

---

# Blend4Web. Руководство пользователя

*Выпуск 14.10*

ООО "Триумф"

30 October 2014

<b>1 Общие сведения</b>	<b>1</b>
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
<b>2 Функционал движка</b>	<b>7</b>
2.1 Текстуры . . . . .	7
2.2 Материалы . . . . .	7
2.3 Освещение . . . . .	7
2.4 Тени . . . . .	8
2.5 Система частиц . . . . .	8
2.6 Рендеринг наружных сцен . . . . .	8
2.7 Постпроцессинговые эффекты . . . . .	8
2.8 Анимация . . . . .	9
2.9 Оптимизация . . . . .	9
2.10 Звук . . . . .	9
2.11 Физика . . . . .	10
2.12 Событийная модель . . . . .	10
2.13 Прочее . . . . .	10
<b>3 Экспресс-установка</b>	<b>12</b>
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

---

<b>4 Развёртывание среды разработки</b>	<b>16</b>
4.1 Установка дистрибутива . . . . .	16
4.2 Выбор браузера . . . . .	16
4.3 Настройка браузера для загрузки локальных ресурсов . . . . .	17
4.4 Запуск просмотрщика сцен . . . . .	21
4.5 Установка аддона движка . . . . .	21
<b>5 Рабочий процесс</b>	<b>24</b>
5.1 Подготовка сцен . . . . .	24
5.2 Экспорт сцен . . . . .	25
5.3 Отображение сцен в просмотрщике . . . . .	26
5.4 Разработка приложения . . . . .	27
<b>6 Просмотрщик сцен</b>	<b>28</b>
6.1 Навигация . . . . .	28
6.2 Боковая панель . . . . .	28
6.3 Индикаторы . . . . .	31
<b>7 Веб-плеер</b>	<b>32</b>
7.1 Использование . . . . .	32
7.2 Навигация . . . . .	33
7.3 Панель управления . . . . .	33
7.4 Атрибуты . . . . .	33
7.5 Ошибки сцены . . . . .	34
<b>8 Аддон</b>	<b>36</b>
8.1 Ошибки инициализации . . . . .	36
8.2 Критические ошибки экспорта . . . . .	37
8.3 Некритические ошибки экспорта . . . . .	41
8.4 Прочие сообщения, требующие внимания пользователя . . . . .	44
<b>9 Объекты</b>	<b>45</b>
9.1 Типы . . . . .	45
9.2 Статические и динамические объекты . . . . .	45
9.3 Настройка . . . . .	46
9.4 Камера . . . . .	51
<b>10 Материалы</b>	<b>57</b>
10.1 Параметры освещения . . . . .	58
10.2 Прозрачность . . . . .	59
10.3 Зеркальное отражение . . . . .	61
10.4 Специальные параметры движка . . . . .	63
10.5 Материалы гало (Halo) . . . . .	65
<b>11 Текстуры</b>	<b>67</b>
11.1 Типы текстур . . . . .	67
11.2 Общие настройки . . . . .	68
11.3 Диффузная текстура (diffuse map) . . . . .	69
11.4 Карта бликов (specular map) . . . . .	69

11.5	Карта нормалей (normal map) . . . . .	70
11.6	Карта высот (height map). Parallax mapping . . . . .	70
11.7	Карта смешивания (stencil map) . . . . .	71
11.8	Карта окружения (environment map) . . . . .	73
11.9	Карта зеркального отражения (mirror map) . . . . .	75
11.10	Текстура неба (skydome) . . . . .	76
11.11	Рендеринг в текстуру (render-to-texture, RTT) . . . . .	77
<b>12</b>	<b>Нодовые материалы</b>	<b>79</b>
12.1	Стандартные ноды . . . . .	79
12.2	Дополнительные ноды . . . . .	80
<b>13</b>	<b>Освещение, тени и фон</b>	<b>86</b>
13.1	Типы затенения . . . . .	86
13.2	Редактор нормалей . . . . .	87
13.3	Освещение от источников света . . . . .	93
13.4	Освещение от окружающей среды . . . . .	94
13.5	Тени . . . . .	96
13.6	Цвет фона . . . . .	102
<b>14</b>	<b>Постпроцессинговые эффекты</b>	<b>104</b>
14.1	Размытие при движении . . . . .	104
14.2	Глубина резкости камеры . . . . .	105
14.3	Взаимное затенение . . . . .	106
14.4	Сумеречные лучи . . . . .	109
14.5	Эффект засветки ярких деталей . . . . .	110
14.6	Свечение вокруг объекта (Glow) . . . . .	111
14.7	Анаглиф стереоизображение . . . . .	112
14.8	Коррекция цвета . . . . .	113
14.9	Сглаживание . . . . .	114
<b>15</b>	<b>Система частиц</b>	<b>116</b>
15.1	Использование . . . . .	117
15.2	Настройка . . . . .	118
15.3	Текстуры в системах частиц . . . . .	122
<b>16</b>	<b>Система частиц для инстансинга объектов</b>	<b>124</b>
16.1	Настройки системы частиц . . . . .	125
16.2	Травяной покров . . . . .	128
16.3	Листья деревьев . . . . .	130
<b>17</b>	<b>Анимация</b>	<b>133</b>
17.1	Управление анимацией . . . . .	133
17.2	Объектная анимация . . . . .	134
17.3	Скинниング и скелетная анимация . . . . .	135
17.4	Вертекальная анимация . . . . .	137
17.5	Параметризация источников звука . . . . .	138
17.6	Анимация ноды Value . . . . .	139

<b>18 Рендеринг наружных сцен</b>	<b>140</b>
18.1 Вода . . . . .	140
18.2 Атмосфера . . . . .	150
18.3 Ветер . . . . .	153
<b>19 Гамма-коррекция и альфа-композитинг</b>	<b>156</b>
19.1 Общее описание . . . . .	156
19.2 Человеческое зрение, мониторы и гамма-коррекция . . . . .	157
19.3 Гамма . . . . .	158
19.4 Коррекция в нодовых материалах . . . . .	158
19.5 Альфа-композитинг . . . . .	159
<b>20 Звуковая подсистема</b>	<b>160</b>
20.1 Настройка звуковых источников . . . . .	160
20.2 Обработка и кодирование . . . . .	163
<b>21 Физика</b>	<b>164</b>
21.1 Подготовка к использованию . . . . .	164
21.2 Статический тип физики . . . . .	165
21.3 Динамический тип физики . . . . .	167
21.4 Ограничители (Constraints) . . . . .	170
21.5 Колесные транспортные средства . . . . .	171
21.6 Плавающие объекты . . . . .	174
21.7 Плавающие транспортные средства . . . . .	175
<b>22 Нелинейная анимация</b>	<b>177</b>
22.1 Редактор NLA . . . . .	177
22.2 Визуальное программирование (NLA Script) . . . . .	179
<b>23 Разработчикам приложений</b>	<b>184</b>
23.1 Простейшее приложение (Hello world!) . . . . .	184
23.2 Загрузка сцены в приложение . . . . .	185
23.3 Быстрое создание приложений . . . . .	185
23.4 Система модулей . . . . .	186
23.5 Управление перемещением объектов . . . . .	187
23.6 Кватернионы . . . . .	188
23.7 Событийная модель . . . . .	190
23.8 Профили качества изображения . . . . .	192
23.9 Файловая структура SDK . . . . .	194
<b>24 Разработчикам движка</b>	<b>197</b>
24.1 Стиль оформления кода . . . . .	197
24.2 Сборка аддона . . . . .	198
24.3 Зависимости . . . . .	199
24.4 Названия функций и переменных . . . . .	199
24.5 Отладка . . . . .	200
24.6 Шейдеры . . . . .	200
<b>25 Работа в команде. Использование git</b>	<b>206</b>

25.1	Общие сведения . . . . .	206
25.2	Типичный рабочий процесс . . . . .	206
25.3	Индивидуальные настройки . . . . .	207
25.4	Проверка статуса . . . . .	207
25.5	Перед коммитом . . . . .	208
25.6	Подготовка к коммиту . . . . .	209
25.7	Коммит . . . . .	210
25.8	Синхронизация между репозиториями . . . . .	211
25.9	Разрешение конфликтов . . . . .	213
25.10	Тэги . . . . .	215
25.11	Другие полезные команды . . . . .	216
<b>26</b>	<b>Проблемы и решения</b>	<b>217</b>
26.1	Проблемы при запуске рендерера . . . . .	217
26.2	Ошибка инициализации WebGL . . . . .	217
26.3	Использование локального web-сервера . . . . .	220
<b>27</b>	<b>Замечания к релизам</b>	<b>222</b>
27.1	v14.10 . . . . .	222
27.2	v14.09 . . . . .	224
	<b>Алфавитный указатель</b>	<b>228</b>



## 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](#)

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 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)

#### 2.2 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)

#### 2.3 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)

## 2.4 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

## 2.5 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, листья деревьев и проч.

## 2.6 Рендеринг наружных сцен

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

## 2.7 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)
- огибание объектов светом (glow)

## 2.8 Анимация

- скиннинг (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)

## 2.9 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)

## 2.10 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 зациклывание 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

## 2.11 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

## 2.12 Событийная модель

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

## 2.13 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
- использование распределенной системы контроля версий git
- модульная структура исходного кода
- мощный шейдерный препроцессор с поддержкой модулей и функциональных блоков (нод)
- удобная система для быстрого развертывания новых 3D приложений
- тесная интеграция с системой моделирования и анимации Blender
- опции для поддержки работы на широком спектре оборудования
- руководство пользователя и документация для программистов
- взаимодействие с пользователем — управление камерой, персонажем, действиями

## Экспресс-установка

---

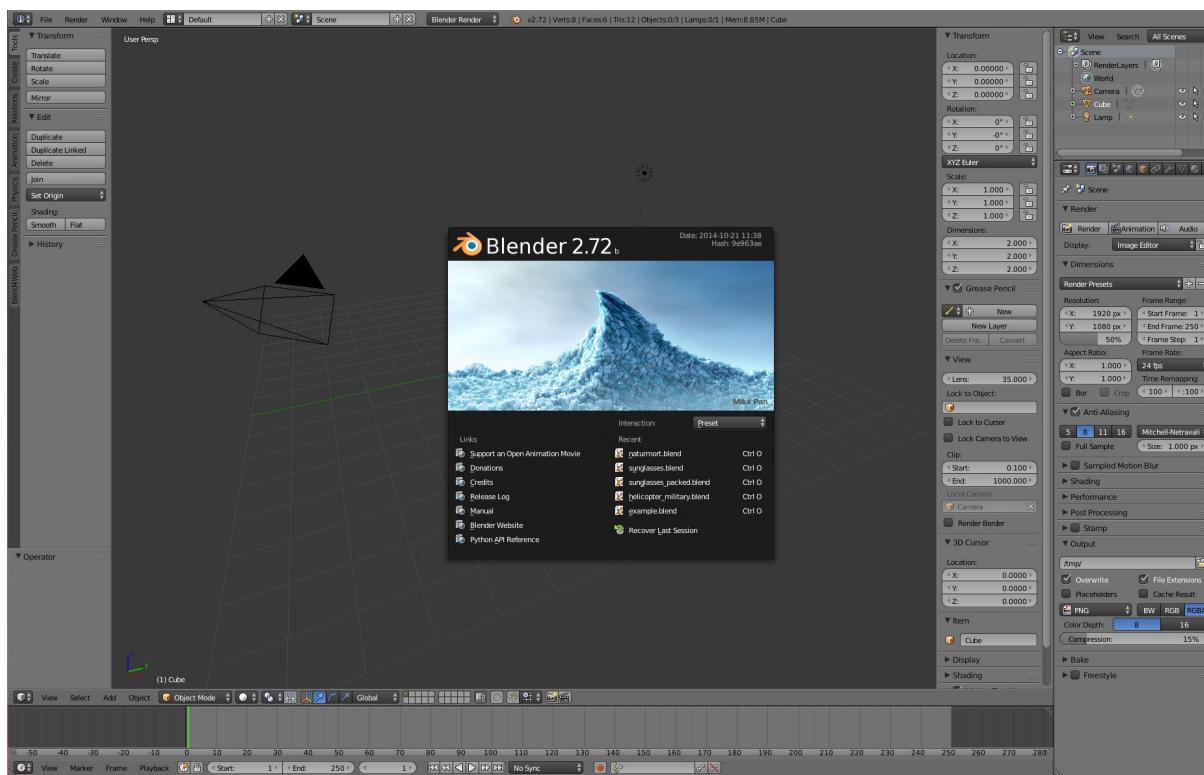
Экспресс-установка аддона Blend4Web подойдет обычным пользователям, которым не требуется разработка полноценных 3D приложений. Основной возможностью в этом случае является экспорт сцены в один HTML-файл с последующим просмотром в браузере, поддерживающем WebGL.

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

### 3.1 Установка программы Blender

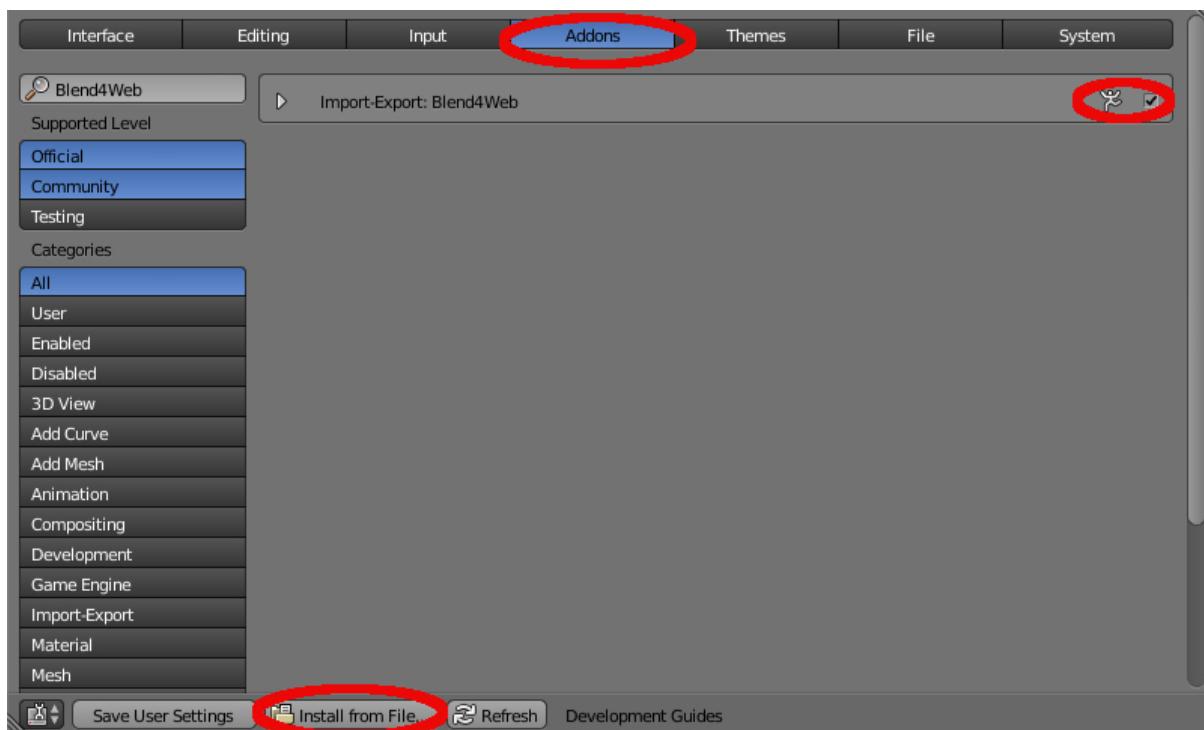
Создание 3D сцен осуществляется в графическом пакете Blender, который является программным продуктом с открытым исходным кодом и распространяется бесплатно.

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](#)

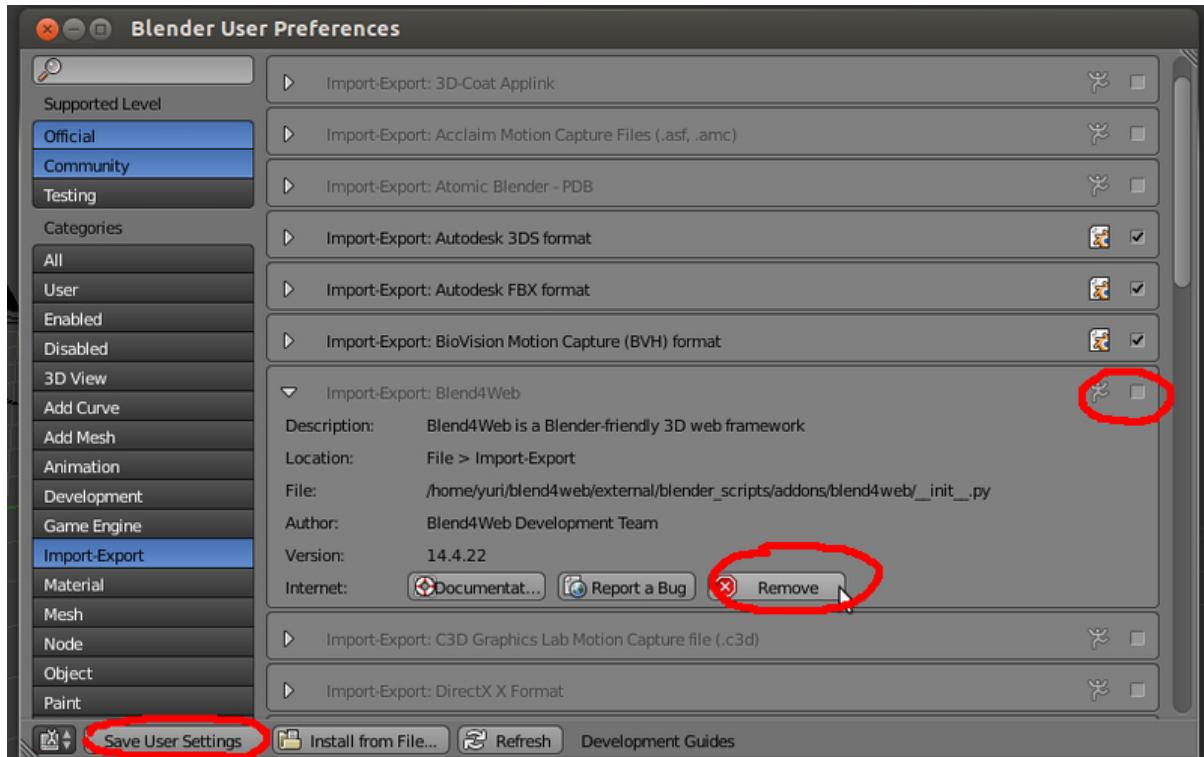
### 3.4 Установка новой версии аддона

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

---

Следующий вариант установки подойдет для разработчиков 3D приложений. Если же нужно простое ознакомление с аддоном Blend4Web, то, возможно, будет удобнее [экспресс-установка](#).

Для работы необходим дистрибутив движка, браузер (настроенный для локального просмотра) и Blender (с установленным аддоном).

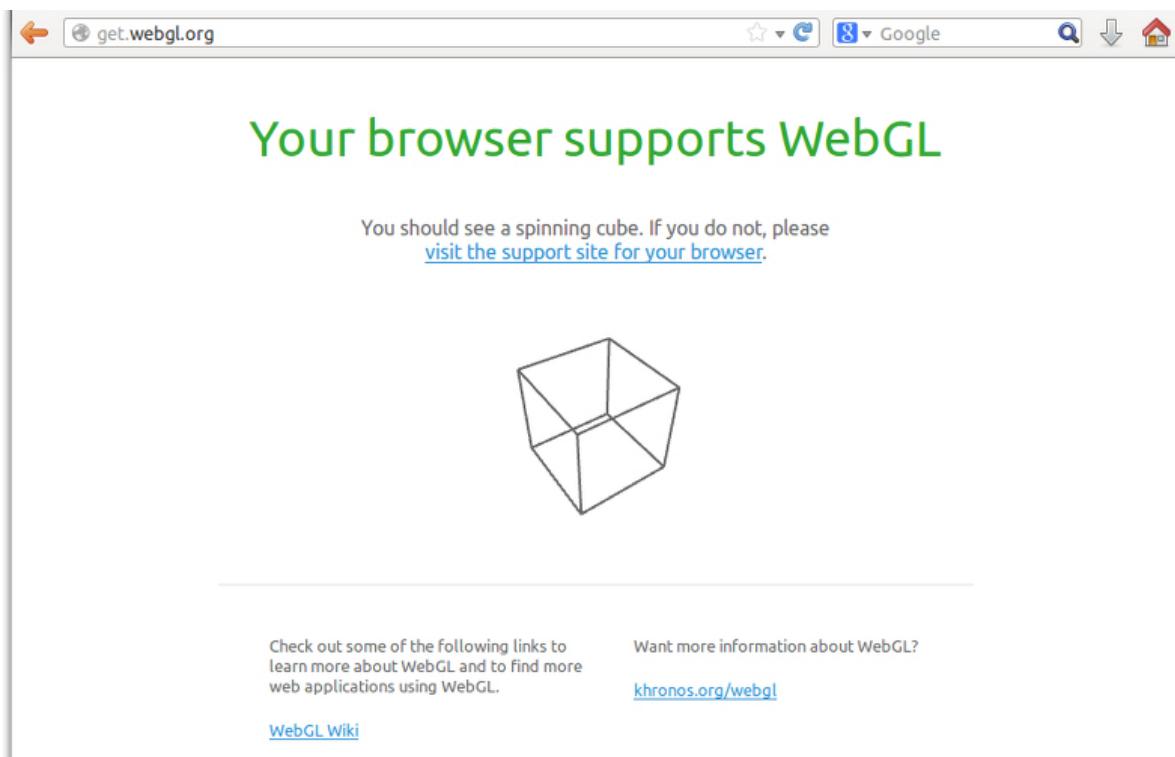
## 4.1 Установка дистрибутива

Стабильные версии дистрибутива поставляются в виде архива (`blend4web_sdk_free.zip` – бесплатный SDK, `blend4web_sdk_pro.zip` – коммерческий SDK). Для установки достаточно распаковать данный архив в любое место на диске.

В дистрибутиве находятся исходный код движка, компактная версия для приложений, скрипты к Blender'у, исходные blend-файлы группы разработки, экспортированные сцены, текстуры и звуковые файлы (см. подробную [структуру репозитория](#)).

## 4.2 Выбор браузера

Для работы движка требуется *браузер с поддержкой WebGL*. Для проверки можно перейти на страницу <http://get.webgl.org/>. Должна появиться надпись зеленого цвета и вращающийся куб:



## 4.3 Настройка браузера для загрузки локальных ресурсов

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 (рекомендуется). Другой способ заключается в использовании [локального web-сервера](#).

---

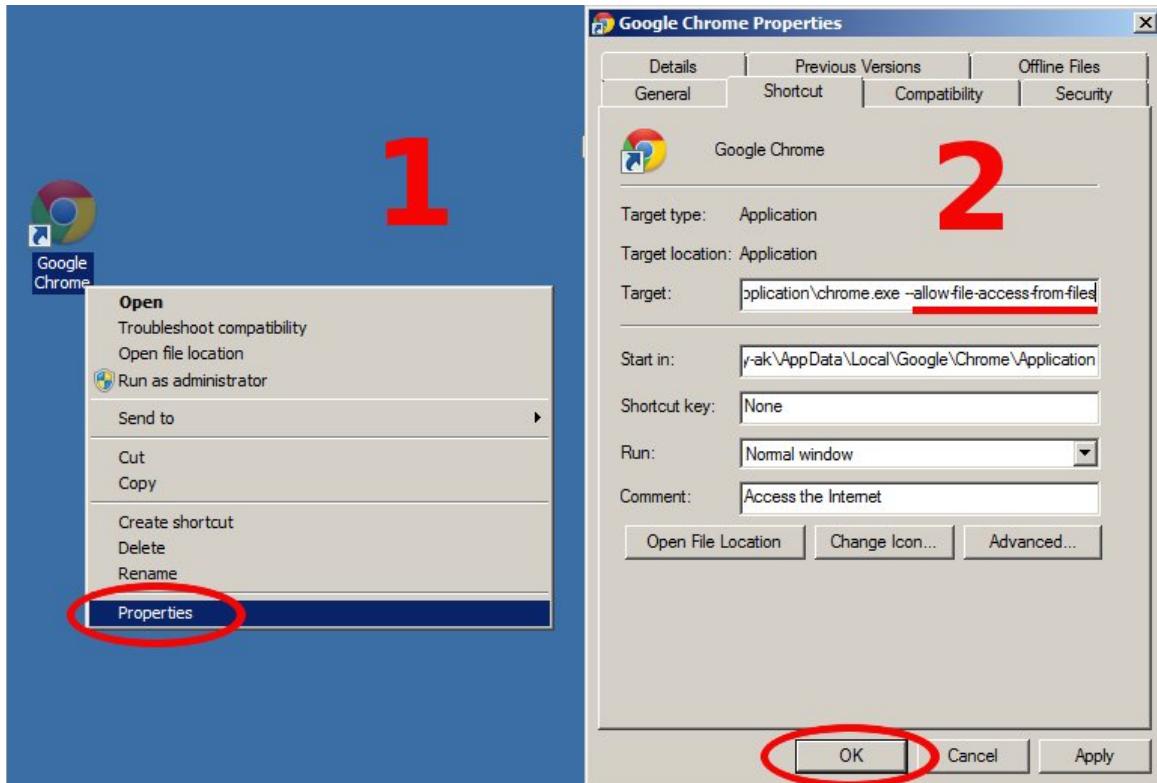
**Примечание:** Рекомендуется использовать такой браузер только для просмотра локального контента, поскольку изменение настроек может привести к понижению безопасности.

---

*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 оригиналную версию ярлыка для запуска браузера в обычном режиме.

*Chrome на OS X:*

Открыть Терминал и запустить браузер с параметром:

```
> /Applications/Google\ Chrome.app/Contents/MacOS/Google\ Chrome --allow-file-access-from-fil
```

*Chrome/Chromium на Linux:*

Запустить браузер с параметром:

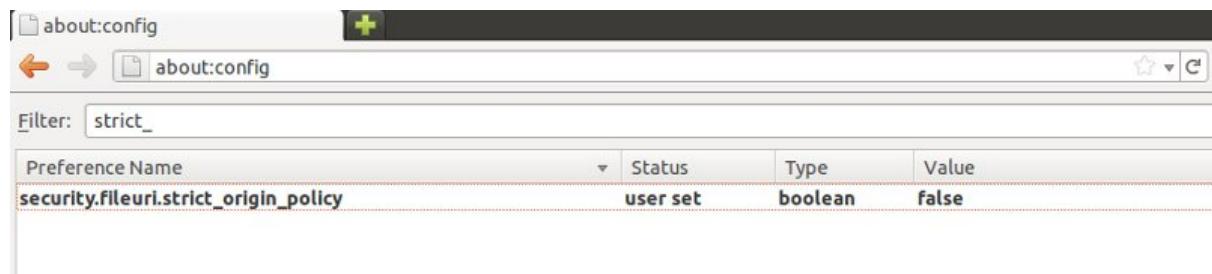
```
> google-chrome --allow-file-access-from-files
```

или:

```
> chromium-browser --allow-file-access-from-files
```

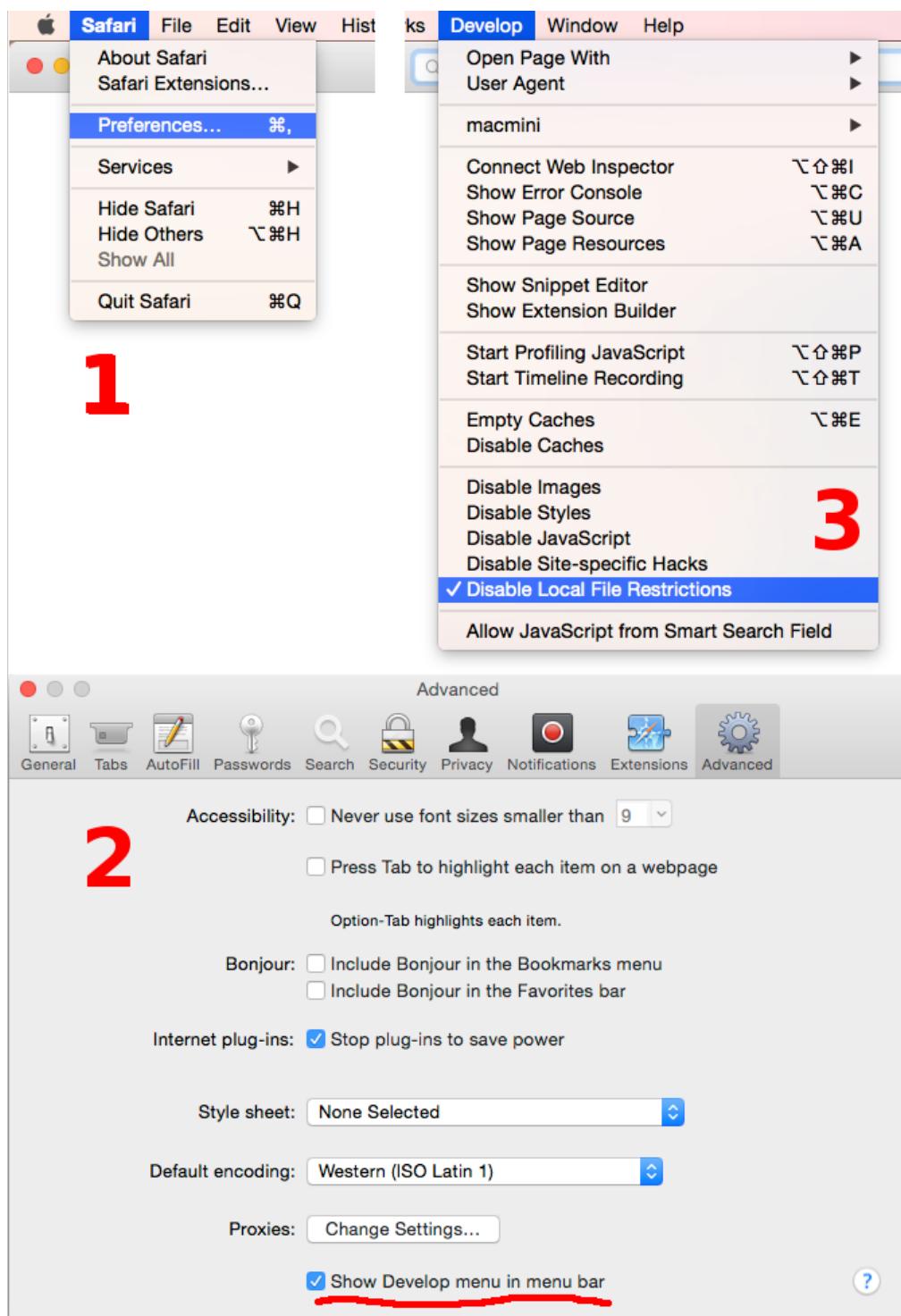
*Firefox на Windows/Linux/OS X:*

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”.



## 4.4 Запуск просмотрщика сцен

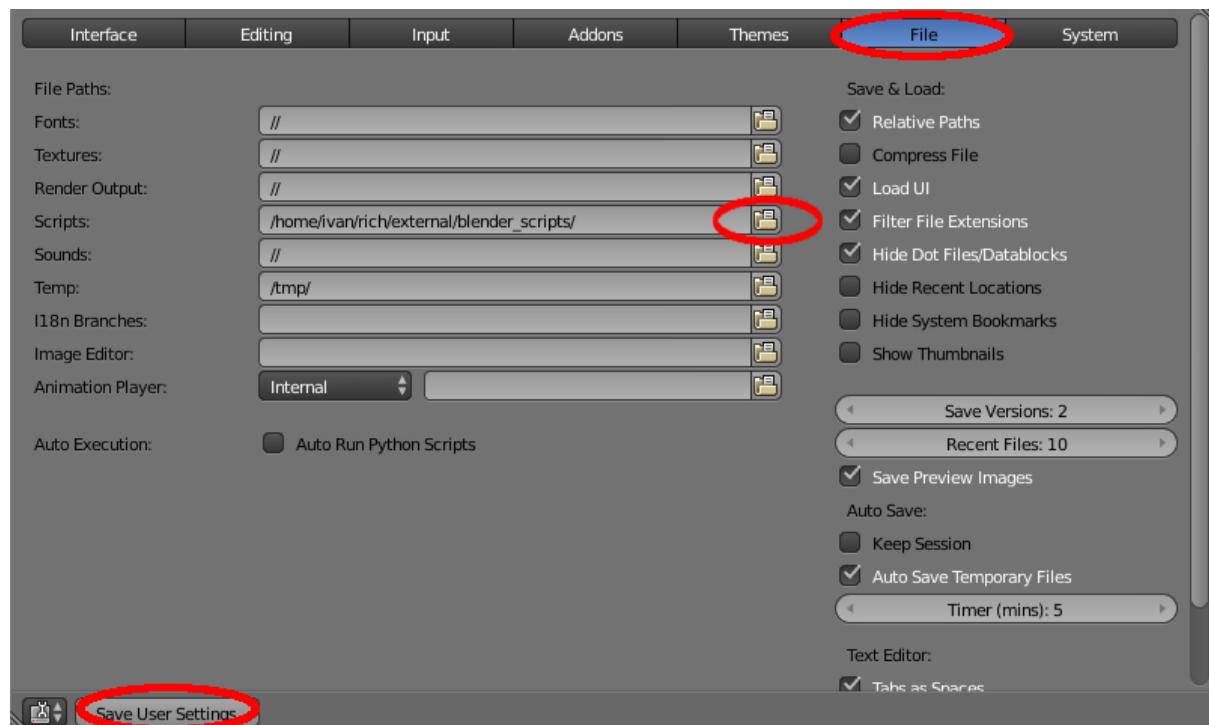
Откройте файл `apps_dev/viewer/viewer_dev.html` в настроенном браузере. Должна отобразиться страница *просмотрщика сцен* с окном рендерера и элементами интерфейса.

**Примечание:** Если страница не отображается корректно, или появляются сообщения об ошибках, необходимо предпринять действия, описанные в разделе *Проблемы при запуске рендерера*.

## 4.5 Установка аддона движка

**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`.

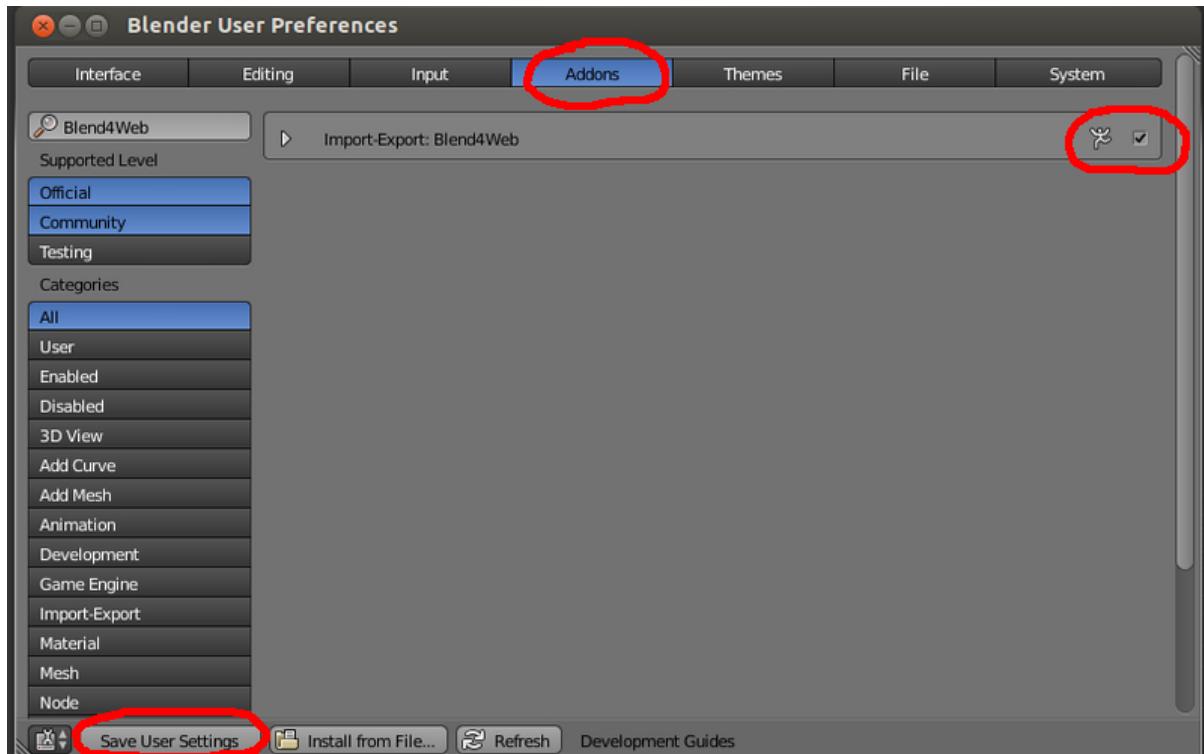


Нажать `Save User Settings` и перезапустить Blender.

**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.



Нажать Save User Settings. Перезапуск Blender не требуется.

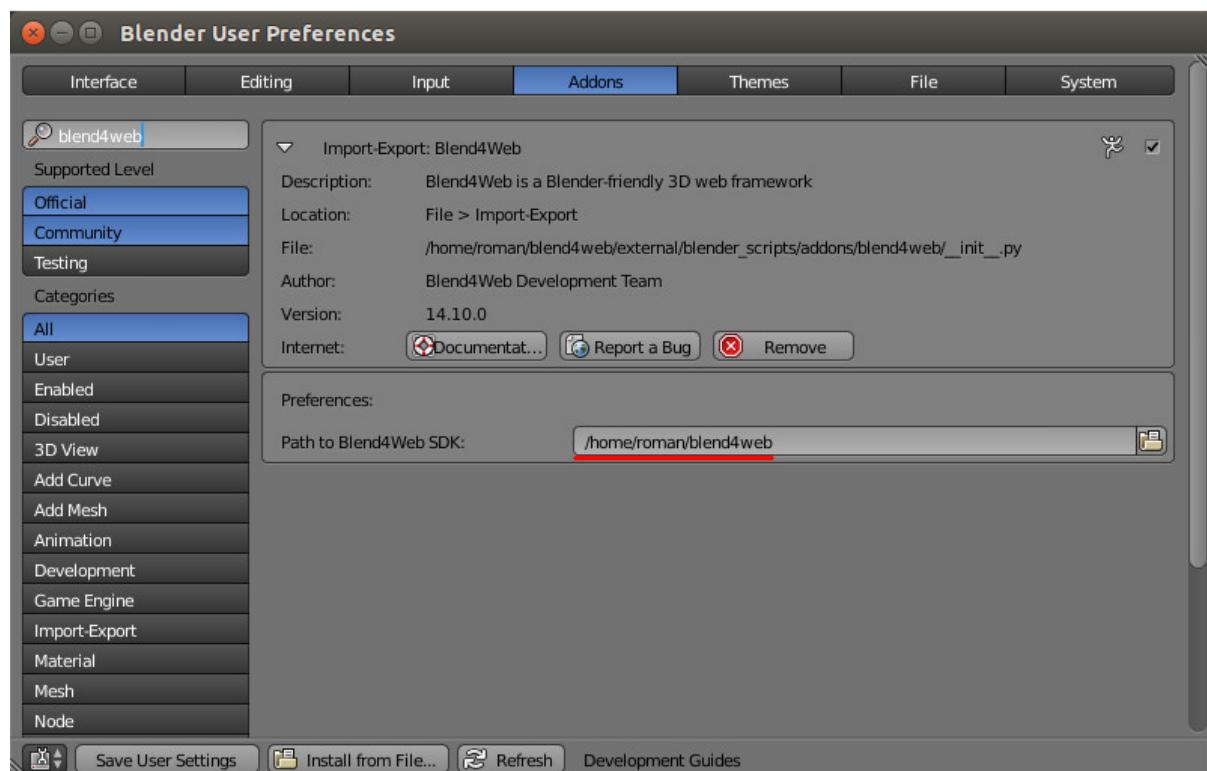
*Для проверки:*

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).

#### 4.5.1 Включение опции экспорта в HTML

Опция Blend4Web (.html) в меню File > Export по умолчанию не активна, в отличие от версии одиночного аддона (см. [экспресс-установка](#)).

В случае необходимости (например, если требуется отладка HTML экспорта) данную опцию можно включить. Для этого указать путь к Blend4Web SDK, в поле Path to Blend4Web SDK.



## 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 `external/blender/имя_проекта`.
2. Файлы текстур и звуков должны быть внешними и находиться в директории `external/deploy/assets/имя_проекта`.
3. Вспомогательные файлы, не предназначенные для загрузки в движок (например, референсы), должны находиться в директории `external/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 `external/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](#).

### 5.2.1 Опции экспорта

***Autosave main 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.

***Filepath*** Путь для экспортного файла. Используется в режиме экспорта из командной строки. В оконном режиме не используется.

## 5.3 Отображение сцен в просмотрщике

Для того, чтобы сцена появилась в списке сцен просмотрщика, нужно вручную добавить запись в текстовой файл `external/deploy/assets/assets.json`.

