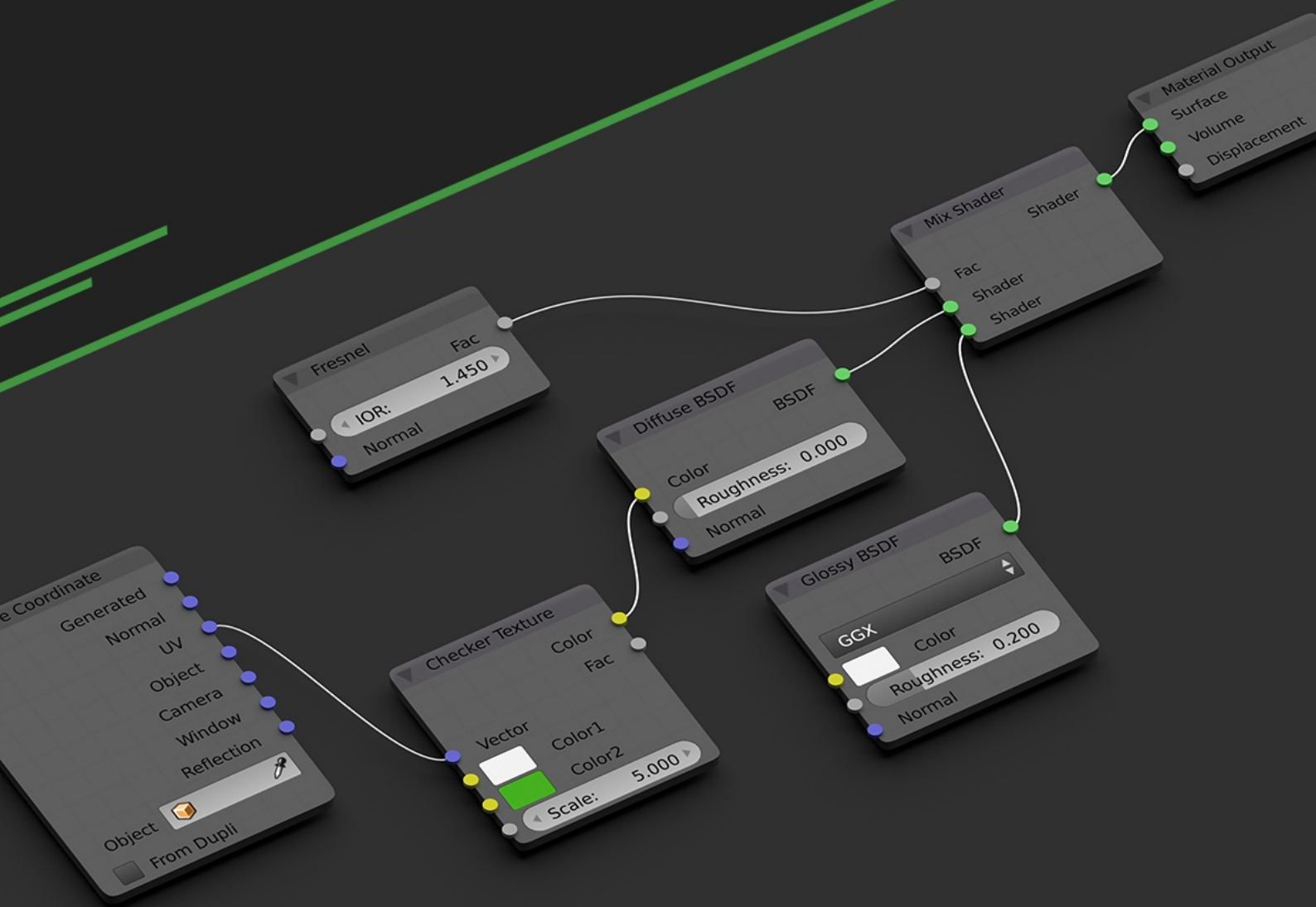




РУКОВОДСТВО ПО CYCLES

АРТЁМ СЛАКВА



Как работает трассировщик пути?

Для каждого пикселя изображения создается луч. Все лучи исходят из камеры и до столкновения с чем-либо являются лучами камеры (camera ray). Если луч попадет на источник света (emission) или шейдер holdout, то он будет удален из сцены, а пиксель получит цвет источника света или станет прозрачным, в случае с holdout. Во всех остальных случаях поведение луча зависит от типа шейдера, назначенного поверхностям, об которые он ударяется. Самым простым примером будет идеальное зеркало (sharp glossy). В данном случае луч будет отражаться согласно простейшему правилу, то есть под тем же углом, под которым и попал на поверхность. После этого луч уже продолжит свой путь как глянцевый луч (glossy ray). Допустим, далее он ударится о диффузную поверхность (diffuse). От данной поверхности луч отразится в случайном направлении. Если к этому моменту времени луч не достигнет максимального количества отскоков для него, то в конечном счете луч попадет на источник света. На этом его путь будет прекращен и пиксель получит цвет в зависимости от того, какой путь преодолел луч. Данный процесс повторяется такое количество раз, сколько сэмплов вы установите на вкладке рендера. В конце принимается среднее значение всех полученных цветов пикселя.

