.

26.4 Проверка статуса

Перед началом, в процессе или после выполнения любых операций рекомендуется проверять текущее состояние репозитория.

Проверить статус можно командой:

```
> git status
```

Результат команды `git status`, если все коммиты проведены и нет новых файлов:

```
# On branch master  
# Your branch is ahead of 'origin/master' by 2 commits.  
#  
nothing to commit (working directory clean)
```

Возможный результат команды `git status`, если имеются изменения. Например, файлы `apps_dev/firstperson/firstperson.js` и `doc_src/git_short_manual.rst` изменены, и создан новый файл `123.txt`:

```
# On branch master  
# Changes not staged for commit:  
#   (use "git add <file>..." to update what will be committed)  
#   (use "git checkout -- <file>..." to discard changes in working directory)  
#  
#       modified:   apps_dev/firstperson/firstperson.js  
#       modified:   doc_src/git_short_manual.rst  
#  
# Untracked files:  
#   (use "git add <file>..." to include in what will be committed)
```

```
#  
#   123.txt  
no changes added to commit (use "git add" and/or "git commit -a")
```

26.5 Перед коммитом

26.5.1 Проверка изменений (текстовых файлов)

Перед совершением коммита в случае текстовых файлов рекомендуется просмотреть внесенные изменения.

Проверить, что изменилось, во всей директории:

```
> git diff
```

или только в определенном файле:

```
> git diff apps_dev/firstperson/firstperson.js
```

Возможный результат команды `git diff` для текстового файла:

```
diff --git a/apps_dev/firstperson/firstperson.js b/apps_dev/firstperson/firstperson.js
index 4381c99..44b3b15 100644
--- a/apps_dev/firstperson/firstperson.js
+++ b/apps_dev/firstperson/firstperson.js
@@ -557,8 +557,9 @@ function enable_camera_control_mode() {
        var cam_view_down = CAMERA_MOVE_UPDOWN * (Math.sin(_passed_time) - 1);

        b4w.camera.translate_view(obj, 0, cam_view_down, cam_view_angle);
-    } else
+    } else {
        b4w.camera.translate_view(obj, 0, 0, 0);
+    }
}
```

26.5.2 Восстановление файлов

Если файл был изменен или удален, но его необходимо восстановить (до состояния, зафиксированного последним коммитом), следует использовать команду:

```
> git checkout doc_src/git_short_manual.rst
> git checkout 123.txt
```

Внесенные изменения будут отменены, поэтому эту команду необходимо выполнять с осторожностью.

26.5.3 Посторонние файлы

Если файл значится в списке **Untracked files** (команда `git status`), но контроль версий для него не нужен, его следует удалить или переместить за пределы рабочей директории.

26.6 Подготовка к коммиту

26.6.1 Добавление файлов

Если изменения устраивают, добавить нужные измененные и/или новые файлы для коммита:

```
> git add apps_dev/firstperson/firstperson.js  
> git add 123.txt
```

Снова проверить статус:

```
> git status
```

Возможный результат команды `git status` после добавления некоторых файлов командой `git add`:

```
# On branch master  
# Changes to be committed:  
#   (use "git reset HEAD <file>..." to unstage)  
#  
#       new file:   123.txt  
#       modified:  apps_dev/firstperson/firstperson.js  
#  
# Changes not staged for commit:  
#   (use "git add <file>..." to update what will be committed)  
#   (use "git checkout -- <file>..." to discard changes in working directory)  
#  
#       modified:  doc_src/git_short_manual.rst  
#
```

Видно, что для коммита добавлены файлы `apps_dev/firstperson/firstperson.js` и `123.txt`, а файл `doc_src/git_short_manual.rst` остался недобавленным. Для упрощения работы рекомендуется либо добавлять такие файлы для коммита, либо отбрасывать их изменения командой `git checkout`.

26.6.2 Удаление файлов

Некоторые файлы могут быть отмечены как удаленные из Git после выполнения команды `git status`, например:

```
# On branch master
# Your branch is ahead of 'origin/master' by 2 commits.
#
# Changes not staged for commit:
#   (use "git add/rm <file>..." to update what will be committed)
#   (use "git checkout -- <file>..." to discard changes in working directory)
#
#       deleted:    123.txt
#
no changes added to commit (use "git add" and/or "git commit -a")
```

В таком случае, если удаление файла должно быть зафиксировано (т.е. войти в коммит), выполнить команду `git rm`, например:

```
> git rm 123.txt
```

Если же файл был удален по ошибке, и его необходимо вернуть, нужно использовать команду `git checkout`.

26.7 Коммит

Выполнить коммит командой:

```
> git commit
```

Появится окно текстового редактора (например, `nano` или `vim`), в котором нужно ввести комментарий к коммиту на английском языке.

```
GNU nano 2.2.6                                         File: .git/COMMIT_EDITMSG

My commit message
# Please enter the commit message for your changes. Lines starting
# with '#' will be ignored, and an empty message aborts the commit.
# On branch master
# Changes to be committed:
#   (use "git reset HEAD <file>..." to unstage)
#
#       new file:   123.txt
#       modified:   apps_dev/firstperson/firstperson.js
#
# Changes not staged for commit:
#   (use "git add <file>..." to update what will be committed)
#   (use "git checkout -- <file>..." to discard changes in working directory)
#
#       modified:   doc_src/git_short_manual.rst
#
^G Get Help          ^O WriteOut        ^R Read File      ^Y Prev Page
^X Exit             ^J Justify         ^W Where Is       ^V Next Page
```

Сохранить изменения и выйти из редактора (в **nano** Ctrl+O, затем Ctrl+X; в **vim** ZZ, или ESC :wq).

После совершения коммита рекомендуется снова проверить статус. Коммит совершен правильно, если команда `git status` отображает `nothing to commit, working directory clean`.

26.8 Синхронизация между репозиториями

26.8.1 Из удаленного - в локальный

После того как все коммиты сделаны, необходимо загрузить изменения из удаленного (“общего”) репозитория в локальный:

```
> git pull
```

Результат команды `git pull`, если в удаленном репозитории нет изменений:

```
Already up-to-date.
```

Результат команды `git pull`, если в удаленном репозитории были изменения, и синхронизация прошла успешно:

```
remote: Counting objects: 151, done.
remote: Compressing objects: 100% (101/101), done.
remote: Total 102 (delta 74), reused 0 (delta 0)
Receiving objects: 100% (102/102), 69.77 MiB | 4.87 MiB/s, done.
Resolving deltas: 100% (74/74), completed with 32 local objects.
From liixer:blend4web
      dbf3877..9f9700c master      -> origin/master
Updating dbf3877..9f9700c
Fast-forward
  apps_dev/firstperson/firstperson.js          |  338 +-+
  .../location_agriculture.blend              | Bin 25601626 -> 25598644 bytes
  ...
  src/controls.js                            |   38 +-+
  src/data.js                                |     5 +
  src/physics.js                            |  185 +-+
19 files changed, 1452 insertions(+), 2767 deletions(-)
create mode 100644    deploy/assets/location_agriculture/textures/rotonda_02_diff.png
```

При желании можно посмотреть, какие изменения были внесены коллегами, командой:

```
> git diff dbf3877..9f9700c
```

Параметр этой команды - в данном случае `dbf3877..9f9700c` - указывает, между какими именно коммитами просматриваются изменения. Этот параметр удобно выделить в результатах команды `git pull` и вставить щелчком мыши (средняя кнопка) в консоли в нужном месте.

Также можно просмотреть лог изменений:

```
> git log
```

Команда `git pull` не всегда приводит в успешной синхронизации. Результат команды `git pull` в случае наличия конфликтов:

```
remote: Counting objects: 11, done.  
remote: Compressing objects: 100% (6/6), done.  
remote: Total 6 (delta 5), reused 0 (delta 0)  
Unpacking objects: 100% (6/6), done.  
From liixer:blend4web  
    ff715c2..dbf316a  master      -> origin/master  
warning: Cannot merge binary files: blender/landscape_objects/Fallen_tree.blend (...)  
  
Auto-merging blender/landscape_objects/Fallen_tree.blend  
CONFLICT (content): Merge conflict in blender/landscape_objects/Fallen_tree.blend  
Automatic merge failed; fix conflicts and then commit the result.
```

Порядок действий при возникновении конфликтов описан далее.

26.8.2 Из локального - в удаленный

Затем нужно загрузить изменения из локального репозитория в удаленный (“общий”), чтобы локальные изменения стали доступными для коллег.

```
> git push
```

Результат команды `git push`, если в удаленном репозитории уже есть все локальные изменения:

```
Everything up-to-date
```

Результат команды `git push`, если синхронизация прошла успешно:

```
Counting objects: 25, done.  
Delta compression using up to 8 threads.  
Compressing objects: 100% (14/14), done.  
Writing objects: 100% (14/14), 1.23 KiB, done.  
Total 14 (delta 11), reused 0 (delta 0)  
To gftteam@lixer:blend4web.git  
    9f9700c..fa1d6ac  master -> master
```

Результат команды `git push`, если синхронизация не прошла, потому что сначала не была выполнена команда `git pull`:

```
To gftteam@lixer:blend4web.git  
! [rejected]      master -> master (non-fast-forward)  
error: failed to push some refs to 'gftteam@lixer:blend4web.git'  
To prevent you from losing history, non-fast-forward updates were rejected  
Merge the remote changes (e.g. 'git pull') before pushing again. See the  
'Note about fast-forwards' section of 'git push --help' for details.
```

Необходимо выполнить команду `git pull`.

Изменения, загруженные в центральный репозиторий, могут быть получены другими участниками разработки с помощью команды `git pull`.

26.9 Разрешение конфликтов

26.9.1 Общие сведения

Конфликты синхронизации происходят, если выполнены оба условия

1. один и тот же файл был изменен как в локальном, так и в удаленном репозитории, и
2. автоматическое слияние изменений не произошло, поскольку изменения находятся в одном и том же месте файла.