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

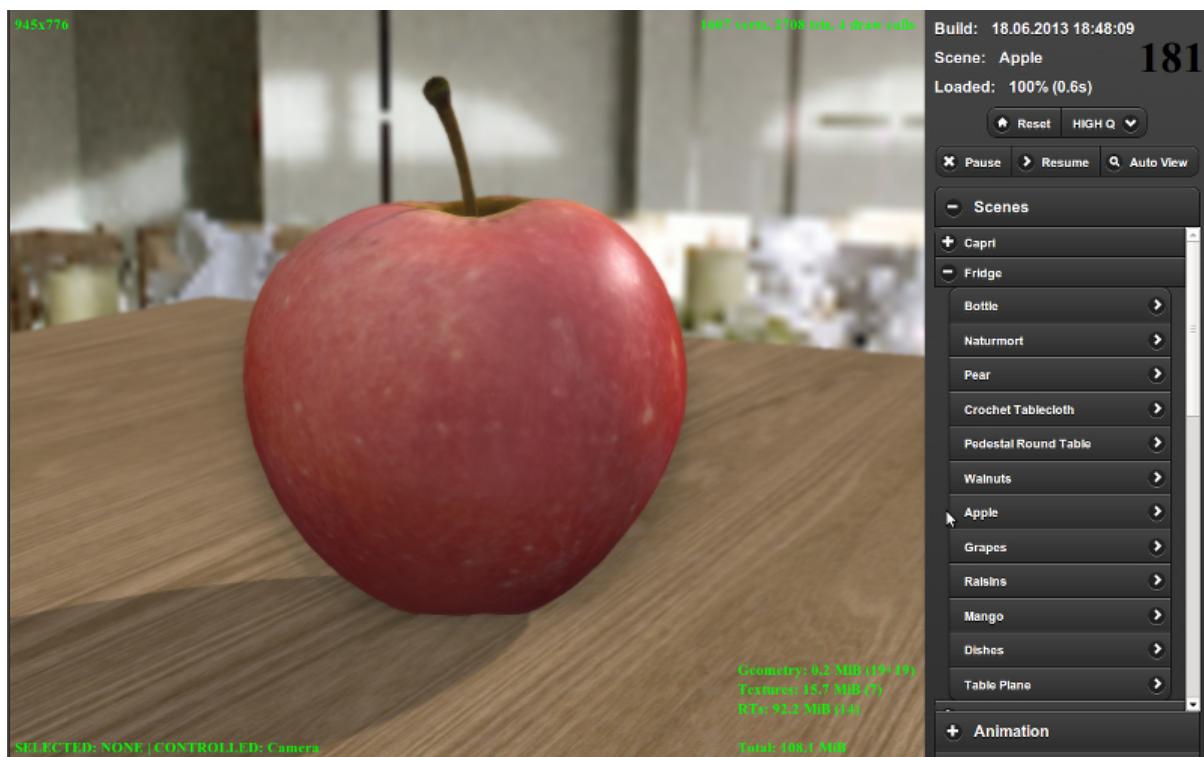
### 5.3.1 Пример

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 просмотрщика 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

---

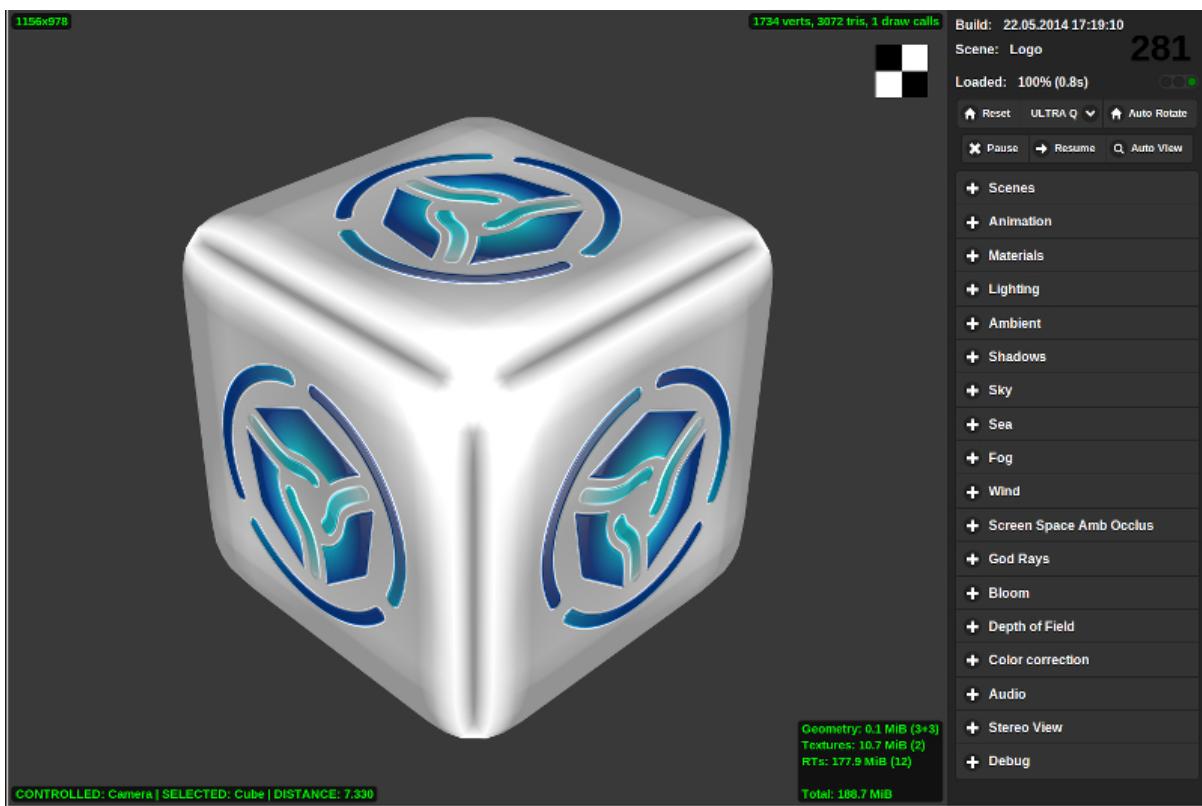
*Запуск просмотрщика сцен.*

### 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.



### 7.1 Использование

Директорию с файлами веб-плеера `external/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 помощи.

## 7.4 Атрибуты

Веб-плеер принимает атрибуты из адресной строки браузера:



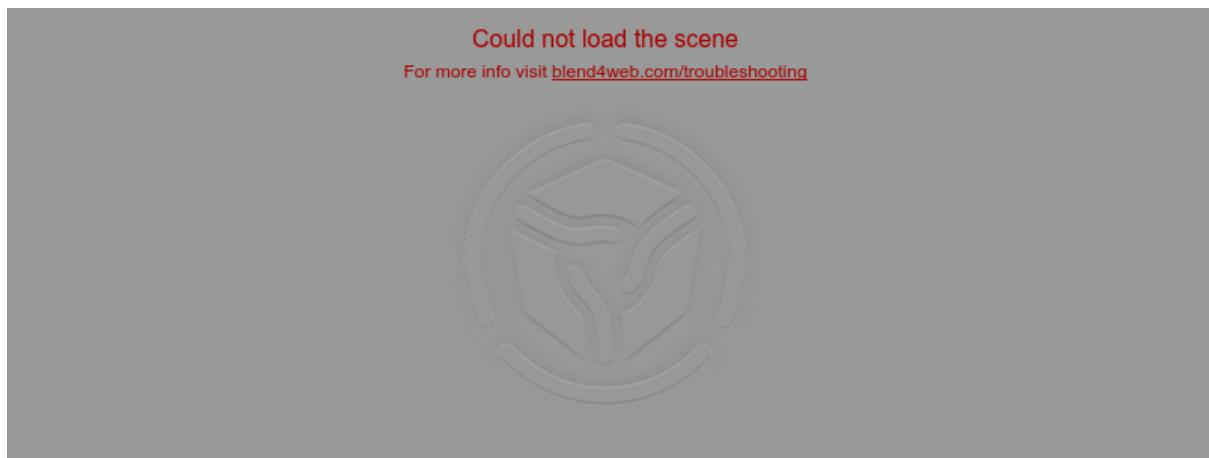
1. для загрузки сцены используется обязательный атрибут `load`, в который помещается относительный путь к JSON файлу.

2. необязательный атрибут `bg` используется для установки фона, заменяющего сцену при возникновении ошибки WebGL.
3. необязательный атрибут `show_fps` применяется для отображения счетчика кадров в секунду в правом верхнем углу плеера.

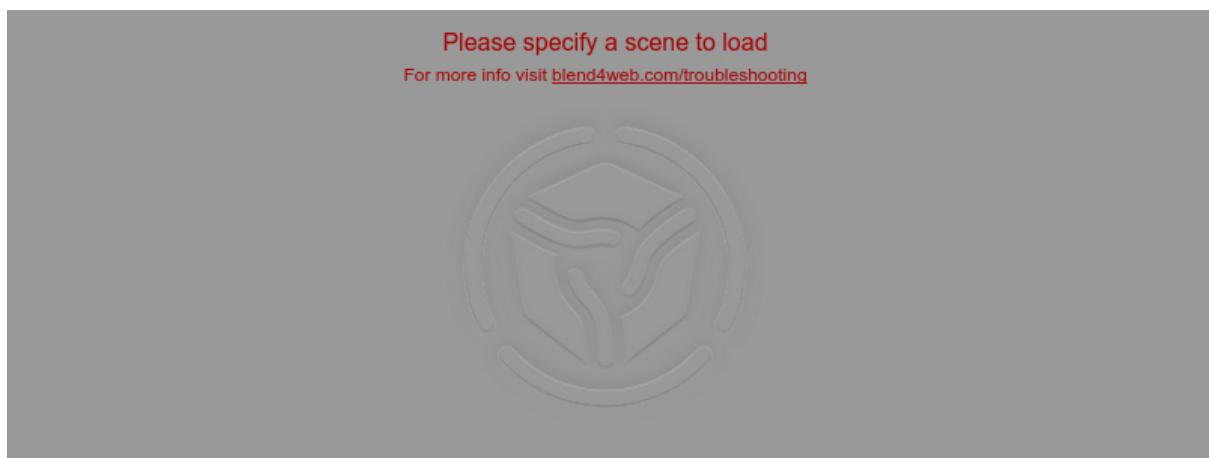
## 7.5 Ошибки сцены

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 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. Warning: Blender version mismatch. Blender VER_REQUIRED is recommended for the Blend4Web addon. Current version is VER_CURRENT.	Ошибка загрузки аддона. Аддон не совместим с платформой PLATFORM. Предупреждение о возможной несовместимости с текущей версией Blender. Для работы рекомендуется версия Blender'a VER_REQUIRED. Текущая версия - VER_CURRENT.

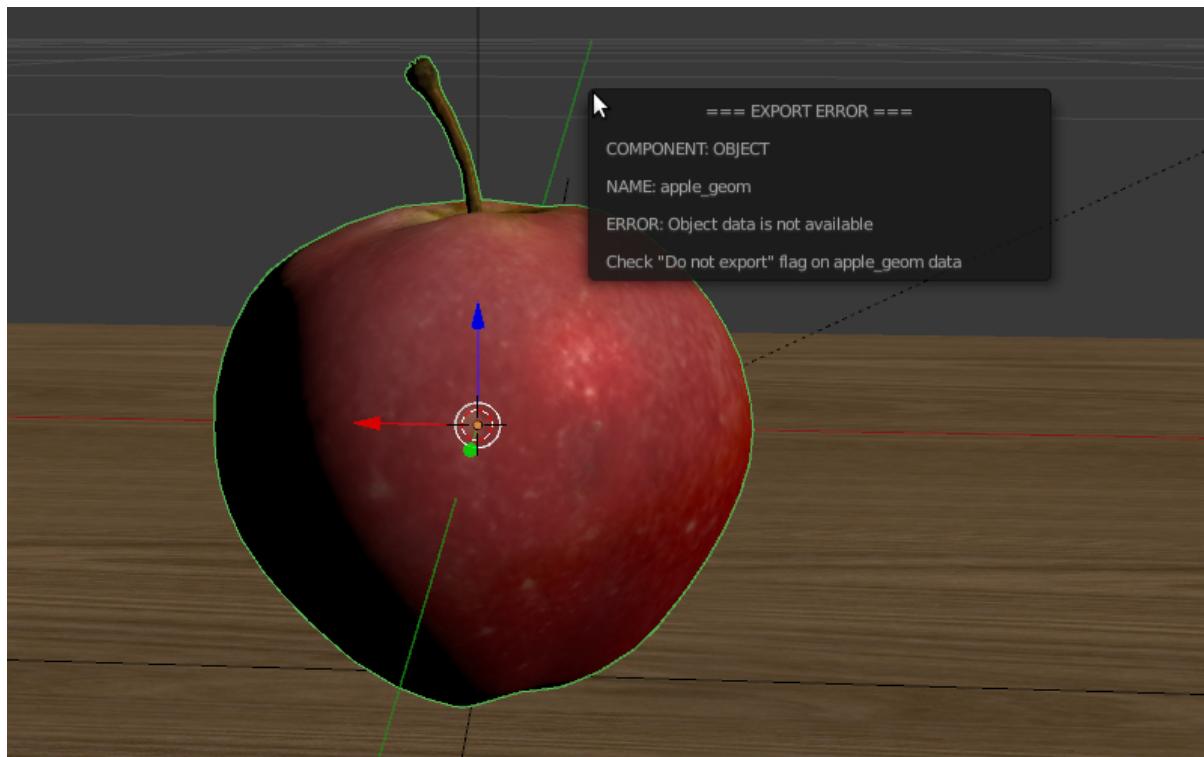
## 8.2 Критические ошибки экспорта

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
Dupli 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 meshes; Check MESH_NAME1 and MESH_NAME2 UV Maps / Vertex colors	Имеются два меша с одинаковым материалом. Такие меша автоматически сливаются в один движком в целях оптимизации, и поэтому должны иметь одинаковое число текстурных разверток (UV Maps) и слоёв вертексовых цветов (Vertex Colors).
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> (т.к. в этом случае при экспорте происходит полное копирование мешей).
Incomplete mesh; Dynamic grass vertex colors required by material settings	Неполный меш: специальный материал для ландшафта использует опции <code>Dynamic grass size</code> и/или <code>Dynamic grass color</code> , но у меша нет слоев вертексового цвета с такими именами.
Incomplete mesh; Material slot is empty	Неполный меш: пустой слот материала.
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 вертексового цвета ( <code>Vertex Color Paint</code> ), но у меша нет слоя вертексового цвета.
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 экспортировался.
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 <code>FROM_NODE</code> и <code>TO_NODE</code> не соответствуют друг другу.
Object constraint has no target	Для ограничителя объекта (вкладка <code>Object Constraints</code> ) не установлено свойство <code>Target Object</code> .
Object data not available; Check the “Do not export” flag on the OBJECT_NAME data	Не доступны данные для объекта. Ошибка, в частности, проявляется, когда у экспортируемого объекта во вкладке <code>Object Data</code> установлено свойство <code>Do not export</code> .
Object-parent relation is not supported; Clear the parent's inverse transform	При использовании отношения родитель-потомок для объекта-потомка требуется сбросить перемещение командой <code>Object &gt; Parent &gt; Clear Parent Inverse (Alt-P)</code> .

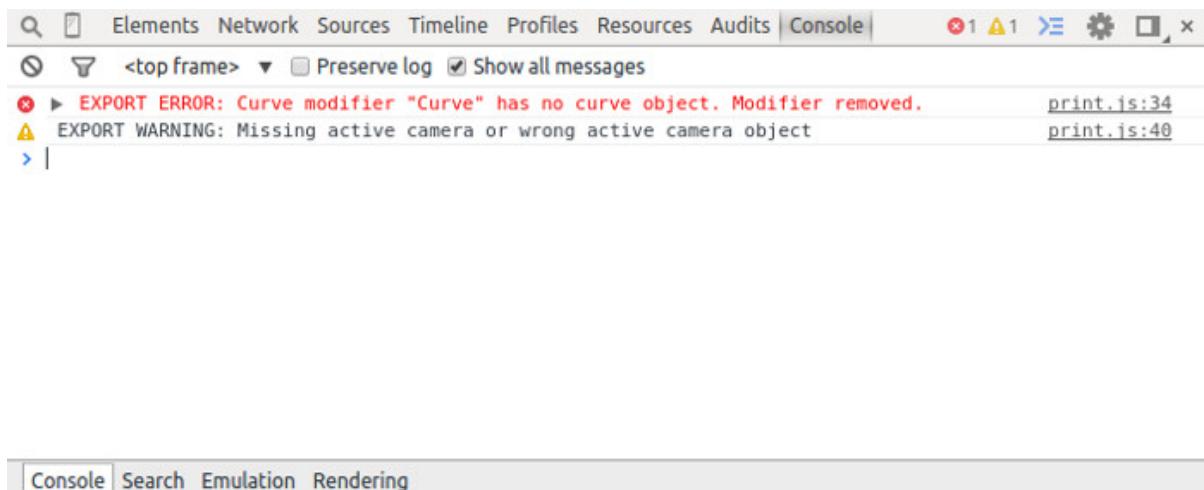
Particle system error; Dupli group isn't specified	Ошибка системы частиц. Не выбрана группа, используемая в качестве частицы.
Particle system error; Dupli object isn't specified	Ошибка системы частиц. Не выбран объект, используемый в качестве частицы.
Particle system error; Dupli object OBJECT_NAME doesn't export	Ошибка системы частиц. Объект OBJECT_NAME, выбранный в качестве частицы, не экспортируется (на нем выбрана опция <code>Do not export</code> ).
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 <code>Do not export</code> , либо объекты имеют неподходящий тип. Поддерживаемые типы: MESH.
Particle system error; 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; The "NAME" vertex color specified in the "to" field is missing in the "OBJECT_NAME" object ("GROUP_NAME" dupli group)	Ошибка системы частиц. Вертексный цвет NAME указанный в поле <code>from</code> , отсутствует в эмиттере OBJECT_NAME. 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.
Particle system error; 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; 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.

Wind bending: vertex colors weren't properly assigned	Настройки процедурной анимации деревьев; должны быть указаны названия всех слоев вертексных цветов ( <b>Main stiffness (A)</b> , <b>Leaves stiffness (R)</b> , <b>Leaves phase (G)</b> , <b>Overall stiffness (B)</b> ), либо только главного ( <b>Main stiffness (A)</b> ), либо ни одного из них.
Wind bending: not all vertex colors exist	Настройки процедурной анимации деревьев: должны существовать все указанные слои вертексных цветов.
Wrong edited normals count; It doesn't match with the mesh vertices count	Число редактируемых нормалей не совпадает с числом вершин меша. Требуется сделать <b>Clean Up</b> либо <b>Save</b> в панели <b>B4W Vertex Normals Editor</b> .
Wrong overridden bounding box; Check the mesh's bounding box values	Указаны неверные размеры при переопределении <b>BoundingBox</b> 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 вертексной анимации, но число вершин покадрово в анимации <b>ANIM_NAME</b> не совпадает с числом вершин меша. Возможное решение - "перезапекание" анимации.

## 8.3 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 (красного цвета) имеет вид:

EXPORT ERROR: Сообщение об ошибке





Сообщение об ошибке	Причина
The NAME action has decimal frames. Converted to integer. 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. Exported UV-layer is missing in node “GEOMETRY”. Material: NAME. Exported UV-layer is missing in texture NAME. [Material: NAME.] The NAME LAMP node has no lamp object. Material: NAME. Ignoring LODs after empty LOD for the NAME object.	Анимация NAME содержит дробные значения кадров. Округлено до целых. В модификаторе NAME типа <b>Armature</b> не указан объект, либо объект не экспортируется. Модификатор удален. В модификаторе NAME типа <b>Curve</b> не указан объект. Модификатор удален. В модификаторе NAME типа <b>Curve</b> указан неподходящий объект. Модификатор удален.  В ноде типа <b>GEOMETRY</b> указан не экспортируемый UV-слой для текстурных координат типа UV. В текстуре указан не экспортируемый UV-слой для текстурных координат типа UV.
Incorrect NLA script, falling back to simple sequential NLA.  No image in the NAME texture. [Material: NAME.] No texture in the texture slot. Material: NAME. No texture in the NAME world texture slot. No texture for the NAME particle settings texture slot. Only 2 UV textures are allowed for a mesh; The mesh has N UVs.	В ноде NAME типа <b>LAMP</b> не указан подходящий объект. В списке LOD объектов, настроенных для объекта NAME, были проигнорированы все LOD объекты, следующие за пустым. Некорректный NLA-скрипт. Вместо него будет использоваться стандартная NLA-анимация. У текстуры отсутствует изображение.  В текстурном слоте материала отсутствует текстура. В текстурном слоте объекта <b>World</b> отсутствует текстура. В текстурном слоте системы частиц отсутствует текстура. Движком поддерживаются только до 2 UV текстур на каждый меш. Меш содержит UV текстуры в количестве N.
The NAME node is not supported. The NAME material will be rendered without nodes. Material: NAME.  Using “REFRACTION” node with incorrect type of Alpha Blend. Material: NAME.	Нода с данным именем не поддерживается движком, поэтому нодовый материал будет отключён. Чаще всего проблемы подобного рода возникают при использовании нод Cycles. Используется нодовый материал с неправильно заданным свойством Alpha Blend. Допускается значение <b>Alpha sort</b> , <b>Alpha blend</b> и <b>Add</b> при использовании ноды “REFRACTION”.
Wrong texture coordinates type in texture NAME. [Material: NAME.]	Для текстур с изображением (image) поддерживаются следующие типы координат: <b>UV</b> , <b>Normal</b> и <b>Generated</b> .

## 8.4 Прочие сообщения, требующие внимания пользователя

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

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.

## 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)

### 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.

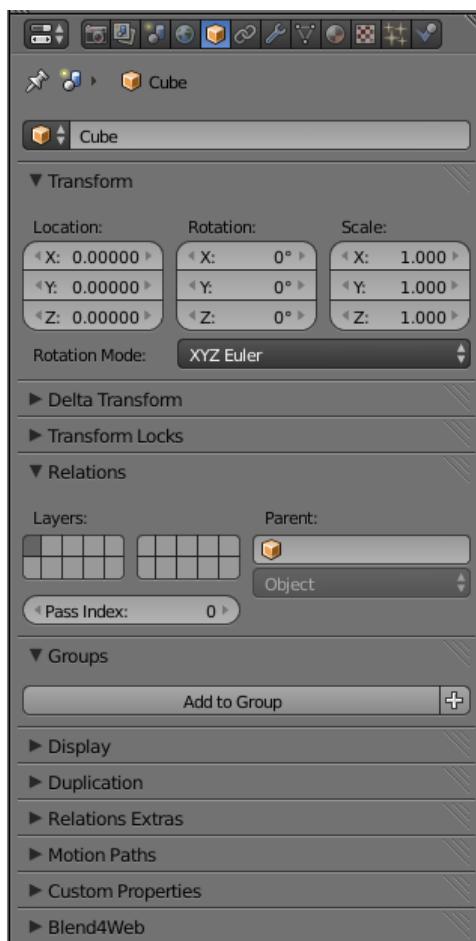
Объекты, имеющие анимацию, физику или родителя, являющегося динамическим объектом, рассматриваются как динамические.

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 > Do not batch`.

## 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



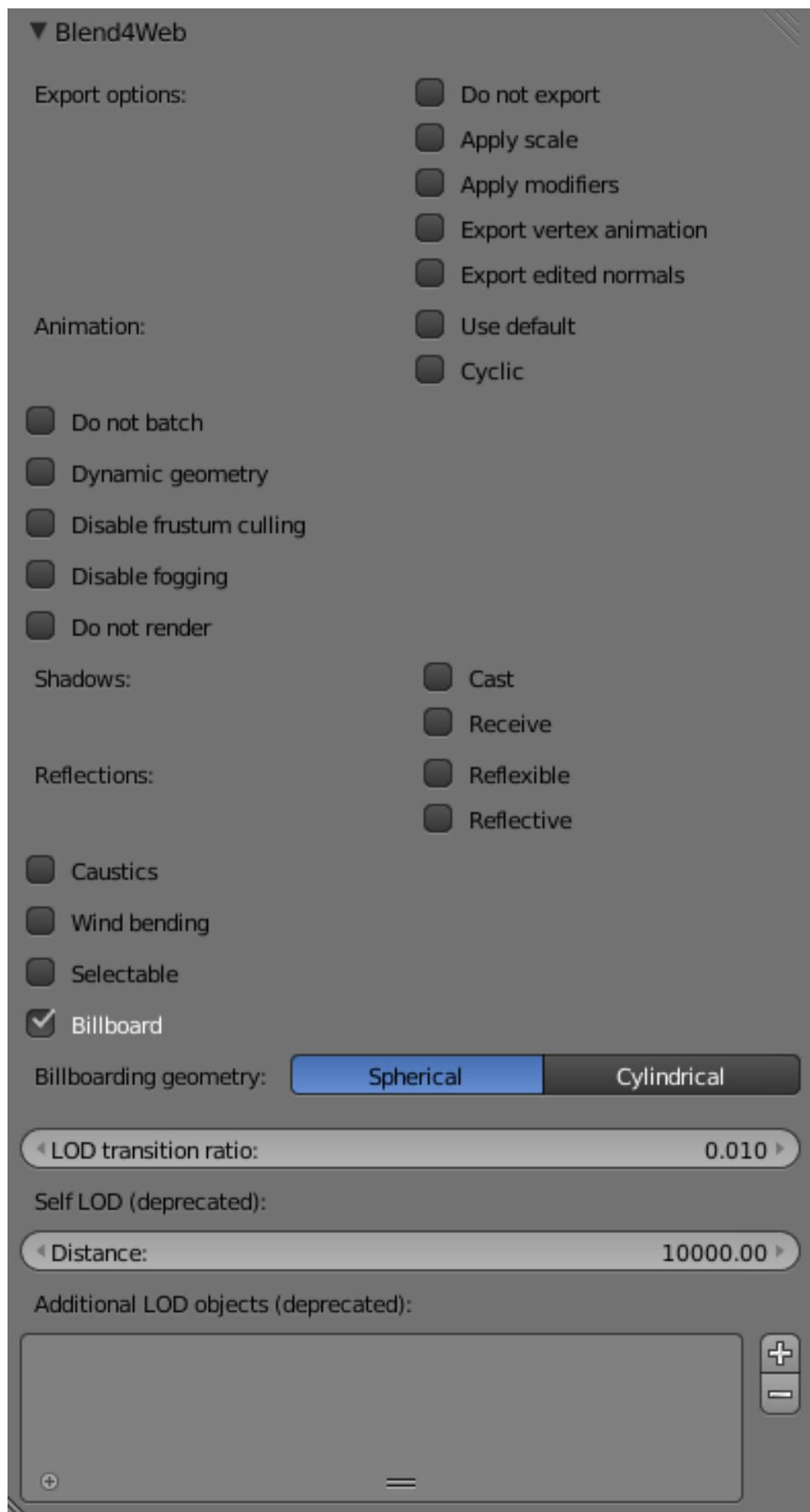
**Transform > Location** Координаты местоположения.

**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 экспортировать.

**Blend4Web > Apply scale** Применить масштабирование объекта при экспорте.  
Также применяет модификаторы.

**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** Экспортировать предварительно созданную и сохраненную вертекскую анимацию.

**Blend4Web > Export edited normals** Экспортировать предварительно отредактированные и сохраненные нормали.

**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, Cyclic - циклически повторять.

**Blend4Web > Do not batch** Принудительно сделать объект *динамическим*.

**Blend4Web > 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, а также его подсвечивание ([эффект Glow](#)).

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

**Blend4Web > Billboarding geometry** 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.

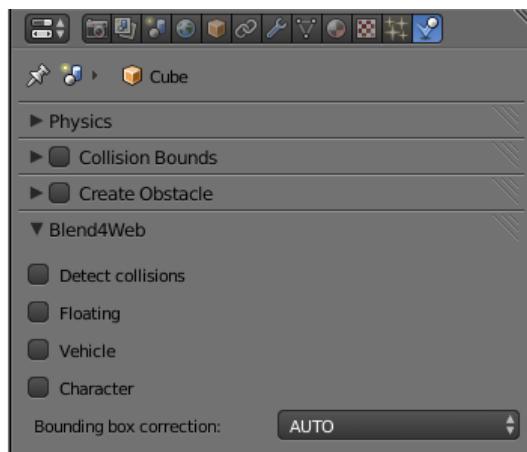
**Blend4Web > Self LOD > Distance** Расстояние от камеры, на котором объект перестает отображаться.

Не рекомендуется, начиная с версии 14.06: реализовано в стандартном инструменте Blender Levels of Detail.

**Blend4Web > Additional LOD objects** Интерфейс добавления низкополигональных объектов для реализации переключения уровня детализации.

Не рекомендуется, начиная с версии 14.06: реализовано в стандартном инструменте Blender Levels of Detail.

### 9.3.2 Вкладка Physics (в режиме Blender Game)



*Blend4Web > Detect collisions* Активировать связанную с объектом физику.

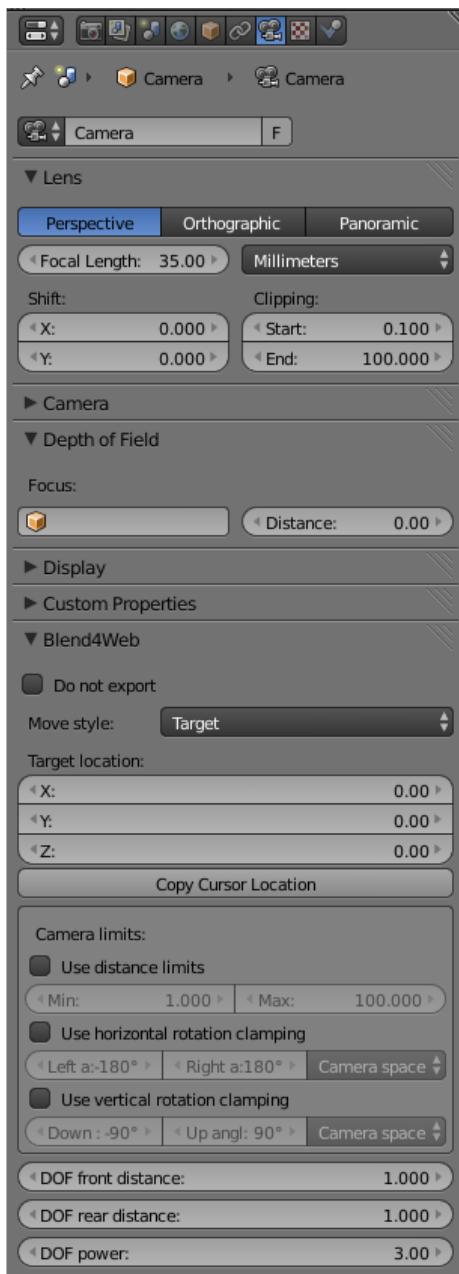
*Blend4Web > Floating* Сделать объект плавучим. Подробное описание настроек плавучих объектов можно найти в разделе *физика*.

*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 Камера

Настройки камеры выставляются в панели Properties на вкладке Object Data.



*Blend4Web > Move style* – режим управления камерой. По умолчанию камера находится в статическом режиме (**Static**), допуская изменение своего положения только через API. В режиме **Target** камера вращается вокруг фиксированной точки. Режим **Eye** позволяет осуществлять вращение и перемещение от первого лица.

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

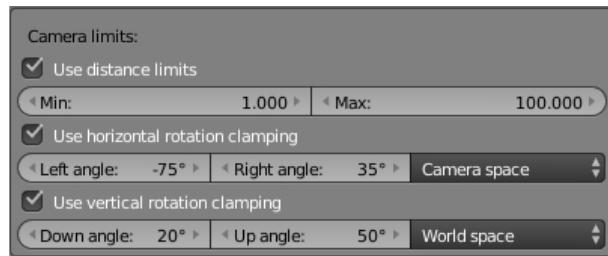
*Blend4Web > DOF front distance* – описано в разделе [Постпроцессинговые эффекты](#)

*Blend4Web > DOF rear distance* – описано в разделе *Постпроцессинговые эффекты*

*Blend4Web > DOF power* – описано в разделе *Постпроцессинговые эффекты*

### 9.4.1 Ограничения движения камеры

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

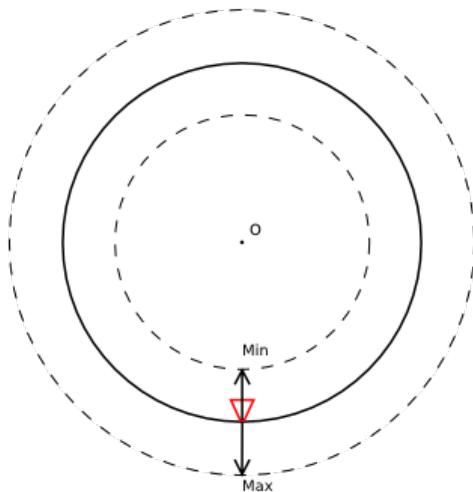


#### Виды ограничений

*Blend4Web > Use distance limits* – доступно в режиме **Target**. Задание минимального и максимального расстояния от камеры до точки вращения.

Варианты интервалов:

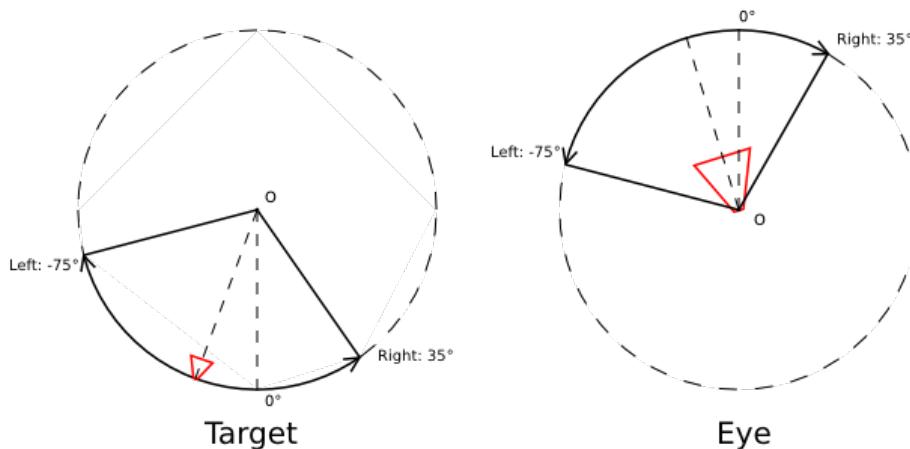
- $Max > Min$  - расстояние от камеры до объекта ограничено интервалом  $[Min, Max]$
- $Max = Min$  - фиксирование камеры на определенной высоте над объектом
- $Max < Min$  - ограничение движения отсутствует



Значения по умолчанию:  $Min = 1$ ,  $Max = 100$ .

*Blend4Web > Use horizontal rotation clamping* – доступно в режимах Target и Eye. Ограничение горизонтального вращения камеры относительно точки вращения (в режиме Target) или относительно своего местоположения (в режиме Eye).

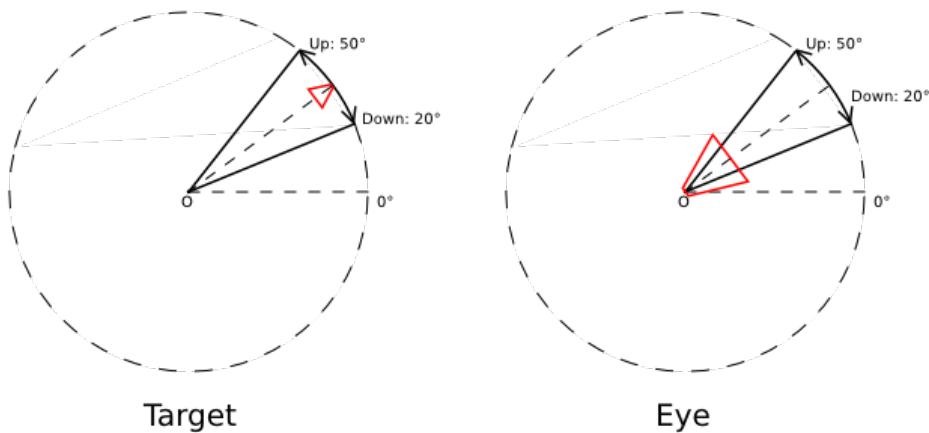
Направление от Left к Right считается положительным и для режима Target совпадает с направлением против часовой стрелки, а для режима Eye - по часовой стрелке:



Значения по умолчанию:  $Left = -180$ ,  $Right = 180$ .

*Blend4Web > Use vertical rotation clamping* – доступно в режимах Target и Eye. Ограничение вертикального вращения камеры относительно фиксированной точки (в режиме Target) или относительно своего местоположения (в режиме Eye).

Направление от Down к Up считается положительным:

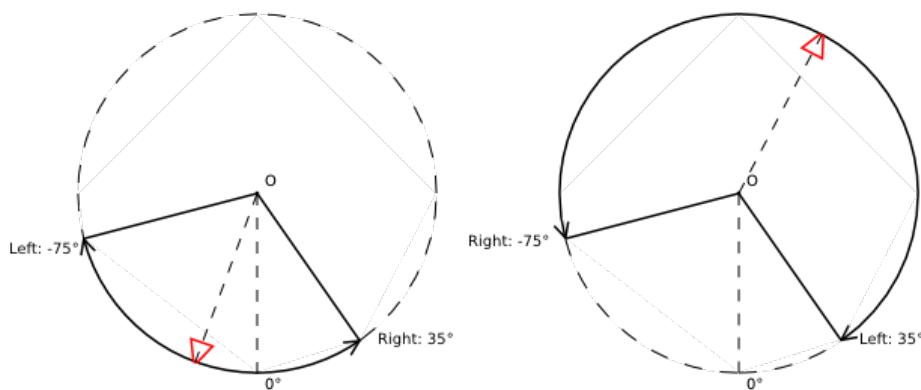


Если включена опция *Use horizontal rotation clamping*, то вертикальное вращение будет ограничено, по крайней мере, интервалом  $[-90, 90]$ .

Значения по умолчанию:  $Down = -90$ ,  $Up = 90$ .

### Особенности значений, ограничивающих поворот

- Перемена мест значений *Left/Right* или *Down/Up* приводит к движению по противоположной дуге окружности вращения.



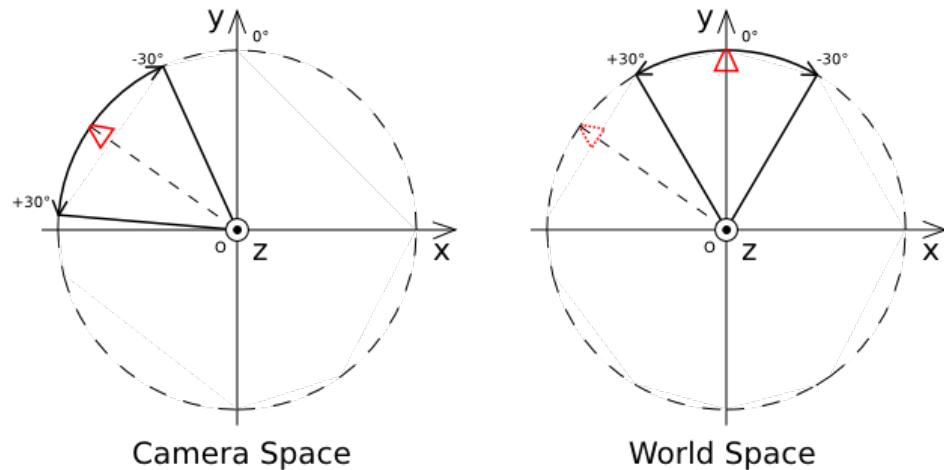
- $Left = Right$ ,  $Up = Down$  - фиксирование камеры соответственно по горизонтали или вертикали.

### Отсчет углов поворота

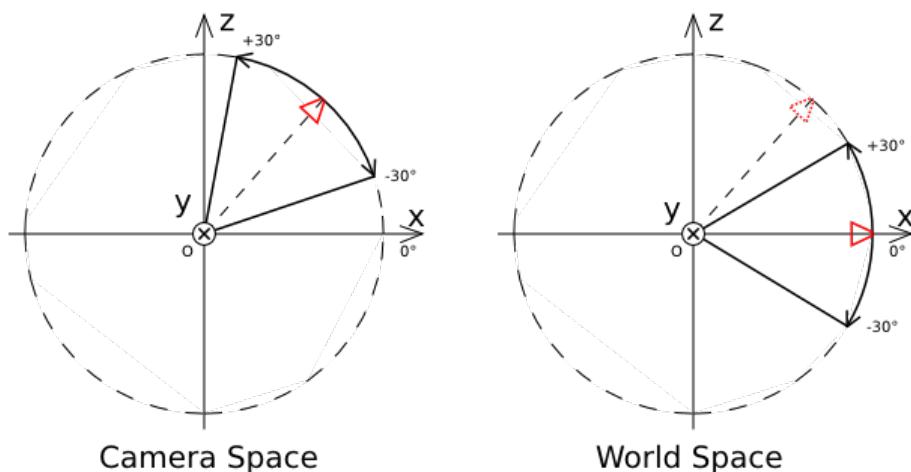
При ограничении горизонтального и вертикального вращения можно выбрать пространство координат:

- **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 горизонтальной плоскости XOY в мировом пространстве.