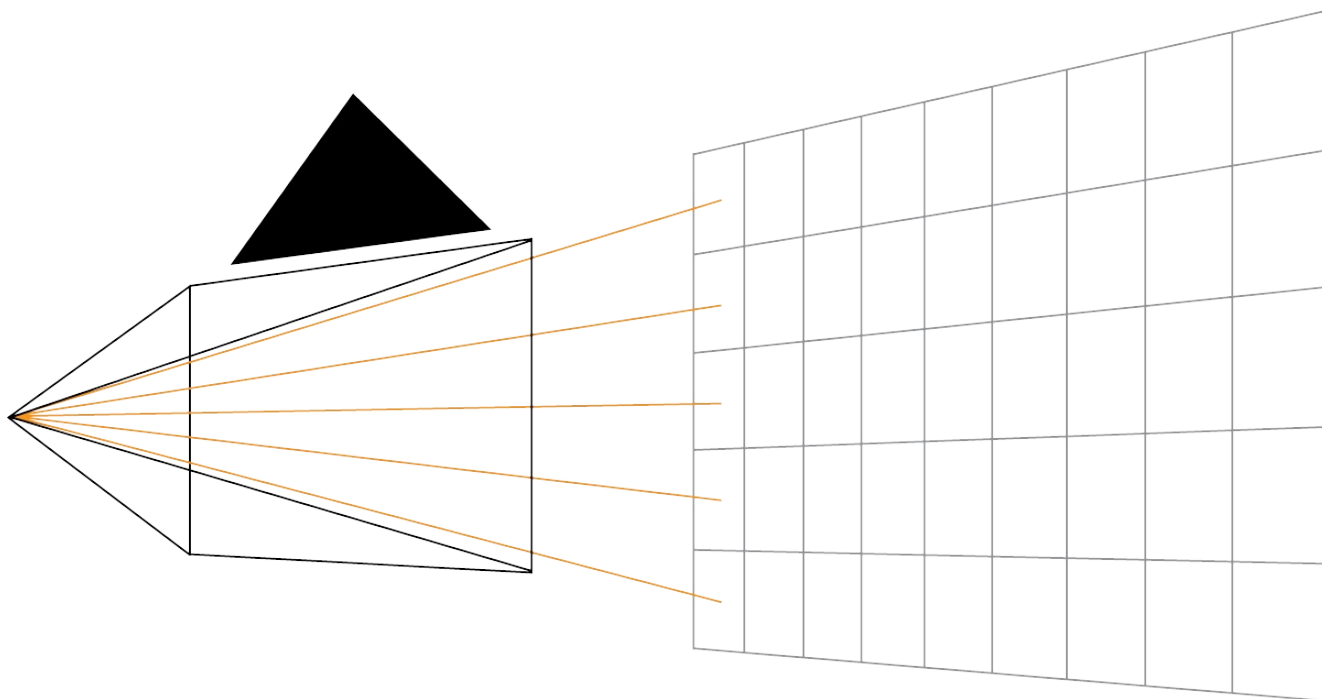


Рис 1.3) Количество клеток на сетке определяет количество пикселей изображения. Для каждой клетки будет выпущено столько лучей из камеры, сколько сэмплов вы установите на вкладке рендера.

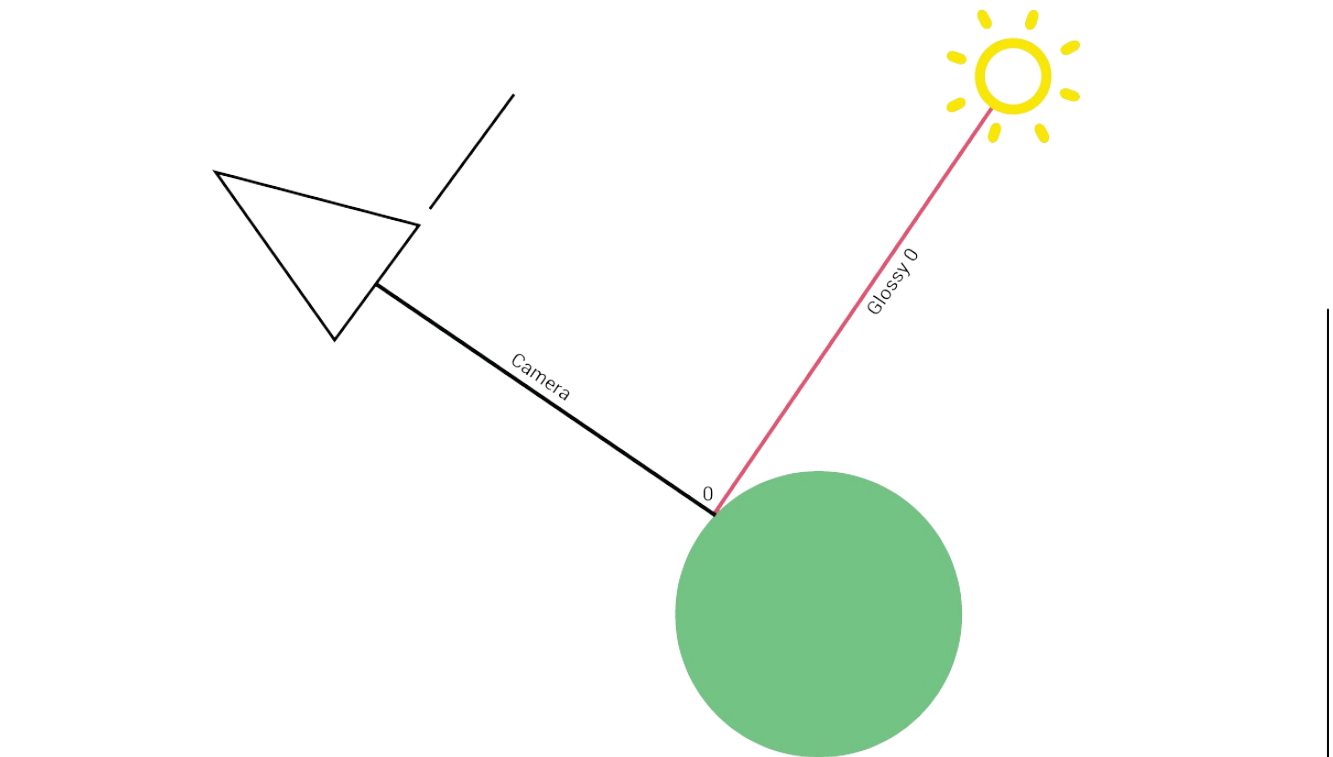


Рис 1.4) Луч, вышедший из камеры, попадает на глянцевую поверхность и отразившись от нее, попадает на источник света.