Типичные случаи:

1. бинарный файл (текстура, blend-файл) независимо изменен двумя участниками разработки
2. в текстовой файл в одной и той же строке были внесены разные изменения
3. один участник разработки изменил файл, а другой - переместил его и т.п.

Хотя конфликты синхронизации - нормальное явление, слишком частое их возникновение замедляет работу. Рекомендуется ставить коллег в известность о начале работ с общими бинарными файлами, а также чаще проводить синхронизацию. Необходимо эффективно распределять работу между участниками разработки, чтобы таких общих файлов было как можно меньше. Этого можно добиться, в частности, подключением всех ресурсов сцены (*linking*) из отдельных blend-файлов в один мастер-файл.

26.9.2 Порядок действий

Не рекомендуется производить какие-либо действия с файлами (изменять, удалять), пока репозиторий находится в конфликтном состоянии.

Первое что необходимо сделать - выполнить команду `git status`.

```
# On branch master
# Your branch and 'origin/master' have diverged,
# and have 7 and 1 different commit each, respectively.
#
# Unmerged paths:
#   (use "git add/rm <file>..." as appropriate to mark resolution)
#
#       both modified:    blender/landscape_objects/Fallen_tree.blend
#
no changes added to commit (use "git add" and/or "git commit -a")
```

Список конфликтующих файлов отображен в разделе **Unmerged paths**.

Дальнейший порядок действий различен для бинарных и текстовых файлов.

26.9.3 Бинарные файлы

На данном этапе конфликтующие бинарные файлы находятся в том состоянии, в котором они находились в локальном репозитории до попытки синхронизации. Файлы полностью функциональны (например, открываются графическими редакторами).

В случае конфликта бинарных файлов необходимо выяснить с коллегами или самостоятельно, какую из версий оставить, а какую отбросить. Выбор осуществляется командой `git checkout`.

Выбрать локальную версию файла (`- --ours`). Его можно открыть и убедиться в этом.

```
> git checkout --ours blender/landscape_objects/Fallen_tree.blend
```

Выбрать удаленную версию файла (`- --theirs`). Его можно открыть и убедиться в этом.

```
> git checkout --theirs blender/landscape_objects/Fallen_tree.blend
```

Снова выбрать локальную версию файла (`- --ours`).

```
> git checkout --ours blender/landscape_objects/Fallen_tree.blend
```

В итоге необходимо остановиться на нужной версии файла. При угрозе потери работы можно сохранить отбрасываемую версию файла вне репозитория.

26.9.4 Текстовые файлы

На данном этапе в конфликтующие текстовые файлы Git'ом вносятся как локальные, так и удаленные изменения одновременно, в особом формате. Такие текстовые файлы как правило, не работоспособны.

Пример. Один участник разработки изменил имя сцены с “Blue Lizard” на “Green Lizard” в файле приложения и загрузил изменения в центральный репозиторий. Другой участник разработки изменил в той же строке “Blue Lizard” на “Red Lizard”, совершил коммит и выполнил команду `git pull`. В результате именно на этого участника ложится ответственность по разрешению конфликта. В его файле приложения будут находиться строки:

```
<<<<< HEAD
        "name": "Red Lizard",
=====
        "name": "Green Lizard",
>>>>> 81bf4e2d5610d500ad4d2a2605ee7e61f759f201
```

В случае конфликта текстовых файлов можно поступить следующим образом. Файлы, содержащие исходный код, необходимо отредактировать с учетом или без учета внесенных обеими сторонами изменений. В то же время экспортированные текстовые файлы сцен (заканчивающиеся на `.json`) проще повторно экспортировать.

26.9.5 Корректирующий коммит

После выбора нужных файлов или редактирования изменений, добавить их для коммита:

```
> git add blender/landscape_objects/Fallen_tree.blend  
> git status
```

Возможный результат выполнения `git status` после добавления конфликтующих файлов для коммита:

```
# On branch master  
# Your branch and 'origin/master' have diverged,  
# and have 7 and 1 different commit each, respectively.  
#  
nothing to commit (working directory clean)
```

Выполнить коммит, комментарий рекомендуется оставить предложенный по умолчанию:

```
> git commit  
> git status  
  
# On branch master  
# Your branch is ahead of 'origin/master' by 8 commits.  
#  
nothing to commit (working directory clean)
```

Конфликты разрешены, изменения из удаленного репозитория успешно применены в локальном репозитории. Теперь изменения в локальном репозитории, - включающие только что разрешенный конфликт, - можно загрузить в удаленный репозиторий командой `git push`.

26.10 Тэги

Тэги (метки) предназначены для указания на определенный коммит, например, с целью обозначения стабилизированной версии продукта.

Просмотреть список тэгов:

```
> git tag
```

Создать тэг для релиза от 3 июня 2013 г., указывающий на коммит со стабильной версией проекта:

```
> git tag R130603 67bb597f7ed1643ed0220d57e894f28662e614e5
```

Просмотреть информацию о коммите тэга:

```
> git show --shortstat R130603
```

Перейти к тэгу...

```
> git checkout R130603
```

...и вернуться:

```
> git checkout master
```

Синхронизировать тэги с удаленным репозиторием:

```
> git push --tags
```

Удалить тэг (при ошибочном создании):

```
> git tag -d R130603
```

26.11 Другие полезные команды

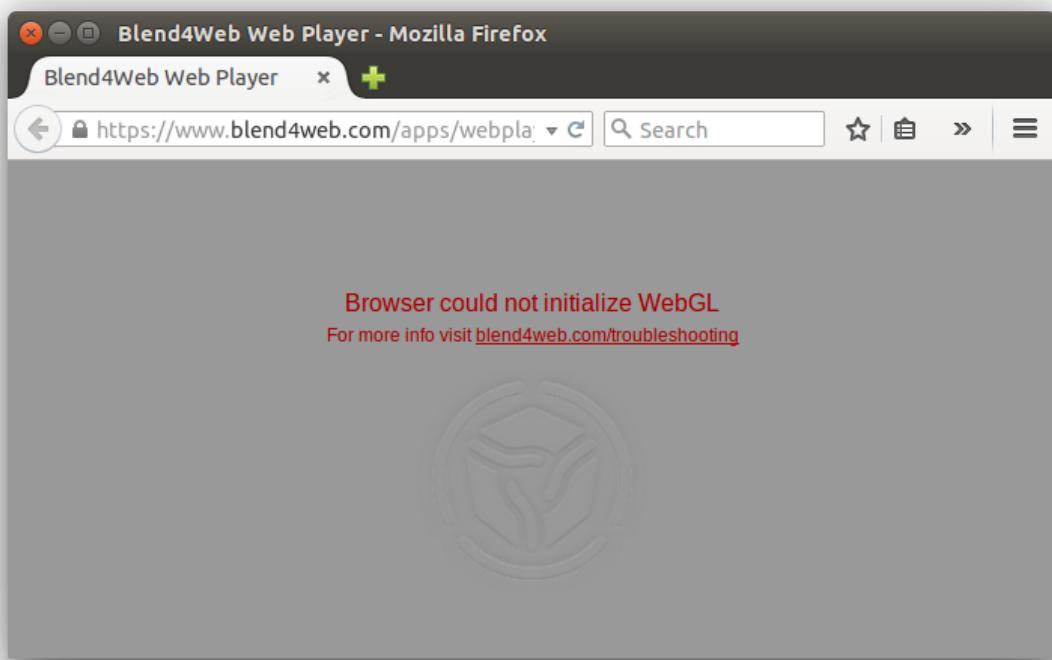
Просмотреть лог за январь 2012 г, показывать имена файлов, без коммитов слияния:

```
> git log --after={2012-01-01} --before={2012-01-31} --name-only --no-merges
```

Проблемы и решения

27.1 Проблемы при запуске движка

1. Появляется сообщение “*Browser could not initialize WebGL.*”



Следует выполнить действия, описанные в разделе *Ошибка инициализации WebGL*.

2. Видны элементы интерфейса или пустой экран, но сцена не отображается. При этом тестовый сайт <http://get.webgl.org/> и другие WebGL приложения работают корректно.

Вероятные причины:

- Не используется локальный веб-сервер или браузер не настроен для работы с локальными ресурсами. См. раздел *Настстройка браузера для загрузки локальных ресурсов*.

- Файлы ресурсов, которые пытается загрузить движок, были перемещены или удалены.
- Используются старые версии драйверов.
- Используются открытые драйвера, не обеспечивающие поддержку WebGL.

Для пользователей Linux - ввиду неполной реализации OpenGL стека в драйверах с открытым кодом в настоящий момент рекомендуется использовать проприетарные драйверы текущей версии для графических процессоров Nvidia и AMD.

- Используется устаревшая операционная система, такая как Windows XP.