Для ограничения по горизонтали:



Для ограничения по вертикали:



Оси координат, отмеченные на рисунках, совпадают с осями мировых координат в Blender'e.

По умолчанию назначается вариант **Camera space**.

### Материалы

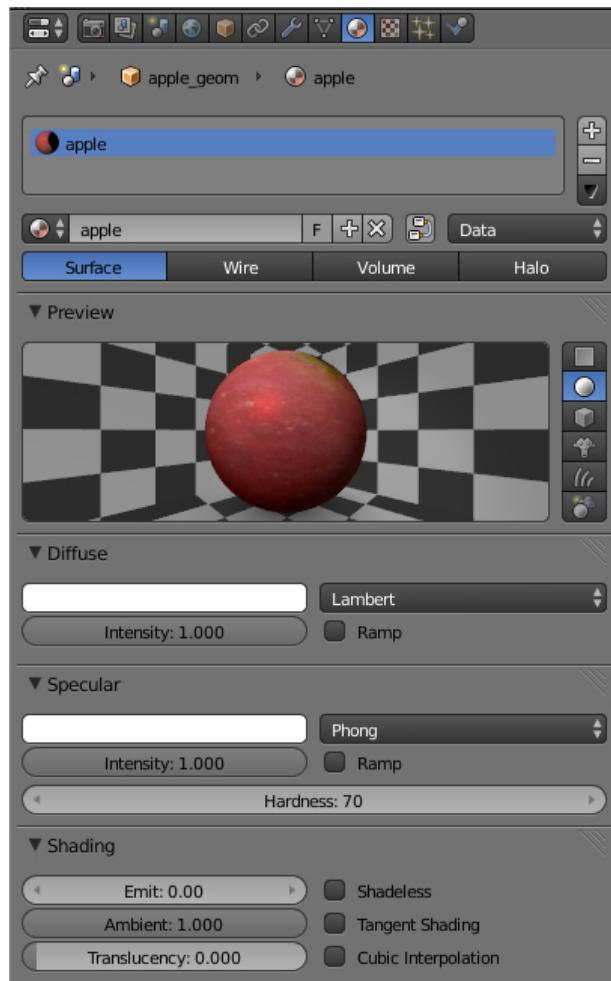
---

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).

## 10.1 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.

## 10.2 Прозрачность

### 10.2.1 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:

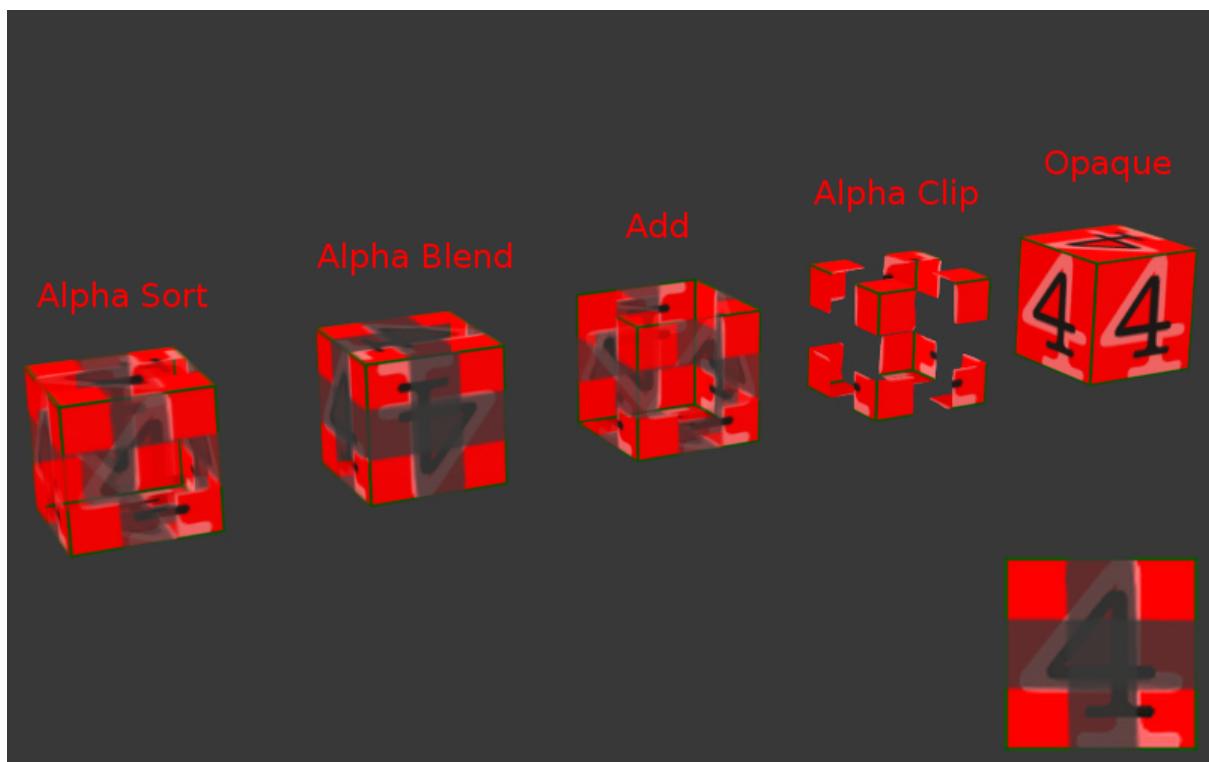
***Alpha Sort*** Прозрачный с градиентом. Для корректного отображения перекрывания одних прозрачных поверхностей другими движком производится сортировка треугольников по дальности от камеры. Операция требует дополнительных затрат вычислительных ресурсов. Рекомендуется применять для замкнутых прозрачных 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.



## 10.2.2 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.

## 10.3 Зеркальное отражение

### 10.3.1 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*

### 10.3.2 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*

### 10.3.3 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  $\theta = 0$ ),

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

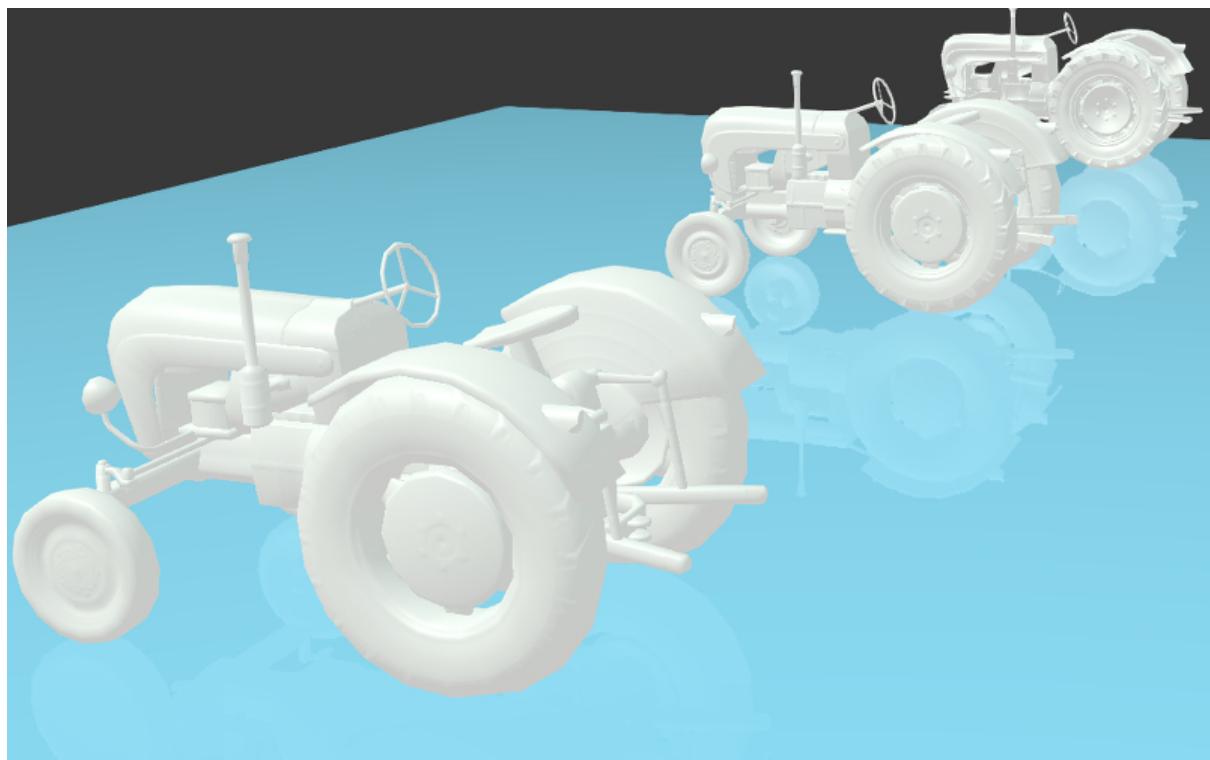
$N$  - показатель степени.

#### Настройка

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

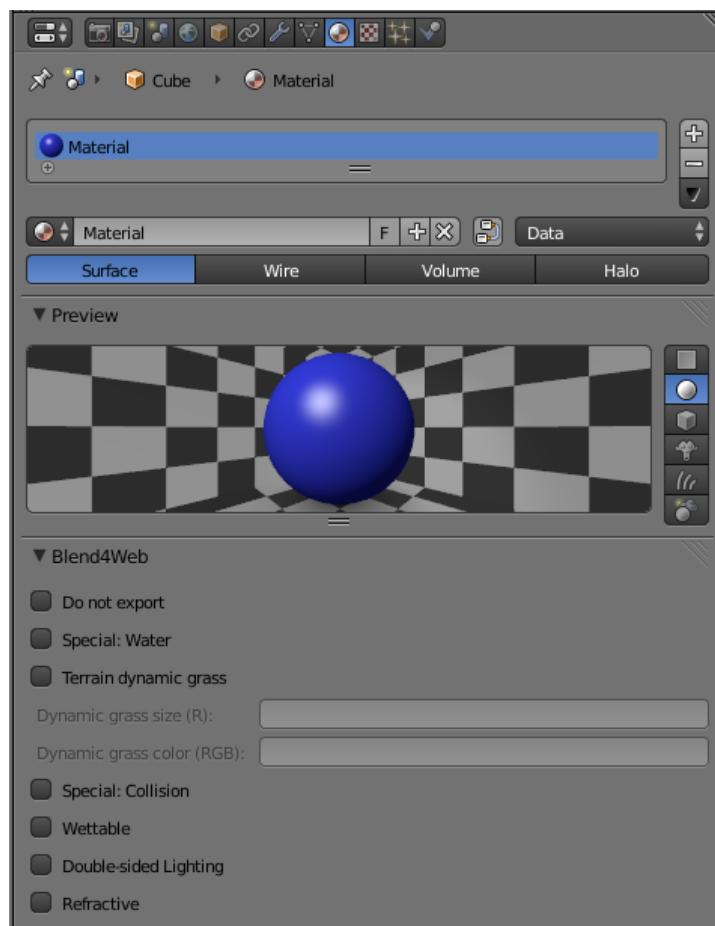
**Mirror > Fresnel** Степень Френеля для отражения. Показатель степени  $N$  в формуле Шлика. В пакете Blender ограничен значениями от 0 до 5. Если этот параметр равен нулю, эффект Френеля не проявляется, происходит *полное отражение* на всех углах. Если этот параметр больше нуля, при обзоре поверхности 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.



## 10.4 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:**

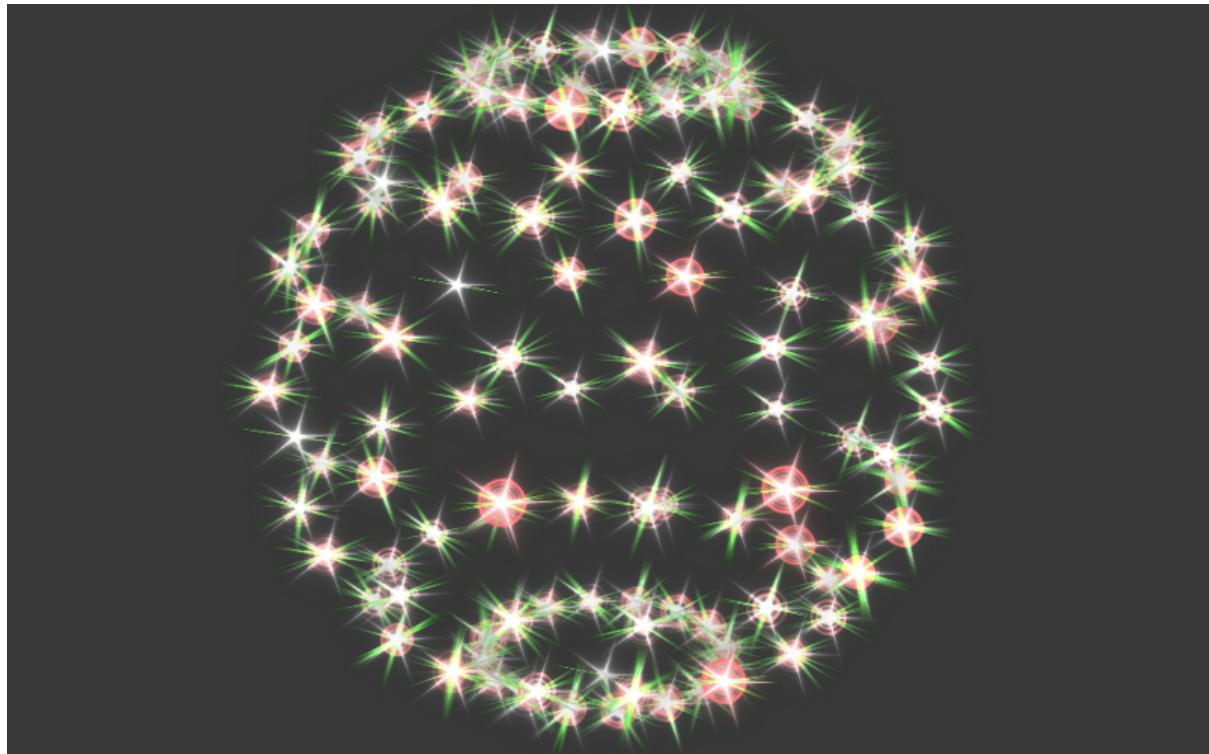
[Прозрачность](#)

## 10.5 Материалы гало (Halo)

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

### 10.5.1 Активация

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



### 10.5.2 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*.

### 11.1 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)*

#### 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, в движке генерируется серая текстура. Также используется для *рендеринга сцены в текстуре*.

#### 4. Blend, градиент

- используется в *системах частиц*

#### 5. Voronoi, процедурная текстура с разбиением Вороного

- используется для *рендеринга воды* с целью настройки каустики

## 11.2 Общие настройки

**Размер** Размер растров 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 экспортовать текстуру. По умолчанию отключено.

**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 > UV translation velocity** Скорость анимации текстурных координат по соответствующим осям. Значения по умолчанию 0.0.

Не рекомендуется, начиная с версии 14.10: указанная функциональность может быть реализована в нодовых материалах с помощью ноды TIME.

**Blend4Web > Water Foam** Текстура пены. Используется материалом для рендеринга воды.

## 11.3 Диффузная текстура (diffuse map)

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

### 11.3.1 Активация

Выставить опцию Diffuse > Color на панели Textures > Influence.

### 11.3.2 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.

## 11.4 Карта бликов (specular map)

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

### 11.4.1 Активация

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

### 11.4.2 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.

## 11.5 Карта нормалей (normal map)

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

### 11.5.1 Активация

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

### 11.5.2 Дополнительные настройки

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

## 11.6 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.

### 11.6.1 Активация

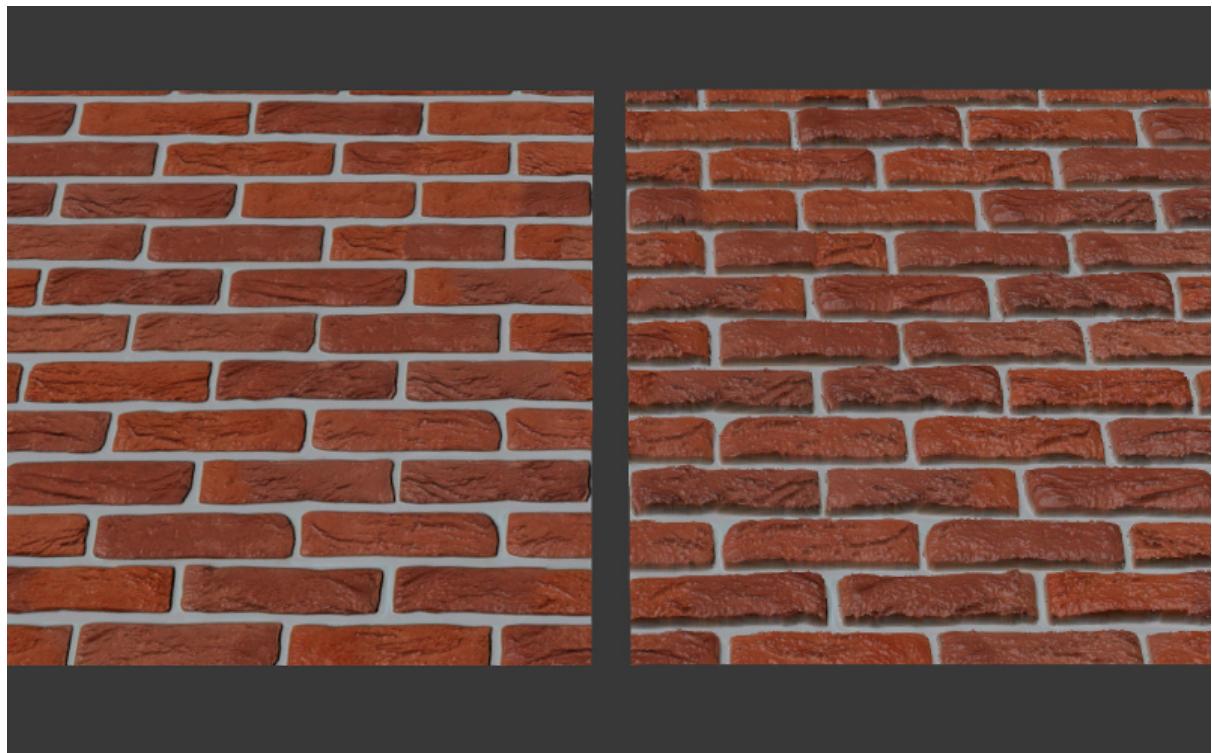
Для карты нормалей дополнительно к опции **Geometry > Normal** на панели **Textures > Influence** выставить опцию **Parallax** на панели **Textures > Blend4Web**.

### 11.6.2 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.



## 11.7 Карта смещивания (stencil map)

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

### 11.7.1 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**.

### 11.7.2 Дополнительные настройки

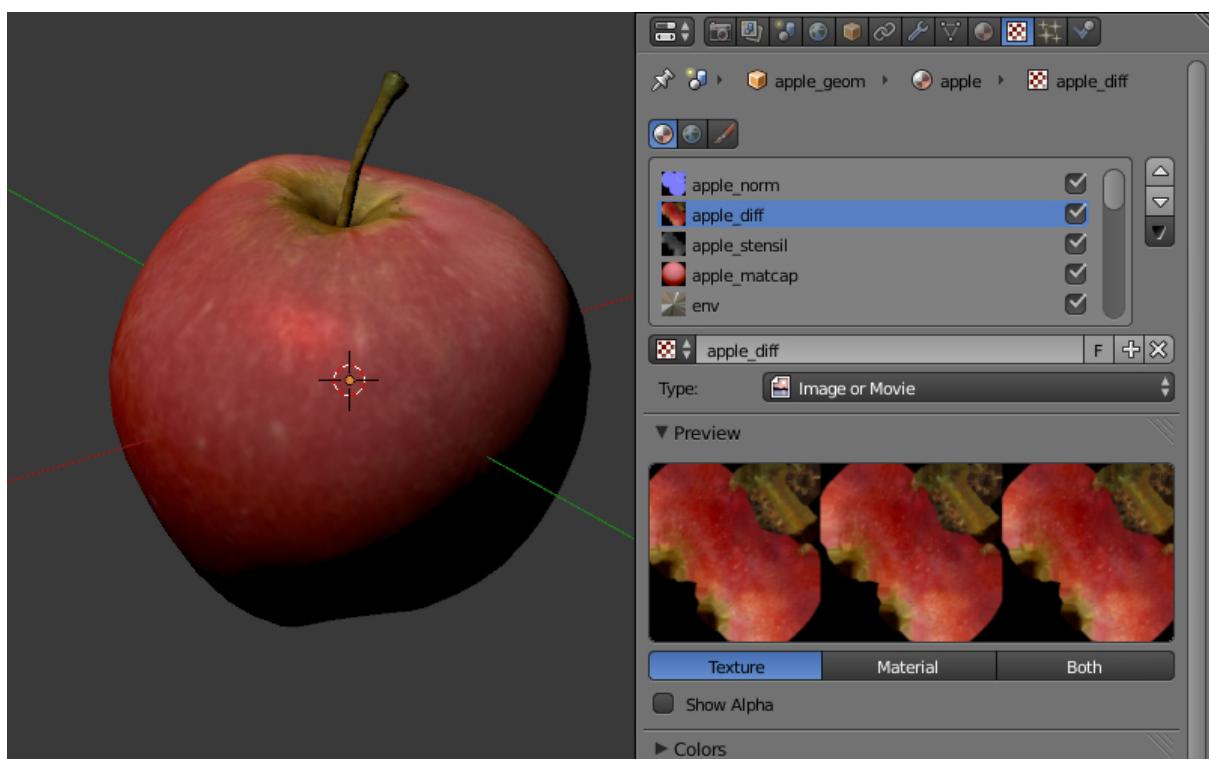
В случае обычных материалов для одной из смешируемых диффузных текстур поддерживается тип текстурных координат **Normal** (“matcap”).

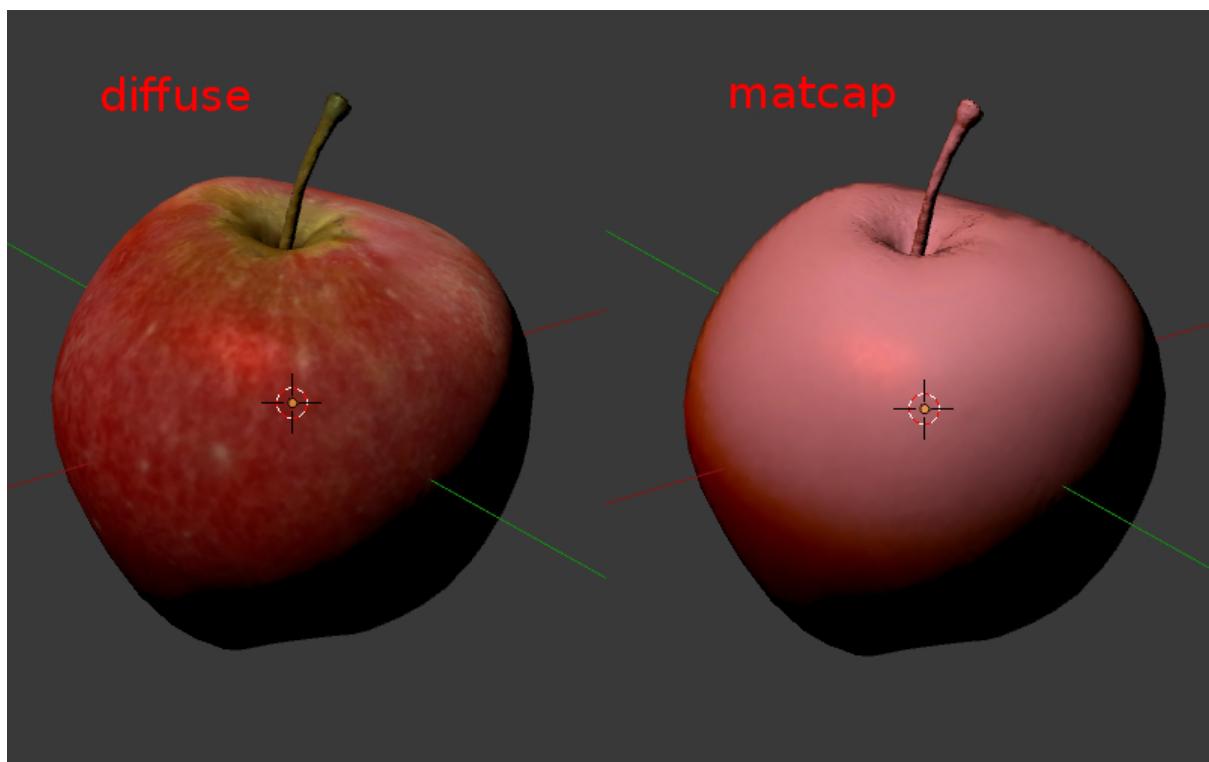
### 11.7.3 Ограничения

В случае обычных материалов движком интерпретируется только красный канал текстуры смеширования. 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.

### 11.7.4 Пример

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



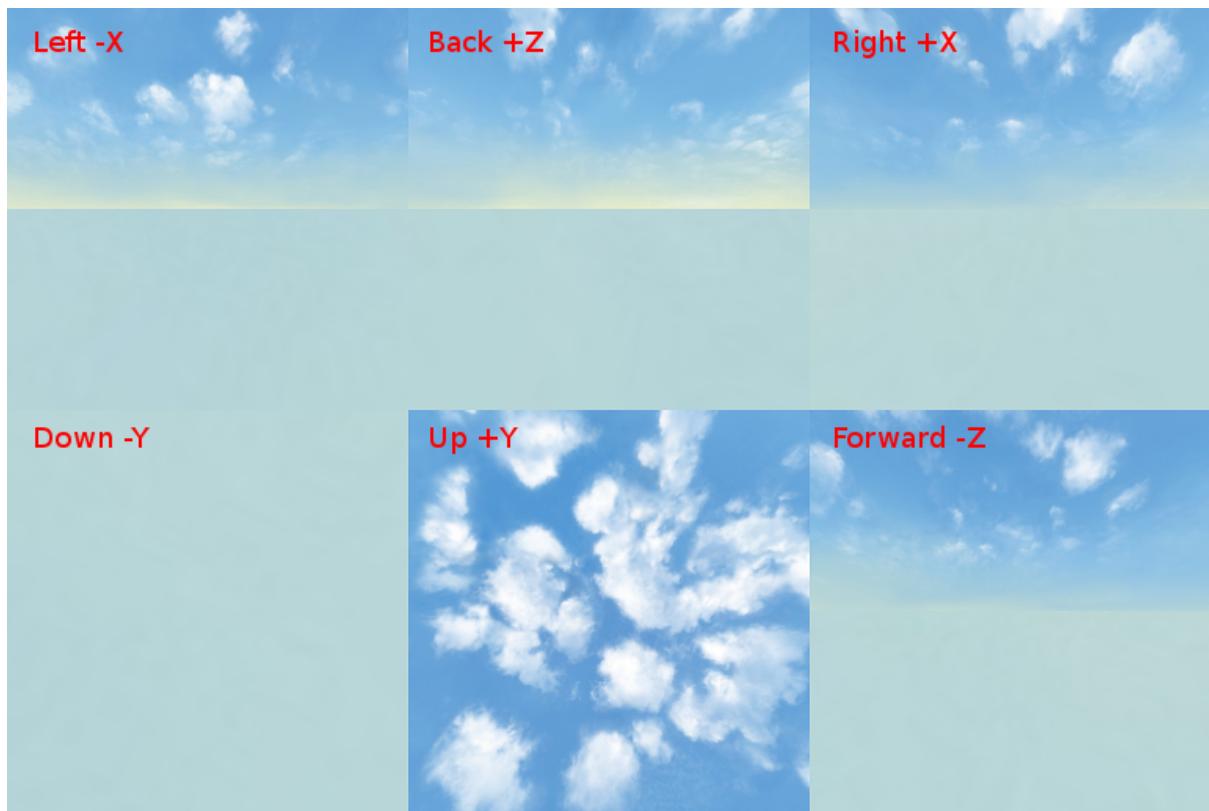


## 11.8 Карта окружения (environment map)

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.).

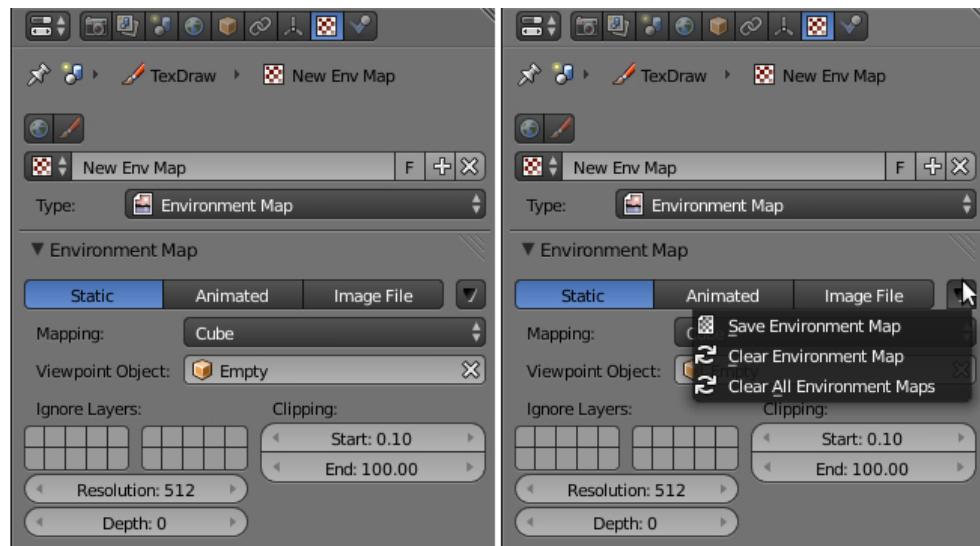
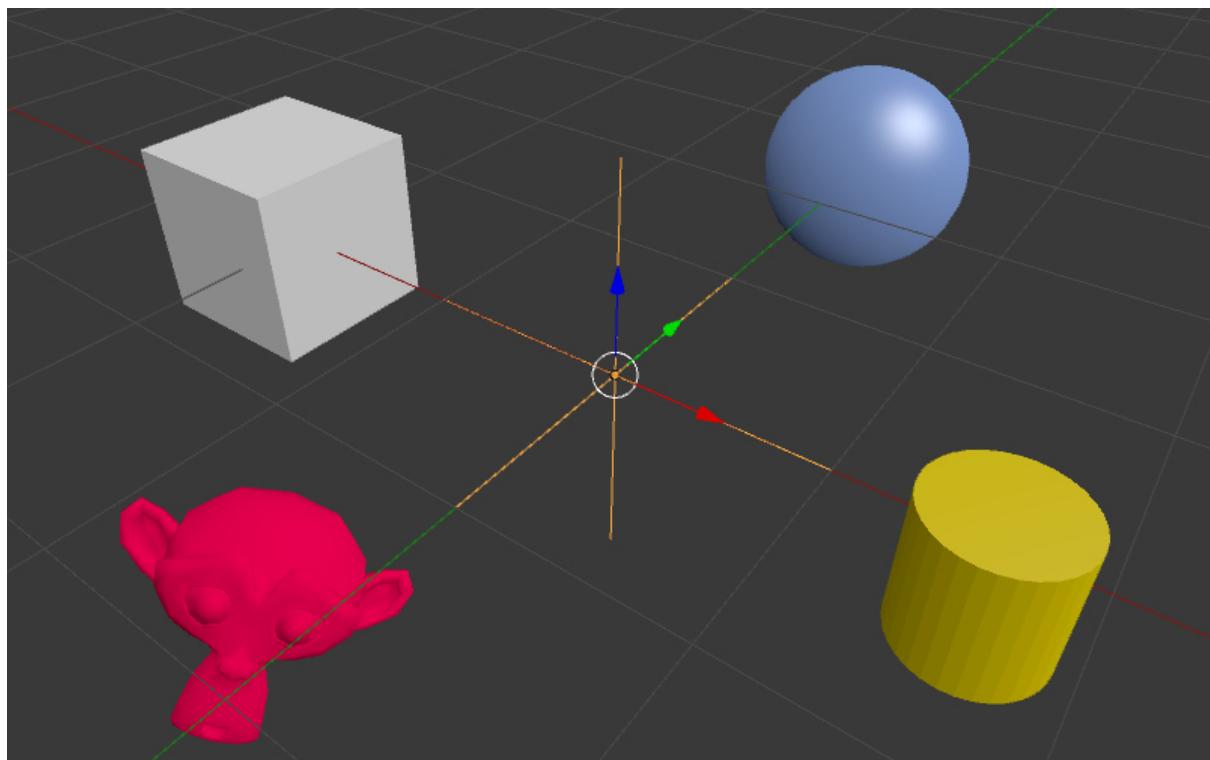
Во избежание проявления швов рекомендуется использовать формат без потери качества (PNG).



### 11.8.1 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.



## 11.9 Карта зеркального отражения (mirror map)

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

### 11.9.1 Активация

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

### 11.9.2 Дополнительные настройки

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

См.также:

*Статическое отражение.*

## 11.10 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*.

### 11.10.1 Активация

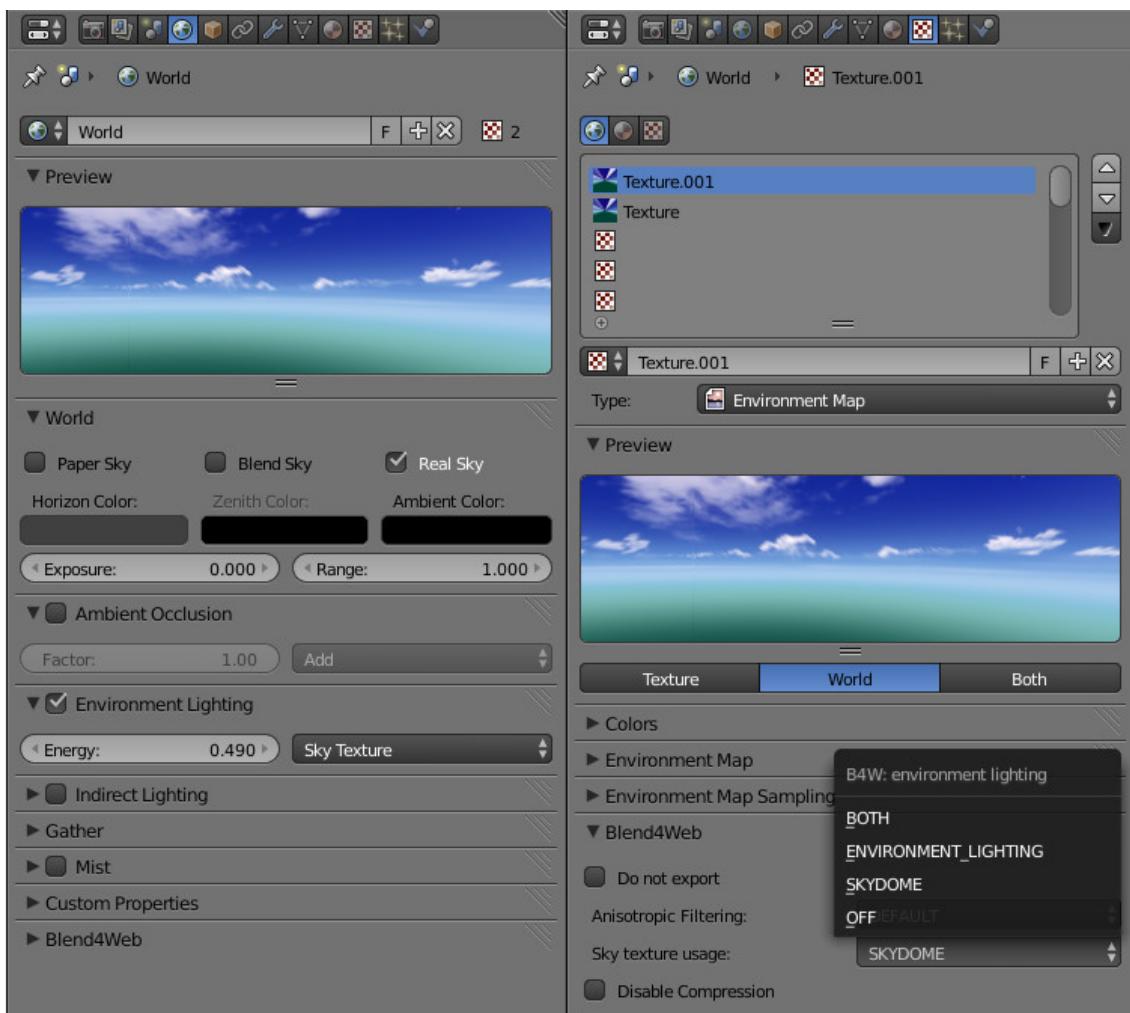
Создать текстуру мира (world texture) с типом “Environment Map”. Выбрать опцию Blend4Web > Sky texture usage > SKYDOME.

---

**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.

---



## 11.11 Рендеринг в текстуру (render-to-texture, RTT)

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

### 11.11.1 Активация

- Создать дополнительную сцену-источник, переименовать для удобства, создать **World**, добавить нужные объекты, настроить вид из камеры.
- В главной сцене для текстуры целевого объекта выставить тип **None**, в поле **Blend4Web > Source scene** указать название сцены-источника. В меню **Mapping > Coordinates** выбрать **UV**. Убедиться, что меш объекта имеет развертку.



### 11.11.2 Ограничения

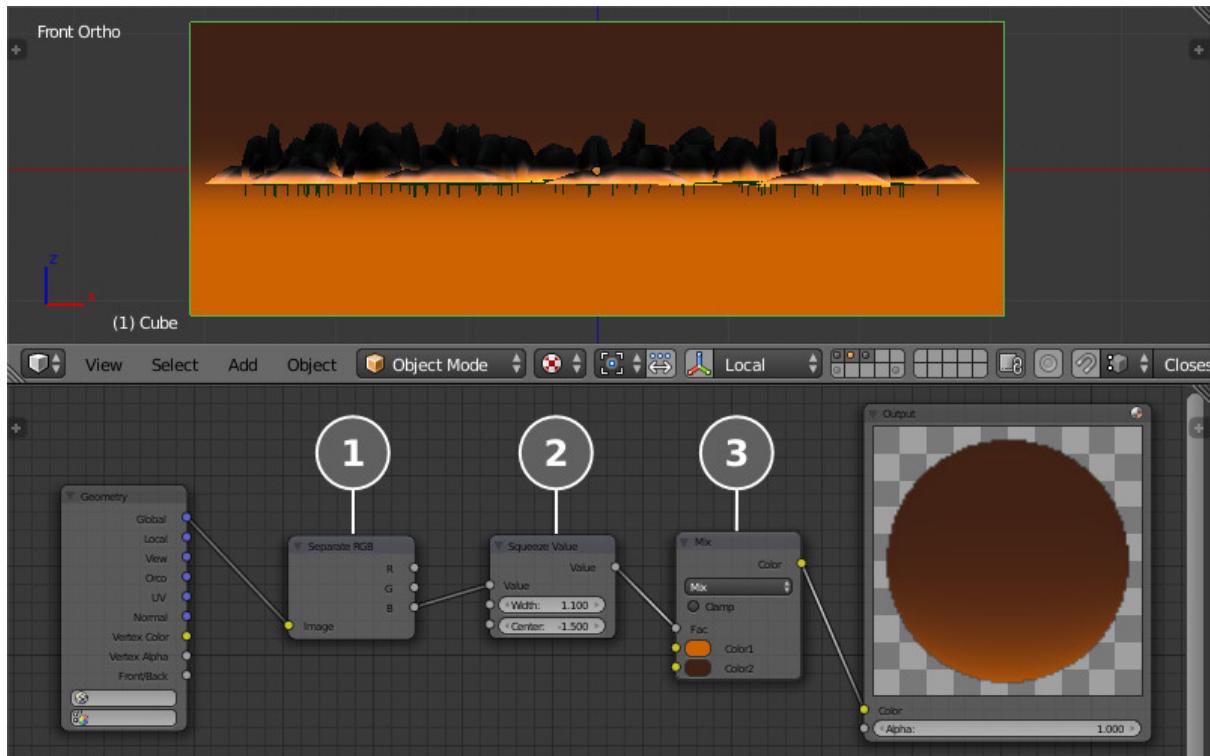
В настоящее время имеется баг, вынуждающий иметь в обеих сценах один общий источник света.