На рисунке 1.4 показан очень простой пример. Луч, вышедший из камеры, попадает на глянцевую поверхность и отражается на источник света. Путь луча после отскока окрашен в красный цвет и называется Glossy 0. Это означает, что луч поменял свой тип, с луча камеры (camera ray) на луч отражения (reflection). 0 означает количество отскоков, сделанных лучом (Cycles начинает считать с 0). После отскока от поверхности, луч изменяет свой тип, в зависимости от типа поверхности, на которую он попал. В данном случае для поверхности назначен шейдер Glossy. Но разные лучи, вышедшие из камеры, могут отскакивать в различных направлениях и попасть на пол, который на рисунке 1.5 имеет диффузную поверхность. В таком случае луч, отразившийся от пола, станет диффузным лучом (diffuse ray).

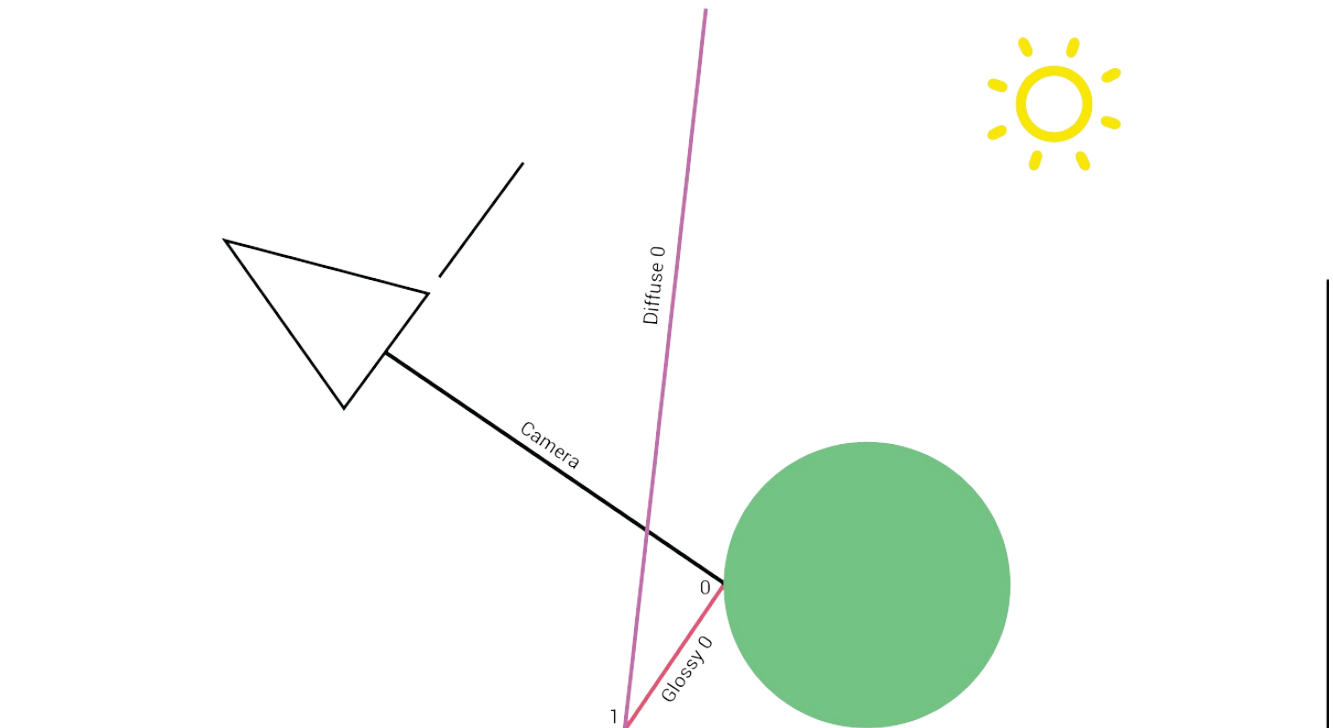


Рис 1.5) Луч, вышедший из камеры, попадает на глянцевую поверхность и отразившись от нее, попадает на диффузную поверхность.

В предыдущем примере объект имел лишь глянцевый шейдер. Но что будет, если объект имеет несколько шейдеров, что практически всегда и бывает с материалами? Чаще всего, это может быть смесь шейдеров diffuse и glossy. В данном случае Cycles будет случайным образом выбирать шейдер из материала. Если используется нод Mix, то шейдер с большим фактором смешивания будет иметь большую вероятность быть выбранным рендером. Давайте изменим глянцевую сферу на стеклянную и посмотрим, что произойдет:

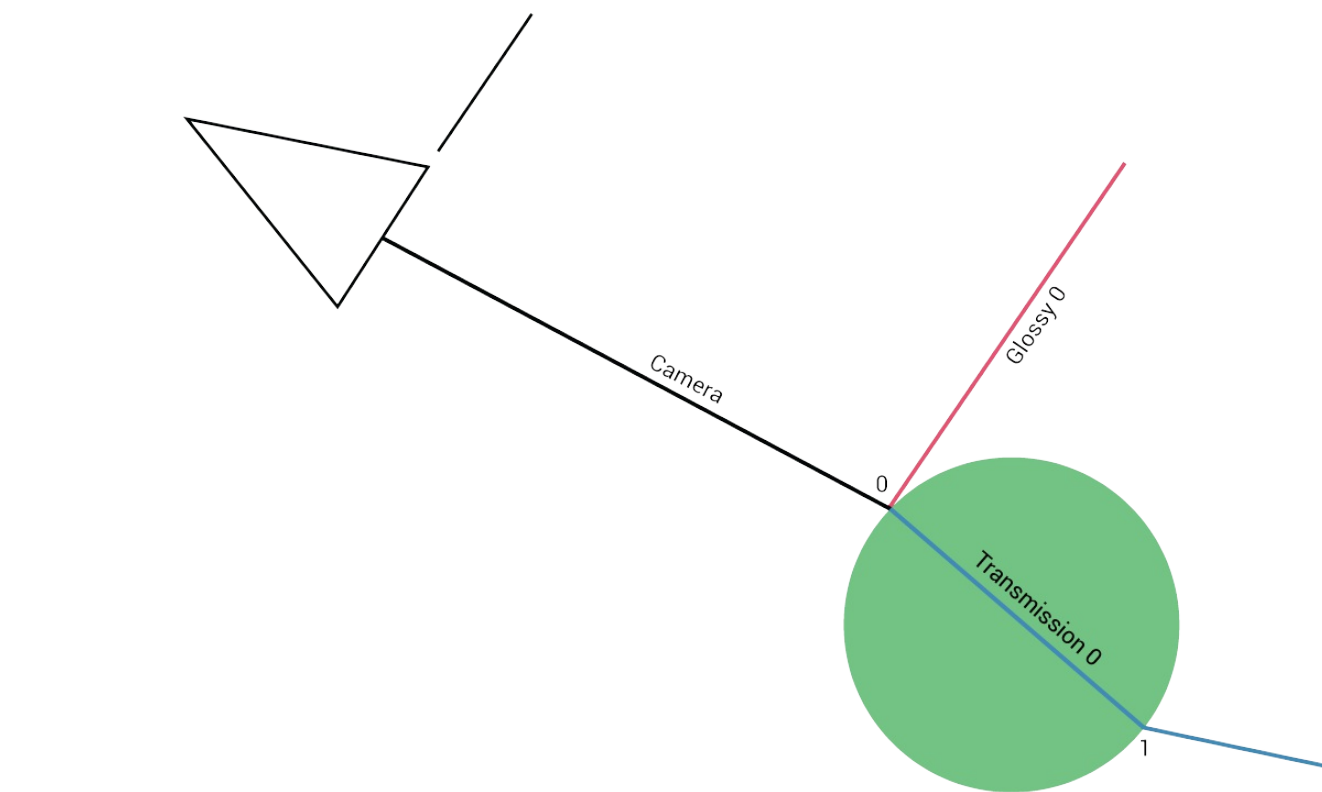


Рис 1.6) Луч, вышедший из камеры, попадает на стеклянную поверхность. В данном случае у луча есть два пути: отразиться от поверхности или пройти сквозь нее.

Как видите, у луча достаточно много вариантов пути даже в простой сцене. Попадая на стеклянную поверхность, луч решает по какому пути ему пойти дальше. Давайте разберем полностью оба маршрута луча:

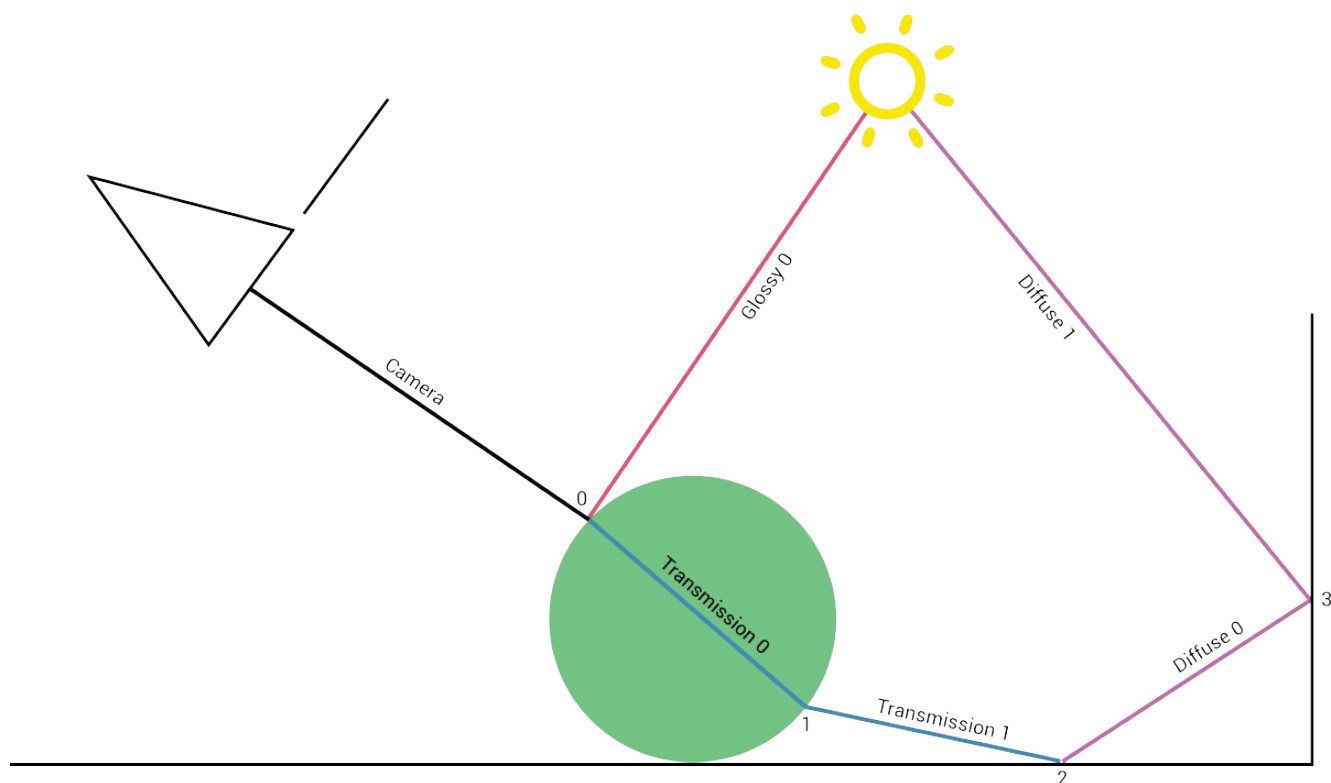


Рис 1.7) Луч, вышедший из камеры, попадает на стеклянную поверхность. В случае отражения от поверхности, он попадет на источник света и прекратит свое существование. В случае прохождения сквозь сферу, луч изменит свой тип на *Transmission* и выйдя из сферы, он все еще будет иметь тип *Transmission* до тех пор, пока не попадет на пол. После этого луч изменится на диффузный и в конце своего пути попадет на источник света, прекратив свое существование.