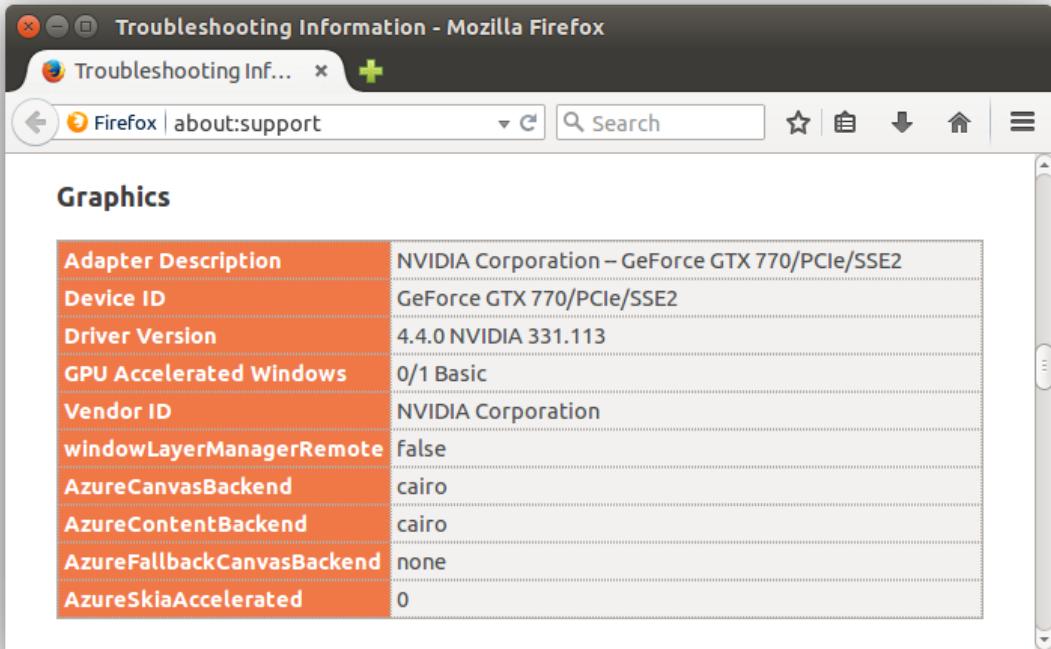
27.2 Ошибка инициализации WebGL

Сайт <http://get.webgl.org/> при просмотре в браузерах Chrome или Firefox последней версии сообщает о проблемах. Что делать?

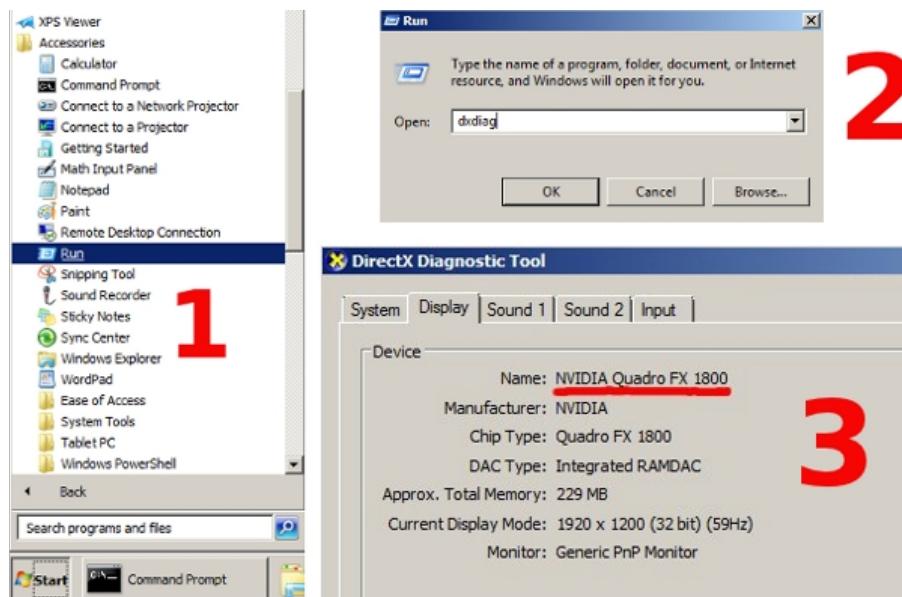
1. Установить доступные обновления для системы (для Windows см. [инструкцию](#)). В случае Windows установить последнюю версию [DirectX](#). Перезагрузить систему.
2. Рекомендуется проводить своевременное обновление драйверов для графических карт. Чтобы определить тип и производителя карты, можно ввести **about:gpu** (или **chrome://gpu**) в адресную строку браузера Chrome...

Driver Information	
Initialization time	66
Sandboxed	true
GPU0	VENDOR = 0x10de, DEVICE= 0x1184
Optimus	false
AMD switchable	false
Driver vendor	NVIDIA
Driver version	331.113
Driver date	
Pixel shader version	4.40
Vertex shader version	4.40
Machine model name	
Machine model version	
GL_VENDOR	NVIDIA Corporation
GL_RENDERER	GeForce GTX 770/PCIe/SSE2

или Firefox...



Для операционных систем семейства Windows можно воспользоваться средством диагностики DirectX **dxdiag**.

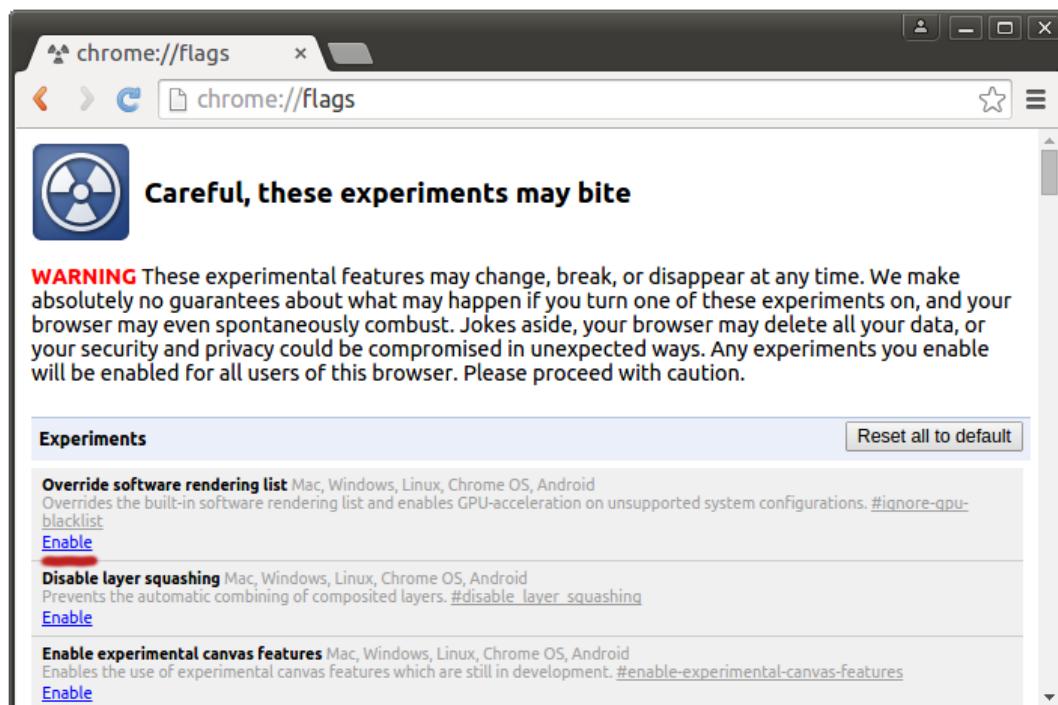


Необходимо загрузить драйверы с соответствующего центра поддержки (например, Intel, Nvidia, AMD/ATI). После установки драйверов перезагрузить систему.

- Если в результате вышеперечисленных действий инициализировать рендеринг не удается (или нет возможности обновить систему), можно попробовать изменить настройки браузера.

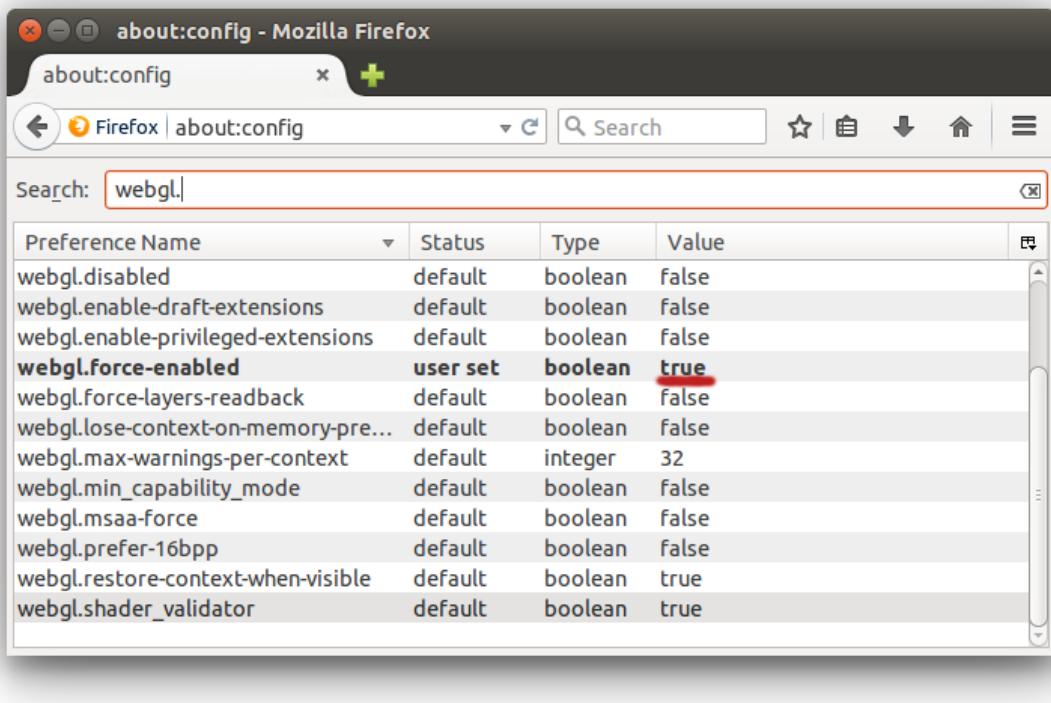
B Chrome:

Ввести **about:flags** (или **chrome://flags**) в адресную строку браузера, нажать **Включить (Enable)** под опцией **Переопределение списка программного рендеринга (Override software rendering list)** и перезапустить браузер.



B Firefox:

Ввести **about:config** в адресную строку браузера, найти параметр **webgl.force-enabled** и переключить его двойным щелчком мыши из **false** в **true**.



Замечания к релизам

28.1 v15.04

28.1.1 Новый функционал

- Добавлена поддержка ключей деформации (*Shape Keys*).