---

## Нодовые материалы

---

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



### 12.1 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**.

## 12.2 Дополнительные ноды

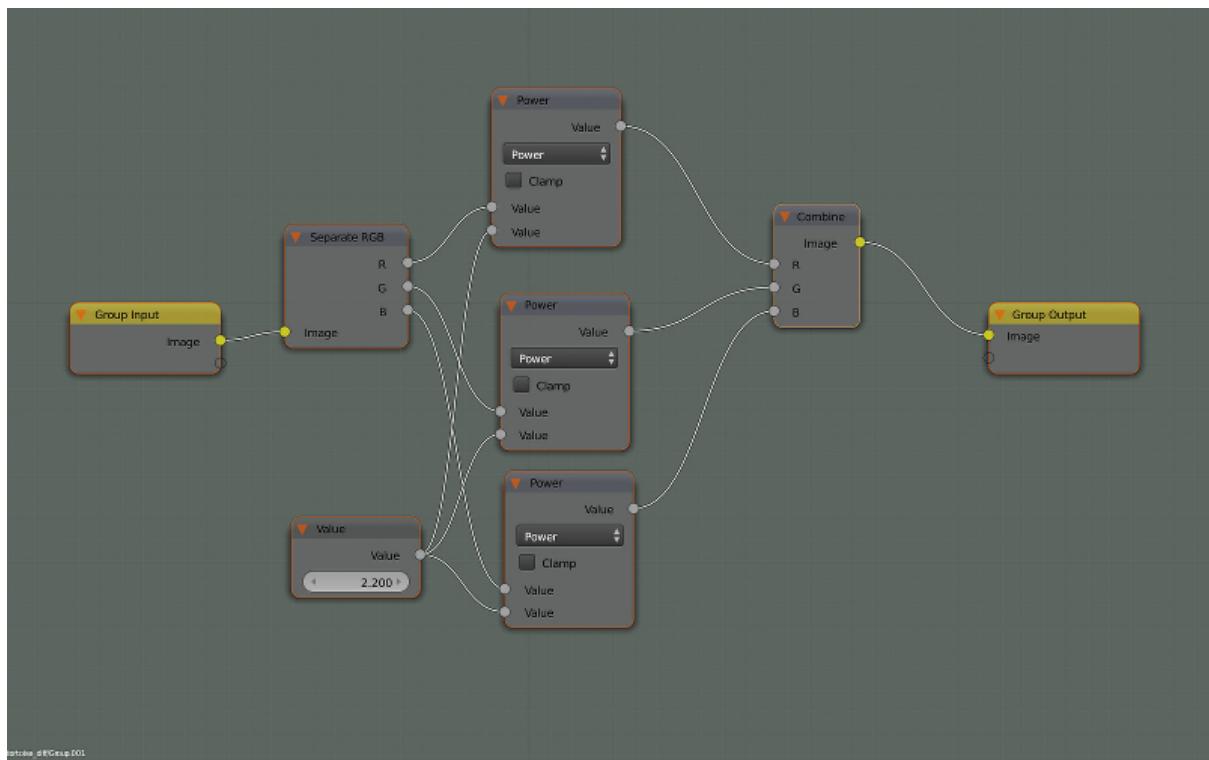
Дополнительные ноды расширяют функционал стандартных с учётом специфики работы движка. Ноды оформляются в виде нодовых групп (**Node groups** или **Node tree**) со специально выбранным именем и форматом входов. Для удобства, все дополнительные ноды собраны в файл `external/blender/dev_nodes/special_nodes.blend`.

**См.также:**

*Анимация ноды Value*

### 12.2.1 LINEAR\_TO\_SRGB и SRGB\_TO\_LINEAR

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

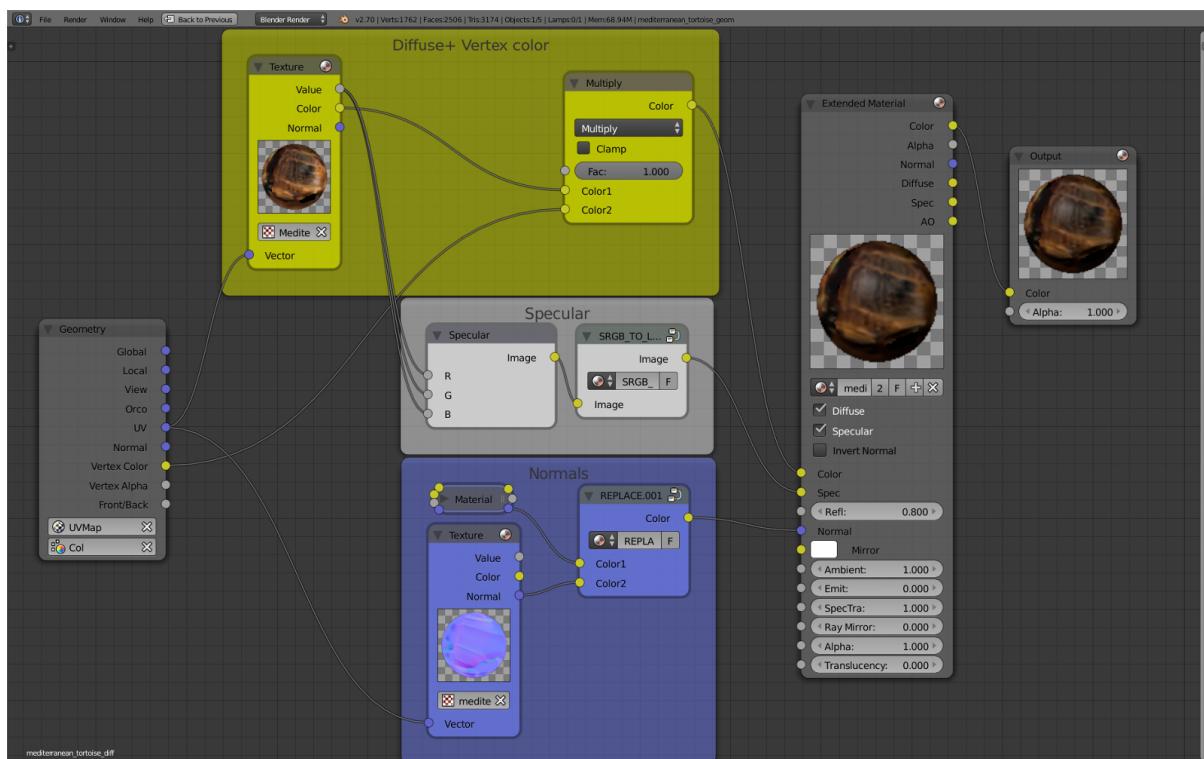


См.также:

*Коррекция в нодовых материалах*

## 12.2.2 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'e 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.

### 12.2.3 CLAMP

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

### 12.2.4 TIME

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

### 12.2.5 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**.

## 12.2.6 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.

## 12.2.7 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**.

## 12.2.8 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*

## 12.2.9 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**.

---

## 12.2.10 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".

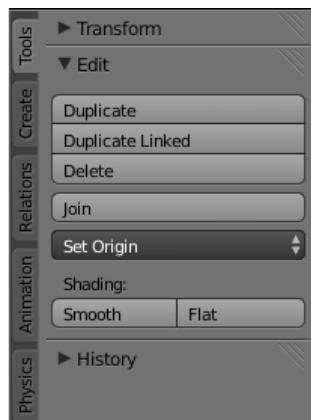
---

## Освещение, тени и фон

---

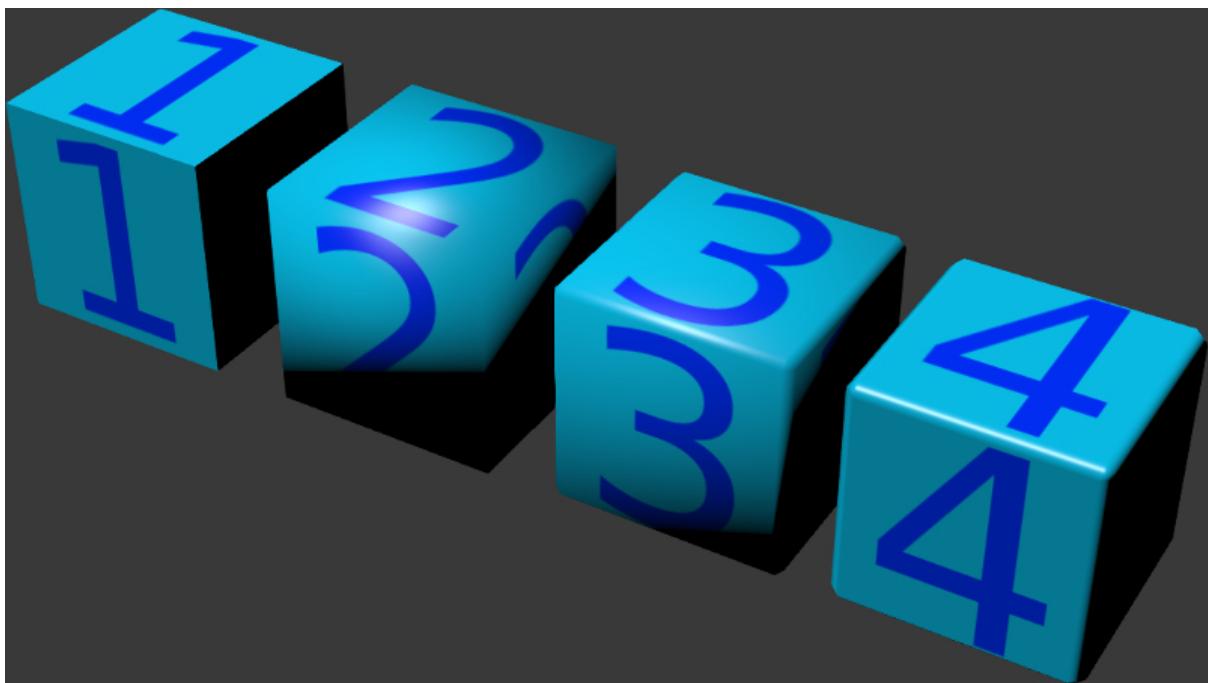
### 13.1 Типы затенения

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:



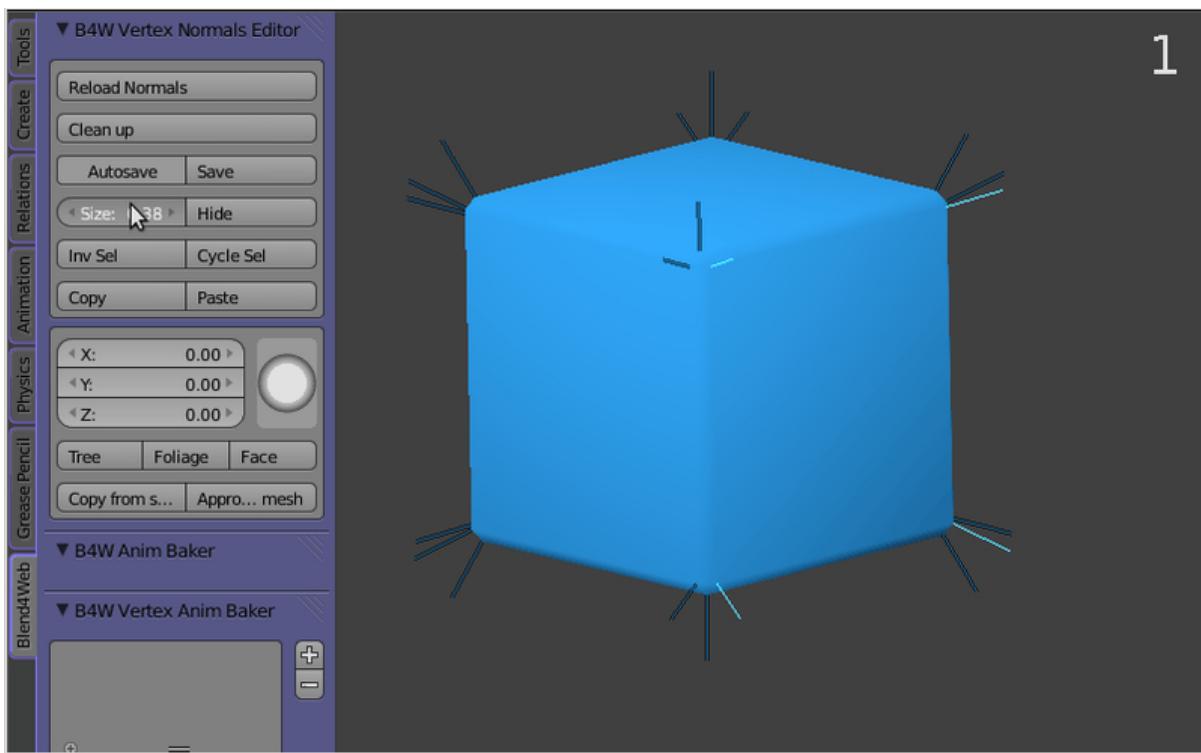
1. Flat Shading
2. Smooth Shading
3. Smooth Shading + фаска
4. Smooth Shading + фаска + редактирование нормалей

## 13.2 Редактор нормалей

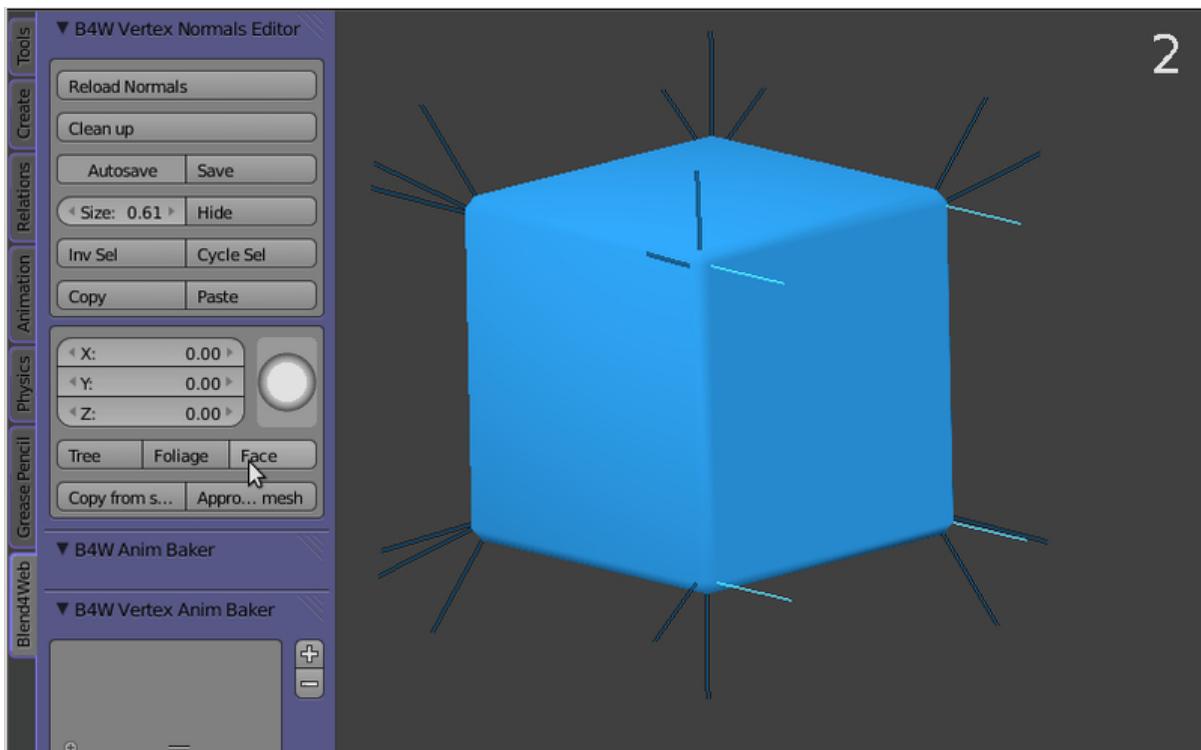
С помощью редактора нормалей можно достичь результат, схожий в результате применения *карт нормалей*. При этом метод редактирования нормалей является более предпочтительным с точки зрения потребления вычислительных ресурсов и видео-памяти.

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

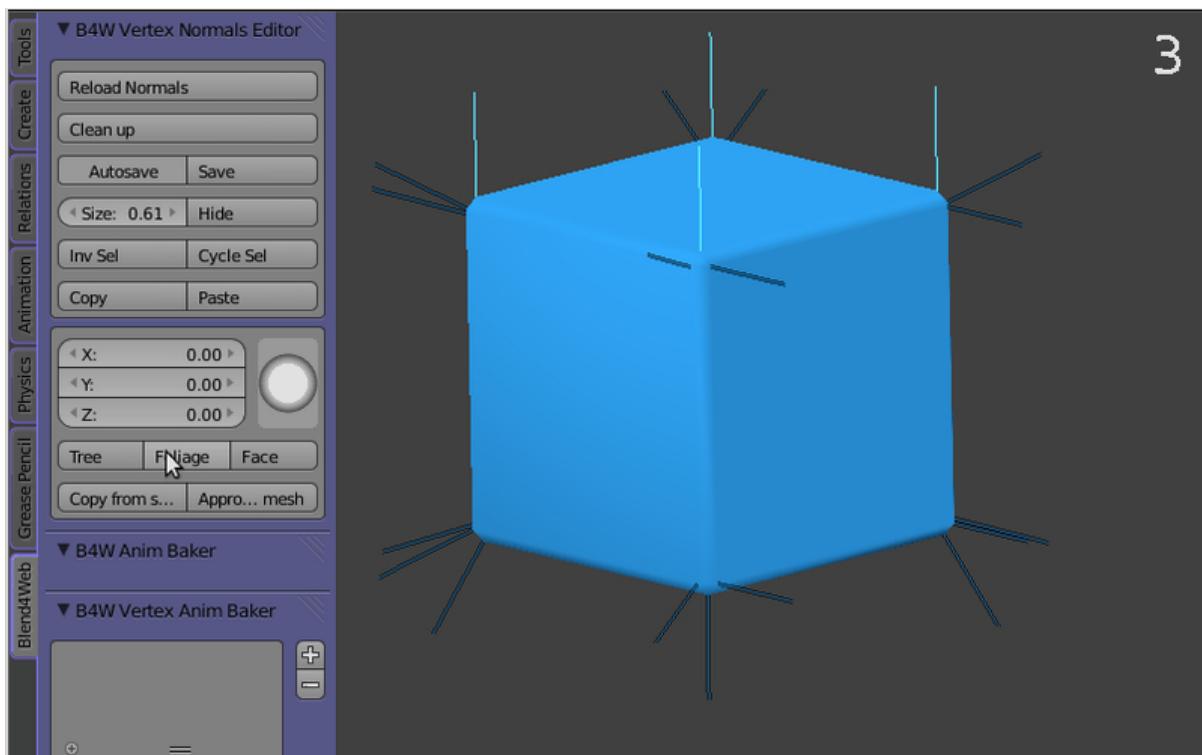
Пример работы редактора нормалей:



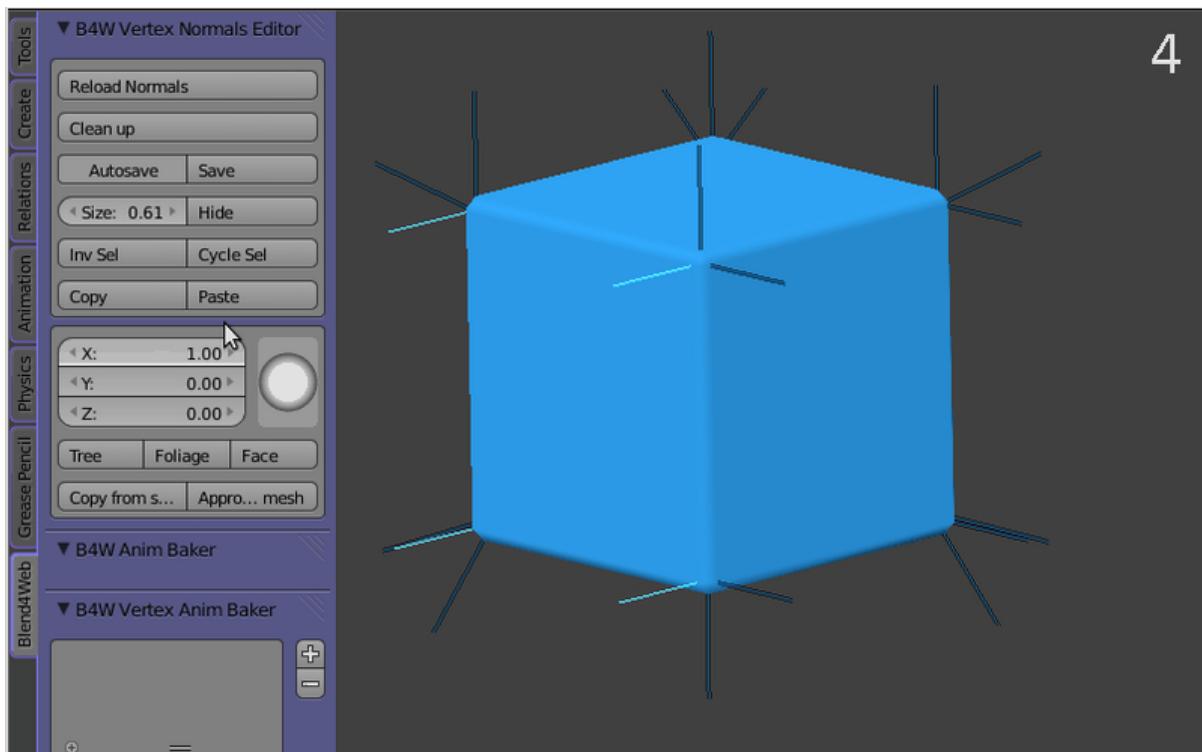
1



2



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.

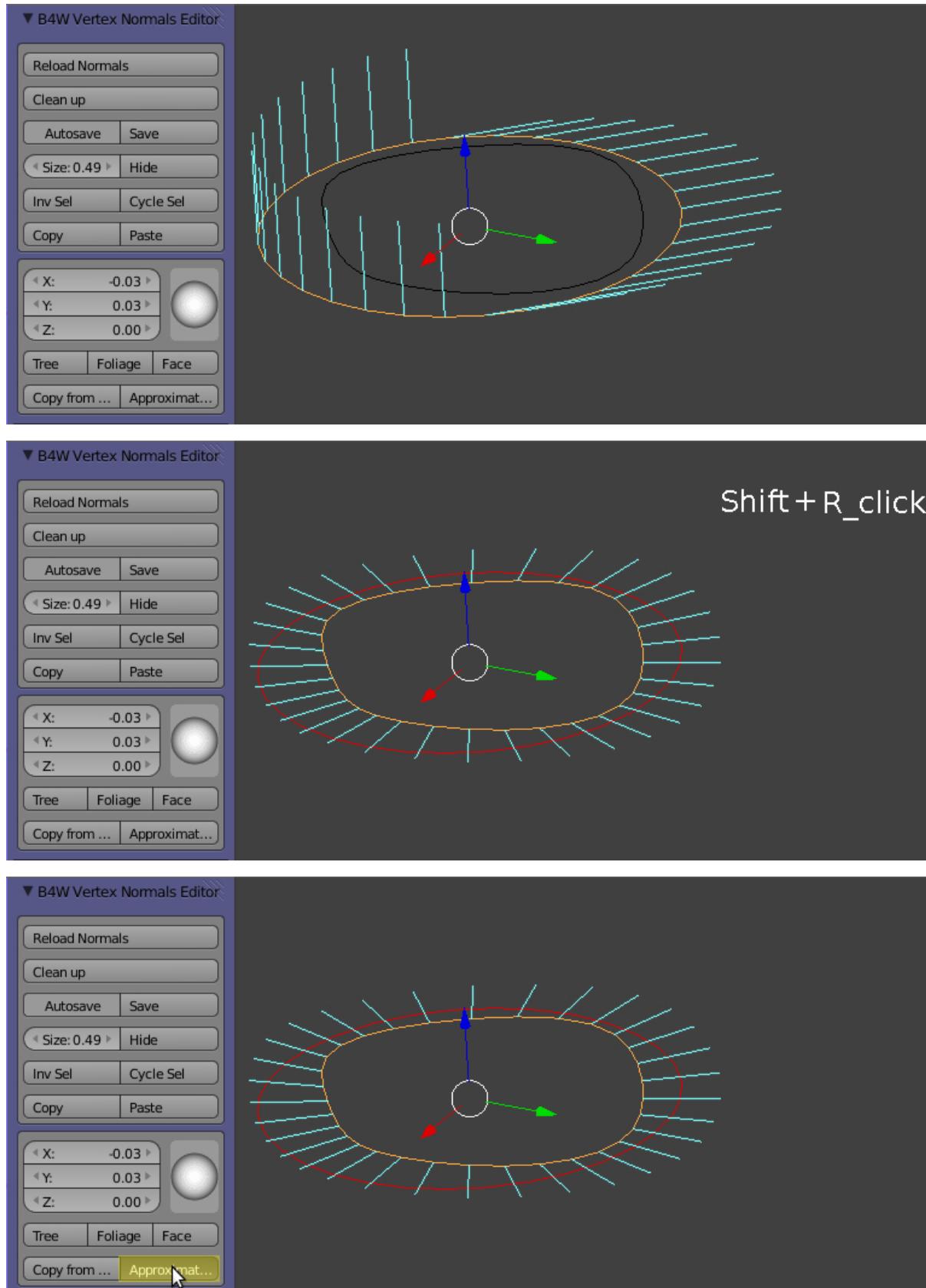
**Шкала XYZ и сфера направления** Инструменты для изменения направления векторов вручную.

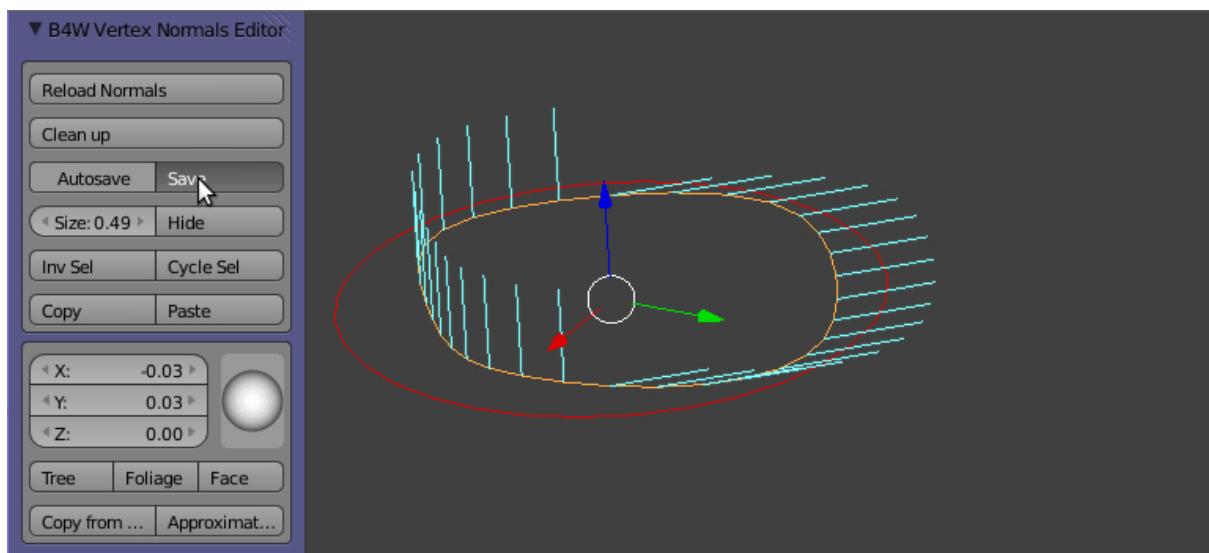
**Tree, Foliage** и **Face** Кнопка **Tree** задает всем выделенным нормалям направление от 3D-курсора, **Foliage** направляет нормали строго вверх (чаще всего используется для настройки частиц *травяного покрова* и *листвы деревьев*), **Face** присваивает векторам направление нормали выделенного полигона.

**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.

## 13.3 Освещение от источников света

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

### 13.3.1 Типы источников света

Поддерживаются источники света следующих типов:

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

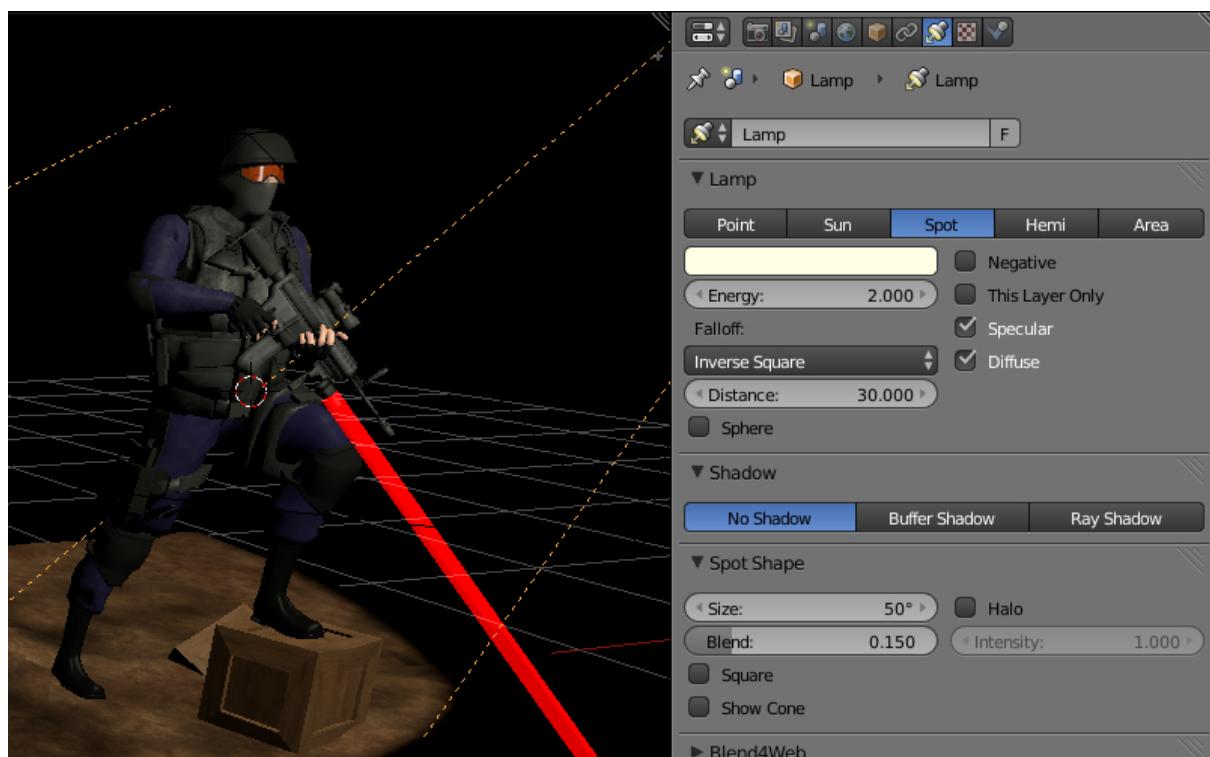
**Sun** “Солнце”. Свет распространяется из бесконечной плоскости прямолинейно в одном направлении, без затухания.

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

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

### 13.3.2 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 экспортировать. 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.

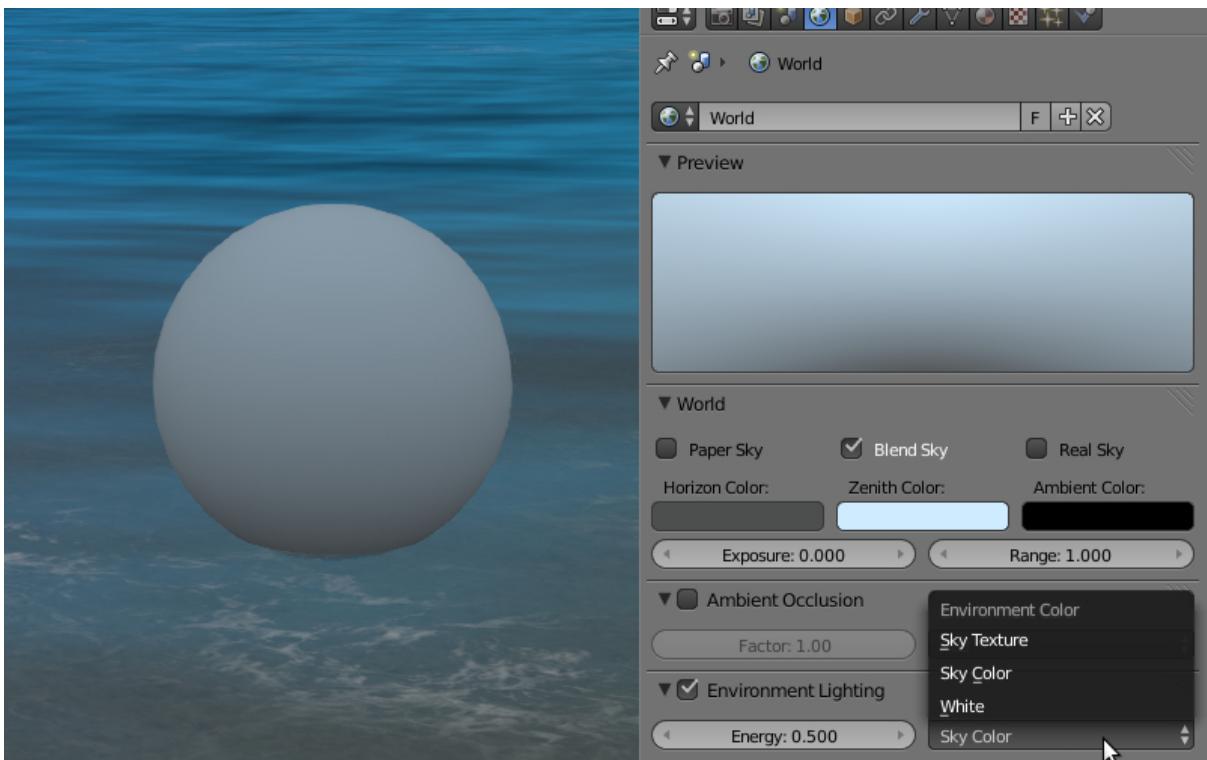
## 13.4 Освещение от окружающей среды

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.

### 13.4.1 Активация

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



### 13.4.2 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**.

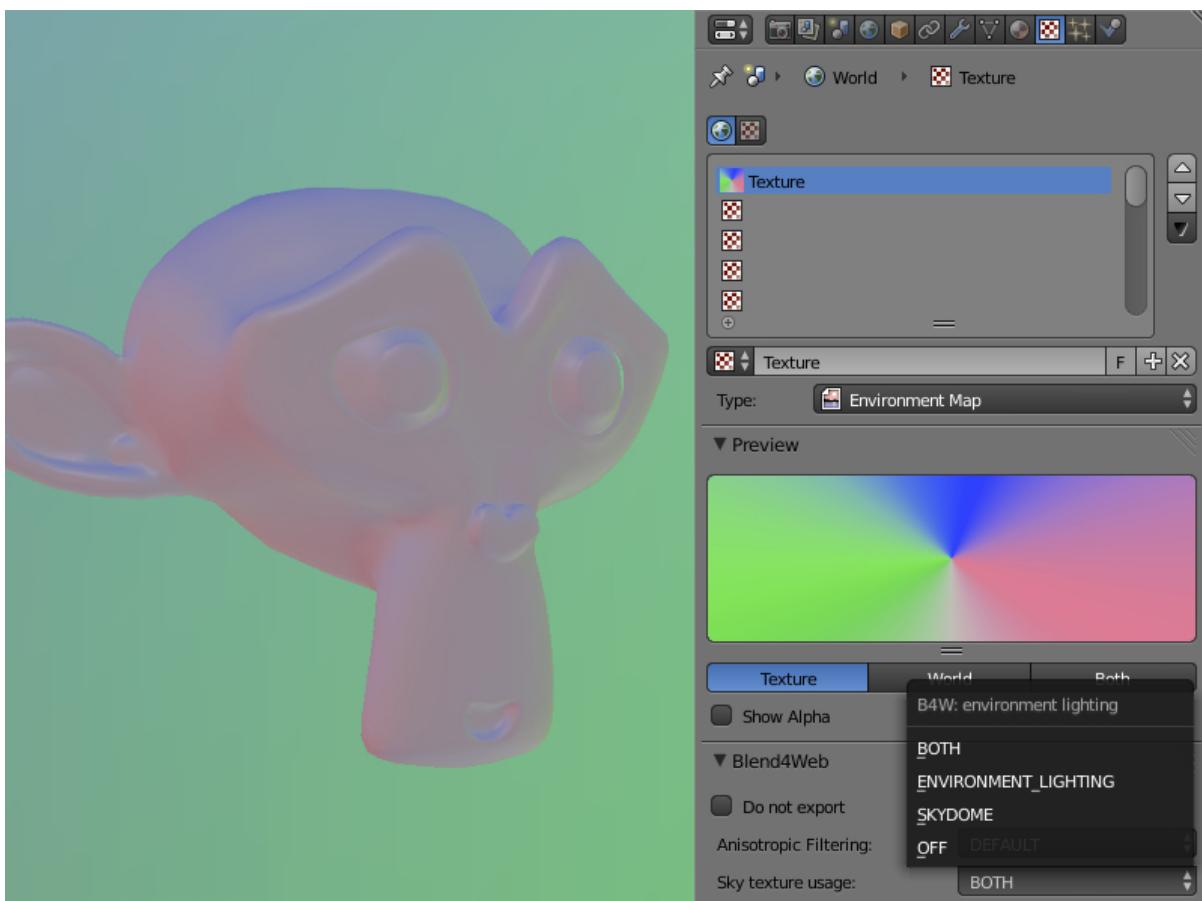
**World > Horizon Color и World > Zenith Color** Если выбрана полусферическая модель рассеянного освещения **Sky Color**, цвет горизонта и цвет зенита задаются цветоподборщиками **World > Horizon Color** (цвет горизонта) и **World > Zenith Color** (цвет зенита). При выборе цвета рекомендуется активировать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

### 13.4.3 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*).



## 13.5 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). Для расчета теней от источников света любого типа используется модель направленного (directional) света типа Sun.

### 13.5.1 Активация

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

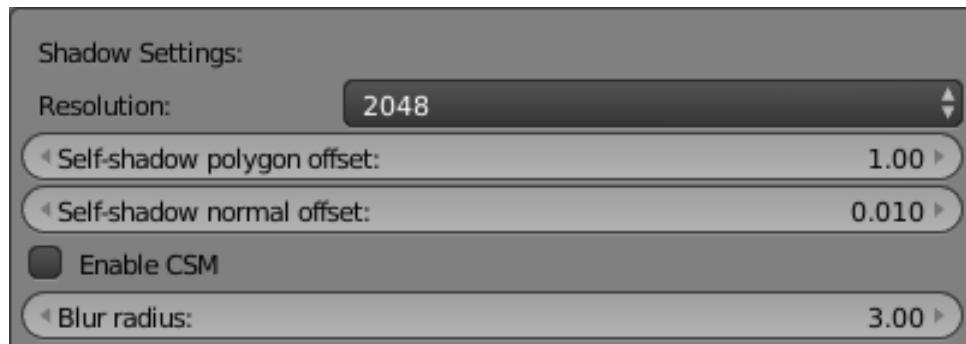
2. На объектах, **получающих** тени, включить опцию Blend4Web > Shadows: Receive во вкладке Object.
3. Убедиться, что включена опция Blend4Web > Render shadows во вкладке Scene.

### 13.5.2 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.

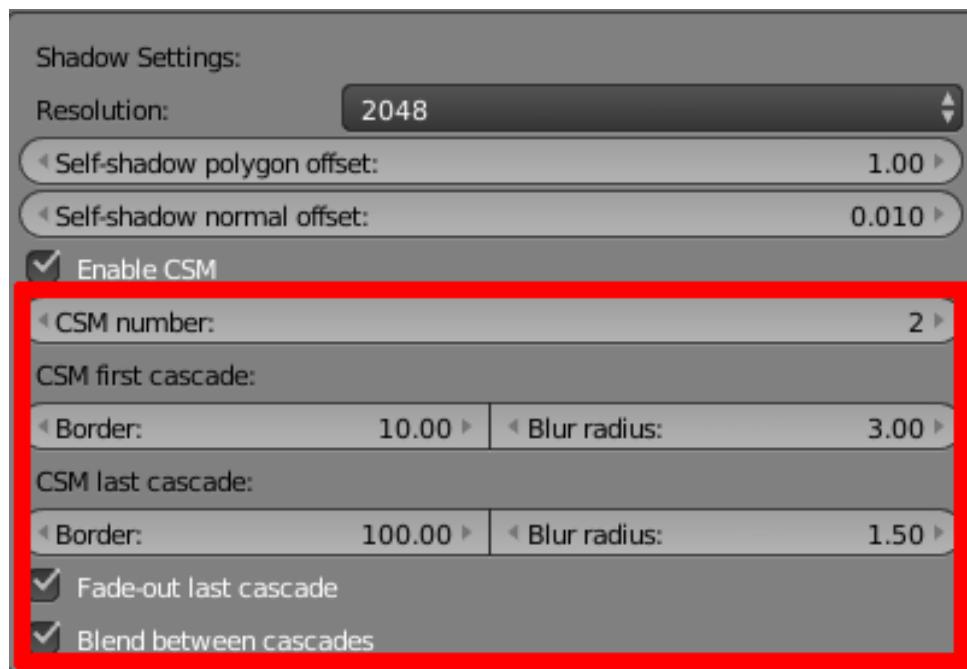
## Стандартная модель

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

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 последующие 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.