Решить отразиться ему от поверхности либо пройти сквозь нее, луч может на основе фактора Френеля. Первый пример мы уже разобрали (отражение луча от поверхности), теперь давайте рассмотрим путь сквозь объект. Попав внутрь и выйдя из объекта, луч будет иметь тип *Transmission* до тех пор, пока не попадет на пол. Так как пол имеет диффузный шейдер, луч соответственно станет диффузным, и продолжит отскакивать от поверхности до попадания на источник света. Весь путь данного луча состоит из 3-х отскоков, но он имеет лишь один отскок *transmission* и один *diffuse*. Если бы вы установили максимальное количество отскоков менее 3-х, то луч прекратил бы свое существование при ударе о стену, и не дошел бы до источника света.

Каждый луч, вышедший из камеры, возвращает значение интенсивности своего пикселя. Этот результат называют сэмплом (sample). Чтобы определить окончательный цвет пикселя, берется среднее значение всех сэмплов, полученных для него.

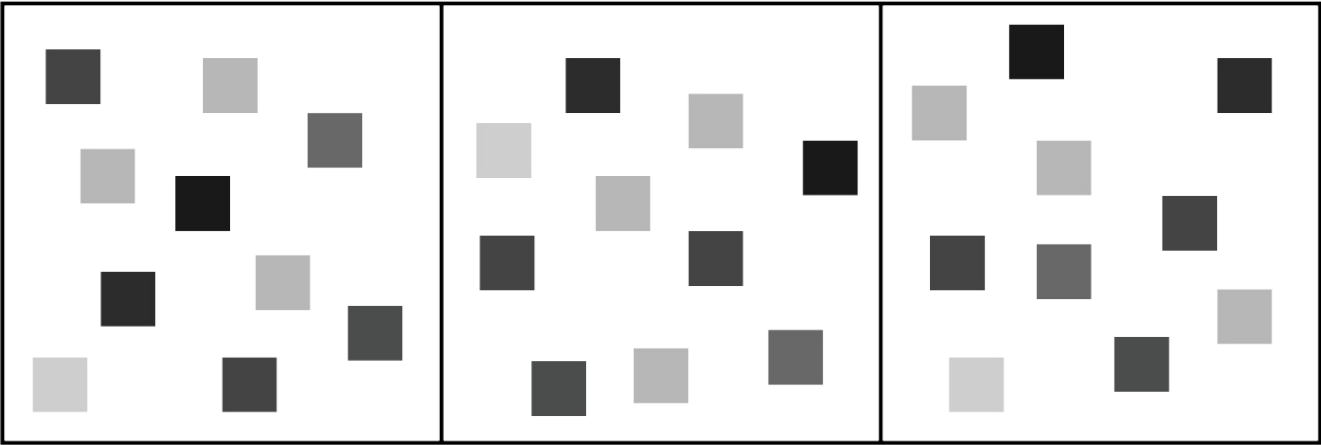


Рис 1.8) Для каждого из трех пикселей хранится 10 значений интенсивности. В результате пиксель получит цвет, основанный на среднем значении интенсивности всех 10-ти сэмплов.