Была осуществлена поддержка *ключей деформации* (*Shape Keys*) в Blender'e. Для применения ключей в модуль `geometry.js` была добавлена функция `set_shape_key_value`. Пример использования данного функционала доступен в приложении *Code Snippets*.

- Добавлена поддержка фона с использованием *Horizon Color* и *Zenith Color*.

Теперь поддерживается настройка фона сцены с использованием *Horizon Color* и *Zenith Color*, а также флагов *Paper Sky*, *Blend Sky* и *Real Sky*.

- Добавлена поддержка ноды *Gamma*.

Силами команды Blend4Web в Blender v2.74 была добавлена поддержка ноды *Gamma*. Теперь эта нода также доступна и в движке.

- Улучшения в инструменте для создания подписей *Anchors*.

В настройках в Blender добавлена возможность ограничивать размеры подписей. Реализована поддержка динамической загрузки/выгрузки подписей. Добавлена возможность скрытия подписей с помощью штатных методов `show()`/`hide()` или в инструменте NLA Script.

- Оптимизация шейдеров.

Улучшение шейдерного компилятора. Добавлен следующий функционал: оптимизация локальных переменных, удаление избыточных фигурных скобок. Улучшение производительности нодовых материалов.

- Оптимизация физического движка *uranium.js*.

Модуль с физическим движком загружается только в тех случаях, когда это необходимо. Размер модуля сокращён на 20%.

- *Расширены инструменты для отладки физики.*

Добавлен метод `physics_stat()` модуля `debug` для вывода статистики о количестве физических объектов (по типам), сложности геометрии и прочей информации. Для визуальной оценки производительности физической симуляции добавлена возможность вывода числа итераций: Physics FPS (активируется в модуле `config`).

- *Новый API для присоединения объектов к камере независимо от соотношений сторон и углов обзора камеры.*

Реализуется с помощью метода `append_stiff_viewport()` модуля `constraints`.

- *Новый модуль для применения трансформаций “tsr.js”.*

Позволяет применять произвольные трансформации к объектам с помощью универсальных векторов TSR, включающих в себя перемещение, масштабирование и поворот. Данные векторы могут быть использованы в качестве более удобной и эффективной замены матриц.

- *Добавлена возможность игнорирования директорий для конвертера ресурсов.*

Была добавлена возможность игнорирования директорий для `конвертера ресурсов`. Для этого надо разместить в этой директории файл с именем `.b4w_no_conv`.

28.1.2 Изменения

- *Изменен внешний вид документации API движка. Появились ссылки для быстрого перехода к описаниям методов и свойств.*
- *Скелетная анимация учитывает взаимное расположение арматуры и объектов.*

Теперь поддерживается родное поведение Blender. Больше нет необходимости размещать арматуру и анимируемый объект в одной точке и с одинаковым поворотом и масштабом.

- *Объектная анимация теперь позволяет анимировать независимо перемещение, поворот и масштаб.*

В объектной анимации теперь нет жесткой простановки ключей на все каналы, что позволяет сохранять изначальные значения в незадрнутых каналах или изменять их через API.

- *Опциональная поддержка фона.*

Активация фона или неба регулируется опцией `Sky Settings > Render Sky` на вкладке `World`. По умолчанию отключено.

- *Физический движок Uranium теперь состоит из двух модулей.*

Для работы физики теперь требуется два модуля: код движка `uranium.js` и файл инициализации памяти `uranium.js.mem`. Оба модуля должны находятся в одной директории.

- Эффект *Glow* и все связанные с ним компоненты движка были переименованы в *Outline*.

Новое имя более ясно описывает суть эффекта: подсвечивание объекта на его границах.

- Применение ограничителя движения при помощи функции `"append_semi_stiff_cam"` модуля `"constraints.js"` теперь разрешено только для камер типа `"EYE"`.
- Изменения локального сервера разработки в *Blender*.

В качестве *локального сервера* теперь вместо стандартного `SimpleHTTPServer` на Python используется `Tornado Web Server`. Новый сервер имеет высокую производительность, а также содержит расширенные настройки для отключения кэширования браузерами.

- Добавлено управление слайдерами с помощью клавиатуры в приложении *Viewer*.

Теперь управление слайдером осуществляется при помощи клавиш `<` и `>`.

- Изменения в методе `"update_object_animation"` модуля `"animation.js"`.

Добавлен необязательный параметр `"force_update"`, заставляющий обновлять состояние анимированного объекта даже при остановленной анимации.

- Изменения API модуля `mouse.js`.

Добавлены методы: `enable_mouse_hover_outline`, `disable_mouse_hover_outline`.

Следующие методы объявлены устаревшими и будут удалены в последующих релизах: `enable_mouse_hover_glow`, `disable_mouse_hover_glow`.

- Изменения API модуля `scenes.js`.

Добавлены методы: `outlining_is_enabled`, `set_outline_intensity`, `get_outline_intensity`, `apply_outline_anim`, `apply_outline_anim_def`, `clear_outline_anim`, `set_outline_color`, `get_outline_color`.

Следующие методы объявлены устаревшими и будут удалены в последующих релизах: `set_glow_intensity`, `get_glow_intensity`, `apply_glow_anim`, `apply_glow_anim_def`, `clear_glow_anim`, `set_glow_color`, `get_glow_color`.

- В панели *Object > Blend4Web* изменены настройки.

Добавлен флаг *Enable Outline*, который устанавливает возможность использования эффекта *подсветки контура* на данном объекте. Так же добавлен флаг *Ouline on Select*, устанавливающий активацию анимации подсветки при выделении объекта (раньше данное поведение определялось флагом *Selectable*).

- В панель *Scene > Blend4Web* добавлены новые настройки.

Добавлено поле *Enable Object Outlining*, которое управляет возможностью подсветки контура объектов. Аналогично, новое поле *Enable Object Selectable* управляет возможностью выделения объектов.

- Теперь свойства объекта *Apply Scale*, *Apply Modifiers*, *Export Vertex Animation*, *Export Edited Normals* и *Export Shape Keys* взаимно исключают друг друга.
- Изменение API модулей.

В API модуля *util.js* добавился метод *is_arbature*, позволяющий проверить, является ли данный объект объектом типа **ARMATURE**.

В API модуля *constraints.js* добавился метод *get_parent*, возвращающий для данного объекта родительский объект.

28.1.3 Исправления

- Исправлена ошибка, приводящая к исчезновению описания объекта в модуле “anchors.js”.
- Исправлена ошибка запекателя скелетной анимации при наличии арматурных объектов на скрытых слоях.
- Исправлено поведение камеры при использовании функции *append_semi_stiff_cam* модуля “constraints.js”.

Исправлено вертикальное выравнивание камеры относительно родительского объекта. Также теперь учитывается начальная ориентация камеры, что может потребовать корректировки лимитов вращения, передаваемых в эту функцию.

- Исправлена ошибка обновления плейлиста, если он оказывался пустым.
- Исправлено некорректное поведение физических объектов при удалении хотя бы одного из них со сцены.
- Исправлен баг при наличии нулевого масштабирования у объекта, размноженного через *DupliGroup*.
- Исправлена ошибка компиляции шейдера воды на Windows и некоторых мобильных устройствах.
- Исправлена ошибка при наличии дублированных ключей анимации.
- В анимации *NLA* теперь можно использовать акторы из разных файлов, имеющие одинаковое имя.

- Устранено размножение обработчиков перемещения мыши при многократном вызове “pointerlock”.
- Исправлен тип прозрачности “Alpha Sort” для динамических объектов.
- Исправлена ошибка сборки аддона, приводящая к неработоспособности на системах Windows без установленного C++ 2010 runtime.
- Исправлен баг рендеринга billboard-объектов на iPad.

28.2 v15.03

28.2.1 Новый функционал

- Инструмент для создания подписей к трёхмерным объектам.

Теперь имеется возможность создавать подписи к трёхмерным объектам (“якори”). Данные якори могут быть трёх различных типов: Аннотации (Annotation) - используется информация из мета-тегов, назначенных на объектах, Элементы (Custom Element) - в качестве якоря можно назначить произвольный HTML-элемент с текущей веб-страницы и Общий (Generic) - якорь, положение которого можно определить с помощью API модуля anchors.

- Анимация и возможность управления через API значениями нод типа Value и RGB.

Теперь помимо анимации нод типа Value возможна анимация нод RGB. Так же в модуле objects.js появились соответствующие функции для изменения значений таких нод.

- Новое приложение “Code Snippets”.

Это [приложение](#) было создано для более быстрого и легкого просмотра уроков по функционалу движка. В нем также можно ознакомиться со скриптами, составляющими основу логики работы приведенных уроков. Приложение доступно из файла index.html, находящегося в корне Blend4Web SDK.

- Новые функции управления эффектом Glow.

В модуль scenes были добавлены новые API: `get_glow_intensity()` и `get_glow_color()`.

- Улучшения в просмотрщике сцен Viewer.

Изменён дизайн кнопки перехода на “домашнюю” сцену, добавлена кнопка “All objects selectable”, позволяющая отключить автоматическое назначение свойства “Selectable” для всех объектов на сцене. Так же в просмотрщике теперь можно увидеть суммарное количество шейдеров, выполняющихся на загруженной сцене.

- Динамическое копирование объектов сцены.

Появилась возможность динамического *копирования и удаления* объектов сцены (инстансинг).

- *Обработка ошибок, связанных со специальной нодой B4W_PARALLAX.*

Добавлена обработка некорректного использования ноды B4W_PARALLAX в виде *некритической ошибки экспорта*.

- *Новые опции в сборщике приложений.*

В сборщике приложений появились новые опции `-j` и `-c`, которые позволяют добавлять в исключения, соответственно, скрипты и стили, не требующие компиляции.

- *Экспериментальный Blend4Web рендер движок.*

Включается в настройках аддона: флаг “Register Blend4Web render engine (Experimental)”. Предназначен для упрощения редактирования свойств объектов. Упрощает интерфейс, отключив неподдерживаемые аддоном панели. В данный момент в режиме Blend4Web рендера отсутствует возможность редактирования шейдерного нодового дерева.