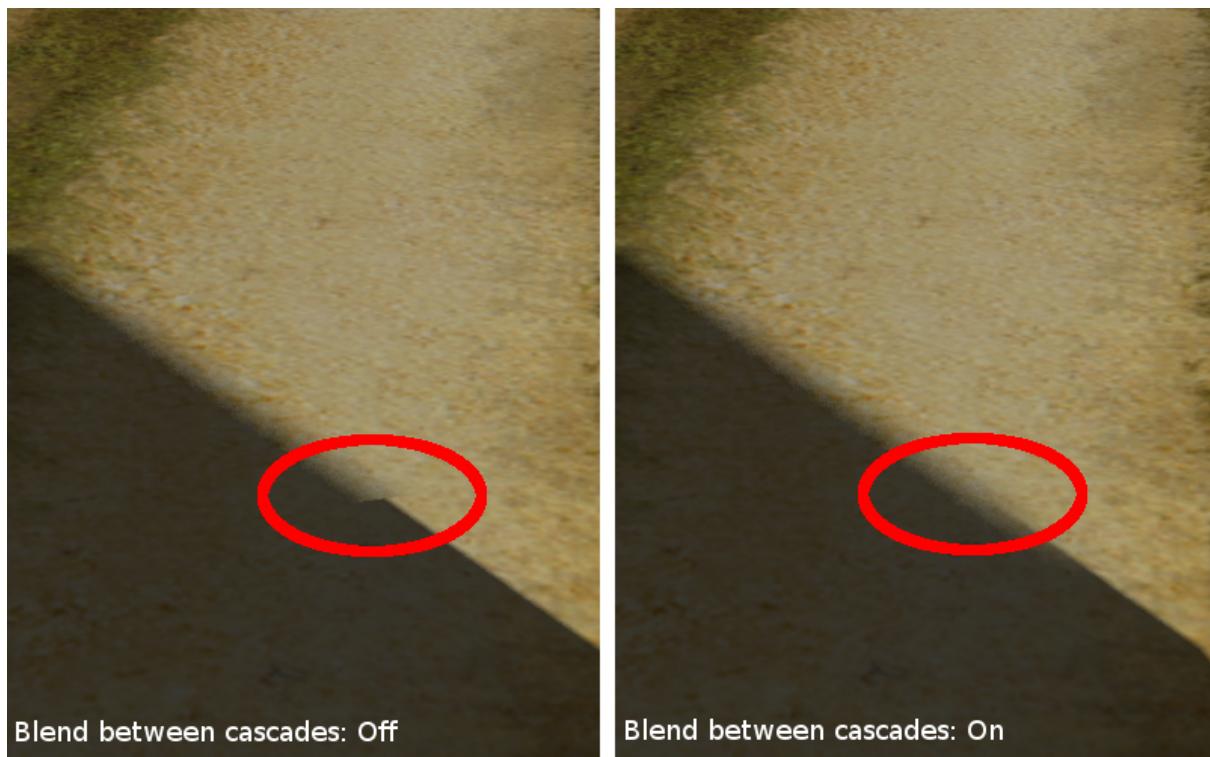
**CSM last cascade blur radius** Коэффициент размытия на последнем каскаде. Значение по умолчанию: 1.5. Коэффициент 0 даст жесткие тени.

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.



## 13.6 Цвет фона

Цвет фона можно задать несколькими способами:

1. Поместить сцену внутрь модели (например, куба или сферы) с направленными внутрь нормальями, с материалом и optionalной текстурой.
2. Использовать *текстуру неба*.
3. Настроить динамически генерируемую *атмосферу*.
4. Установить параметр движка `background_color`, используя программный метод `config.set()`. 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");