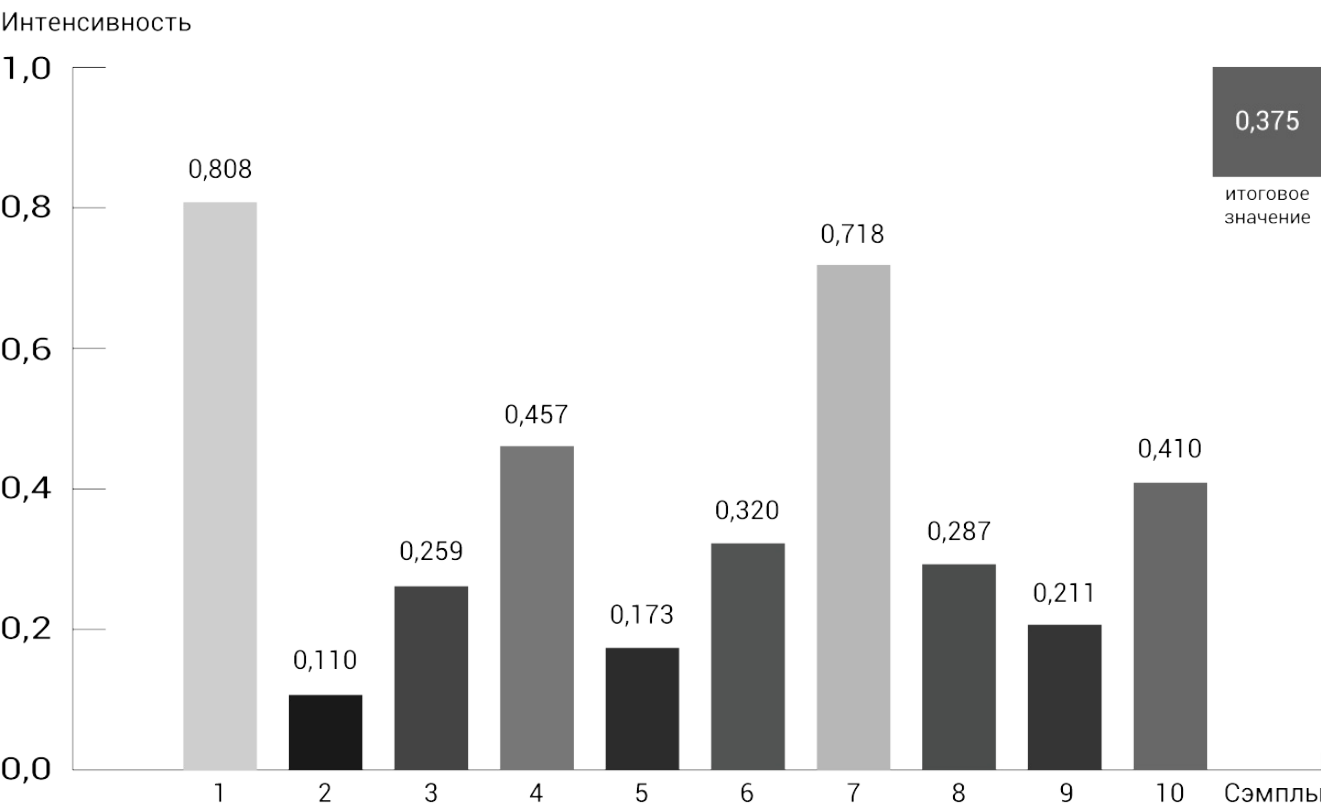


Рис 1.9) Определение итогового значения интенсивности пикселя при 10-ти сэмплах.

Определение пути луча при попадании на поверхность

На рисунках выше может показаться, что путь луча хорошо определен, но они иллюстрируют лишь один из возможных его путей. Помимо случайного выбора шейдера поверхности при попадании на нее, Cycles также случайным образом выбирает направление луча. Практически все шейдеры имеют компонент, который указывает Cycles возможные направления пути луча при попадании на поверхность. Исходя из этой информации, Cycles случайным образом выбирает направление луча в соответствии с заданным диапазоном. В этом диапазоне различные углы отскоков имеют различные вероятности. Так, например, при попадании на диффузную поверхность, вероятности всех углов одинаковы, в то время как для глянцевого шейдера вероятность отразиться под углом падения значительно выше, чем под любым другим.

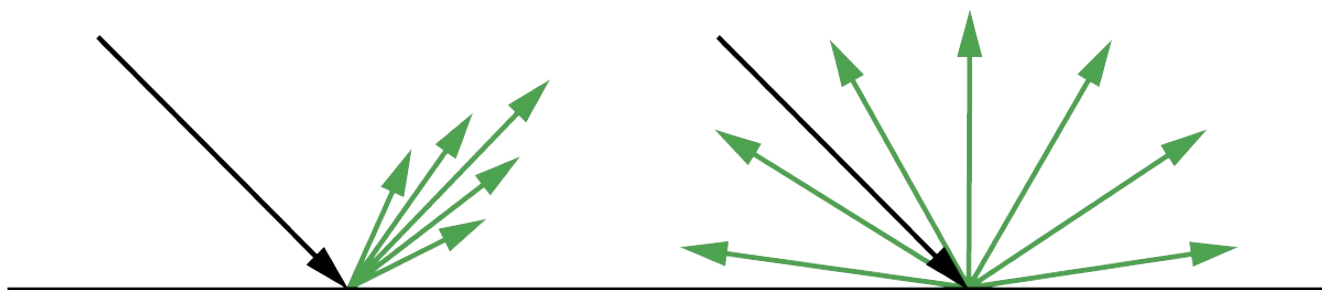


Рис 1.10) Слева — схематическое отображение вариантов отражения луча для глянцевой поверхности, справа — для диффузной. Длина зеленой стрелки соответствует вероятности отражения (чем длиннее, тем вероятнее).

С новой информацией в голове, давайте вернемся к старым примерам и немного их усложним. Мы сталкиваемся со следующей ситуацией: вероятность того, что луч, прошедший сквозь объект, попадет на источник света очень мала, потому как при каждом отскоке у него есть масса других

вариантов выбора пути:

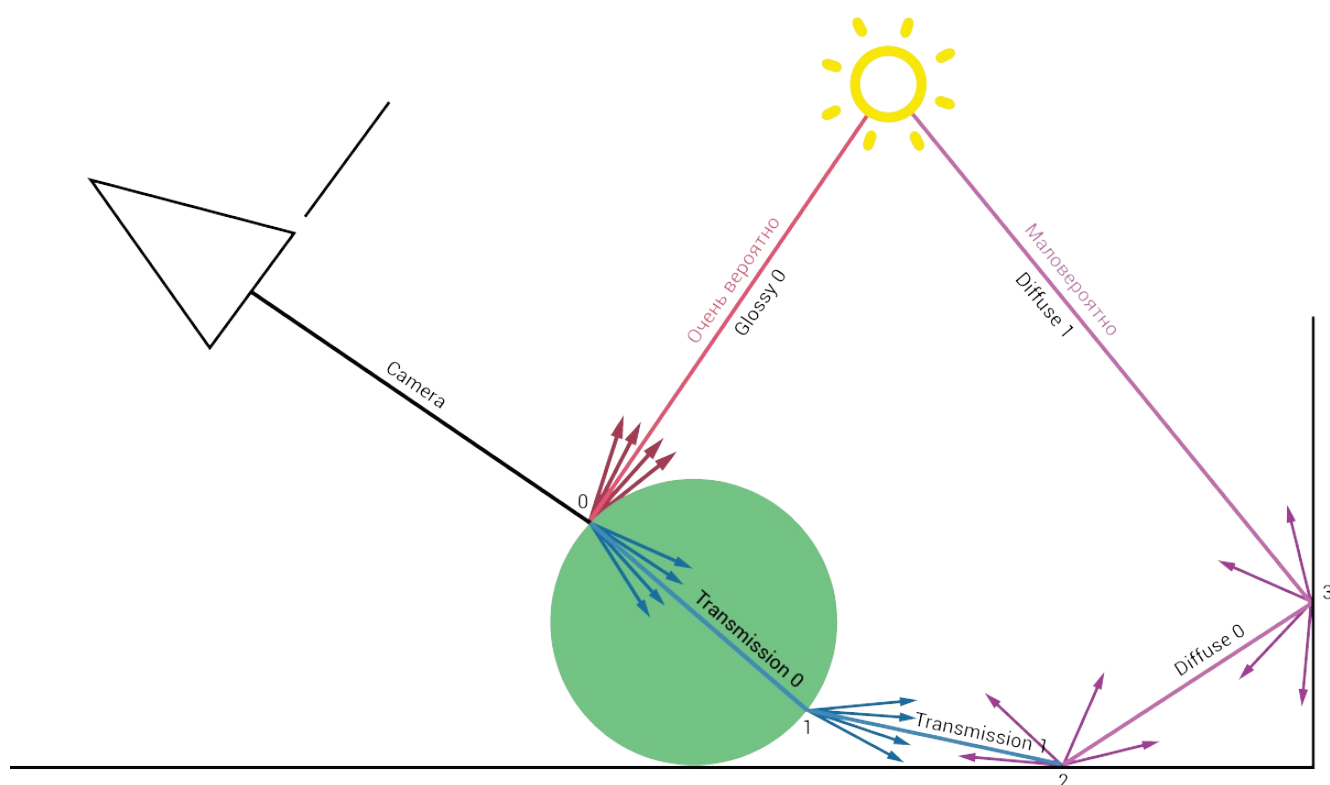


Рис 1.11) На рисунке показаны возможные варианты направления луча после каждого отскока. Как видите, вероятность того, что луч пройдет по заданному нами пути крайне мала.

Как видите, вероятность добраться до задней стенки у луча не большая, но исходя из рисунка, мы видим, что задняя часть сцены должна быть освещена. Так как же этого достичь? Использовать пятизначные значения количества сэмплов? К счастью, в Cycles есть умный способ сбора информации об освещении для каждого отскока, который называется Light Sampling.

Light Path

Лучи, путешествующие по сцене, имеют различные типы. Все они исходят из камеры и именно там берут свое начало. При попадании на поверхность, они разделяются на различные типы. Всего существует 4 типа лучей:

- **Camera ray** (лучи, исходящие из камеры)
- **Reflection ray** (лучи, отраженные от поверхности)
- **Transmission ray** (лучи, проходящие сквозь поверхность)
- **Shadow ray** (лучи, попадающие на источник света и создающие тени)

В свою очередь, Reflection ray и Transmission ray могут разделяться на:

- **Diffuse ray** (лучи, созданные диффузными поверхностями)
- **Glossy ray** (лучи, созданные глянцевыми поверхностями)
- **Singular ray** (лучи, созданные идеально гладкими поверхностями)



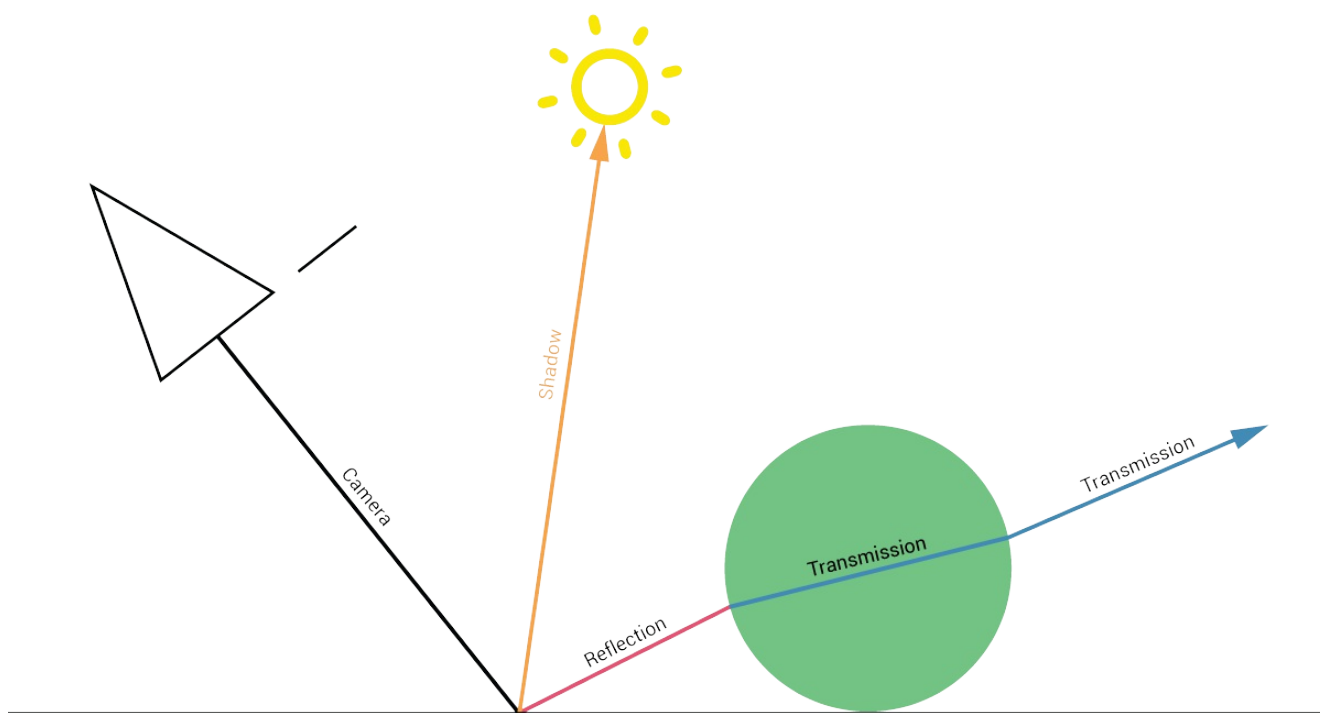


Рис 4.10) *Четыре основных типа луча.*

С помощью данного нода мы можем получить доступ ко всем типам лучей. Приставка "Is" означает, что на выходе вы получите 0 или 1 (черное или белое) и ничего более.