28.2.2 Изменения

- *Изменение точки отсчета лимитов перемещения камеры.*

Выставление лимитов перемещения камеры через API теперь полностью соответствует *отсчету значений* в системе координат движка. Выставление в Blender'е горизонтальных лимитов для камеры типа TARGET в пространстве координат World Space изменилось, поэтому может потребовать перенастройки для старых сцен.

- *Обновление документации по настройкам камеры.*
- *Горизонтальные и вертикальные лимиты вращения камеры теперь полностью независимы друг от друга.*
- *Изменения API модуля camera.js.*

API модуля camera.js претерпело ряд изменений.

Добавились новые методы: `is_target_camera`, `is_eye_camera`, `is_hover_camera`, `rotate_camera`, `rotate_target_camera`, `rotate_eye_camera`, `rotate_hover_camera`, `get_camera_angles`, `hover_cam_set_translation`.

Объявлены устаревшими и будут удалены в последующих релизах: `rotate_pivot`, `rotate_hover_cam`, `rotate`, `set_eye_params`, `get_angles`, `translate_hover_cam_v`, `set_hover_cam_angle`.

Методы `set_ortho_scale` и `get_ortho_scale` при применении их к камере отличного от Orthographic типа теперь выводят сообщение об ошибке. Также изменено поведение метода `get_hover_angle_limits`,

который теперь возвращает угловые лимиты для HOVER камеры в формате [down, up] вместо [up, down] как было раньше.

- *Значительно оптимизирована система освещения.*

Многие свойства источников света теперь обрабатываются на этапе компиляции. Теперь нет ограничения в 4 источника для некоторых мобильных устройств.

- *Изменён способ вёрстки приложений, использующих модуль app.*

При инициализации приложения с использованием модуля app размеры создаваемого элемента <canvas> теперь полностью определяются размерами элемента-контейнера. Это нужно иметь в виду, например, в случае использования элемента <div> в качестве контейнера, поскольку по умолчанию его высота равна нулю. Задать правильные размеры контейнера можно с использованием CSS, например с помощью inline-стиля. Также, при любых преобразованиях контейнера теперь необходимо вызывать функцию модуля *app.resize_to_container()*. Аналогичного эффекта можно добиться, если задать опцию *autoresize* при инициализации приложения (в функции *app.init()*). Поддержка низкоуровневого метода изменения размеров элемента <canvas> с использованием функции *main.resize()* сохранена.

- *Функция модуля app enable_controls() теперь вызывается без параметров.*
- *Удалены устаревшие методы API.*

Модуль *textures.js*: *stop_video*.

Модуль *scenes.js*: *add_object*, *get_screen_scenes*, *set_light_pos*, *set_light_direction*, *set_dir_light_color*, *get_lights_names*, *remove_all*, *check_collision*, *check_ray_hit*, *get_appended_objs*, *get_object_by_empty_name*.

Модуль *physics.js*: *set_character_dist_to_water*.

Модуль *material.js*: *set_batch_param*, *set_max_bones*, *max_bones*.

Модуль *main.js*: *set_shaders_dir*, *set_texture_quality*.

Модуль *data.js*: *get_bpy_world*.

Модуль *controls.js*: *sensor_make_positive*, *sensor_make_negative*.

Модуль *camera.js*: *change_eye_target_dist* (также удалена устаревшая константа *MS_CONTROLS*).

- *Новые условия для изменения положения объектов через API.*

Функции модуля *transform.js*, связанные с изменением положения объекта, доступны только для *динамических объектов*.

- *Разрешено использование ноды типа TEXTURE с отсутствующей текстурой.*

Отображение нодового материала в этом случае целиком соответствует Blender'у.

- *Обновление механизма проверки совместимости версий экспорттированных файлов и движка.*

Движок будет сигнализировать о несовместимости загружаемой сцены специальными *сообщениями в консоли браузера*.

- *Свойство объекта “Do not batch” переименовано в “Force Dynamic Object”*

Данная опция сообщает движку о том, что объект обязан быть динамическим независимо от других настроек. Теперь она носит более понятное имя.

28.2.3 Исправления

- *Исправлена ошибка автоповорота камеры в веб-плеере.*
- *Исправлена ошибка полноэкранного режима в веб-плеере.*
- *Исправлена ошибка определения угловых координат камеры в некоторых положениях.*
- *Исправлена ошибка автоворота камеры при наличии горизонтальных лимитов, ограничивающих поворот.*
- *Исправлена ошибка множественного добавления специальных нод в blend-файл.*
- *Исправлена ошибка замены материала на объекте функцией “inherit_material” из модуля “material.js”.*
- *Исправлены ошибки при отражении зеркальной поверхностью объекта, имеющего свойства, измененные с использованием модуля “material.js”.*
- *Исправлена генерация сферических WIREFRAME-объектов в режиме отладки сцены.*
- *Исправлена процедура оптимизации нод типа TEXTURE в нодовом материале.*
- *Исправлена работа опции “Clamp” в ноде типа MixRGB(Linear Light).*
- *Исправлена ошибка, возникающая при экспорте, если на сцене присутствуют объекты с разделяемым мешем и настройками “Apply Scale”, “Apply Modifiers”, “Export Vertex Animation” или “Export Edited Normals”.*
- *Исправлена ошибка при использовании опции “Blend4Web > Preserve global orientation and scale” на ряде мобильных устройств.*
- *Исправлена ошибка рендеринга тумана в отдельных версиях Chrome и Firefox под Windows.*

28.2.4 Известные проблемы

- *Проблемы с обновлением аддона в Blender.*

Настоятельно не рекомендуется устанавливать новую версию аддона Blender, не удалив предварительно старую версию, особенно это касается систем Windows.

- *Не работают видеотекстуры в Firefox при экспорте сцены в формате HTML.*

28.3 v15.02

28.3.1 Новый функционал

- *Добавлена возможность запуска локального сервера разработки непосредственно из Blender.*

При использовании Blend4Web SDK имеется возможность запуска *локального сервера разработки*. Сервер предоставляет быстрый доступ к содержимому Blend4Web SDK, а также позволяет *автоматически открывать экспортированные сцены* в просмотрщике сцен.

- *Поддержка теней от источников света типа Spot.*

Ранее на лампах Spot тени строились так же, как для ламп типа Sun. Теперь расчет производится аналогично тому, как это делается в Blender, с учетом расхождения лучей света в пространстве.

- *Добавлена/расширена поддержка объектов типа “Metaball”, “Surface” и “Curve”.*

Была добавлена поддержка объектов типа Metaball, Surface и Curve. Объекты этих типов автоматически конвертируются в меши при экспорте. Поддержка использования объектов типа Curve в модификаторах сохранена.

- *В веб-плеер добавлены кнопки социальных сетей.*

Данные кнопки позволяют разместить ссылку на 3D сцену, а также ее описание в одной из четырёх популярных социальных сетей.

- *Добавлена возможность редактирования файла списка сцен для просмотрищика непосредственно в Blender.*

Была добавлена возможность *быстрого редактирования файла assets.json* для приложения Viewer через Blender, если используется Blend4Web SDK.

- *Добавлена возможность изготовления скриншотов в приложении Viewer.*
- *В веб-плеер добавлена опция fallback_video.*

С помощью опции `fallback_video=/path/to/video/` можно указывать видео-файл, который будет проигрываться вместо 3D контента на системах без поддержки WebGL.

- *Расширение функционала рендеринга в текстуру.*

Добавлена поддержка рендеринга сцен сразу в несколько текстур. Сцены теперь могут иметь любую степень вложенности.

- *Для билбордов появилась возможность сохранять ориентацию и масштабирование объектов в мировом пространстве.*

Для этого нужно выставить опцию *Blend4Web > Preserve global orientation and scale* на панели настроек объекта.

- *Улучшения на главной веб-странице SDK.*

Теперь на главной странице можно узнать версию используемого SDK и проверить систему на совместимость с WebGL.

- *Добавлена поддержка флага Clamp в нодах MATH и MIX_RGB.*

Функциональность была реализована сначала в Blender 2.73, и теперь в движке.

- *Улучшено качество рендеринга при отсутствии аппаратной поддержки текстуры глубины.*

Увеличены возможности рендеринга для устройств, не поддерживающих текстуру глубины. Стали доступны эффекты, такие как отражение, свечение, засветка, размытие при движении, стлаживание.

- *Добавлена новая документация.*

Добавлена документация по модулям `vec3`, `vec4`, `quat`, `mat3`, `mat4`, а также глобальному пространству имён `b4w`. Улучшен дизайн веб-страниц документации.

- *Добавлена возможность иметь несколько запущенных копий движка на одной странице.*

Несколько приложений теперь могут работать одновременно, указывая имя пространства имён при инициализации.

- *Возможность использования SDK в Apple OS X.*

В OS X теперь доступны все функции SDK, включая сборку движка и приложений, конвертацию ресурсов и генерирование документации.

- *В модуль camera добавлен метод set_trans_pivot().*

Данная функция позволяет устанавливать произвольное положение опорной точки и положение камеры типа Target.

- *В утилиту “project.py“ добавлено новое свойство “version“.*

Использование данного свойства позволяет добавлять версию к скриптом и стилям скомпилированного приложения.

28.3.2 Изменения

- Теперь в приложение можно добавлять опции с одинаковыми именами через адресную строку браузера.

В функции `get_url_params()` модуля `app` появился необязательный параметр `allow_param_array`, по умолчанию равный `false`. При назначении этому параметру `true` одинаковые опции будут объединены в массив, в противном случае будет использоваться последняя.

- Улучшения работы источников звука типа “Background Music“.

Теперь для таких источников поддерживается возможность указания параметров задержки и времени воспроизведения. Также появилась возможность зацикливать их воспроизведение.