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

m_main.init(...);
```

5. 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`). 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

---

### 14.1 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.

#### 14.1.1 Активация

Выставить опцию Enable Motion Blur на панели Scene > Blend4Web.

#### 14.1.2 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.



## 14.2 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.

### 14.2.1 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.

### 14.2.2 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.



## 14.3 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.

### 14.3.1 Активация

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

### 14.3.2 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.

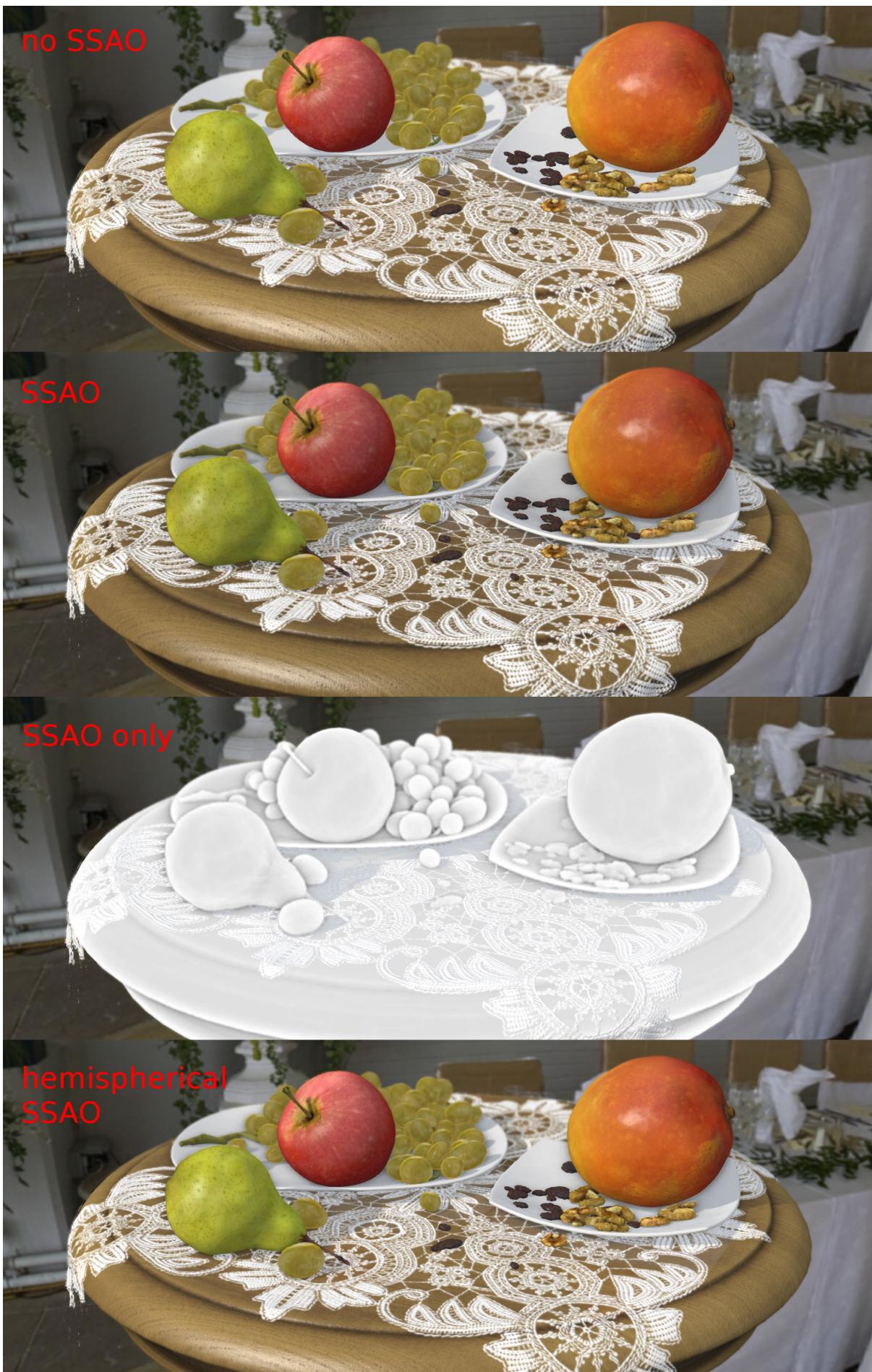
**Use Blur Depth Test** Если активировано - используется размытие SSAO на основе буфера глубины. Иначе - размытие по квадарту 4x4 по соседним пикселям.

**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 сэмпов (чем больше, тем лучше качество, но меньше производительность). Значение по умолчанию 16.



## 14.4 Сумеречные лучи

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

### 14.4.1 Активация

Выставить опцию Enable God Rays на панели Scene > Blend4Web.

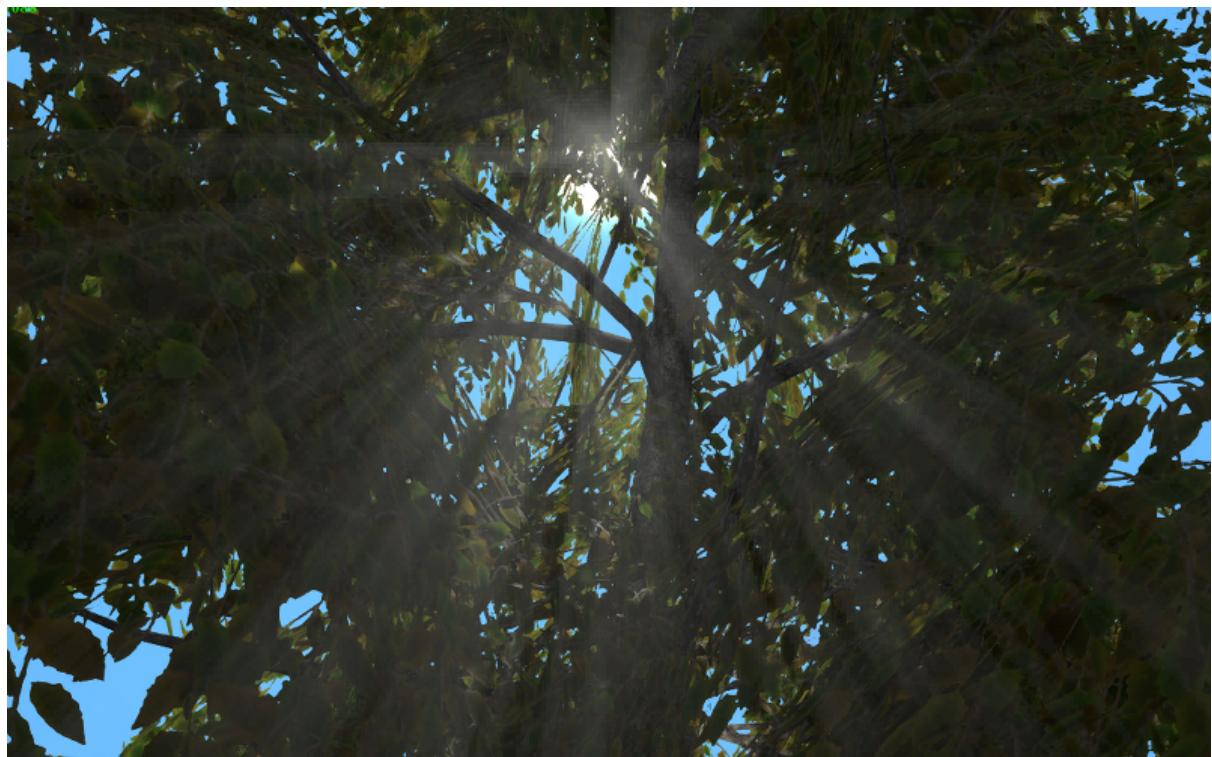
### 14.4.2 Дополнительные настройки

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.



## 14.5 Эффект засветки ярких деталей

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

### 14.5.1 Активация

Выставить опцию Enable Bloom на панели Scene > Blend4Web.

### 14.5.2 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.



## 14.6 Свечение вокруг объекта (Glow)

Эффект Glow заключается в подсвечивании конкретного объекта по контуру некоторым цветом. В результате вокруг объекта будет создан светящийся ореол.

### 14.6.1 Активация

Эффект Glow активируется программно через API. Может быть реализован как эффект постоянного свечения, так и затухающего, пульсирующего и любой другой модели. Чтобы разрешить его использование на конкретном объекте, необходимо выставить опцию **Selectable** на панели **Object > Blend4Web**.

### 14.6.2 Дополнительные настройки

На панели **Object > Blend4Web**:

**Glow duration** Длительность Glow-анимации, сек. Значение по умолчанию 1.

**Glow period** Период повторения Glow-анимации, сек. Значение по умолчанию 1.

**Glow relapses** Количество итераций Glow-анимации. В случае 0 анимация будет повторяться бесконечно. Значение по умолчанию 0.

На панели **World > Blend4Web**:

**Objects glow color** Общий цвет эффекта для всех объектов. Значение по умолчанию (1,1,1).

**Glow factor** Толщина и яркость ореола, окружающего объект. Падает с уменьшением параметра. Значение по умолчанию 1.

При управлении через API данные настройки воспринимаются как настройки по умолчанию.



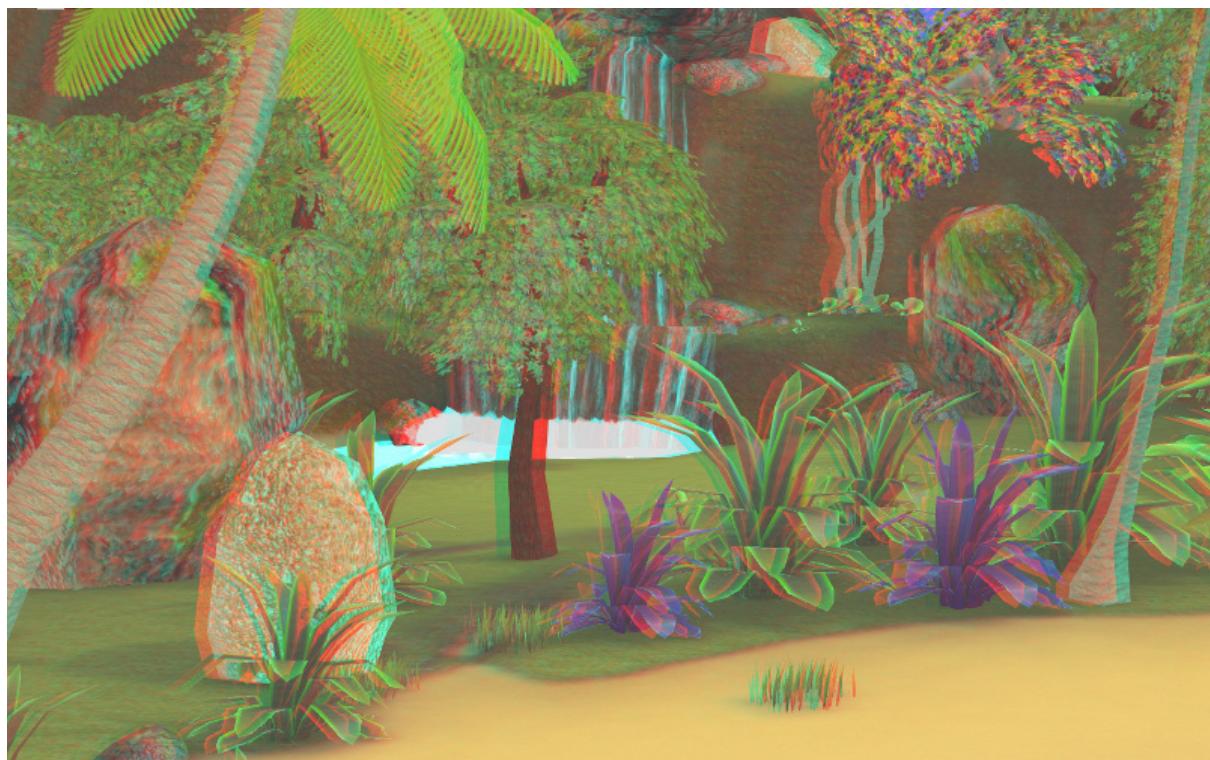
## 14.7 Аналиф стереоизображение

### 14.7.1 Активация

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

### 14.7.2 Дополнительные настройки

Нет.



## 14.8 Коррекция цвета

### 14.8.1 Активация

Выставить опцию Enable Color Correction на панели Scene > Blend4Web.

### 14.8.2 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.



## 14.9 Сглаживание

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

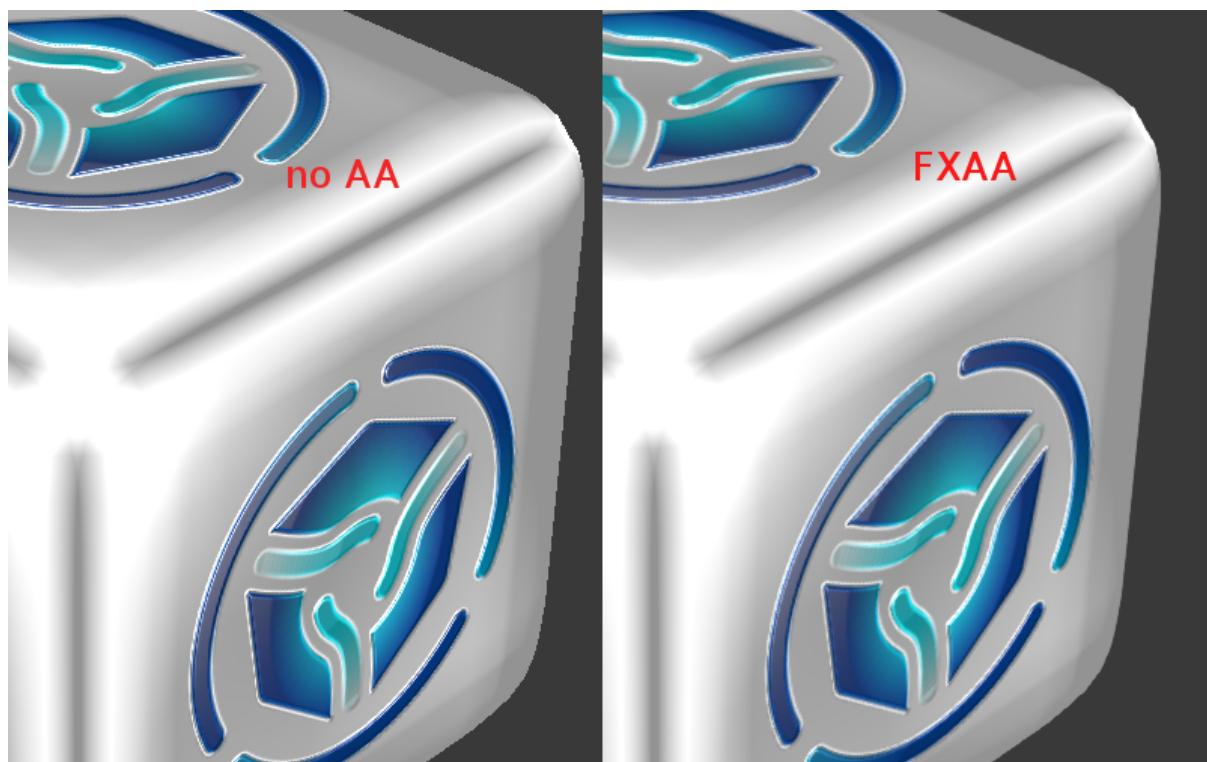
### 14.9.1 Активация

Выставить опцию `Enable Antialiasing` на панели `Scene > Blend4Web`.

### 14.9.2 Дополнительные настройки

Метод сглаживания назначается одновременно с выбором профиля работы движка.

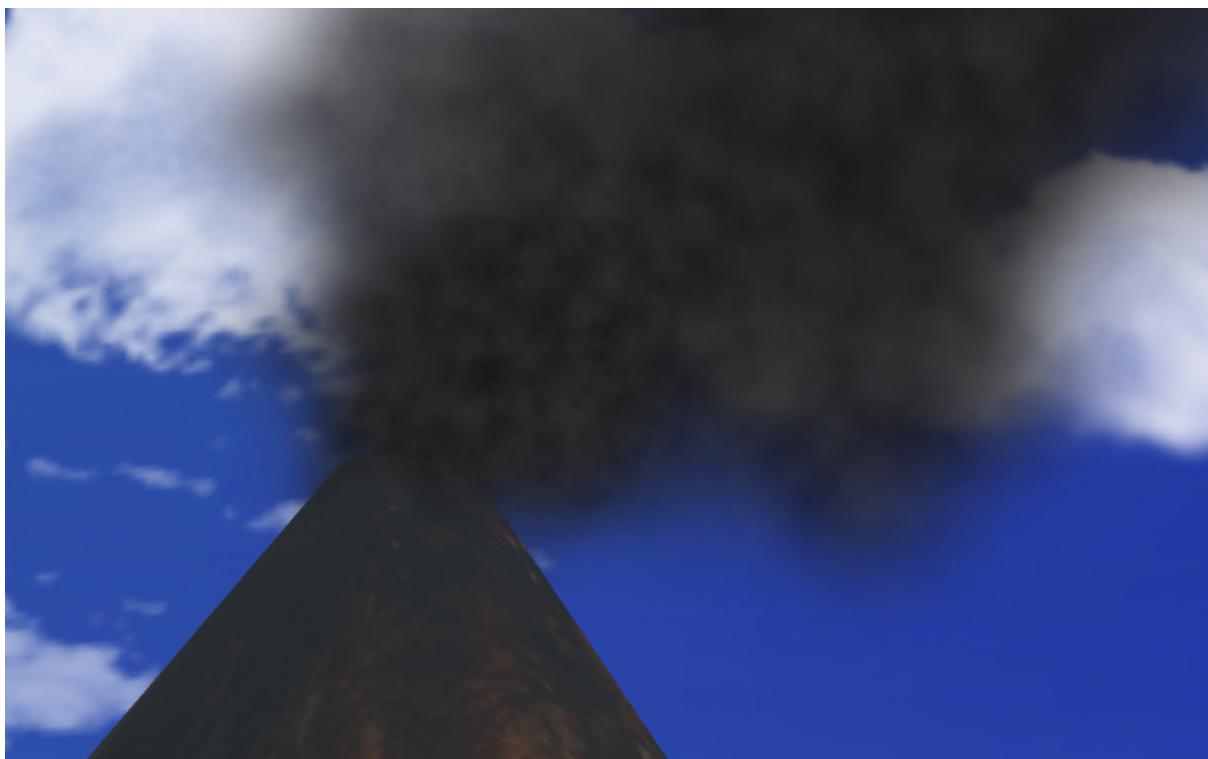
- *низкое качество* - антиалиасинг отключен
- *высокое качество* - метод антиалиасинга FXAA (Fast Approximate Anti-Aliasing), предложенный Nvidia
- *максимальное качество* - метод антиалиасинга SMAA (Enhanced Subpixel Morphological Anti-Aliasing), предложенный Crytek



### Система частиц

---

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.

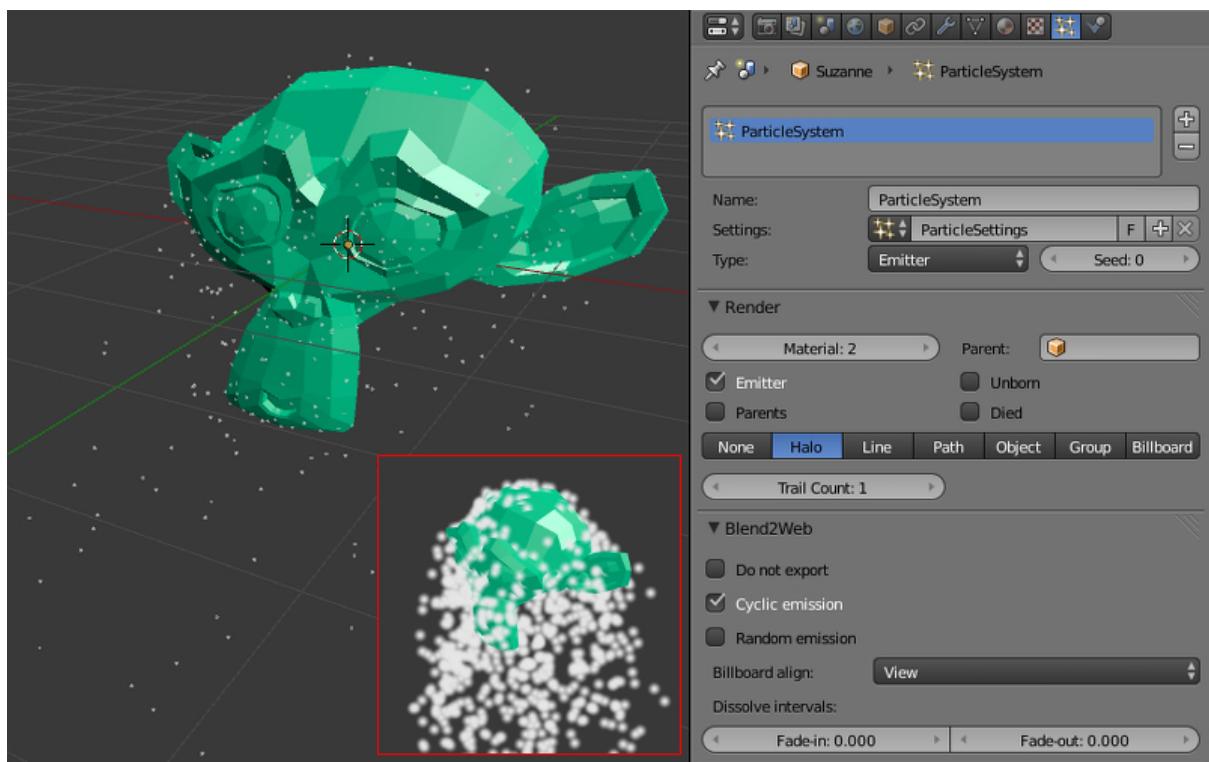
## 15.1 Использование

### 15.1.1 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**.
  - “некиклическая анимация” - для эмиттера выставить опцию **Blend4Web > Animation > Use default**.

### 15.1.2 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.



## 15.2 Настройка

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

### 15.2.1 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.

### 15.2.2 Настройки испускания

**Emission > Number** Количество частиц. Значение по умолчанию 1000.

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

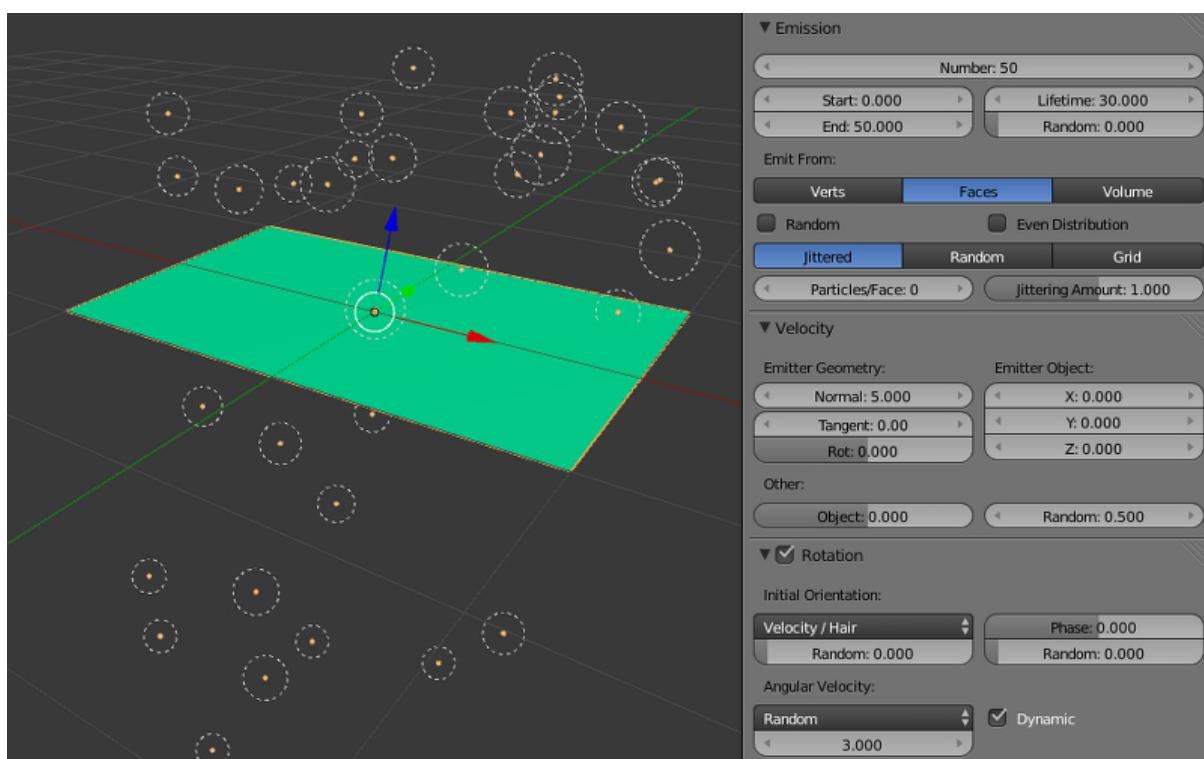
**Emission > End** Последний кадр, после которого прекращается испускание частиц. Значение по умолчанию 200.0.

**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.



### 15.2.3 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.

### 15.2.4 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.

## 15.2.5 Настройки физики

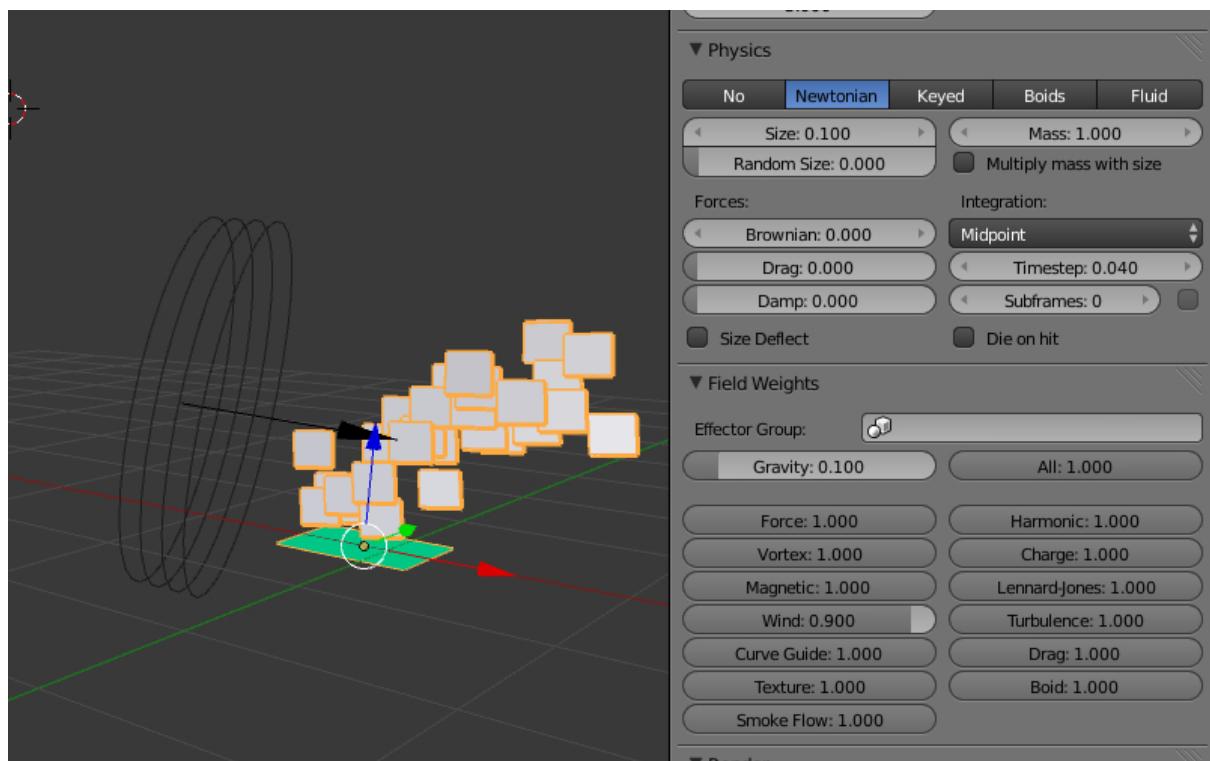
Поддерживаются только:

**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.



## 15.2.6 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**.

### 15.2.7 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.

### 15.2.8 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.

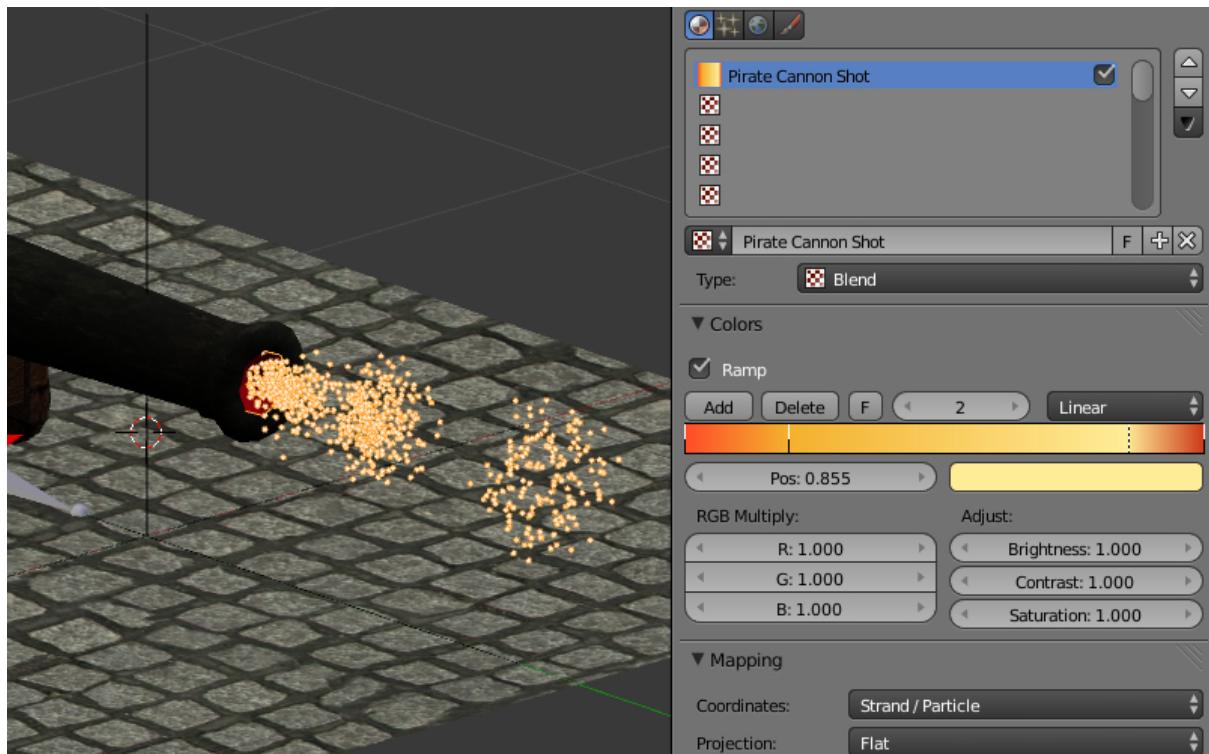
**Blend4Web > Coordinate system** Система координат испускаемых частиц: **Local** - использовать локальную систему координат эмиттера, **World** - использовать мировую систему координат.

## 15.3 Текстуры в системах частиц

### 15.3.1 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.

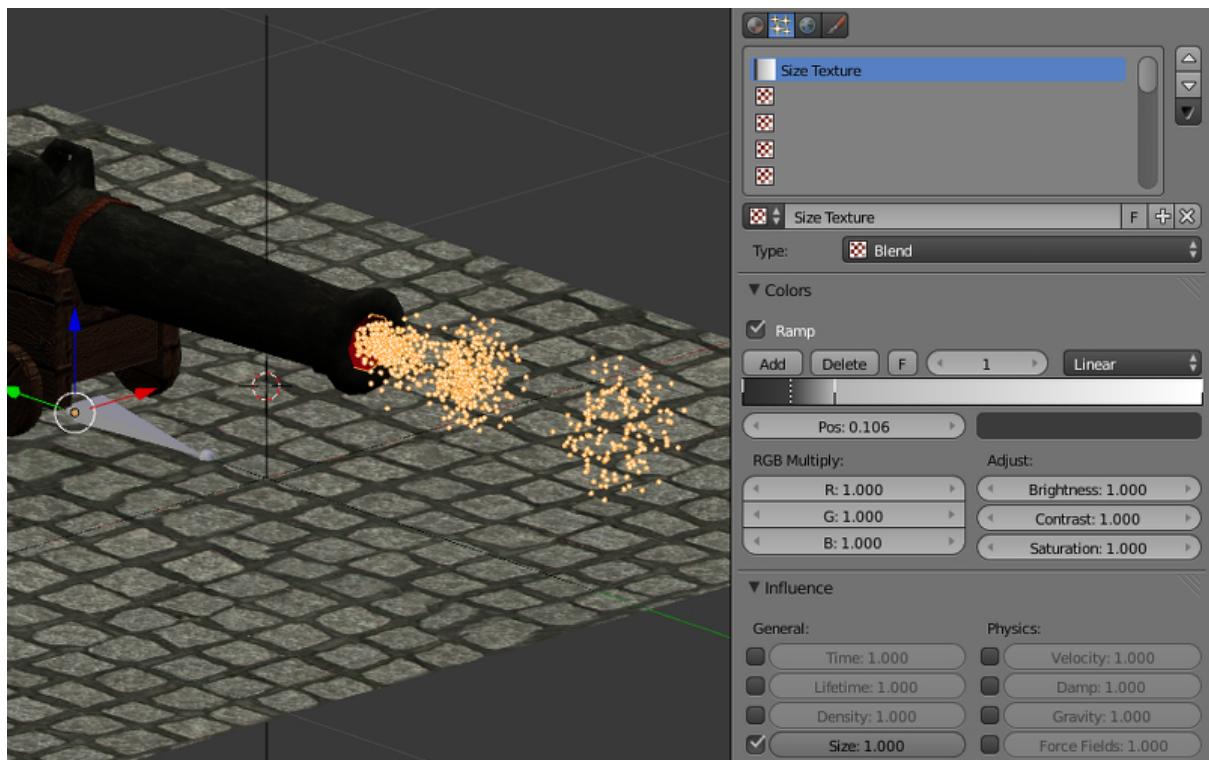


### 15.3.2 Текстуры системы частиц

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 для инстансинга объектов

---

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.

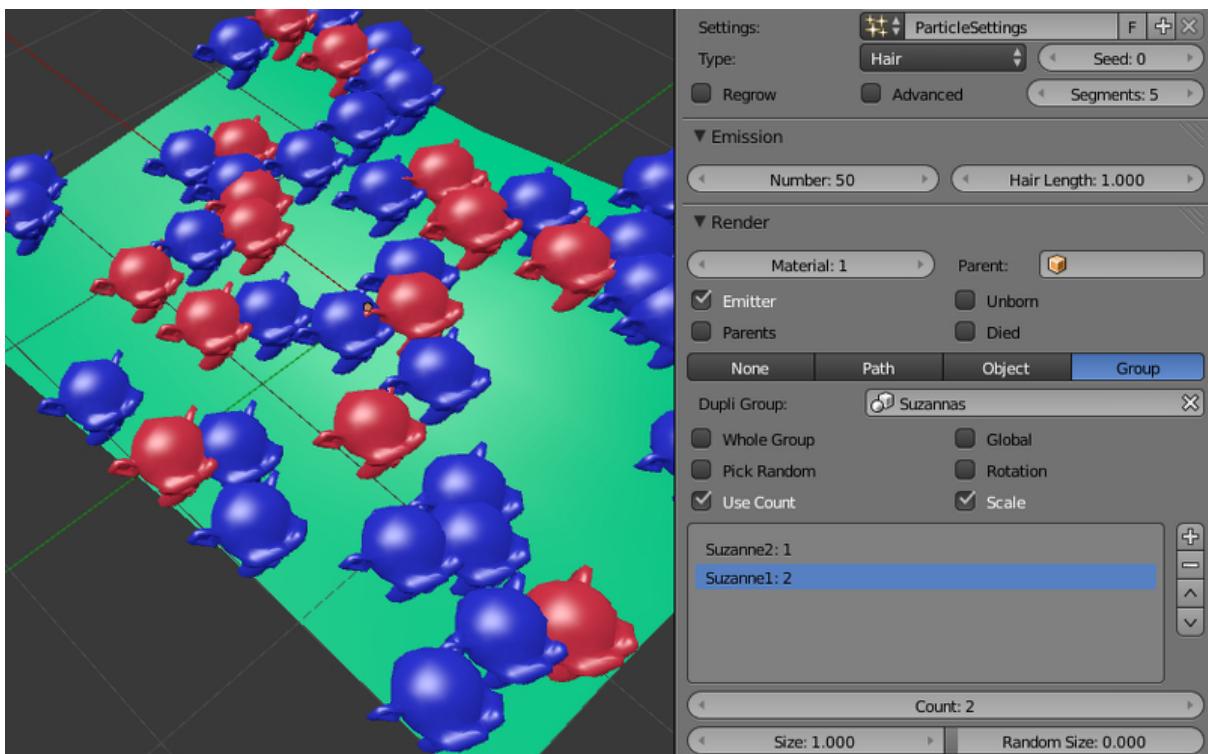
## 16.1 Настройки системы частиц

### 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)

- **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, отсчитываемый от нуля

- **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:

- **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; существует 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 type)**

- **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.

## 16.2 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

- На отдельном объекте-плоскости создать систему частиц для инстансинга объектов. Включить опцию Blend4Web > Dynamic Grass.
- 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).

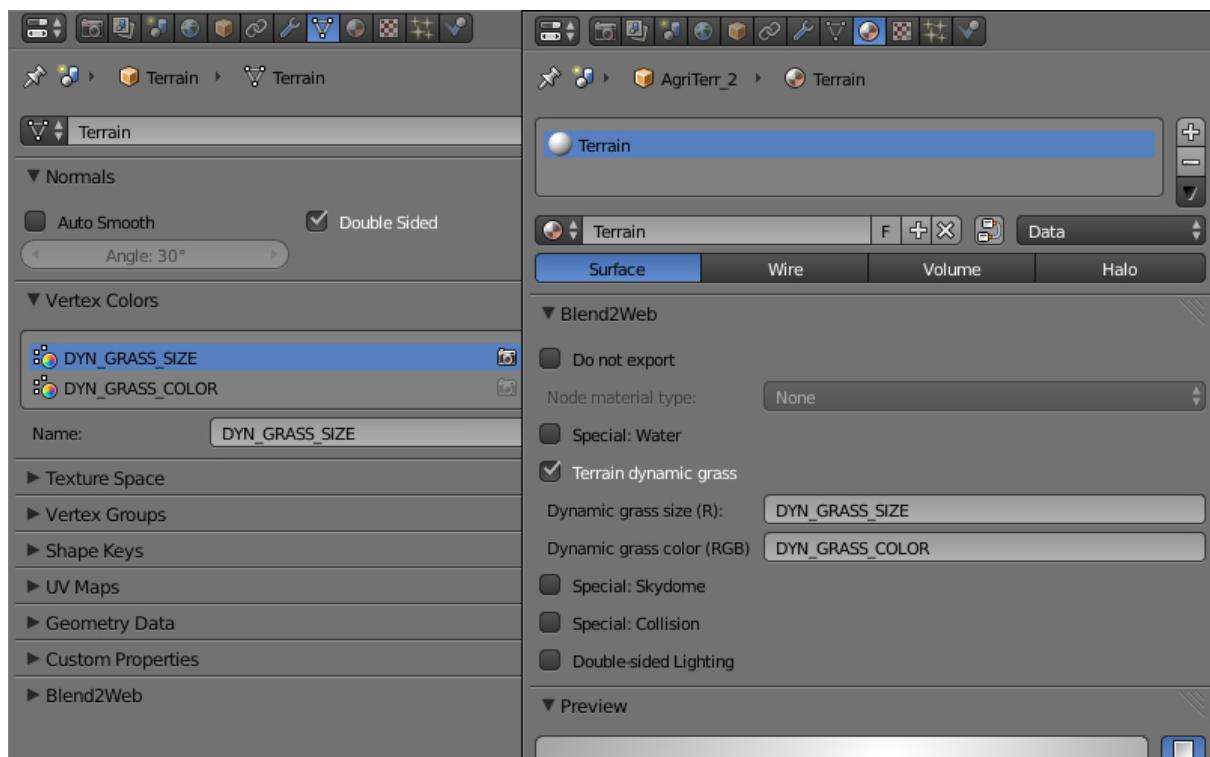
На **материале** ландшафта при включении опции Blend4Web > Terrain dynamic grass становятся активными текстовые поля:

**Dynamic grass size (R)** Название слоя вертекского цвета меша ландшафта, предназначенного 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).



## 16.3 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 случайного поворота

Если включена опция Blend4Web > Initial random rotation, то рекомендуется выставить вертикальную ось случайного 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 `Vertex Group > Density` выбрать вертексную группу эмиттера с вершинами для расположения частиц. Также нужно отключить опцию Blend4Web > `Random location and size`.

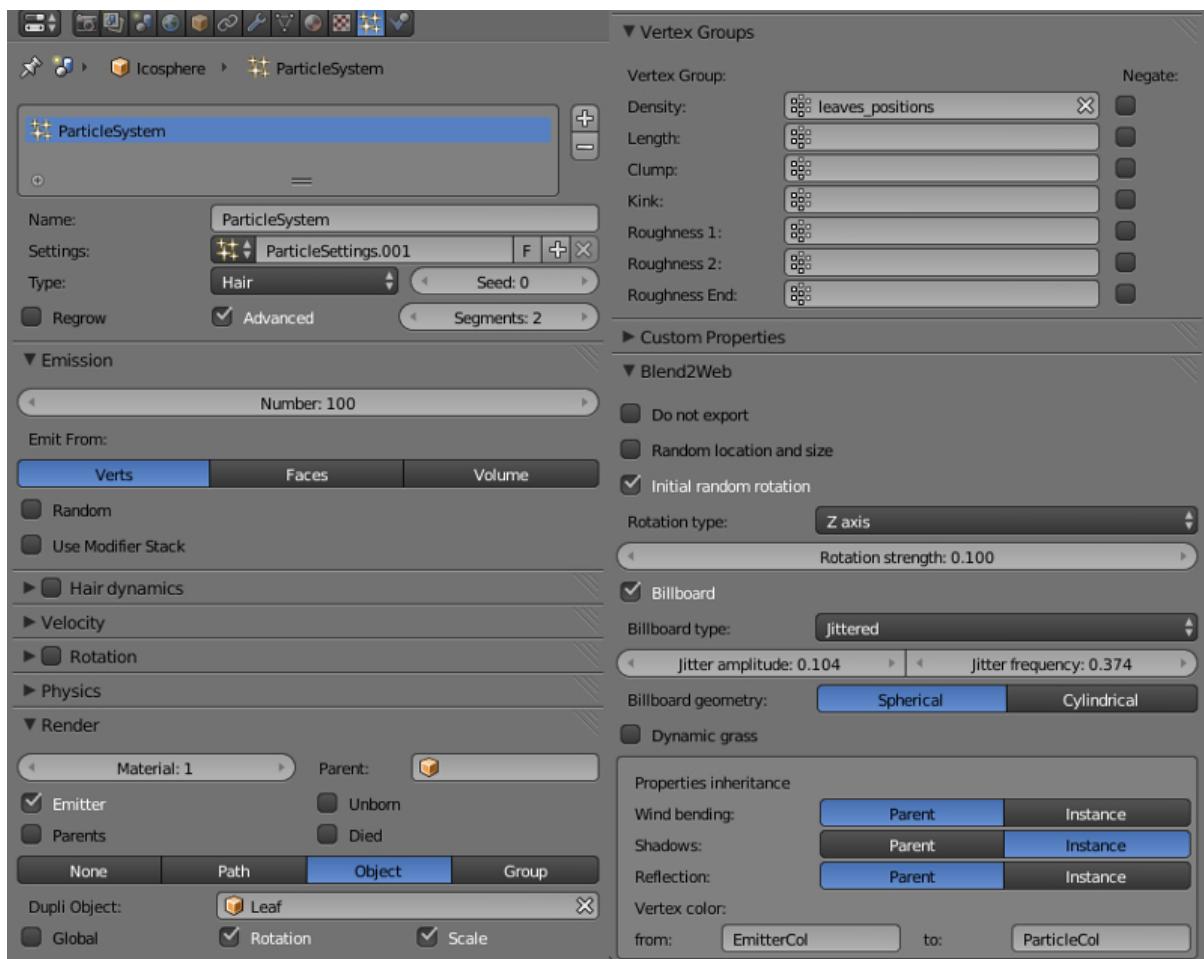
#### 4. Настройки Wind Bending

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 вертексный цвет для бендинга - Blend4Web > `Main Bending > Main stiffness`.

#### 5. Настройки наследования вертексного цвета

Для наследования частицами вертексного 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*.

## 17.1 Управление анимацией

Управление анимацией в движке осуществляется одним из двух способов:

1. Автоматически, с помощью указания свойств `Animation: Use default` и `Animation: Cyclic` 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*.

## 17.2 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 `Do not batch` на вкладке свойств объекта.

### 17.3 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.

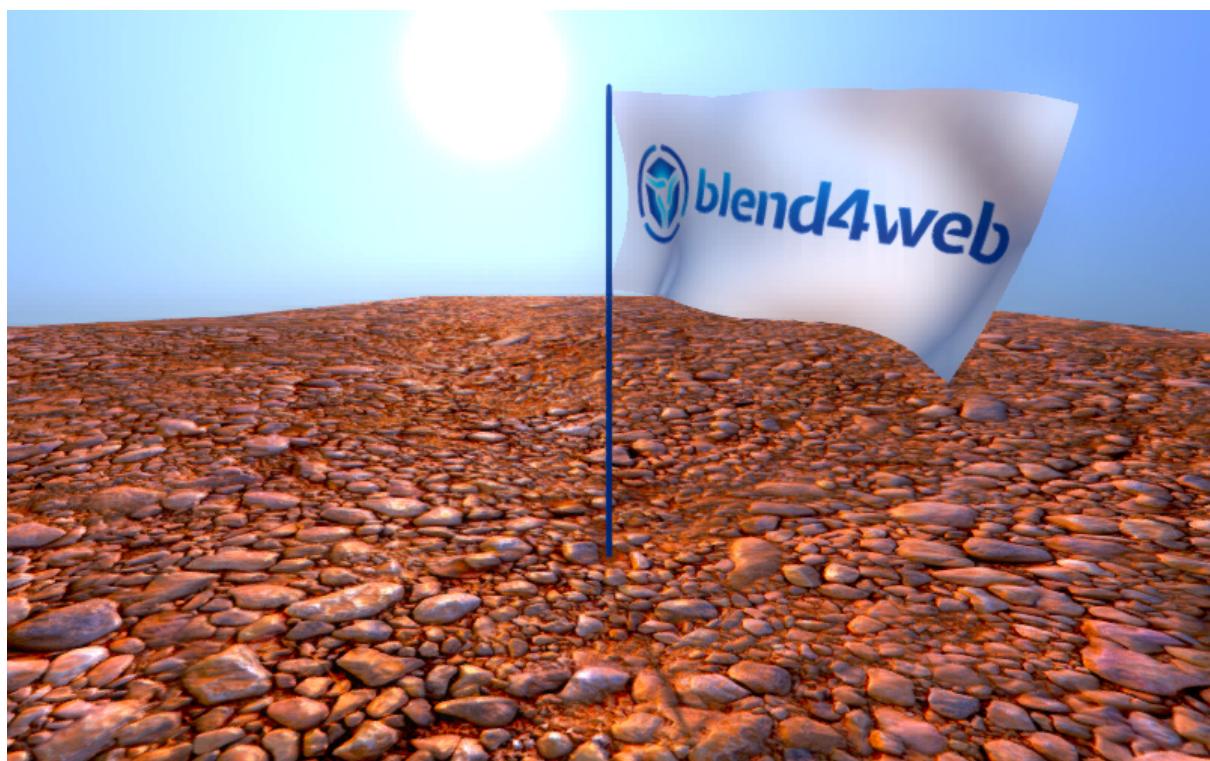
**Примечание:** Движок поддерживает не более 4-х вертексных групп на каждом из вертексов, эти группы отбираются по величине влияния или “веса” вертекса. В процессе загрузки исходного файла со сценой “веса” вертексов проходят через процедуру нормализации, т.е. их сумма приводится в единице.

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



---

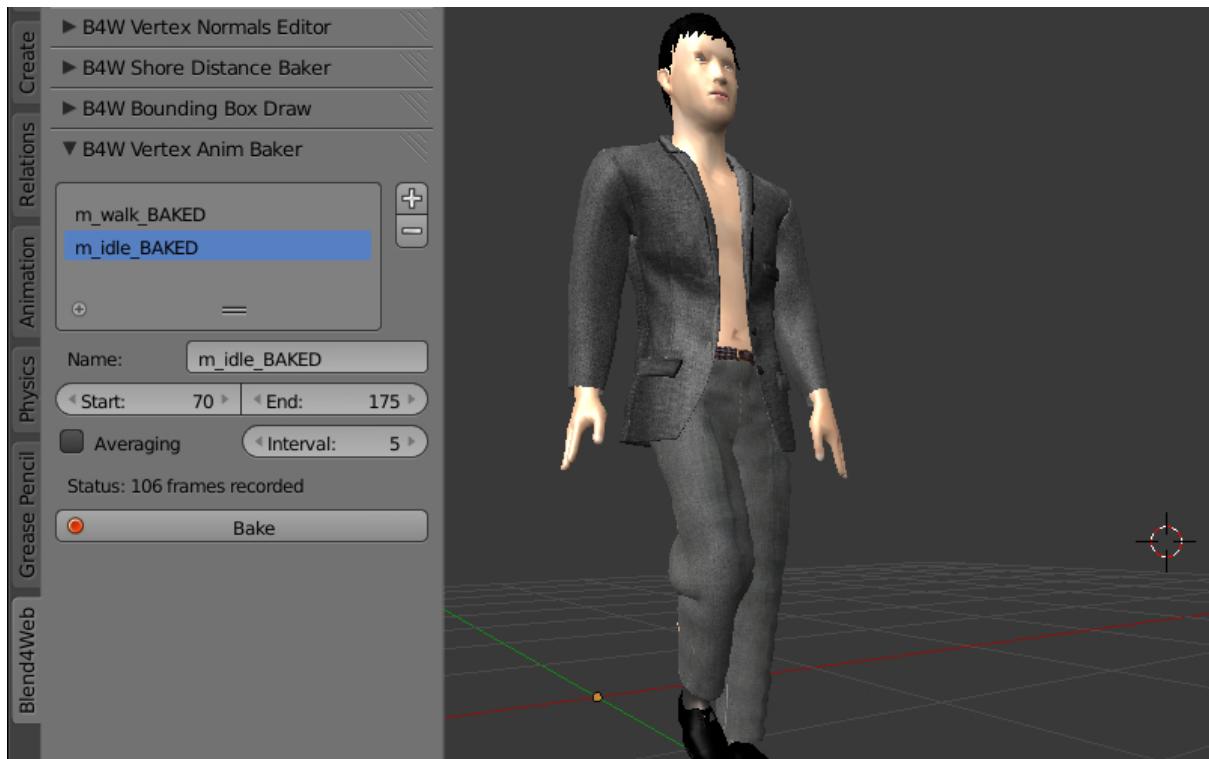
## 17.4 Вертексная анимация



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.



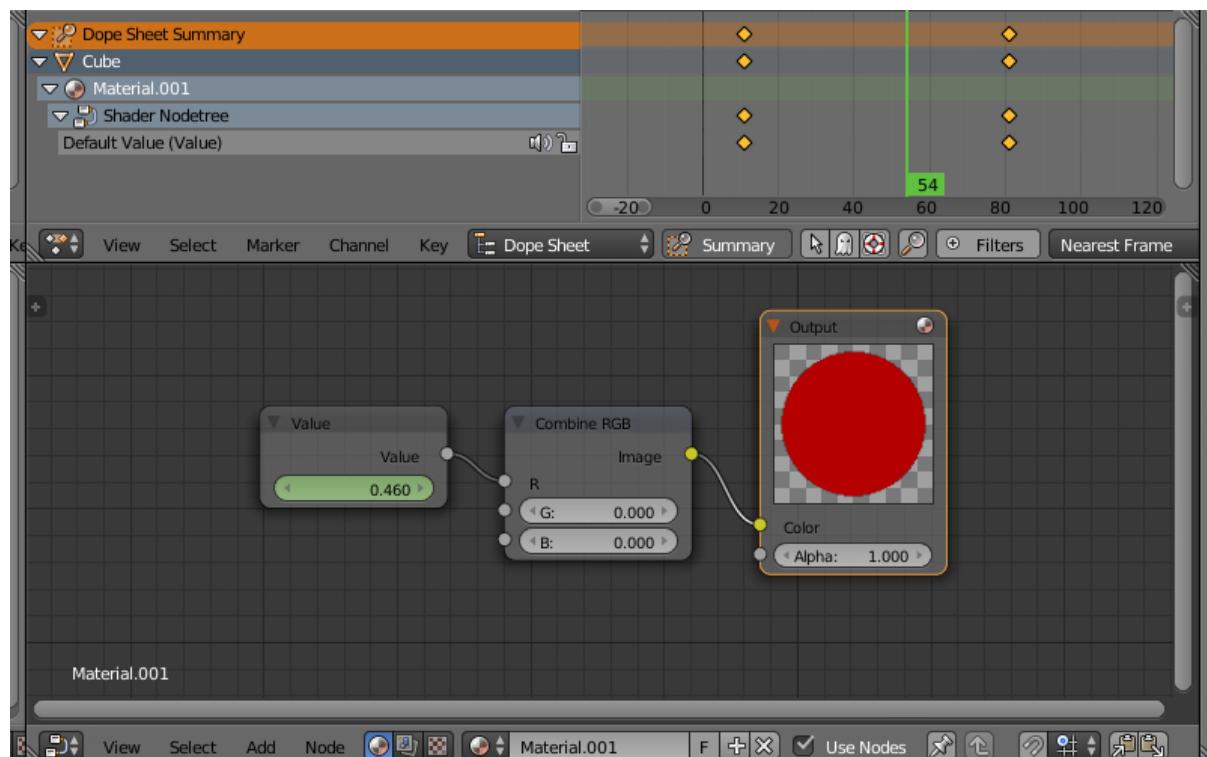
## 17.5 Параметризация источников звука

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.

## 17.6 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:**

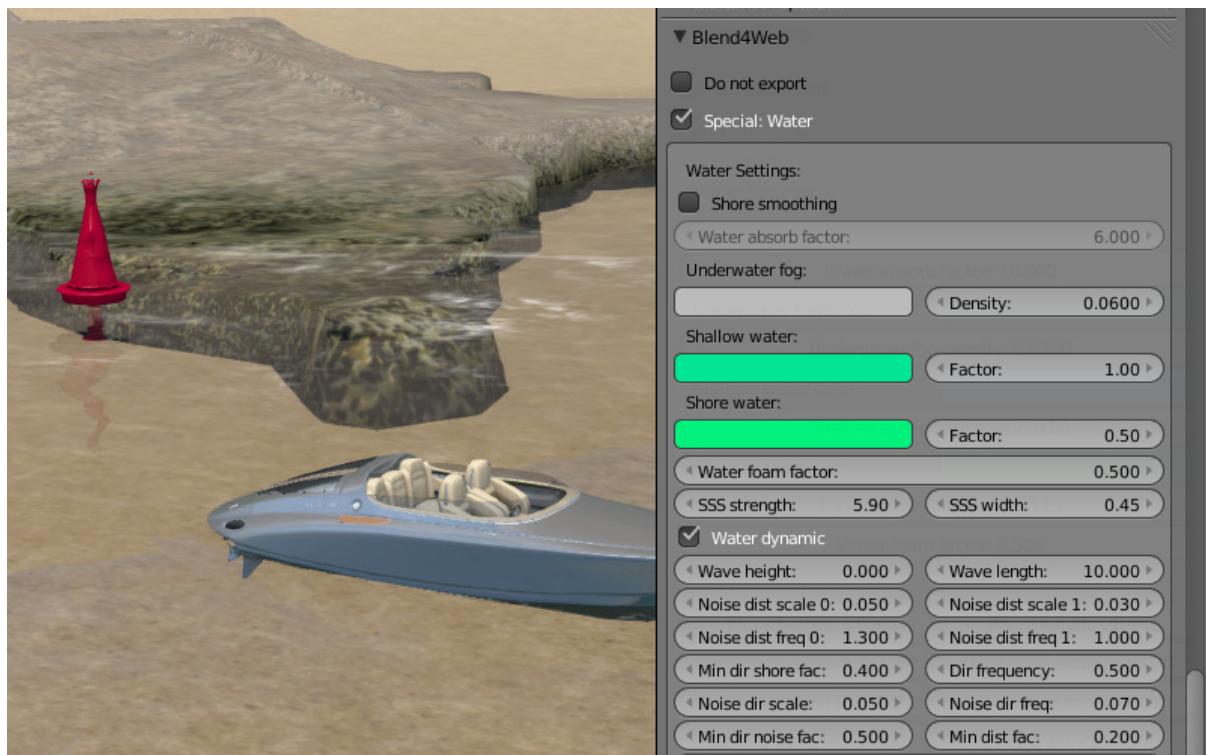
[Нодовые материалы](#)

## Рендеринг наружных сцен

### 18.1 Вода

#### 18.1.1 Активация

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



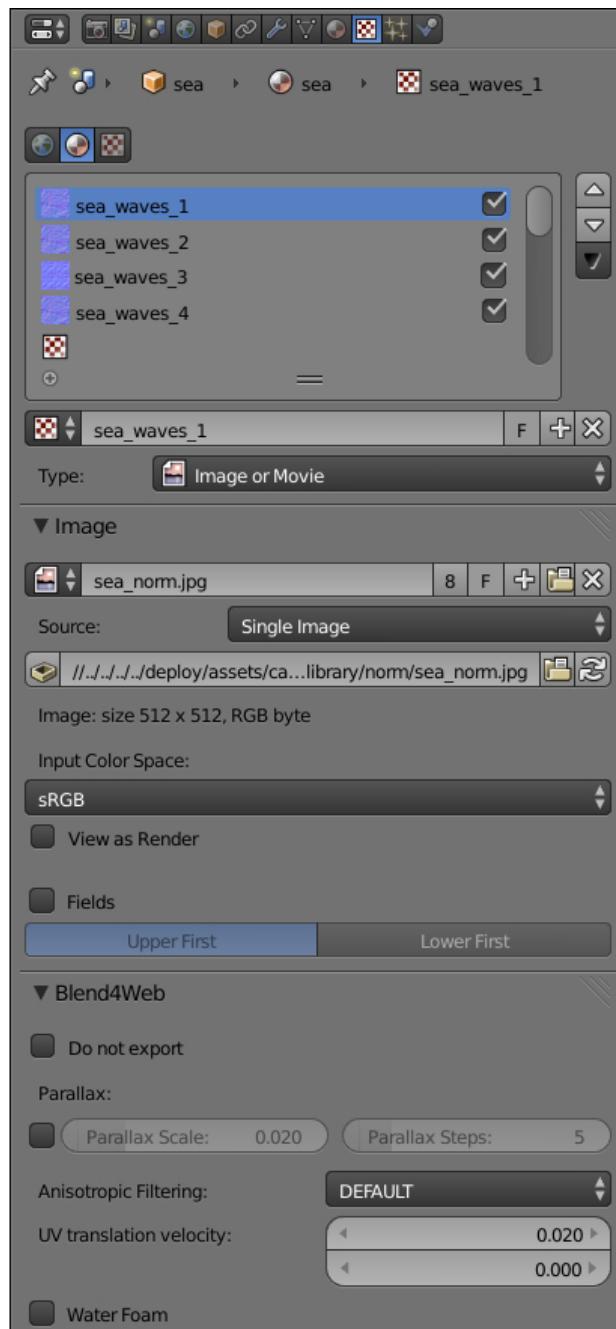
#### 18.1.2 Базовые настройки

**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](#).

### 18.1.3 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 и Blend4Web > UV translation velocity. Меш для воды должен иметь текстурную развертку.



### 18.1.4 Смачивание поверхностей

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

### 18.1.5 Отражение и эффект Френеля

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



### 18.1.6 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.

### 18.1.7 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 применения цвета воды на береговой линии.

### 18.1.8 Преломление

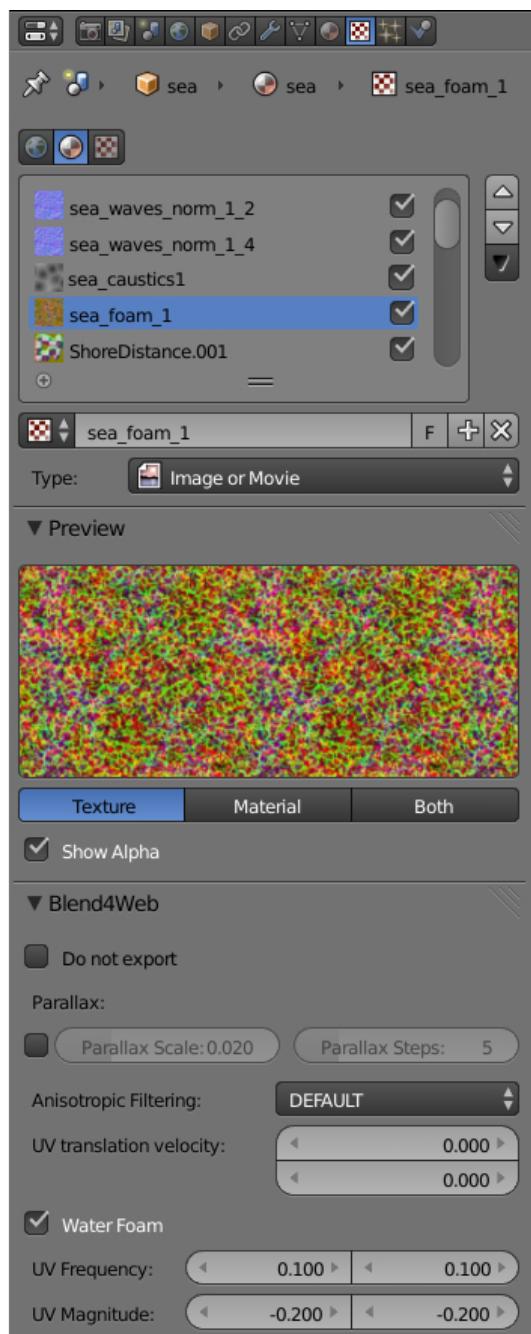
Во вкладке **Scene** включить опцию **Blend4Web > Render refractions**.



### 18.1.9 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

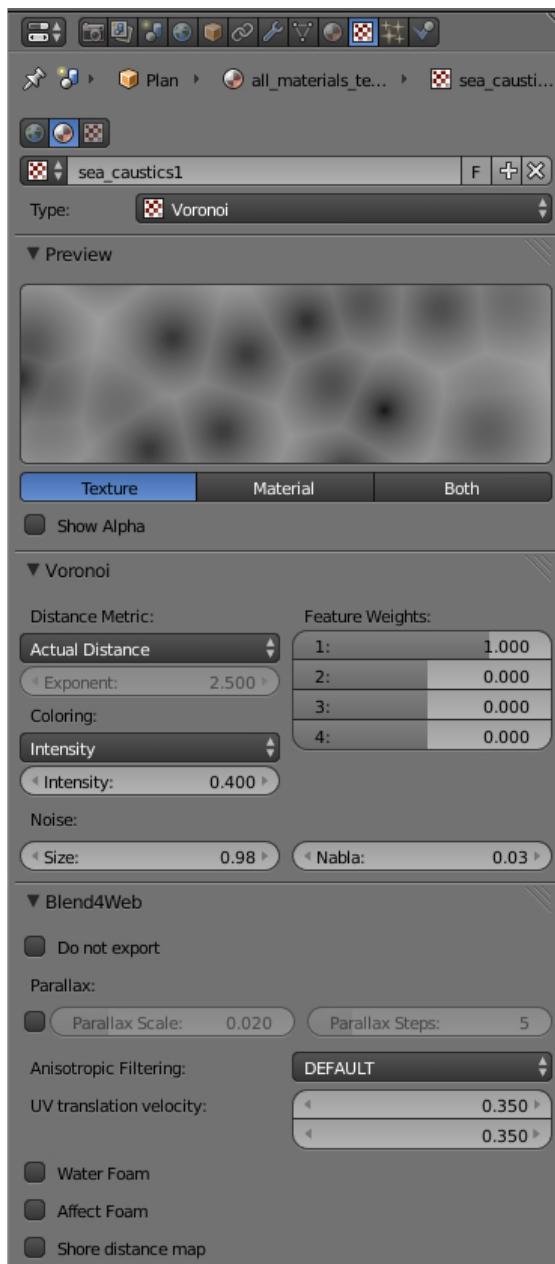
*Blend4Web > Water Settings > Water foam factor* Фактор общего влияния пены. Значение по умолчанию 0.5.

### 18.1.10 Каустика и хроматическая аберрация

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



## Настройка

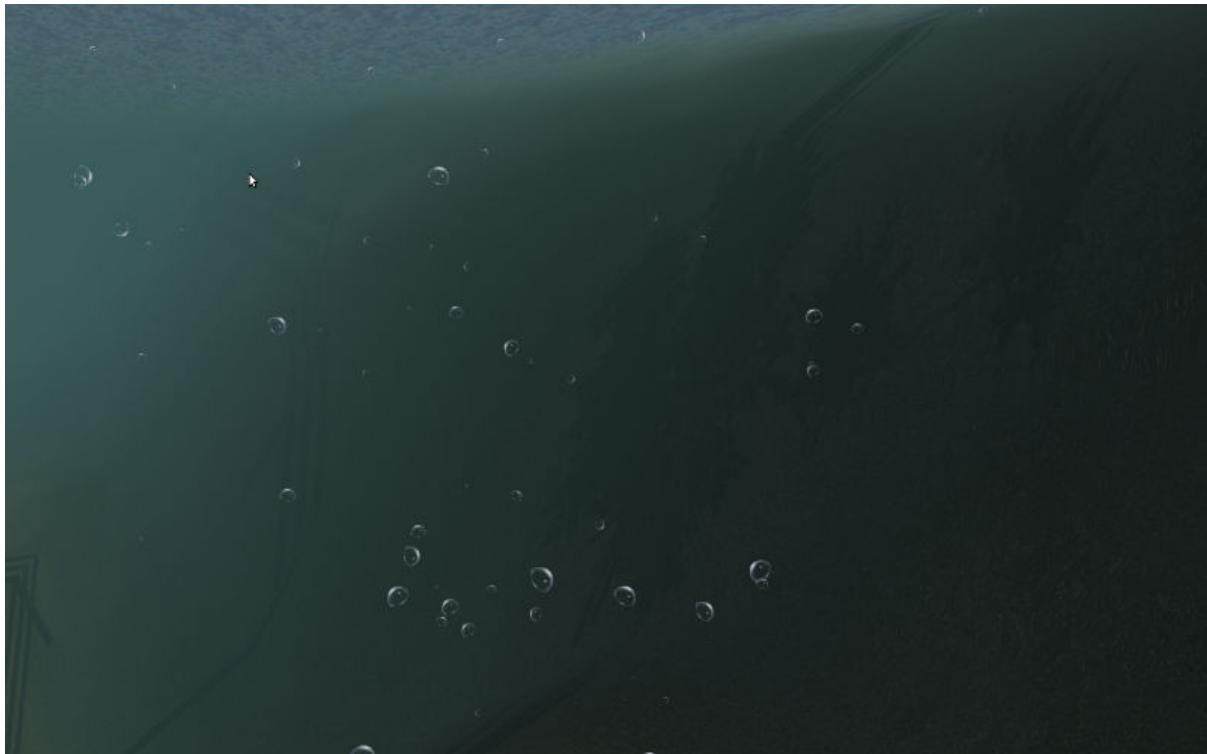


**Voronoi > Coloring: Intensity** Фактор влияния каустики. Значение по умолчанию 1.0.

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

**Blend4Web > UV translation velocity** Скорость анимации текстурных координат. Значение по умолчанию (0.0, 0.0).

### 18.1.11 Подводная среда



#### Настройки видимости (“туман”)

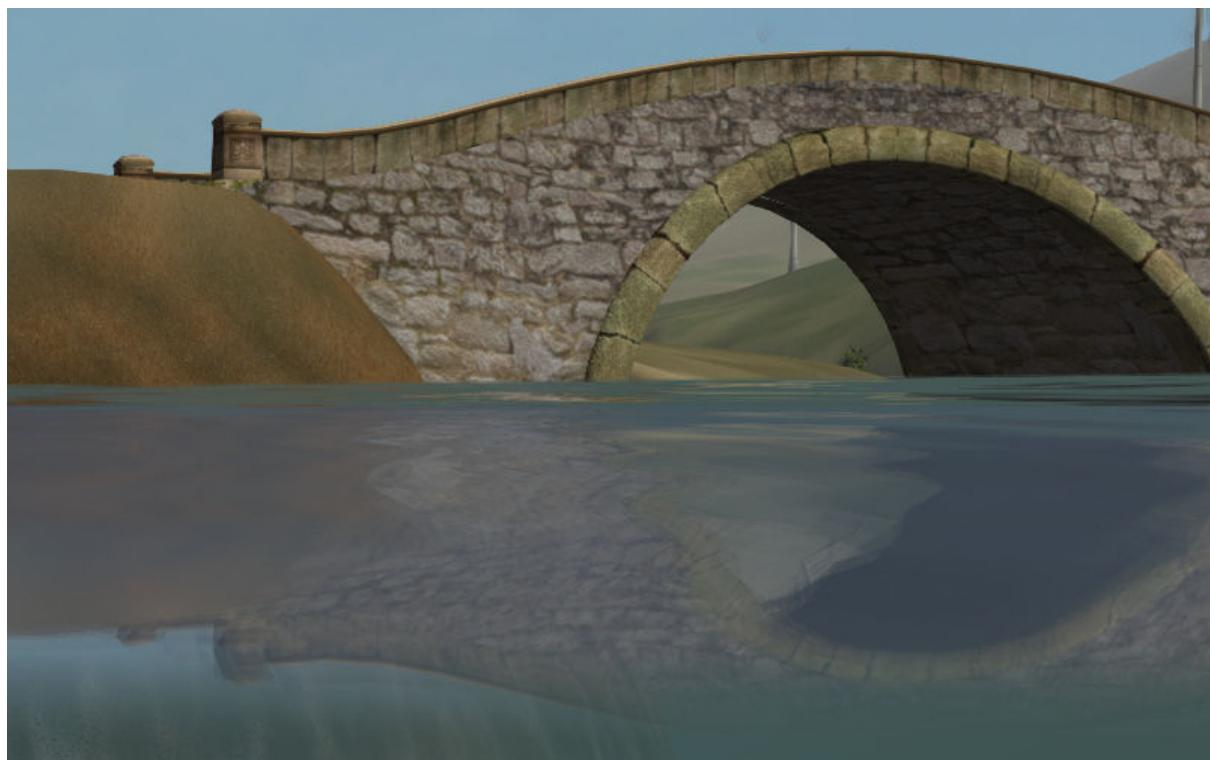
*Blend4Web > Water Settings > Underwater fog density* Экспоненциальный фактор, влияющий на плотность и расстояние. Значение по умолчанию 0.06.

*Blend4Web > Water Settings > Underwater fog color* Цвет тумана. Значение по умолчанию (0.5, 0.5, 0.5) (серый).

Применяются также настройки *сумеречных лучей*.

### 18.1.12 Граница сред

Выключить опцию Game Settings > Backface Culling.

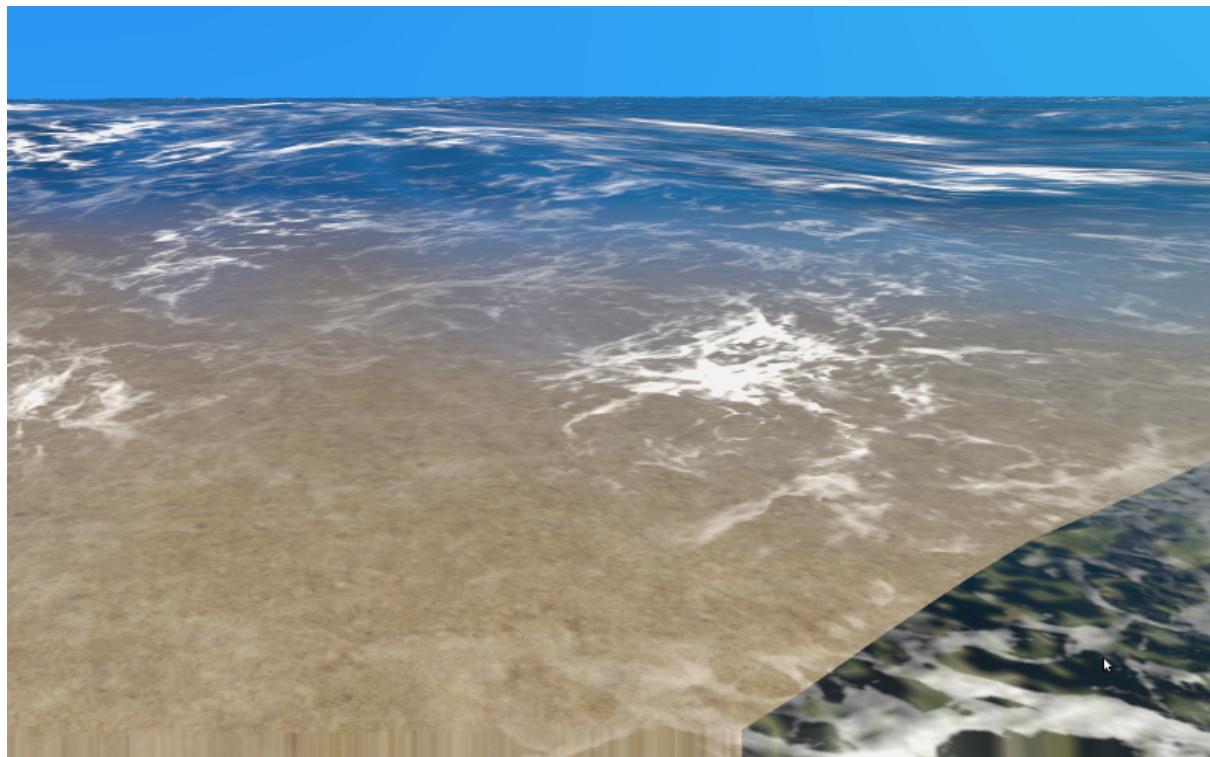


### 18.1.13 Объемные волны

#### 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

## 18.1.14 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.

## 18.2 Атмосфера

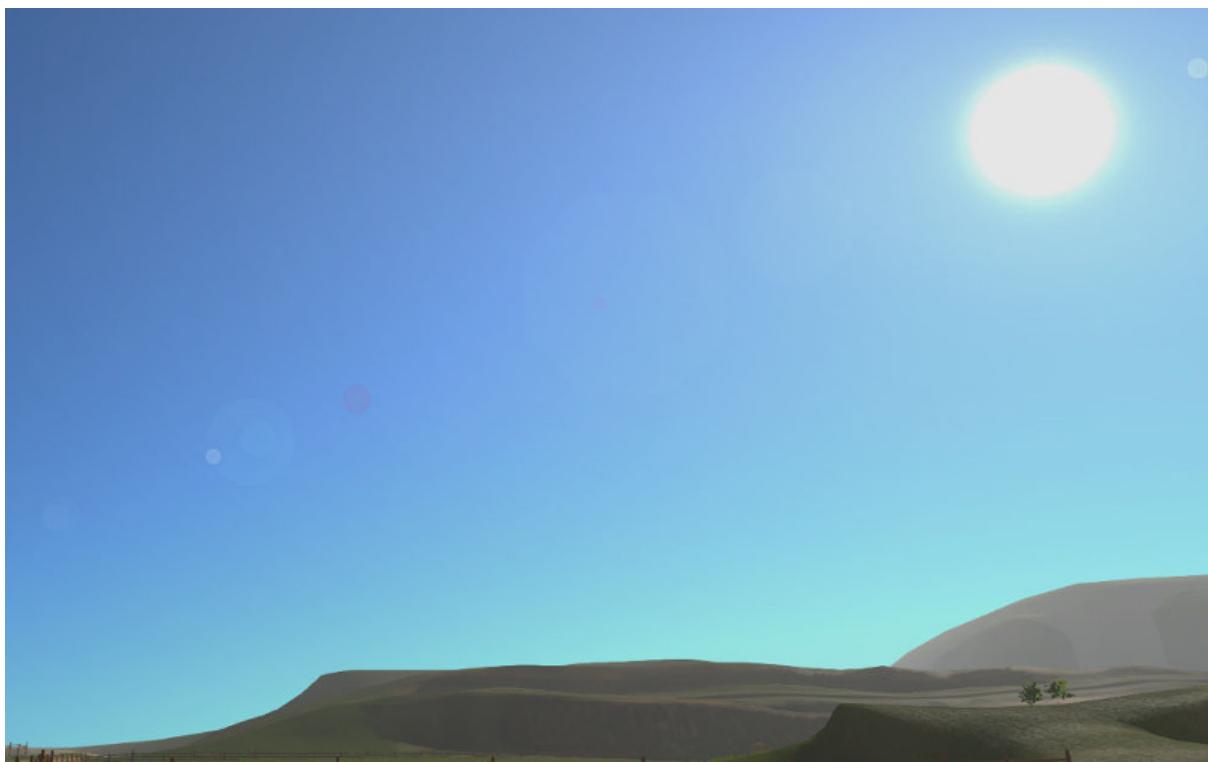
### 18.2.1 Рассеивание

Во вкладке **World** выставить опцию **Sky Settings > Procedural skydome**. 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 коэффициент рэлеевского рассеяния. Значение по умолчанию 0.35.

**Sky Settings > Mie collection power** Степенной коэффициент рассеяния Ми. Значение по умолчанию 0.5.

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

## 18.2.2 Туман

Настраивается во вкладке **World**.

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

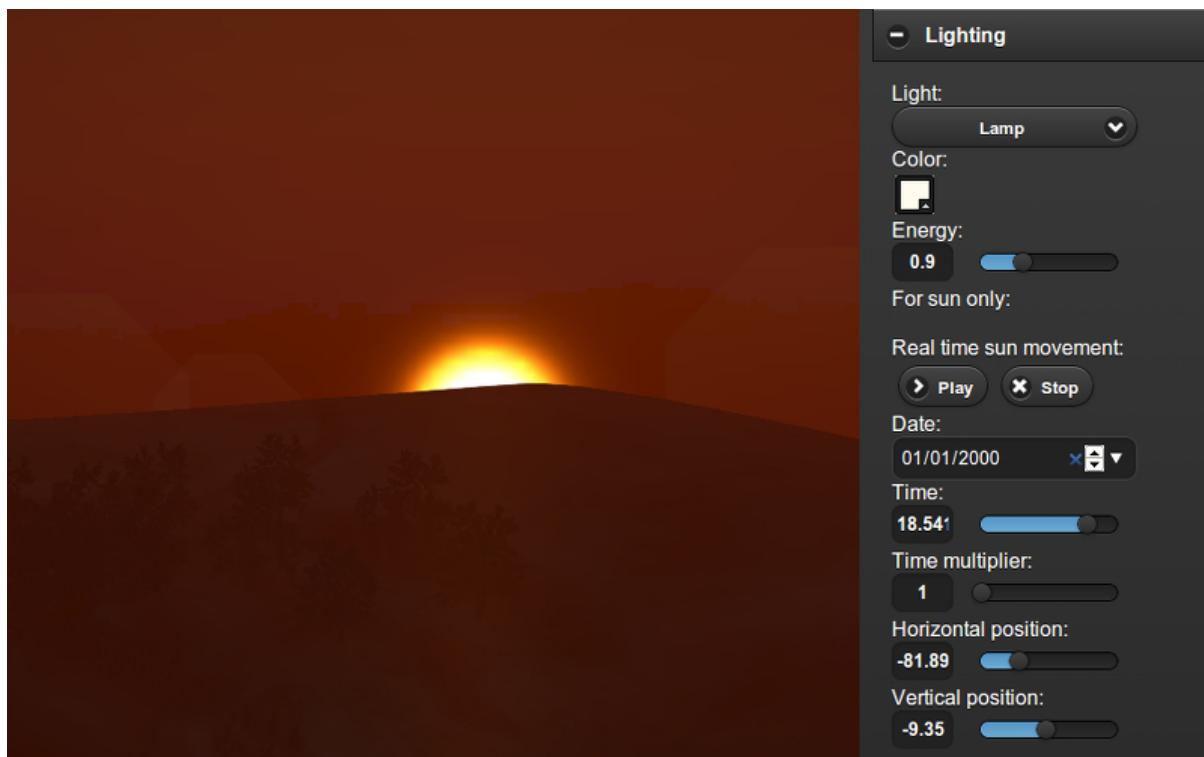
**Blend4Web > Fog Settings > Fog color** Цвет тумана. Значение по умолчанию (0.5, 0.5, 0.5) (серый).

При использовании динамического неба цвет тумана определяется цветом неба.

## 18.2.3 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*.



## 18.2.4 Звезды

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



## 18.3 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)

### 18.3.1 Активация

Добавить на сцену объект - силовое поле типа Wind.

### 18.3.2 Настройка

**Направление** Направление задается посредством вращения объекта - силового поля.

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

### 18.3.3 Анимация травы и крон деревьев

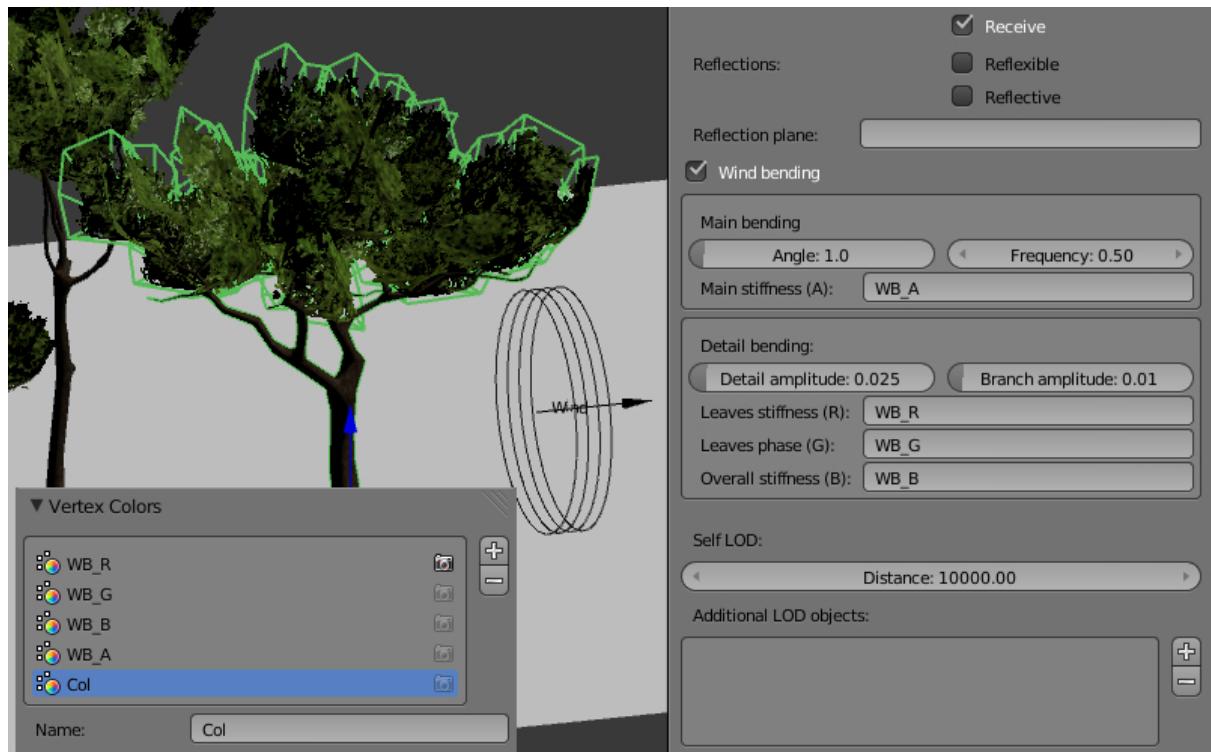
Подготовка ресурсов для рендеринга травы описана в разделе [Травяной покров](#).

## 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 “детализованного” отклонения под действием ветра (в градусах). Значение по умолчанию 0.1.

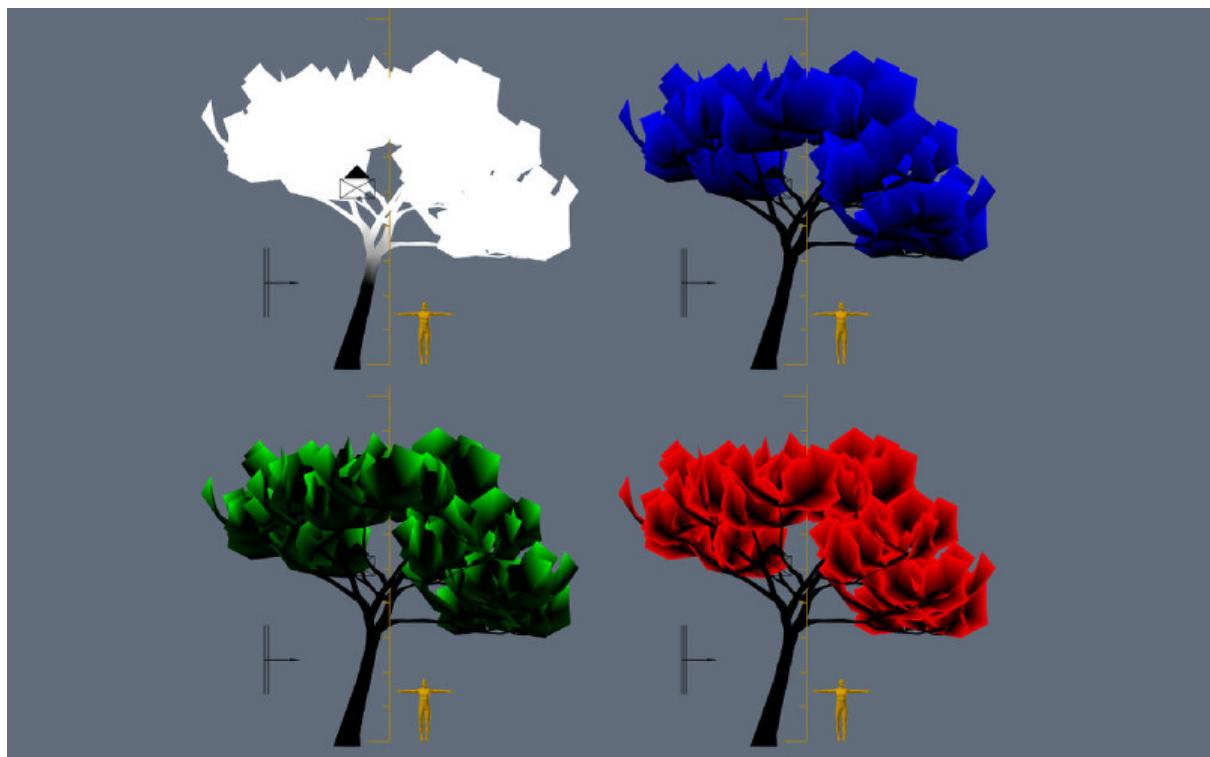
**Detail bending > Branch amplitude** Амплитуда угла отклонения ветвей под действием ветра (в градусах). Значение по умолчанию 0.3.

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

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

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

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



---

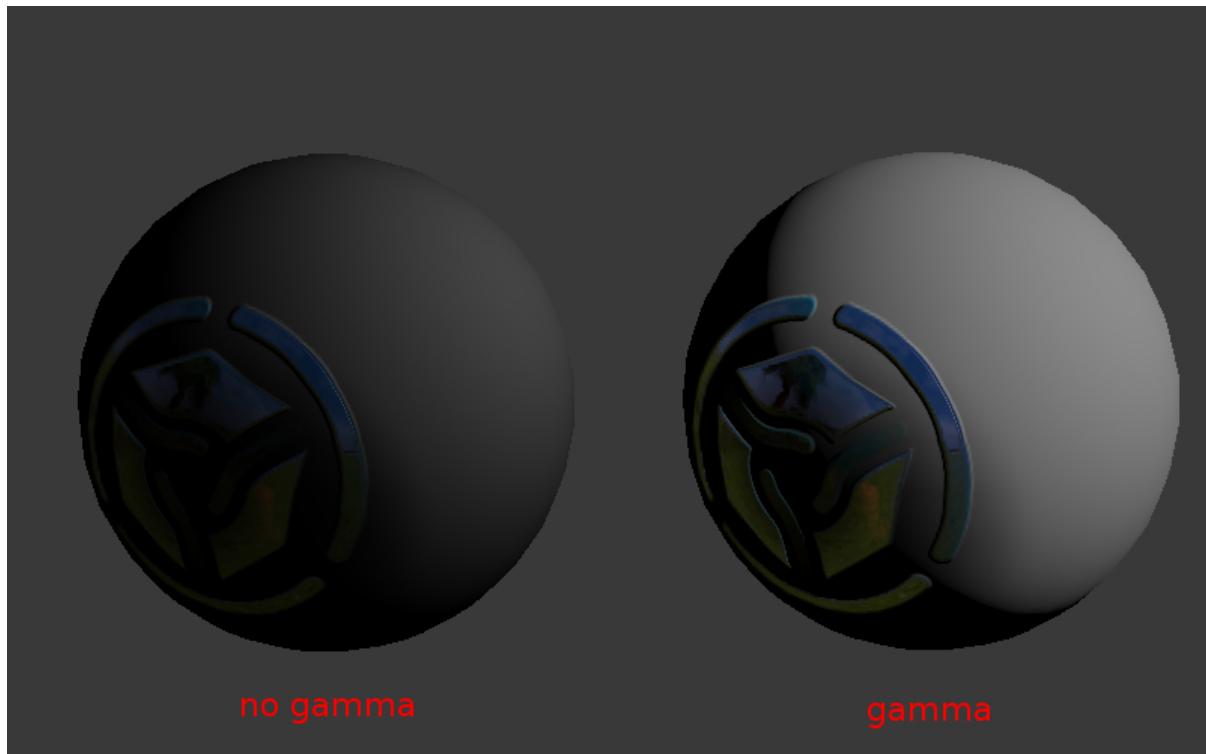
## Гамма-коррекция и альфа-композитинг

---

### 19.1 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

## 19.2 Человеческое зрение, мониторы и гамма-коррекция

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.

## 19.3 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).

## 19.4 Коррекция в нодовых материалах

### 19.4.1 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.

### 19.4.2 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*.

### 19.4.3 Карты нормалей

При использовании карт нормалей никакие преобразования не производятся.

#### 19.4.4 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
Вертексы цвет для окраски	SRGB_TO_LINEAR
Вертексы цвет для маски	не требуется
Карта нормалей	не требуется

## 19.5 Альфа-композитинг

### 19.5.1 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  $\alpha_b$  во втором слагаемом. Таким образом, для осуществления альфа-композитинга должно быть известно не только значение  $\alpha_a$  пикселя-источника, но и значение  $\alpha_b$  пикселя, поверх которого осуществляется рендеринг.

В случае предварительного умножения значений  $\alpha$  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  $\alpha_o$ :

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

Предварительное умножение цветовых каналов на значения  $\alpha$  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  $\alpha_o$  на каждой последующей итерации.

### 19.5.2 Реализация

Функция смешивания в движке Blend4Web имеет вид:

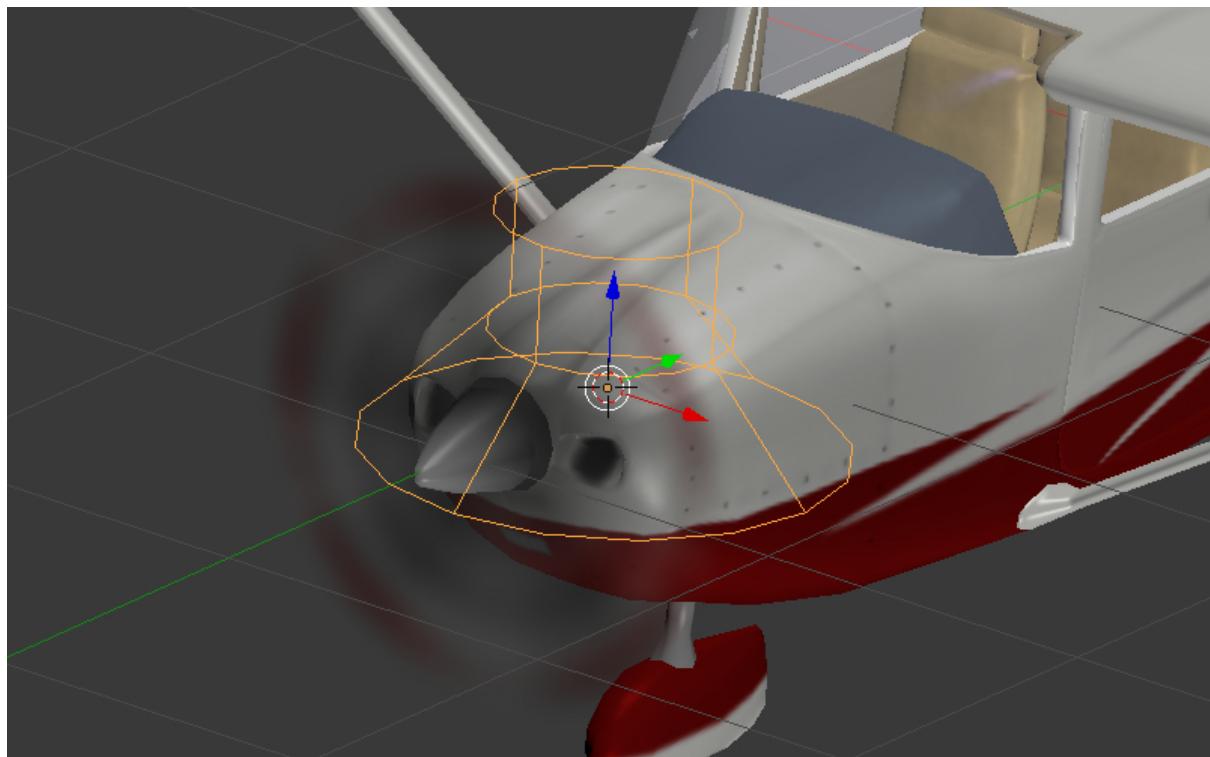
```
gl.blendFunc(gl.ONE, gl.ONE_MINUS_SRC_ALPHA);
```

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  $\alpha$ .

## Звуковая подсистема

---

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



### 20.1 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



## 20.2 Обработка и кодирование

### 20.2.1 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.

В состав дистрибутива включен Python скрипт (`scripts/converter.py`) для конвертации исходных файлов в другие форматы с целью расширения спектра поддерживаемых платформ. Для конвертации необходимо выполнить команду:

```
> ./converter.py convert_sounds
```

Преобразование происходит по схеме:

- ogg -> mp4
- mp3 -> ogg
- mp4 -> ogg

Конвертация ресурсов происходит с потерей качества, поэтому сконвертированные файлы получают суффикс `.lossconv`.

---

## Физика

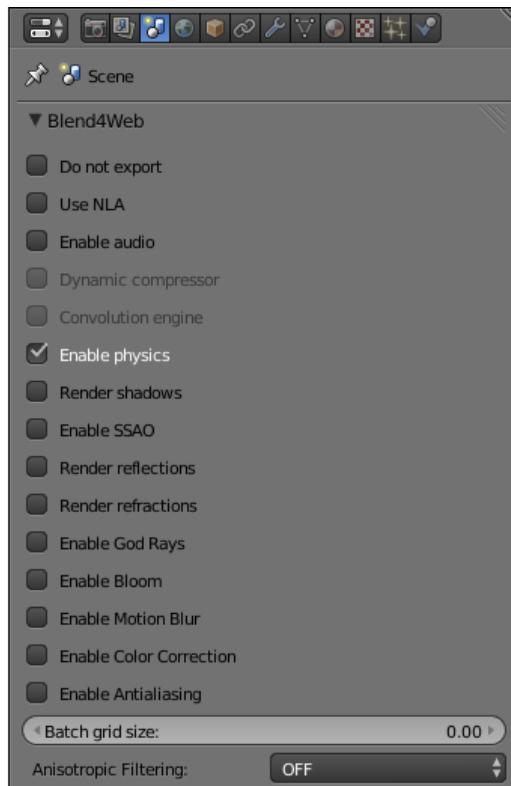
---

### 21.1 Подготовка к использованию

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. Подключение физической подсистемы и указание пути загрузки модуля `uranium.js` осуществляется приложениями посредством внешнего API:

```
b4w.config.set("physics_enabled", true);
b4w.config.set("physics_uraniun_path", "../../external/deploy/apps/common/uranium.js");
```

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

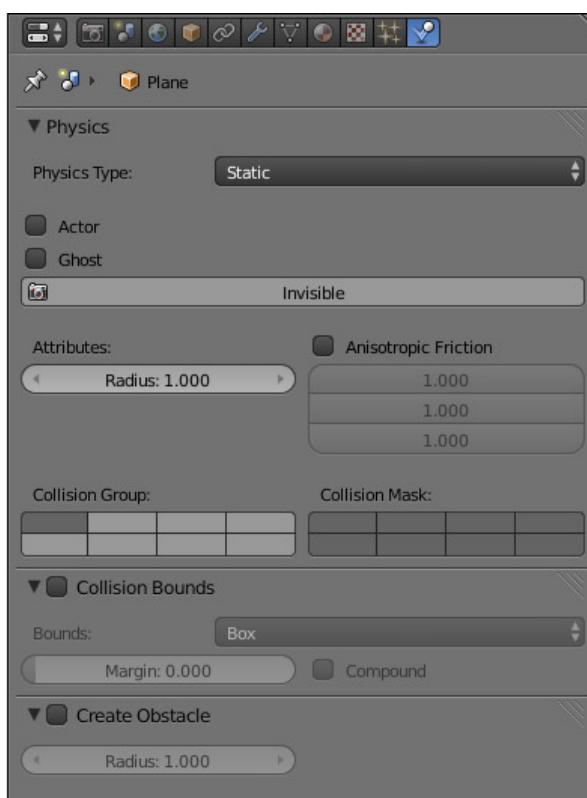


Настройка физических параметров производится в режиме Blender Game.



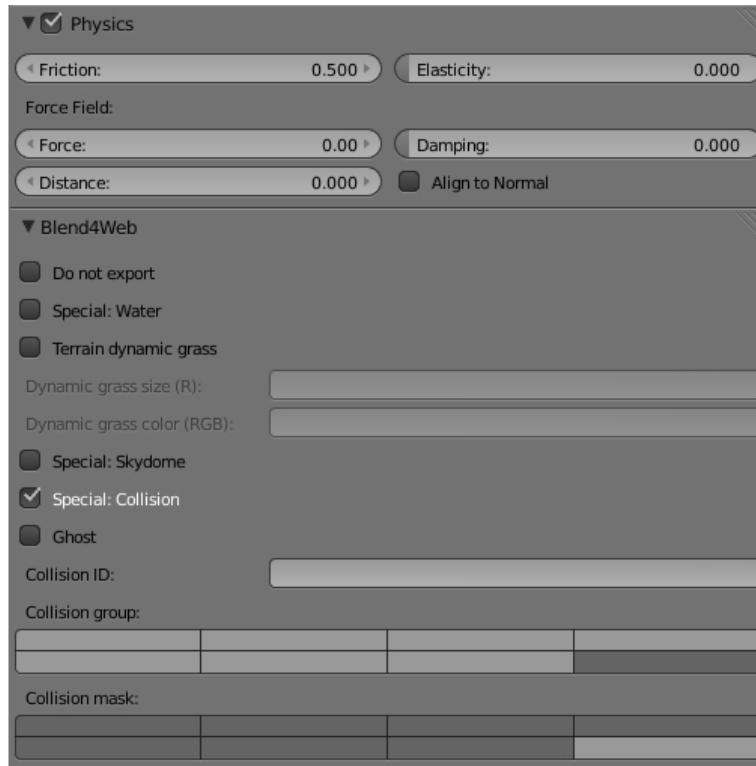
## 21.2 Статический тип физики

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).



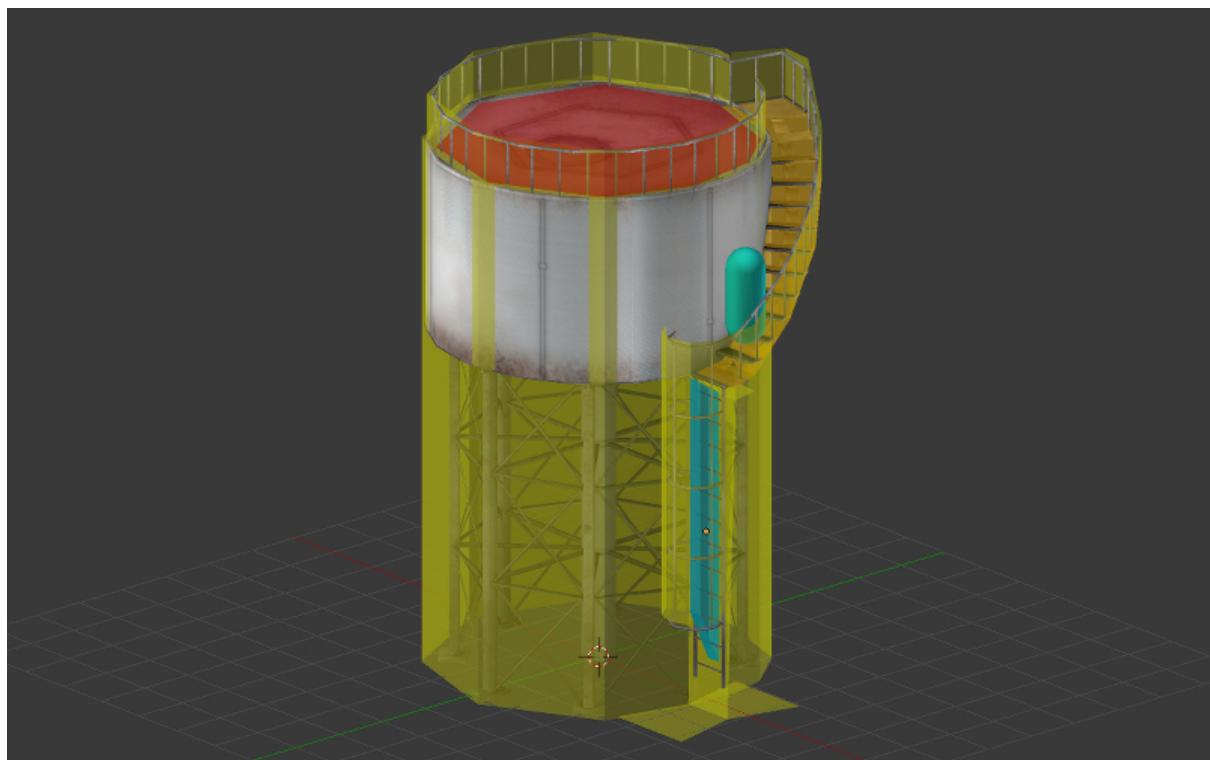
Меш может быть покрыт одним или несколькими физическими материалами. Во вкладке Material должна быть включена опция Blend4Web > Special: Collision. Также во вкладке Material на панели Physics (в режиме Blender Game) распола-

гаются 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.

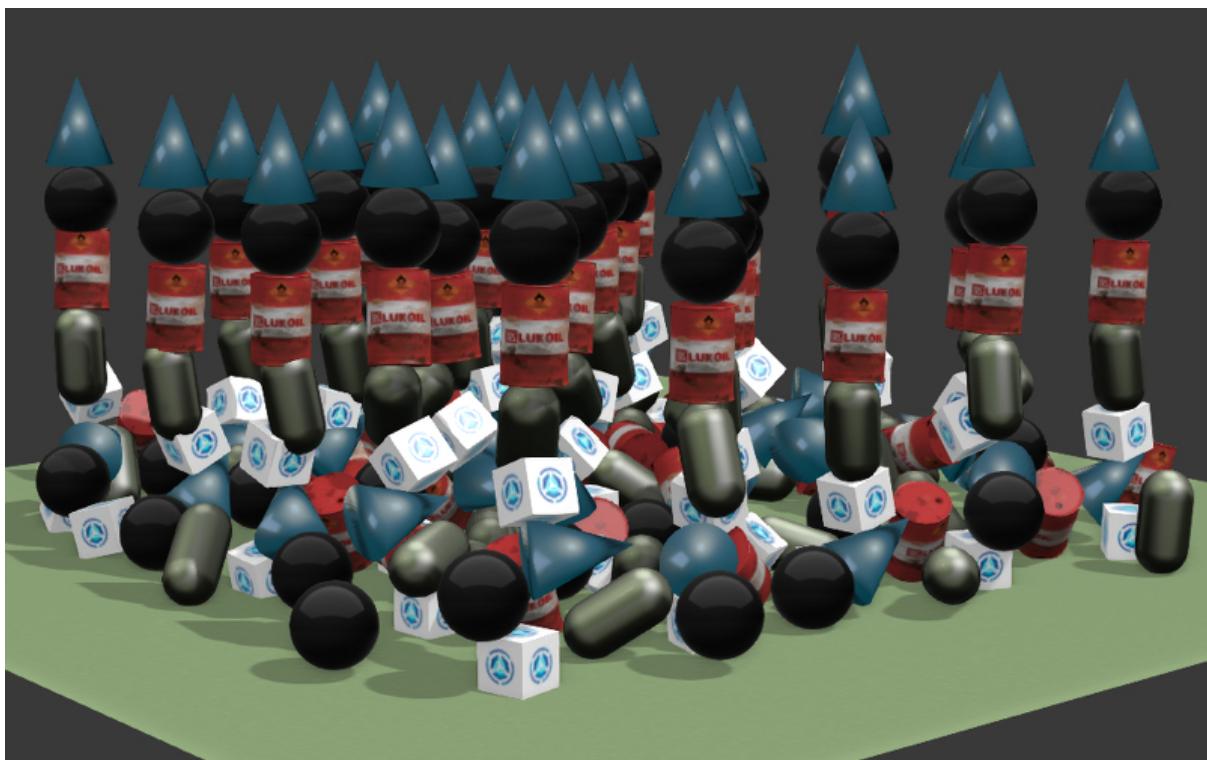
Опция **Ghost** исключает материал из физических взаимодействий, но сообщает приложению о контакте с ним. Пример - определение, что игровой персонаж находится на вертикальной лестнице.



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

## 21.3 Динамический тип физики

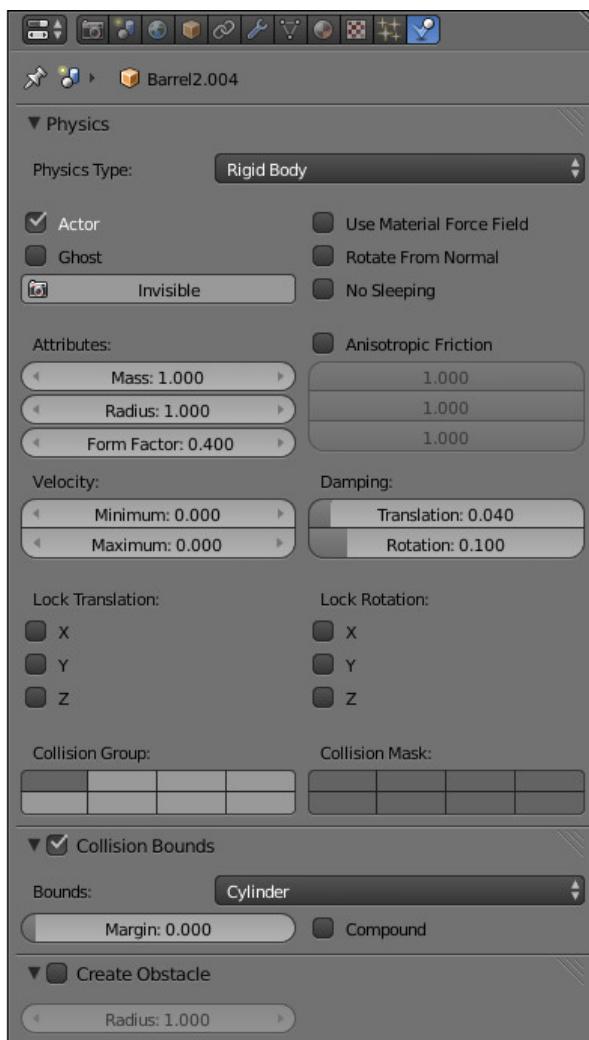
Предназначен для симуляции движения жесткого тела.



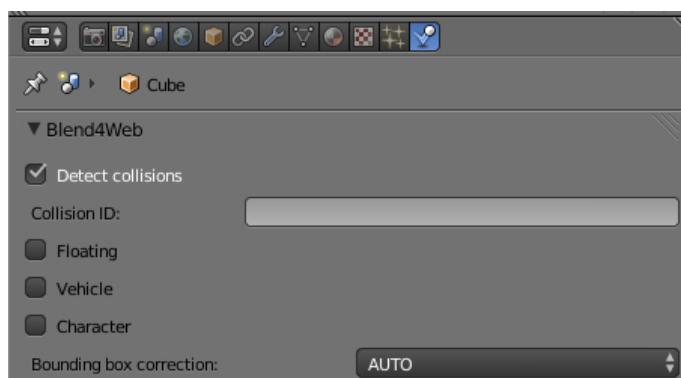
В настройках физики такого объекта для опции **Physics Type** может быть выбрано значение **Rigid Body** (с вращениями) или **Dynamic** (без вращений). В настройках **Collision Bounds** может быть выбран тип коллайдера, поддерживаются: **Box**, **Capsule**, **Sphere**, **Cylinder**, **Cone**. Другие 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.

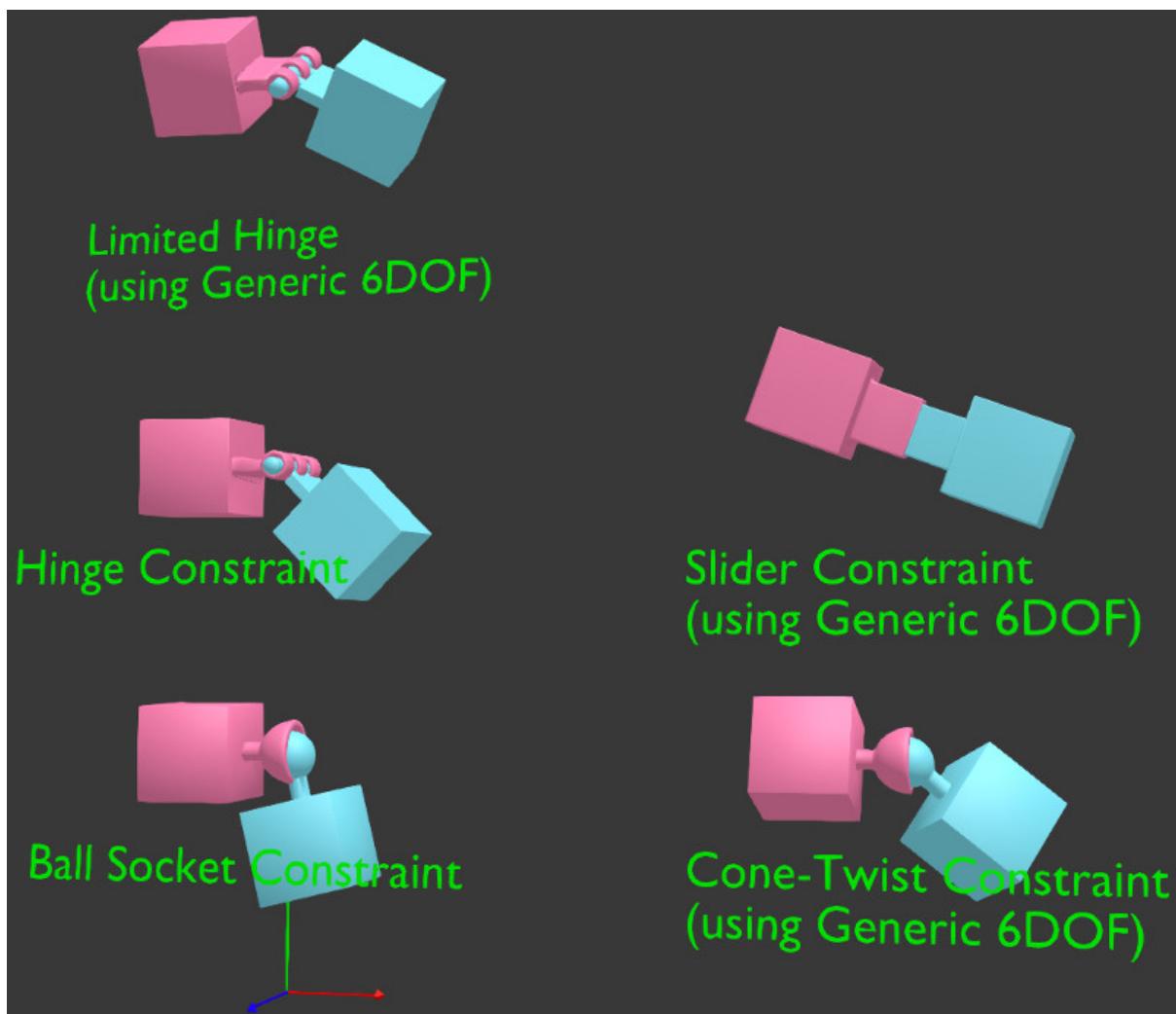


Для материала такого объекта поддерживаются: трение (Friction), упругость (Elasticity). В случае использования на одном меше нескольких материалов физические настройки считаются с первого из них.

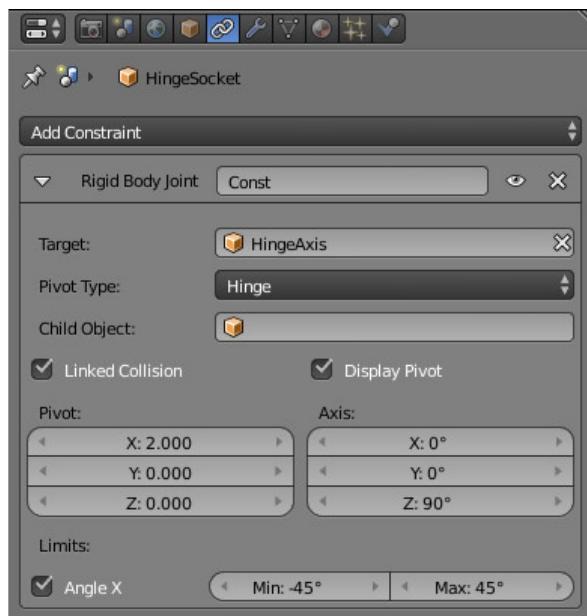
Для объекта-камеры должна использоваться настройка Physics Type = Dynamic, должен быть установлен флаг Detect collisions.

## 21.4 Ограничители (Constraints)

Физические ограничители используются для уменьшения числа степеней свободы объектов.

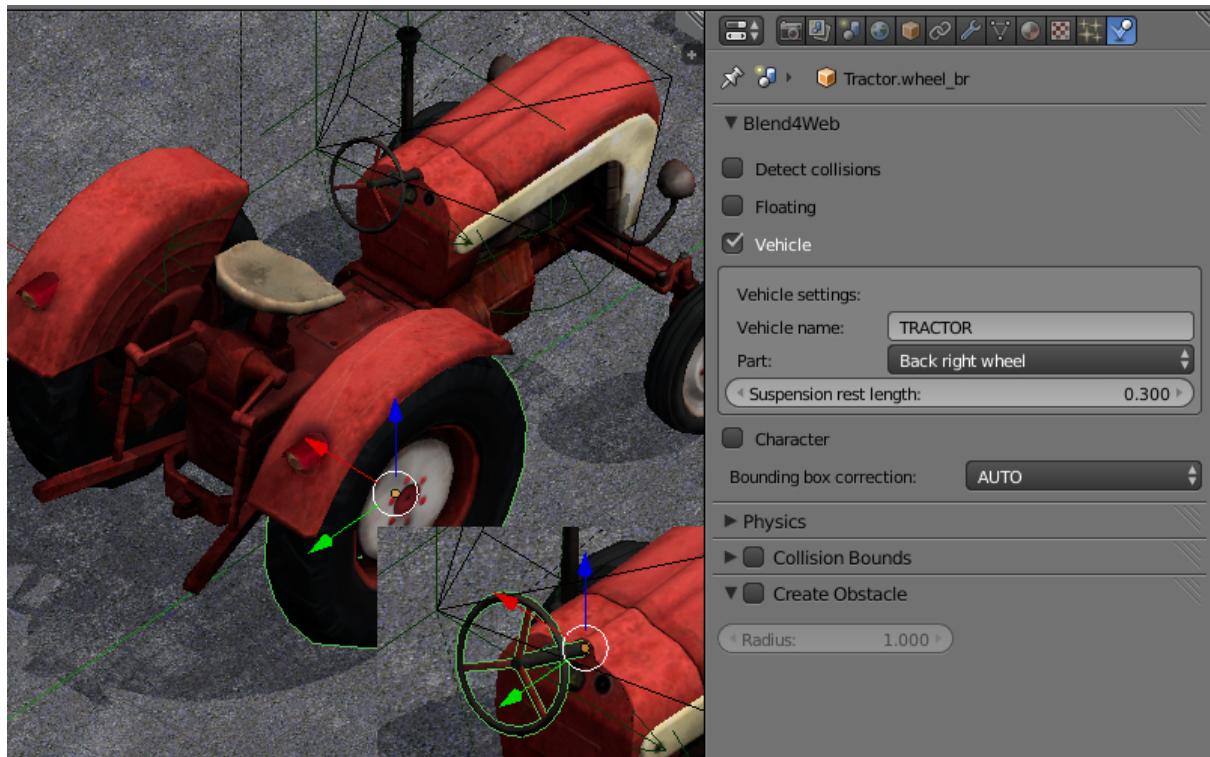


Установка физического ограничителя (**Rigid Body Joint**) на объект происходит в панели **Object Constraints**. Поддерживаемые типы (**Pivot Type**): **Ball**, **Hinge**, **Cone**, **Twist**, **Generic 6 DoF**. Физический ограничитель можно установить на один из двух взаимодействующих объектов, при этом 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.



## 21.5 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**.

### 21.5.1 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

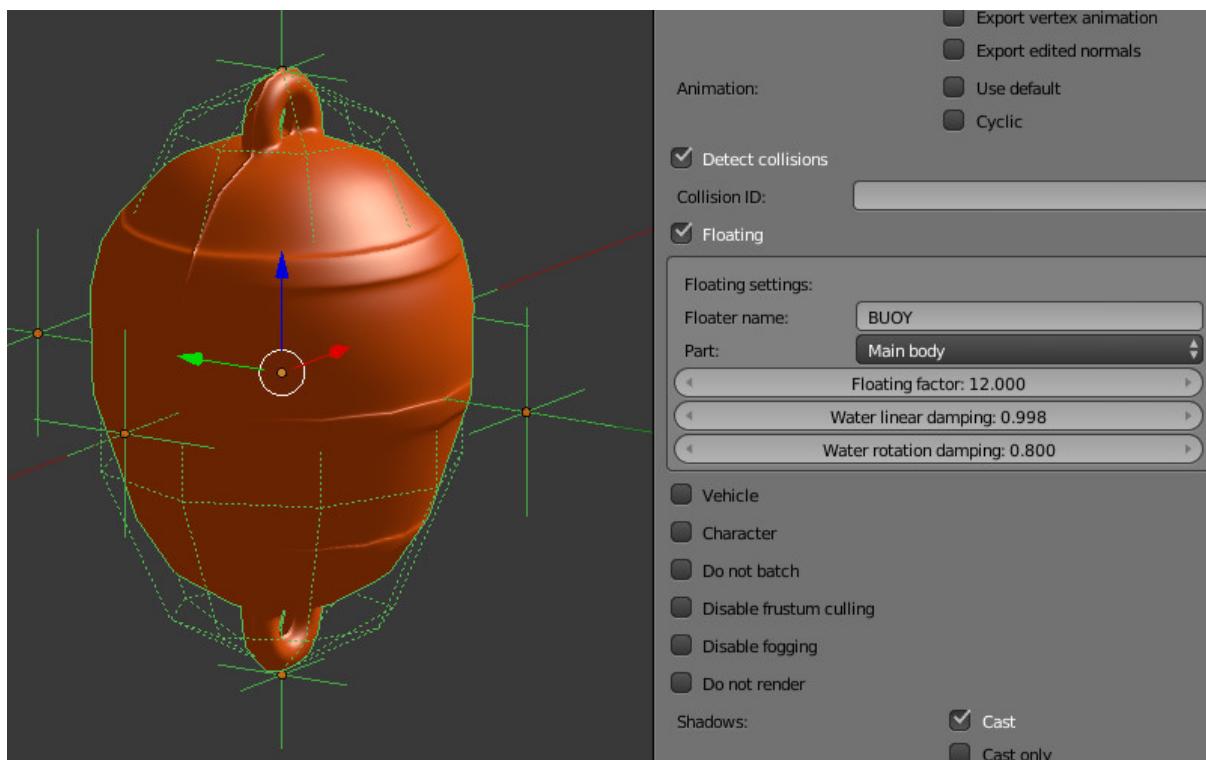
Для рулевого колеса (**Steering wheel**) необходимо указать максимальный угол поворота (**Steering max**) и передаточное отношение угла поворота руля к передним колесам (**Steering ratio**). Максимальное значение угла поворота указывается в оборотах. Один оборот равен 360 градусам. Таким образом, поставив **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.

## 21.6 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**

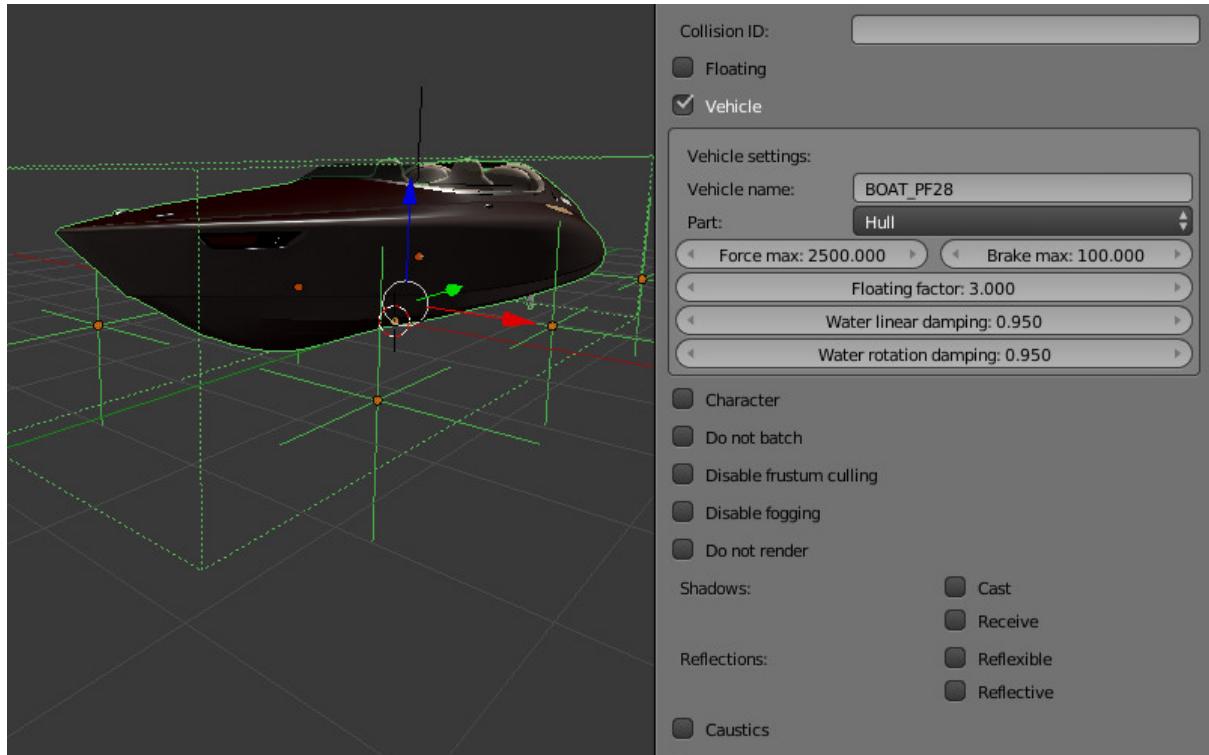
### 21.6.1 Параметры настройки плавающего объекта

**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.

## 21.7 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**.

### 21.7.1 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.

---

## Нелинейная анимация

---

### 22.1 Редактор NLA

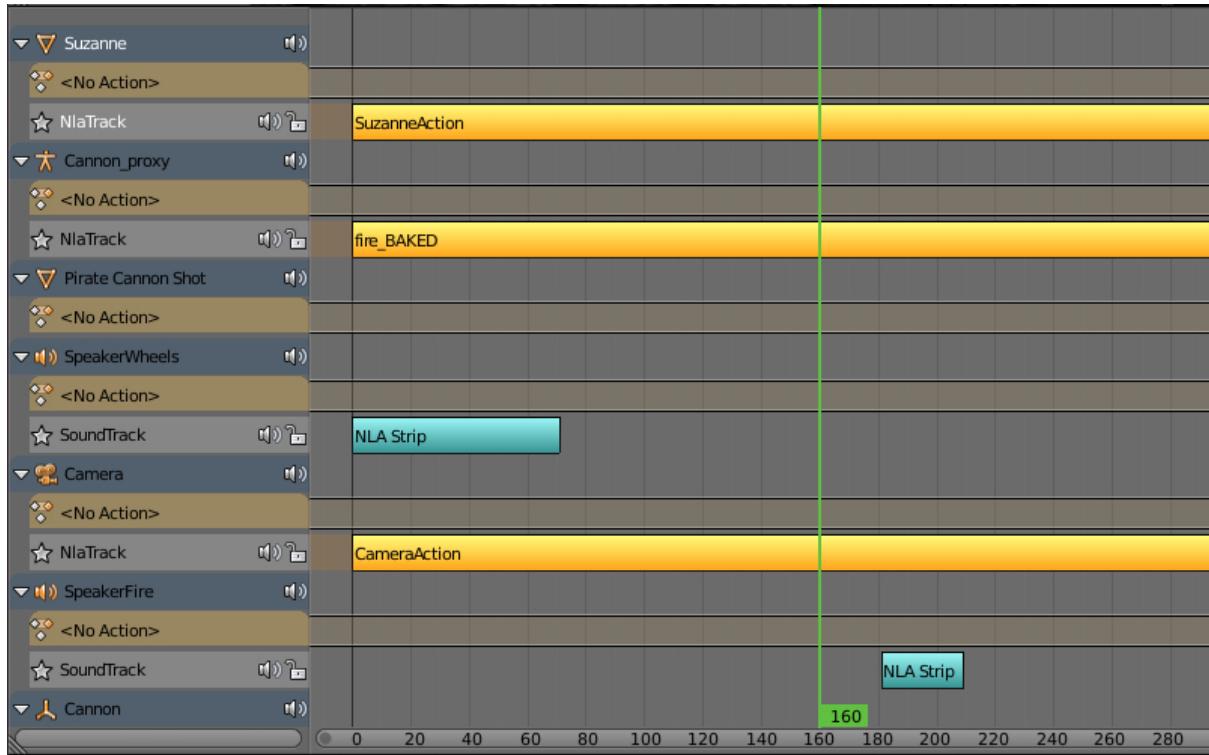
Редактор нелинейной анимации, имеющийся в программе Blender, позволяет в удобной форме задавать поведение сцены. С его помощью можно реализовать несложные сценарии. Таким образом исключается необходимость программирования простых сцен и приложений.



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)



### 22.1.1 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.

### 22.1.2 Дополнительные настройки

Настройка сцены *Blend4Web > Cyclic NLA* позволяет активировать режим циклической NLA-анимации.

### 22.1.3 Ограничения

- Вертекальная анимация не поддерживается.
- Масштабирование и зацикливание отдельных полос анимации (Strip) не поддерживается.
- Одновременное проигрывание различных видов анимации для одного объекта не поддерживается.

## 22.2 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.

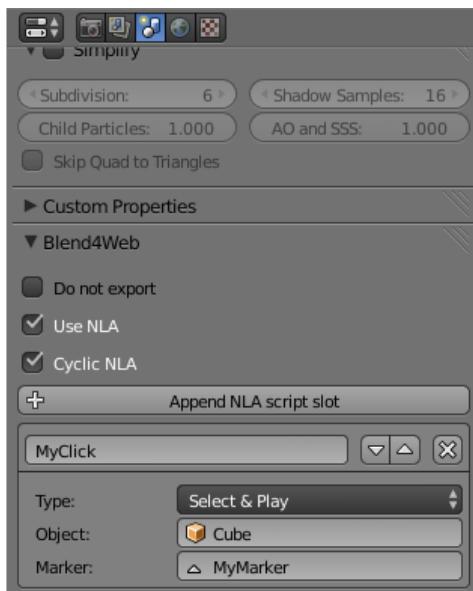
---

**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.).

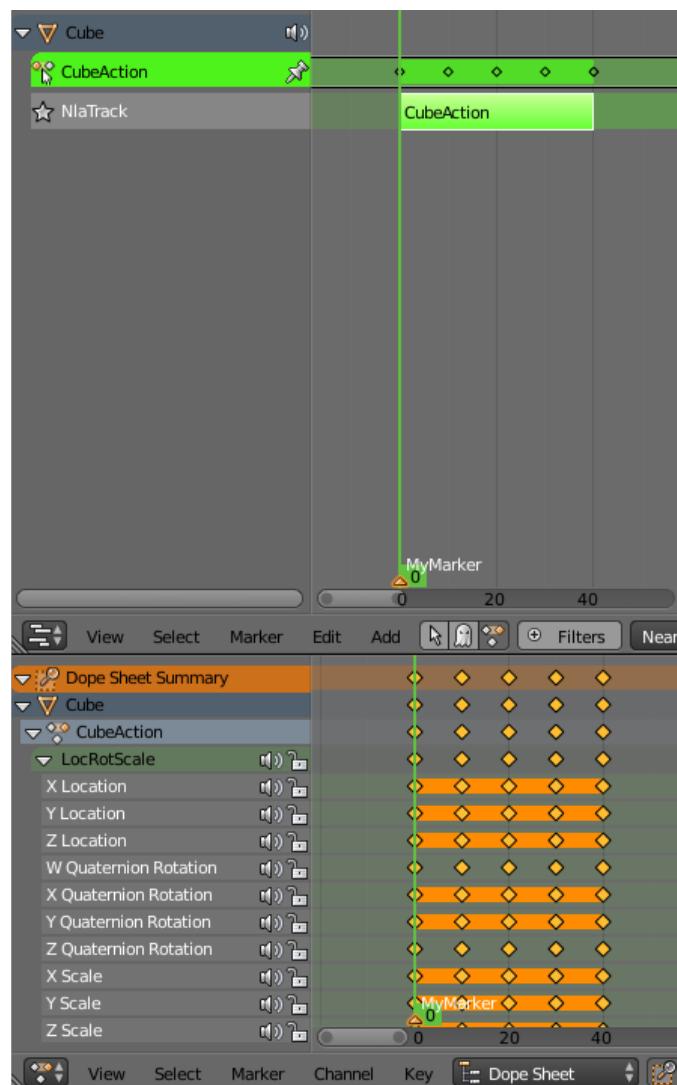
Пример использования NLA Script:



Ниже рассмотрены все возможные варианты слотов.

### 22.2.1 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.



## 22.2.2 Select & Play

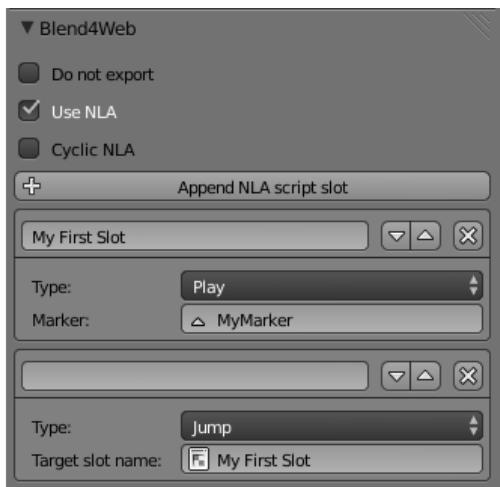
Ожидать, когда пользователь выберет объект (с помощью клика мышью в версии движка для десктопа, либо нажатия на экране в мобильной версии). Если выбран объект, указанный в параметрах слота, начать играть анимацию аналогично слоту **Play**, если же был выбран любой другой объект - немедленно передать 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*.

### 22.2.3 Jump

Перейти к указанному слоту.



### 22.2.4 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).

### 22.2.5 Conditinal 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.

### 22.2.6 Register Store

Записать числовое значение в регистр.

### 22.2.7 Math Operation

Выполнить математическую операцию и сохранить результат в регистр. Любой из параметров (операндов) может быть либо числовым значением, либо регистром.

## 22.2.8 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

---

### 23.1 Простейшее приложение (Hello world!)

Простейшее приложение на движке 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>
```