- Специальные ноды аддона Blend4Web теперь присутствуют в blend-файле по умолчанию.

Теперь нет необходимости добавлять *специальные ноды* аддона в файл. Это доступно как при использовании Blend4Web SDK, так и при экспресс-установке аддона.

- Изменен экспорт пустых объектов типа “Mesh”.

Теперь объекты типа `Mesh`, не имеющие полигонов, экспортируются как `Empty`.

- Изменения модуля “light.js”.

Была добавлена новая функция `get_light_type`, а также теперь в функции `get_light_params` и `set_light_params` передается не имя объекта, а сам объект типа `LAMP`. Также теперь доступны для изменения через эти функции свойства `spot_blend`, `spot_size`, `distance` для источника света типа `SPOT`. Для источника света типа `POINT` теперь доступно для изменения свойство `distance`.

- Улучшение рефракции при низких настройках качества.

При активации профиля качества `LOW`, теперь используется упрощённая модель рефракции (без искажений).

- Оптимизация нодовых шейдеров.
- Теперь режим автоматического вращения камеры отключается при касании сенсорного экрана.

28.3.3 Исправления

- Исправлено поведение прелоадера для веб-плеера.

При открытии веб-плеера в новой вкладке могли наблюдаться артефакты на логотипе B4W.

- Исправлена ошибка рендеринга однокаскадных теней, связанная с возникновением жесткой неосвещенной линии по границе всего каскада.
- Исправлена ошибка, связанная с коллизией имен при линковке объектов в Blender'е.
- Исправлена ошибка с недостаточной оптимизацией приложений из состава SDK.
- Исправлена проблема экспорта мешей с плоским шейдингом (*flat shading*) в Linux x32.
- Исправлено некорректное поведение Target камеры в отдельных случаях.
- Исправлена ошибка с попыткой использования карты теней размером большим, чем поддерживаемый конкретным устройством.
- Исправлена ошибка, приводящая к снижению FPS в Firefox 35/36 под Windows при включении теней.

28.3.4 Известные проблемы

- Проблемы с обновлением аддона в Blender.

Настоятельно не рекомендуется устанавливать новую версию аддона Blender, не удалив предварительно старую версию, особенно это касается систем Windows.

- Не работают видеотекстуры в Firefox при экспорте сцены в формате HTML.

28.4 v15.01

28.4.1 Новый функционал

- Поддержка режима панорамирования для устройств с сенсорным экраном.

Движение камеры в режиме панорамирования осуществляется параллельным перемещением двух пальцев по поверхности экрана.

- Поддержка объектов типа “Text”.

Данные объекты теперь автоматически конвертируются в меш при экспорте.

- Расширение функционала инструмента NLA Script.

Добавлены новые логические слоты: Show Object и Hide Object используются для скрытия и отображения объектов, Page Redirect - для перенаправления на другие веб-страницы, Page Param - для сохранения произвольного параметра веб-страницы в выбранном числовом регистре. Упрощено использование слотов Select & Jump и

Select & Play - теперь для выделяемых объектов нет необходимости указывать свойство **Selectable**.

- *Поддержка экранов сверхвысокого разрешения (HIDPI, Retina).*

Режим HIDPI позволяет достичь существенного улучшения качества картинки на устройствах с высоким разрешением. Режим включается автоматически при запуске приложений с настройками качества ULTRA. При необходимости, высокое разрешение может быть включено и для других профилей качества.

- *Поддержка масштабирования для ортографической камеры.*

Добавлено API для изменения масштабирования камеры типа **Orthographic** (**Orthographic scale** в Blender'e).

- *В веб-плеер добавлена опция autorotate.*

Опция autorotate используется для автоматического включения вращения камеры сразу после загрузки сцены.

- *В функцию “enable_camera_controls” из аддона “app.js” добавлен облегченный режим управления с клавиатуры.*

Включить режим можно передав необязательный параметр **disable_letter_controls**. При этом будет отключено управление посредством алфавитно-цифровых клавиш клавиатуры (WASD и т.д.). Указанная функциональность может быть использована для тех случаев, когда в приложении необходимо использовать клавиши для целей, не связанных с перемещением камеры.

- *Поддержка работы с гироскопом на мобильном устройстве.*

Добавлены два сенсора для работы с гироскопом на мобильных устройствах. Первый сенсор позволяет оперировать с разностью текущего положения устройства с предыдущим, создается функцией **create_gyro_delta_sensor** из модуля “controls.js”. Второй сенсор - с текущим значением угла. Создается функцией **create_gyro_angles_sensor** из модуля “controls.js”. Стоит отметить, что все значения вычисляются в радианах. Также был создан аддон “gyroscope.js”, в котором реализовано движение камеры при угловых перемещениях устройства. С примером работы данного функционала для мобильных устройств можно ознакомиться в нашей программе “Viewer”, включив опцию “Gyroscope” в меню программы.

- *Добавлено новое свойство “Do not Render“ в настройках материала.*

Включение данного свойства позволяет не отображать на сцене все объекты, использующие этот материал.

- *Поддержка видеотекстур на IE 11 и iPhone.*

Поддержка добавлена посредством создания простого формата видео-секвенции *.seq. Более полную информацию можно посмотреть в *соответствующем разделе документации*.

- *Поддержка тега “title” в веб-плеере.*

Тег “title” для веб-плеера, необходимый для вывода названия сцены в браузере, теперь берется из JSON-файла сцены. С более подробной информацией можно ознакомиться в *соответствующем разделе документации*.

- *Поддержка мета-тегов в Blender.*

В Blender появилась возможность добавлять мета-теги к сцене и к объектам. Для сцены это “title” и “description”, для объекта “title”, “description” и “category”.

- *Добавлена возможность вызова пользовательских функций каждый кадр.*

Для возможности создания более сложных приложений в модуле “main.js” была добавлена функция `append_loop_cb`, позволяющая вызывать переданную в нее функцию каждый кадр. При этом в эту функцию каждый кадр будут передаваться два параметра: время с начала старта приложения и разница во времени между текущим кадром и предыдущим. Обе величины измеряются в секундах. Для того, чтобы больше не вызывать переданную функцию необходимо удалить ее вызовом функции `remove_loop_cb` из модуля “main.js”.

- *Добавлена анимация простого экрана загрузки приложения.*

Для создания приложения с анимированным экраном загрузки в функцию `create_simple_preloader` из модуля “preloader.js” необходимо передать опцию “`preloader_fadeout`” со значением “`true`”.

- *Добавлена возможность экспорта конвертированных медиаданных в HTML-файл.*

Теперь при экспорте в `html` имеется возможность записать конвертированные данные в файл. Для этого при экспорте необходимо задать свойство “`Export converted media`” в *опциях экспорта*.

- *Добавлена возможность использования `min50` и `dds` текстур при просмотре сцен через `webplayer`.*

Для этого необходимо *передать атрибут* “`compressed_textures`” при запуске приложения.

28.4.2 Изменения

- *Упрощена файловая структура SDK.*

Директория `external` больше не существует, всё её содержимое перемещено на уровень выше - в корневую директорию SDK. Файл со списком сцен для просмотрищика `assets.json` теперь находится в директории `apps_dev/viewer`.

- *Изменено поведение автоворота камеры (экспериментально).*

При наличии ограничений на горизонтальное вращение камера, приближаясь к границам, плавно замедляется и продолжает движение в обратном направлении.

- *Изменено поведение настройки “Special: Collision“ в настройках материала.*

Ранее включение опции автоматически приводило к сокрытию объектов, теперь для этой цели необходимо использовать настройку материала **Do not Render**.

- *Изменен суффикс конвертированных медиаданных.*

Суффикс `*.lossconv.*` заменен на `*.altconv.*`.

- *Изменена работа опции “Do not render” на объекте.*

Теперь выставление данной опции на объекте не приводит к потере физических свойств объекта. Объект всего лишь становится невидимым.

- *Доработана комплектация SDK.*

В бесплатной и коммерческой версиях SDK были добавлены новые сцены с примерами использования движка; устаревшие и малоинформационные сцены исключены.

28.4.3 Исправления

- *При выставлении в Blender’е камере типа “Panoramic” в движке принудительно используется камера типа “Perspective”.*
- *Исправлено дрожание камеры типа “Target” в отдельных случаях.*
- *Небольшие исправления в работе аддона “B4W Anim Baker”.*
- *Исправлена проблема со звуком на сценах с несколькими камерами.*
- *В модуле “controls” улучшена стабильность работы сенсоров типа “Timer”.*
- *Устранена ошибка, возникающая при просмотре сцены через IE при экспорте в HTML.*
- *Оптимизирована работа видеотекстуры. Теперь не производится обновление видеотекстуры при остановке видео.*
- *Исправлена ошибка рендеринга нодового материала, содержащего ноду “REFRACTION“.*

28.4.4 Известные проблемы

- *Проблемы с обновлением аддона в Blender.*

Настоятельно не рекомендуется устанавливать новую версию аддона Blender, не удалив предварительно старую версию, особенно это касается систем Windows.

- Не работают видеотекстуры в Firefox при экспорте сцены в формате HTML.
- Текущая реализация depth-текстур в Firefox 35 приводит к значительному снижению FPS и другим неисправностям.

В различных сценах наблюдается снижение FPS, например, при включении теней. Также некорректно отображаются тени для текстур с использованием прозрачности, например, ALPHA_CLIP. Соответствующая [ошибка](#), возможно, будет исправлена в ближайшее время в будущих обновлениях браузера.

28.5 v14.12

28.5.1 Новый функционал

- Поддержка настройки скоростей камеры.

Осуществлена поддержка пользовательской *настройки скоростей* перемещения, вращения, масштабирования камер типа Target, Hover, Eye. Настройка может быть выполнена как в интерфейсе Blender'a, так и посредством API Blend4Web.

- Поддержка MIP-текстурирования для Canvas-текстур.

Осуществлена поддержка MIP-текстурирования для *Canvas-текстур*.

- Полная поддержка ноды “MAPPING”.

Осуществлена поддержка всех возможных значений *Vector type* для ноды MAPPING: Texture, Point, Vector и Normal.

- Glow-эффект при наведении курсора мыши на объект.

Для эффекта подсветки объекта под курсором мыши доступно API модуля `mouse.js`. Управление эффектом осуществляется с помощью функций `enable_mouse_hover_glow()` и `disable_mouse_hover_glow()`. При этом на объекте должно быть выставлено свойство `Object > Blend4Web > Selectable`.

- Новая система сборки приложений.

Теперь пользователи имеют возможность вести разработку приложений непосредственно в составе SDK, благодаря новой утилите `project.py`. [Данный скрипт](#) позволяет собирать приложения вместе с движком, минимизировать JavaScript и CSS-файлы, а также экспортить готовое приложение для последующего развертывания на сервере.

28.5.2 Изменения

- Удалена поддержка устаревшей опции текстур “UV translation velocity”.

Вместо неё рекомендуется использовать функционал нодовых материалов.
- Удалена поддержка устаревшего интерфейса “Levels of Detail”.

Аналогичный функционал теперь реализуется штатным инструментом Blender “Levels of Detail”.
- Аддон *pointerlock.js* переименован в *mouse.js*.
- Добавлено срабатывание события *mouseup* при покидании курсора мыши вьюпорта приложения.

Таким образом была исправлена наблюдавшаяся ранее проблема с нарушением контроля камеры.
- Сообщение об ошибке “Clear parent inverse” более не появляется.

Ранее при использовании отношения родитель-потомок требовалось сбрасывать перемещение, вращение и масштаб объекта-потомка (*Object > Parent > Clear Parent Inverse*). Указанная трансформация теперь поддерживается движком.
- Использование “Apply scale” теперь не приводит к применению модификаторов.

Для применения модификаторов по-прежнему следует использовать флаг *Apply modifiers*.
- Использование карты нормалей в нодовом материале больше не требует наличия нод *Material* или *Extended Material*.

В некоторых случаях (например, рефракция) карты нормалей могут применяться в материалах без освещения.

28.5.3 Исправления

- Исправлена ошибка воспроизведения звуков через инструмент *NLA*.

Ошибка проявлялась вследствие недостаточной точности чисел с плавающей запятой.
- Исправлен некорректный рендеринг источников света на мобильных устройствах.
- Исправлено наложение тени при нескольких активных источниках освещения.

Теперь тень накладывается так же как в Blender, не затеняя те участки, куда попадает свет от других источников.
- Исправлена ошибка отображения нодовых материалов.

Ошибка возникала при использовании под MATERIAL и MATERIAL_EXT с добавленным по ссылке (из другого .blend-файла) материалом.

- *Исправлен сброс позы арматуры при использовании “B4W Animation Bake”.*

Теперь при использовании инструмента *запекания анимации* арматурный объект остаётся в позе, в которой он находился до запекания.

- *Исправлено резкое движение камеры после старта приложения.*
- *Исправлена ошибка, связанная с неверным определением лимитов горизонтального вращения камеры.*
- *Исправлена ошибка, связанная с экспортом неиспользуемых текстур.*

28.5.4 Известные проблемы

- *Проблемы с обновлением аддона в Blender.*

Настоятельно не рекомендуется устанавливать новую версию аддона Blender, не удалив предварительно старую версию, особенно это касается систем Windows.

- *Не работают видеотекстуры в Firefox при экспорте сцены в формате HTML.*

28.6 v14.11

28.6.1 Новый функционал

- *Поддержка видео-текстур.*

Осуществлена поддержка *видео-текстур* для текстур типа **Image or Movie**.

- *Частота кадров.*

Добавлена поддержка изменения частоты кадров для анимации и видеотекстур. Частоту кадров можно задать опцией **Scene > Dimensions > Frame rate**.

- *Поддержка текстур типа Canvas.*

В качестве *текстуры* используется HTML-элемент **Canvas**. Работа с ней *описана* в документации.

- *Camera panning.*

В режиме вращающейся вокруг точки камеры (**Target**) теперь имеется возможность перемещения точки вращения в плоскости обзора (т.н. панорамирование) при зажатой правой или средней кнопках мыши. Функция включена по умолчанию и при необходимости выключается в настройках в Blender'e.

- *Новый режим управления камерой - Hover.*

Реализован режим скольжения камеры над плоскостью (*Hover*), включающий возможность приближения и удаления. Этот режим камеры позволяет реализовывать сценарии удобного просмотра протяженных в двух измерениях сцен (помещений, игровых уровней).

- В *SDK* добавлена корневая веб-страница *index.html* для упрощения навигации по дистрибутиву.
- Поддержка преобразования форматов видео-файлов в конвертере ресурсов.
- Добавлена система сборки, ранее отсутствовавшая в публичных дистрибутивах *SDK*.
- Поддержка режима экспортта “*Strict mode*” в аддоне.

Активация режима *Strict mode* позволяет вывести все возможные ошибки и предупреждения, связанные с некорректной настройкой сцены. Опция полезна для финальной отладки сцены с целью получения максимально корректных и оптимизированных ресурсных файлов.

- Поддержка воспроизведения звуков на устройствах с *iOS*.

28.6.2 Изменения

- Опция “*bg*” веб-плеера переименована в “*fallback_image*”.

Также претерпела изменение логика поведения опции. При указании *fallback_image* информация об отсутствии поддержки WebGL теперь не выводится, вместо неё пользователю демонстрируется изображение.

- В веб-плеере опция отключения звука теперь не показывается для сцен, не имеющих источников звука.
- Улучшена стабильность работы стековых материалов.
- В сенсор “*mouse_down*” добавлен код на жатой кнопки мыши, который доступен в поле *payload* сенсора.
- Значительно оптимизирован экспорт систем частиц типа *Hair*.

28.6.3 Исправления

- Карты нормалей теперь работают с типами текстурных координат *Generated* и *Normal*.

При использовании карт нормалей теперь нет необходимости использовать UV-развертку.

- Исправлена проблема с неверными путями к физическому движку в веб-плеере.

Ошибка проявлялась при перемещении файла uranium.js из директории с главным HTML-файлом веб-плеера.

- В аддоне исправлена проблема с упакованными текстурами, проявляющаяся при экспорте при включённой опции “Automatically Pack Into .blend”.

28.6.4 Известные проблемы

- Проблемы с обновлением аддона в Blender.

Настоятельно не рекомендуется устанавливать новую версию аддона Blender, не удалив предварительно старую версию, особенно это касается систем Windows.

28.7 v14.10

28.7.1 Новый функционал

- Новый веб-плеер.

Новый минималистичный дизайн веб-плеера удачно сочетается с любыми трёхмерными сценами, имеет упрощённый интерфейс и встроенную подсказку по управлению. Веб-плеер работает на всём спектре оборудования, включая мобильные устройства.

- Улучшение системы теней.

Добавлена возможность выбора некаскадной модели теней на основе одной оптимизированной карты теней. Такая модель проще в настройке и хорошо подходит для небольших сцен. Подробности в [документации](#).

- Многочисленные улучшения в системе нелинейной анимации (NLA).

Появилась возможность создания сложной логики в инструменте NLA Script с помощью блоков Conditional Jump, Register Store и Math Operation, а также переменных, сохраняемых в регистрах.

Возможность использования в NLA всех типов анимации, поддерживаемых движком, включая воспроизведение звуков, вертекстную анимацию и эмиссию частиц. Анимацию различных типов теперь возможно проигрывать параллельно.

- Поддержка биллбординга объектов.

Объектам добавлены [опции](#), позволяющие настроить биллбординг.

- Поддержка режима “XYZ Euler” для анимации поворота.

Объектная и скелетная анимация теперь поддерживают режим **XYZ Euler** для анимации поворота.

- Поддержка текстурных координат типа *GENERATED*.
- Добавлена поддержка загрузки ресурсов с удаленного сервера (*Cross-origin resource sharing*).
- Упрощен процесс экспортации сцен.

Ряд ошибок, возникающих при экспорте материала, теперь не блокирует экспорт. Вместо этого при загрузке сцены данный материал будет отображаться розовым цветом. Более подробное описание ошибок доступно в [документации](#).

- Добавлена поддержка опции “*Do not export*” для системы частиц.
- Улучшена стабильность работы приложений на устройствах с мобильной операционной системой *Apple iOS*.

28.7.2 Изменения

- В SDK изменился путь для настройки экспортации в *HTML*.

В настройках аддона, установленного через SDK, теперь достаточно указывать путь непосредственно к корневой директории SDK. Ранее требовалось указывать путь к приложению embed. Обратите внимание, старое поведение более не поддерживается.

- Объявлена устаревшей опция текстур “*UV translation velocity*”.
- Опция будет удалена начиная с версии 14.12.
- Удалена опция “*Do not export*” на панели “*Object data*”.
 - Опция *Blend4Web>Animation>Cyclic* на вкладке свойств объектов удалена.

Вместо неё следует использовать опцию *Blend4Web > Animation > Behavior*, расположенную там же. В сценах, где циклическая анимация назначается по умолчанию, возможно нарушение работы анимации, соответственно в них необходимо установить поведение в *Cyclic*.

- Изменена реализация алгоритма *SSAO*.

Новая реализация работает значительно быстрее и показывает лучшее качество изображения. Параметры настройки алгоритма также изменились. Подробнее в [документации](#).

28.7.3 Исправления

- Исправлена ошибка рендеринга материала типа *HALO*.
- Исправлена ошибка рендеринга объекта при выставлении опции “*Force Dynamic Object*”, если он имел родительский объект.

- Исправлена ошибка назначения горячих клавиш на экспорт аддона Blend4Web.