Базовый код движка подключается с помощью тега `<script src="...">`. Далее, приложение ожидает окончания загрузки страницы и выводит сообщение с текущей версией 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](#).

Файлы со скомпилированным кодом движка находятся в директории SDK `external/deploy/apps/common`. Файл `b4w.min.js` содержит в себе базовую версию, в случае необходимости использования аддонов, вместо него используется `b4w.full.min.js`.

## 23.2 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.

## 23.3 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: "body_id",
    callback: load_cb
});

function load_cb() {
    m_data.load("some_scene.json");
}

</script>
</head>

<body id="body_id"></body>

</html>
```

В данном случае модуль `app` создаст элемент `<canvas>` внутри контейнера с указанным идентификатором `body_id`, осуществит инициализацию движка при загрузке страницы и сообщит о её окончании с помощью обработчика `load_cb`.

## 23.4 Система модулей

Несмотря на то, что движок предоставляет прикладному программисту API в объёме десятков модулей, в процессе работы он занимает в глобальном пространстве имён 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();
```

## 23.5 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* имеются 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).

## 23.6 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  $A^*B$  является поворотом на А. То есть у объекта уже имеется некий поворот Б, который мы дополняем новым поворотом на А.

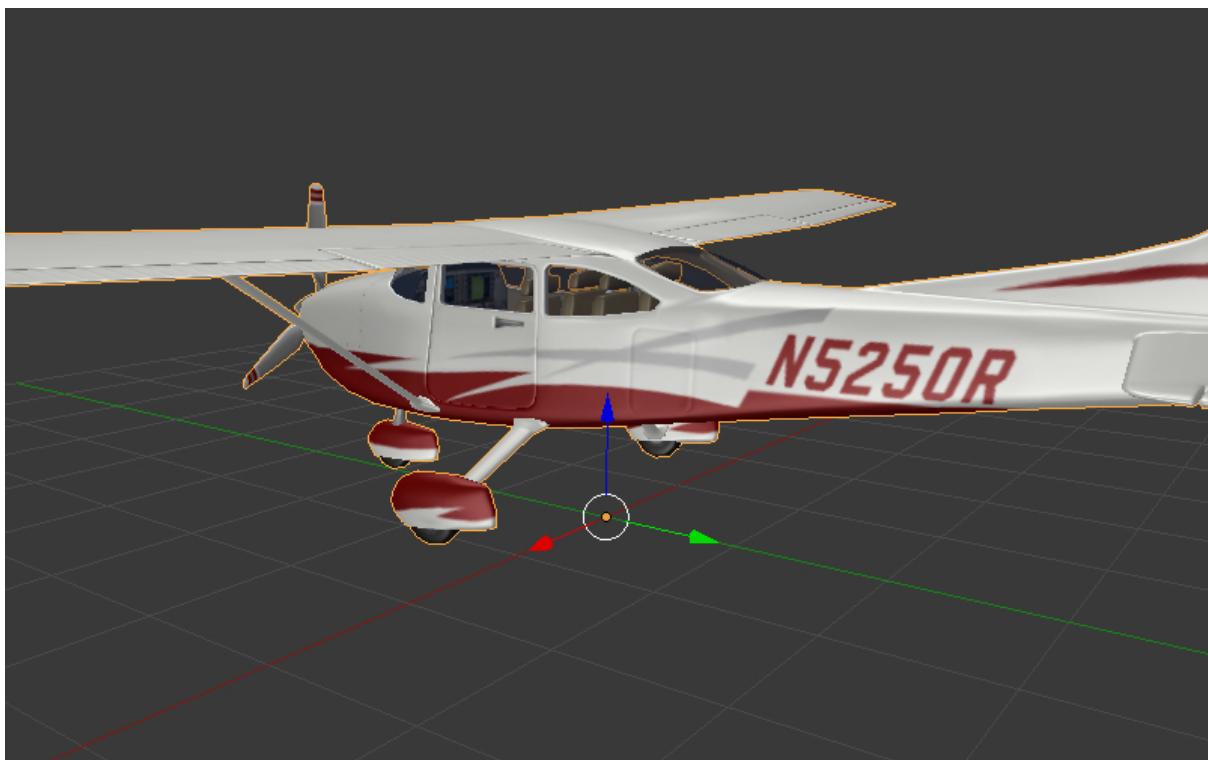
***quat.setAxisAngle*** Кватернион представляет собой иную форму записи поворота относительно произвольной оси (вектора) на произвольный угол. Положительное направление поворота отсчитывается против часовой стрелки, если смотреть с конца вектора. Например вызов `quat.setAxisAngle([1, 0, 0], Math.PI/3, 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.

### 23.6.1 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);
```

## 23.7 Событийная модель

Событийная модель предоставляет унифицированный интерфейс для описания изменения состояний 3D сцены, упрощая обработку событий физики и действий пользователя.

### 23.7.1 Сенсоры

Основным блоком событийной модели является сенсор (sensor). Сенсор является программной сущностью, и может быть только активным (1, единица) или неактивным (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.

### 23.7.2 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");
}

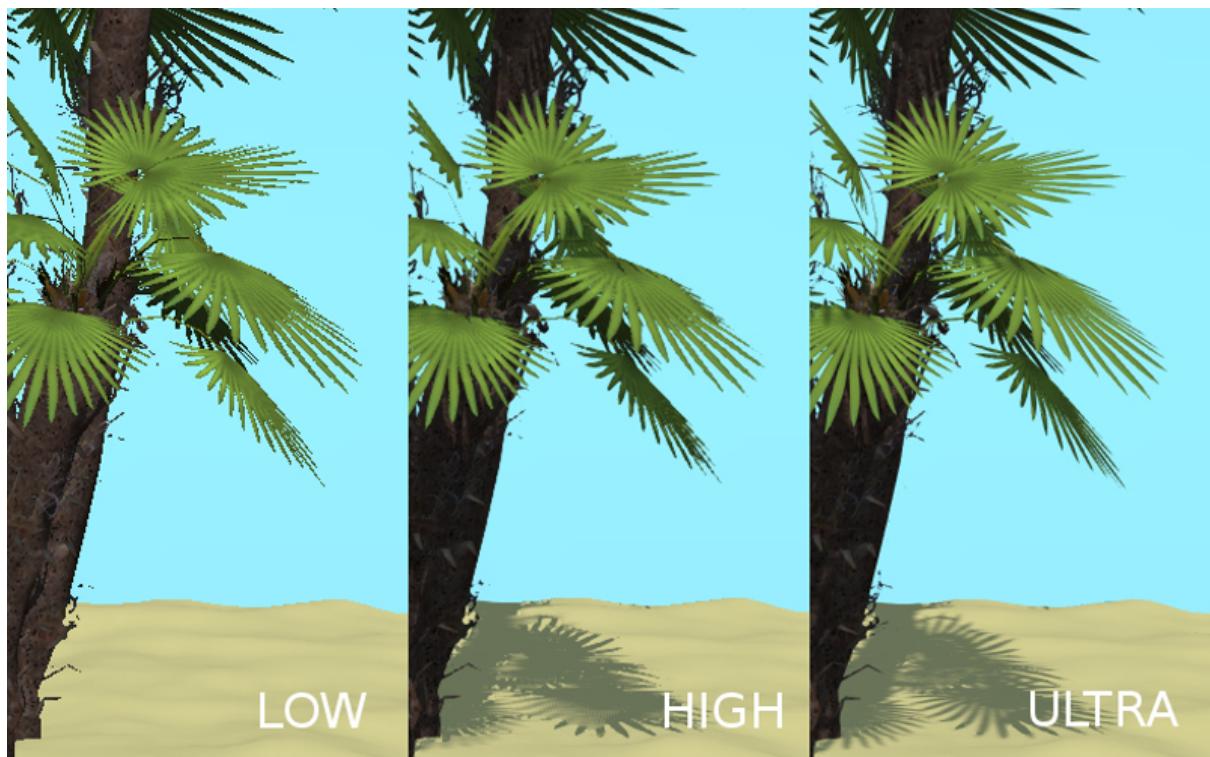
// create the manifold
m_controls.create_sensor_manifold(stone, "IMPACT", m_ctl.CT_SHOT,
    impact_sens_array, impact_sens_logic, impact_cb);
```

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).

## 23.8 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
});
```

## 23.9 Файловая структура SDK

**apps\_dev** исходный код приложений (не все приложения доступны в бесплатной версии)

**closure-compiler** компилятор Google Closure, файлы исключений к нему, генераторы файлов исключений

**csrc** исходный код бинарной части экспортера движка и других утилит на языке C

**doc\_src** исходный код настоящего руководства пользователя на языке разметки reST

### **external**

**blender** исходные файлы сцен в формате Blender (не все сцены доступны в бесплатной версии)

**blender\_scripts** экспортер и вспомогательные скрипты для Blender'a

### **deploy**

**api\_doc** документация API движка для разработчиков в формате HTML (собирается автоматически, на основе исходного кода движка)

**apps** 3D-приложения, предназначенные для развертывания, директория дублирует *apps\_dev*

**common** Файлы скомпилированного движка. Используются всеми приложениями из состава SDK (отсюда и название).

**assets** загружаемые ресурсы: сцены, текстуры, звуковые файлы (не все ресурсы доступны в бесплатной версии)

**assets.json** метаданные с информацией о сценах, загружаемых просмотрщиком сцен Viewer

**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

**reexporter.py** и **Makefile** Python-скрипт и файл сборки для автоматического экспорта всех сцен в *external/deploy/assets*

### **glsl\_utils**

**compiler** компилятор GLSL-шейдеров движка

**out** содержит результат компиляции GLSL-шейдеров движка

**pegjs** грамматики парсер-генератора PEG.js для реализации препроцессора GLSL, а также скрипт для генерации модулей парсеров из этих грамматик

**index.html** web-страница со ссылками на 3D-приложения (недоступен в бесплатной версии)

**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 вызова Google Closure compiler с целью минификации и обfuscации 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

**gpu\_shader\_analyzer\_server.py** скрипт, запускающий локальный веб-сервер, который осуществляет подсчет сложности шейдеров

**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

**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 экспотируемые файлы)

**screencast.sh** скрипт для записи видео с экрана

**shaders** GLSL-шейдеры движка

**src** основной исходный код ядра движка

**addons** исходный код дополнений (не входят в ядро движка, подключаются отдельно)

**ext** исходный код внешних объявлений (формирует API движка)

**third\_party** код сторонних библиотек

**uranium** исходный код и скрипты сборки физического движка Uranium (форк Bullet)

**VERSION** содержит текущую версию движка

---

## Разработчикам движка

---

### 24.1 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).

#### 24.1.1 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.

## 24.2 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 корня репозитория:**

- скрипт сборки: `./csrc/b4w_bin/build.py`
- аддон Blend4Web: `./external/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.

## 24.3 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
LLVM, Clang	llvm, clang	сборка Uranium
Emscripten	из исходных текстов	сборка Uranium
ImageMagick, GraphicsMagick	imagemagick, graphicsmagick	конвертация ресурсов
NVIDIA Texture Tools	libnvtt-bin	конвертация ресурсов
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	сборка документации (документация на API)

## 24.4 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

## 24.5 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 график 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.

## 24.6 Шейдеры

### 24.6.1 Обфускатор

Используемые в движке шейдеры подвергаются обработке обфускатором. Для запуска обфускации требуется выполнить одну из команд в корне репозитория:

- **make compile\_shaders** - проверка, обфускация и экспорт скомпилированных шейдеров
- **make verify\_shaders** - только проверка и обфускация

Обфускатор служит для сокращения объема, оптимизации и затруднения понимания GLSL-кода. На данный момент в нем реализованы следующие процедуры:

- удаление лишних пробелов, переводов строк и повторяющихся символов „,”

- замена пользовательских идентификаторов более короткими односимвольными, двухсимвольными и т.д. именами
- вывод сообщений о неиспользуемых переменных и функциях (dead code)
- проверка синтаксиса шейдеров
- поддержка import/export-механизма и проверка шейдеров на соответствие ему

В процессе обфускации сначала осуществляется синтаксический анализ (парсинг) текста шейдера. Соответствующий парсер создается автоматически на основе грамматики с помощью генератора [PEG.js](#). Далее по данным парсинга производится оптимизация и валидация 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

## 24.6.2 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 объявление, 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.

### 24.6.3 Рекомендации и ограничения по использованию обфускатора

В связи с наличием препроцессинга, необходимостью совместной обработки нескольких шейдеров и include-файлов, а также особенностями реализации обфускатора 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 обфускации 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`. Также 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
...
}
```

#### 24.6.4 Поддержка WebGL-расширений

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

#### 24.6.5 Ошибки обфускатора

В случае ошибки обфускатор 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. Использование git

---

### 25.1 Общие сведения

Для организации командной работы над проектом может быть использована система контроля версий файлов git.

Основная функция git - сохранение истории изменений с возможностью возврата к предыдущим версиям файлов. Другая важная функция - синхронизация с автоматическим слиянием изменений.

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.

### 25.2 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.

Для git имеются графические оболочки, предоставляющие пользовательский интерфейс к командам. Тем не менее, непосредственный ввод команд git достаточно удобен.

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
```

## 25.3 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.

## 25.4 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")
```

## 25.5 Перед коммитом

### 25.5.1 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);
+    }
}
```

### 25.5.2 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.

### 25.5.3 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.

## 25.6 Подготовка к коммиту

### 25.6.1 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`.

### 25.6.2 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`.

## 25.7 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`.

## 25.8 Синхронизация между репозиториями

### 25.8.1 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 lixer: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    external/deploy/assets/location_agriculture/textures/rotonda_02_diff.p
```

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: external/blender/landscape_objects/Fallen_tree.blend (...)
Auto-merging external/blender/landscape_objects/Fallen_tree.blend
CONFLICT (content): Merge conflict in external/blender/landscape_objects/Fallen_tree.blend
Automatic merge failed; fix conflicts and then commit the result.
```

Порядок действий при возникновении конфликтов описан далее.

## 25.8.2 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 gfxteam@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 gfxteam@lixer:blend4web.git
 ! [rejected]      master -> master (non-fast-forward)
error: failed to push some refs to 'gfxteam@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`.

## 25.9 Разрешение конфликтов

### 25.9.1 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.

### 25.9.2 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:    external/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.

### 25.9.3 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 external/blender/landscape_objects/Fallen_tree.blend
```

Выбрать удаленную версию файла (`- --theirs`). Его можно открыть и убедиться в этом.

```
> git checkout --theirs external/blender/landscape_objects/Fallen_tree.blend
```

Снова выбрать локальную версию файла (`- --ours`).

```
> git checkout --ours external/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.

### 25.9.4 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 экспортанные текстовые файлы сцен (заканчивающиеся на `.json`) проще повторно экспортовать.

## 25.9.5 Корректирующий коммит

После выбора нужных файлов или редактирования изменений, добавить их для коммита:

```
> git add external/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`.

## 25.10 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
```

## 25.11 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

---

### 26.1 Проблемы при запуске рендерера

1. Появляется сообщение “*Browser could not initialize WebGL.*”

Следует выполнить действия, описанные в разделе *Ошибка инициализации WebGL*.

2. Видны элементы интерфейса и фон, но сцена по умолчанию не отображается (куб с логотипом). При этом тестовый сайт <http://get.webgl.org/> и другие WebGL приложения работают корректно.

Вероятные причины:

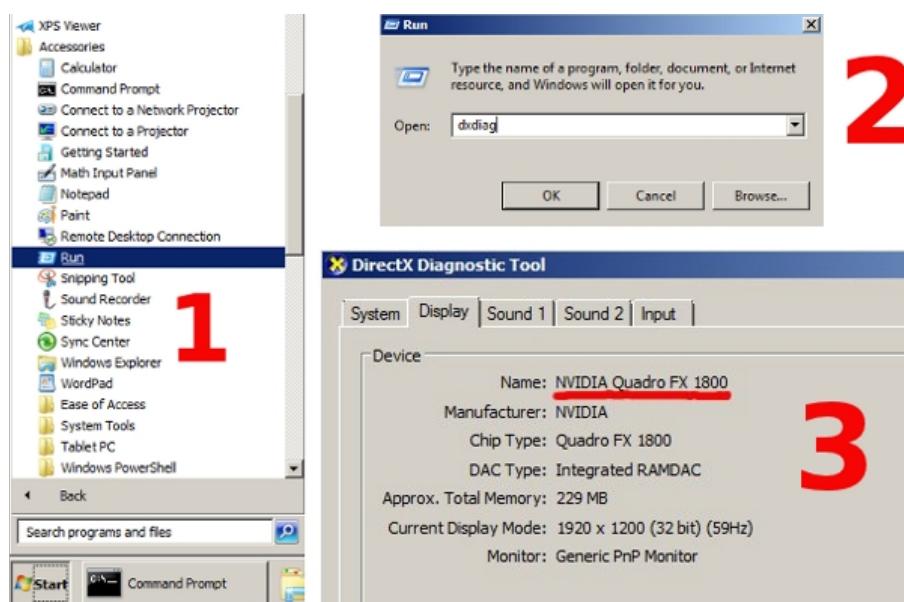
- Браузер не настроен для работы с локальными ресурсами. См. раздел *Настройка браузера для загрузки локальных ресурсов*.
- Файлы ресурсов, которые пытается загрузить рендерер, были перемещены или удалены.
- Используется комбинация браузера, драйверов и операционной системы, не обеспечивающая полную поддержку WebGL (примеры: Internet Explorer 11 / Windows, открытые драйверы / Linux).

### 26.2 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...

или воспользоваться средством диагностики **DirectX dxdiag** (только для Windows).

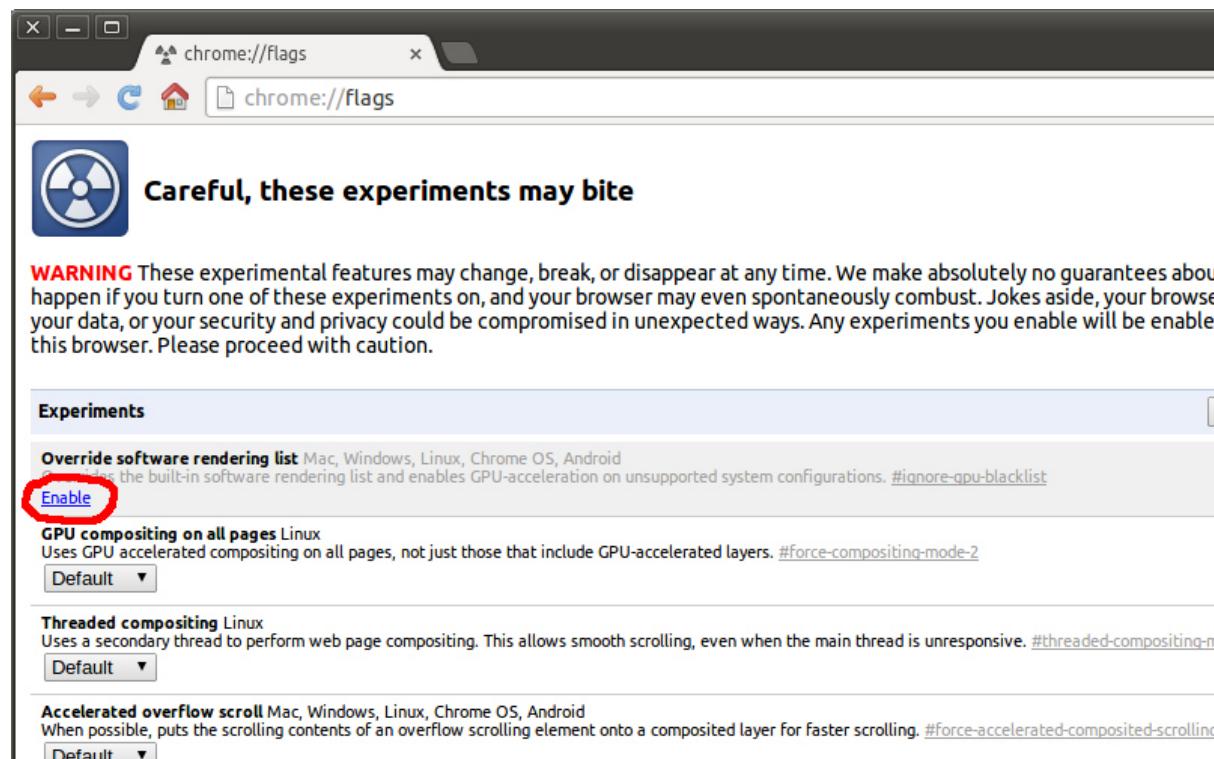


Необходимо загрузить драйверы с соответствующего центра поддержки (например, Intel, Nvidia, AMD/ATI). После установки драйверов перезагрузить систему.

- 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`.

Preference Name	Status	Type	Value
webgl.can-lose-context-in-foreground	default	boolean	true
webgl.default-no-alpha	default	boolean	false
webgl.disable-extensions	default	boolean	false
webgl.disabled	default	boolean	false
webgl.enable-draft-extensions	default	boolean	false
<b>webgl.force-enabled</b>	<b>user set</b>	<b>boolean</b>	<b>true</b>
webgl.force-layers-readback	default	boolean	false
webgl.lose-context-on-heap-minimize	default	boolean	false
webgl.max-warnings-per-context	default	integer	32
webgl.min_capability_mode	default	boolean	false
webgl.msaa-force	default	boolean	false
webgl.prefer-16bpp	default	boolean	false
webgl.prefer-native-gl	default	boolean	false
webgl.shader_validator	default	boolean	true

**Примечание:** 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.

## 26.3 Использование локального web-сервера

Простым вариантом обеспечения просмотра локальных ресурсов в браузерах может быть запуск web-сервера из стандартной библиотеки [Python](#).

*На Windows:*

1. Загрузить и инсталлировать последнюю версию Python с [официального сайта](#). На сегодняшний день это версия 3.4, и по умолчанию установка произойдет в директорию Python34 на диске С.
2. Запустить командную строку (Command Prompt).
3. Выполнить команды:

```
> c:  
> /Python34/python -m http.server
```

4. Перейти на страницу <http://localhost:8000>, на которой выбрать нужный файл для отображения.

*На Linux:*

```
> python -m SimpleHTTPServer
```

или:

```
> python3 -m http.server
```

Можно указать порт дополнительным параметром:

```
> python -m SimpleHTTPServer 8080
```

---

## Замечания к релизам

---

### 27.1 v14.10

#### 27.1.1 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, вертексную 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 экспорта 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 частичек.*
- *Улучшена стабильность работы приложений на устройствах с мобильной операционной системой Apple iOS*

## 27.1.2 Изменения

- *В SDK изменился путь для настройки экспорта 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 14.11.
- *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](#).

## 27.1.3 Исправления

- *Исправлена ошибка рендеринга материала типа HALO.*
- *Исправлена ошибка рендеринга объекта при выставлении опции “Do not batch”, если он имел родительский объект.*
- *Исправлена ошибка назначения горячих клавиш на экспорт аддона Blend4 Web.* 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 запечатления при помощи инструмента “*B4W Vertex Anim Baker*”.

## 27.1.4 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.

# 27.2 v14.09

## 27.2.1 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 языка.*  
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.** 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 профилей высокого качества (включая тени, динамическое отражение и антиалиасинг) для iOS.**

## 27.2.2 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_uraniunm`...** 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.**

### 27.2.3 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 “Do not batch” для объектов, анимирующихся с использованием NLA.*
- *Исправлена ошибка для систем частич, в которых дублируемый объект входит в иерархию с другими объектами.*

### 27.2.4 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.
- **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

- аддон, 35
  - ошибки инициализации, 36
  - установка, 13
  - установка новой версии, 14
- анаглиф, 112
- анимация, 132
- береговая линия, 149
- браузер, 4
  - настройка, 16
- браузерные технологии, 4
- цвет фона, 102
- драйверы, 5
- движок, 1
- эффект Френеля, 61
- экспорт, 14, 25
  - ошибки, 37
  - предупреждения об ошибках экспорта, 41
- глубина резкости камеры (depth of field), 105
- графический движок, 1
- инстансинг, 123
- интерактивная графика, 5
- источники света, 92
- коррекция цвета, 113
- кватернион, 187
- материалы, 56
  - гало, 65
  - ноды, 78–80
  - параметры освещения, 57
  - прозрачность, 59
  - специальные параметры, 63
  - зеркальное отражение, 60
- нормали, 87
- редактор, 87
- обfuscатор шейдеров, 200
- директивы import/export, 201
- ограничения, 201
- ошибки, 203
- освещение, 85
- параметры берега, 149
- просмотрщик, 27
  - добавление сцен, 25
  - запуск, 20
- прозрачность, 59
  - настройка, 60
  - типы, 59
- размытие при движении (motion blur), 104
- сенсор, 190
  - множество, 190
- сглаживание, 114
- система частиц, 115
  - использование для инстансинга, 123
- стереоизображение, 112
- сумеречные лучи (crepuscular rays), 109
- свечение (glow), 110
- текстуры, 66
  - диффузная, 69
  - карта бликов, 69
  - карта нормалей, 70
  - карта окружения, 73
  - карта смешивания, 71
  - карта высот, 70
  - карта зеркального отражения, 75
  - настройки, 68
  - небо, 76
  - рендеринг в, 77
  - типы, 67
- трехмерный движок, 1
- веб-плеер, 31
- видео-карта, 5

- взаимное затенение (screen-space ambient occlusion), 106
- замечания к релизам, 222
- затенение, 86
- типы, 86
- зеркальное отражение, 60
- динамическое, 61
- эффект Френеля, 61
- статическое, 61
- 3D моделирование, 3
- B**
- Blend4Web, 1
- Blender, 3
- установка, 12, 21
- D**
- diffuse map, 69
- DOF, 105
- E**
- environment map, 73
- G**
- git, 205
- добавление и удаление файлов, 208
- индивидуальные настройки, 207
- коммит, 210
- подготовка к коммиту, 207
- проверка статуса, 207
- разрешение конфликтов, 212
- синхронизация между репозиториями, 210
- тэги, 215
- god rays, 109
- H**
- halo, 65
- height map, 70
- M**
- matcap, 68
- material capture, 68
- mirror map, 75
- N**
- normal map, 70
- P**
- parallax mapping, 70
- R**
- render-to-texture, 77
- RTT, 77
- S**
- sensor, 190
- manifold, 190
- skydome, 76
- specular map, 69
- SSAO, 106
- stencil map, 71
- W**
- WebGL, 2
- поддержка в браузерах, 2
- преимущества, 3
- расширения, 203