Устранена ошибка, не позволяющая назначить горячие клавиши на экспорт в форматы JSON и HTML: File->Export->Blend4Web(.json) и File->Export->Blend4Web(.html).

- Исправлена ошибка загрузки текстур, превышающих допустимый размер для данного устройства.
- Исправлены ошибки анимации нодовых материалов, приводившие к нестабильной работе движка.
- Устранена ошибка нодовых материалов, содержащих сложные группы нод (Node Groups).
- Исправлены ошибки компиляции шейдеров на устройствах с мобильной графикой Qualcomm Adreno 305.
- Исправлена ошибка рендеринга при использовании ноды REFRACTION на непрозрачных материалах.
- Исправлен сброс значения текущего кадра анимации после её запечатывания при помощи инструмента “B4W Vertex Anim Baker”.

28.7.4 Известные проблемы

- Проблемы с обновлением аддона в Blender.

Настоятельно не рекомендуется устанавливать новую версию аддона Blender, не удалив предварительно старую версию, особенно это касается систем Windows.

- Карты нормалей не работают с типом текстурных координат Generated.

При использовании карт нормалей необходимо использовать UV-развертку.

28.8 v14.09

28.8.1 Новый функционал

- Поддержка типа ABSOLUTE для ноды MATH.
- Поддержка специализированной ноды LEVELS_OF_QUALITY.

Позволяет управлять сложностью материала в зависимости от профиля качества, указанного пользователем при загрузке движка.

- Поддержка специализированной ноды SMOOTHSTEP.

Упрощает создание некоторых эффектов в нодовом материале.

- *Поддержка нодовых групп.*

Нодовые группы позволяют повторное использования блока нод одного материала в составе другого.

- *Возможность вывода промежуточных результатов рендеринга с целью отладки.*

Результат рендеринга конкретной стадии теперь можно вывести поверх основного изображения. Настройка осуществляется в модуле `config.js` опциями группы `debug_subs`.

- *Реализация логики управления NLA-анимацией в Blender с помощью визуального редактора.*

В интерфейс Blender добавлен инструмент NLA Script, который позволяет с помощью визуальных блоков реализовывать простые сценарии, например осуществить управление текущей анимацией в ответ на действия пользователя.

- *Многочисленные улучшения системы сенсоров.*

В модуле `controls` разрешено регистрировать множества сенсоров глобально, без привязки к конкретному объекту, для чего в соответствующие API необходимо подавать параметр `null`. Обеспечена более предсказуемая и надёжная обработка логики сенсоров, в соответствии с последовательностью создания их множества. Обработчики событий, связанных с действиями пользователя теперь назначаются с помощью функций вида `register_<inputtype>_events()`. В данные функции теперь можно подавать флаг `prevent_default`, позволяющий заблокировать стандартное поведение браузера, происходящее после возникновения соответствующих событий.

- *Поддержка использования физики в веб-плеере.*

Работает только в версии веб-плеера с раздельной загрузкой JSON-файлов. В одиночных HTML файлах физика не поддерживается.

- *Поддержка смешивания различных скелетных анимаций.*

В модуле `animation.js` добавились API, позволяющие осуществлять плавные переходы между скелетными анимациями: `get_skel_mix_factor()` - для получения текущего значения коэффициента смешивания и `set_skel_mix_factor()` - для его назначения.

- *Поддержка анимации ноды Value в нодовых материалах.*

Функциональность работает аналогично другим видам анимации. Поддерживается работа в NLA.

- *Поддержка параметров Specular и Diffuse в источниках освещения.*

- *Возможность рендеринга полупрозрачного объекта поверх остальных объектов на сцене.*

Активируется с помощью опции `Render above all` для прозрачных материалов (т.е. не `Opaque`).

- *Автоматическое применение масштаба к мешу объекта.*

Реализуется включением опции `Apply scale` в настройках объекта.

- *Поддержка профиля высокого качества (включая тени, динамическое отражение и антиалиасинг) для iOS.*

28.8.2 Изменения

- *Улучшение рендеринга теней.*

Существенно изменена система отрисовки теней, которая теперь основывается на технике `Stable Cascaded Shadow Maps`. Техника позволяет существенно уменьшить мерцание краев теней при движении камеры. Между каскадами реализовано сглаживание. Также тени последнего каскада плавно исчезают при удалении. Для рендеринга мягких теней реализована техника `Percentage Closer Shadows`. Переработана и упрощена система пользовательских настроек теней. Теперь можно задавать размеры карт теней, степень размытия, настройки для компенсации ошибок самозатенения. Новые настройки подробно [документированы](#).

- *Настройка качества графики в веб-плеере теперь сохраняется для каждой сцены независимо.*
- *Изменено поведение настроек конфигурации приложения: `physics_uraniun_path`, `smaa_search_texture_path` и `smaa_area_texture_path`.*

Данные настройки конфигурации теперь рассчитываются автоматически в зависимости от местоположения запускаемого HTML-файла, если они не были переопределены при инициализации приложения.

- *Завершён переход на систему модулей, подключаемых через вызов `b4w.require()`.*

Это также означает, что начиная с текущей версии в релиз-версии движка модули нельзя вызывать с помощью устаревшего интерфейса `b4w.<module>`. С целью совместимости создан аддон `ns_compat.js`, подключение которого позволяет восстановить старое поведение.

- *Добавлена возможность скрытия панели управления в веб-плеере.*
- *Скелетная анимация применяется только к объекту арматуры.*

Нет необходимости применять скелетную анимацию к объектам типа `MESH`. Если они привязаны к какой-либо анимированной арматуре, скиннинг будет происходить автоматически.

- *Демонстрационные приложения и обучающие материалы приведены в соответствие с новым функционалом.*

28.8.3 Исправления

- *Индикатор загрузки не скрывался, если в процессе происходила ошибка загрузки ресурса (текстуры либо звукового файла).*
- *Исправлены лаги при масштабировании и повороте на touch-устройствах.*
- *Устранено дрожание камеры типа TARGET при небольших поворотах.*
- *Исправлено управление камерой типа EYE на мобильных устройствах.*
- *Улучшено управление в приложении Ферма в браузере Safari.*
- *Исправлены ошибки при использовании неподдерживаемых моделей освещения на нодовых материалах.*
- *Для объектов без материалов теперь работает опция “Selectable”.*
- *Более нет необходимости включать “Force Dynamic Object” для объектов, анимирующихся с использованием NLA.*
- *Исправлена ошибка для систем частиц, в которых дублируемый объект входит в иерархию с другими объектами.*

28.8.4 Известные проблемы

- *Проблемы с обновлением аддона в Blender.*

Настоятельно не рекомендуется устанавливать новую версию аддона Blender, не удалив предварительно старую версию, особенно это касается систем Windows.

- *Смешивание скелетной анимации не работает в некоторых браузерах.*

Если при использовании API смешивания скелетной анимации возникают необработанные ошибки, следует переопределить стандартную функцию Math.sign следующим образом:

```
var m_util = require("util");
Math.sign = m_util.sign;
```

- *genindex*
- *search*

Symbols

- Управление перемещением объектов, 59
аддон, 37
 ошибки инициализации, 42
 установка, 13
 установка новой версии, 14
анаглиф, 135
анимация, 154
береговая линия, 171
браузер, 4
браузерные технологии, 4
цвет фона, 125
драйверы, 5
движок, 1
эффект Френеля, 80
экспорт, 14, 24
 ошибки, 44
 предупреждения об ошибках экспорта, 47
глубина резкости камеры (depth of field), 129
графический движок, 1
интерактивная графика, 5
источники света, 115
компиляция шейдеров, 227
 директивы import/export, 229
 обfuscation, 228
 оптимизация, 228
 ошибки, 231
 расширения WebGL, 231
 рекомендации и ограничения, 229
 валидация, 228
коррекция цвета, 136
кватернион, 60
материалы, 75
 гало, 84
ноды, 101–103
параметры освещения, 76
прозрачность, 78
специальные параметры, 82
зеркальное отражение, 79
нормали, 110
редактор, 110
освещение, 108
параметры берега, 171
подсветка контура (outline glow), 133
просмотрщик, 27
 добавление сцен, 24
 запуск, 21
прозрачность, 78
 настройка, 79
 типы, 78
размытие при движении (motion blur), 128
сенсор, 212
 множество, 213
сглаживание, 136
система частиц
 флюиды, 137
 инстансинг, 145
стереоизображение, 135
сумеречные лучи (crepuscular rays), 132
текстуры, 85
 диффузная, 88
 карта бликлов, 88
 карта нормалей, 89
 карта окружения, 94
 карта смешивания, 90
 карта высот, 89
 карта зеркального отражения, 96
 настройки, 87
 небо, 97
 рендеринг в, 98

- типы, 86
трехмерный движок, 1
веб-плеер, 31
версия
 ошибки, 43
видео-карта, 5
взаимное затенение (screen-space ambient occlusion), 130
замечания к релизам, 250
затенение, 109
 типы, 109
зеркальное отражение, 79
 динамическое, 80
 эффект Френеля, 80
 статическое, 80
3D моделирование, 3
- B**
- Blend4Web, 1
Blender, 3
 установка, 12, 16
- D**
- diffuse map, 88
DOF, 129
- E**
- environment map, 94
- G**
- git, 233
 добавление и удаление файлов, 237
 индивидуальные настройки, 235
 коммит, 238
 подготовка к коммиту, 236
 проверка статуса, 235
 разрешение конфликтов, 241
 синхронизация между репозиториями, 239
 тэги, 243
god rays, 132
- H**
- halo, 84
height map, 89
- M**
- matcap, 87
material capture, 87
- mirror map, 96
N
normal map, 89
P
parallax mapping, 89
R
render-to-texture, 98
RTT, 98
- S**
- sensor, 212
 manifold, 213
skydome, 97
specular map, 88
SSAO, 130
stencil map, 90
- W**
- WebGL, 2
 поддержка в браузерах, 2
 преимущества, 3