Is Camera Ray

Пожалуй, наиболее часто используемый выход данного нода. Он позволяет получить доступ к тем лучам, которые вышли из камеры и еще не попали на поверхность. С помощью данного выхода вы можете сделать объект светящимся (emission), но никак не влияющим на остальную сцену (то есть не освещающим ее). Или сохранить время рендеринга, сделав объект (bump) видимым лишь для камеры и не влияющим на остальные предметы. Сокращение времени происходит за счет того, что движку не приходится вычислять огромное количество отскоков света от неровной поверхности.

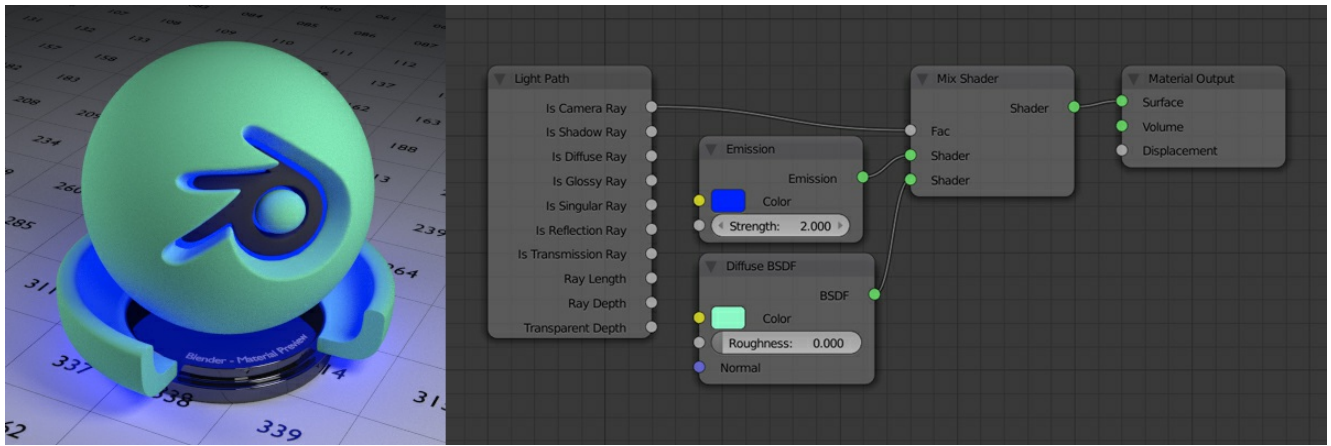


Рис 4.11) При первом попадании на объект, выбирается шейдер Diffuse (1), для всех остальных типов лучей используется шейдер Emission (0).

Is Shadow Ray

Лучи, которые попали бы на лампу, но не могут этого сделать, так как на их пути есть непрозрачный объект. Стоит отметить, что это работает непосредственно для ламп, а не для поверхностей с шейдером Emission до тех пор, пока не будет включена опция MIS (включена по умолчанию). С помощью этого выхода вы можете изменить цвет тени объекта.

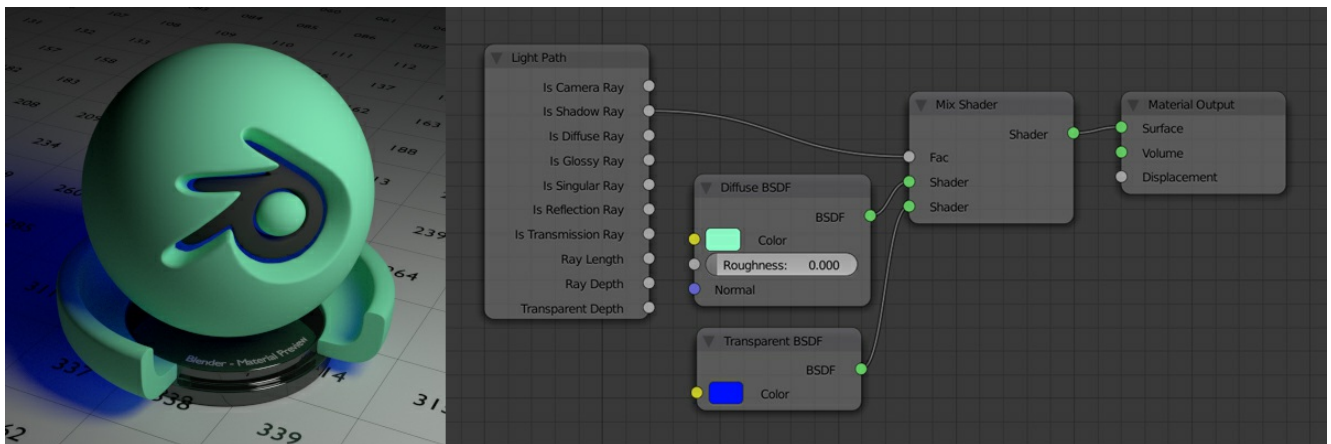


Рис 4.12) С помощью данной связки узлов вы можете изменить цвет тени. Обратите внимание на то, что тень окрашивается в синий цвет лишь в нижней части объекта. Вверху сфера не синяя, потому что на нее не попадет свет, а не потому что на нее падает тень. В данном случае тень падает на пол.

Is Diffuse Ray

Лучи, которые попали на диффузную поверхность. Также используется в качестве оптимизации, делая глянцевые поверхности диффузными для

окружающих объектов, чтобы избежать каустики.

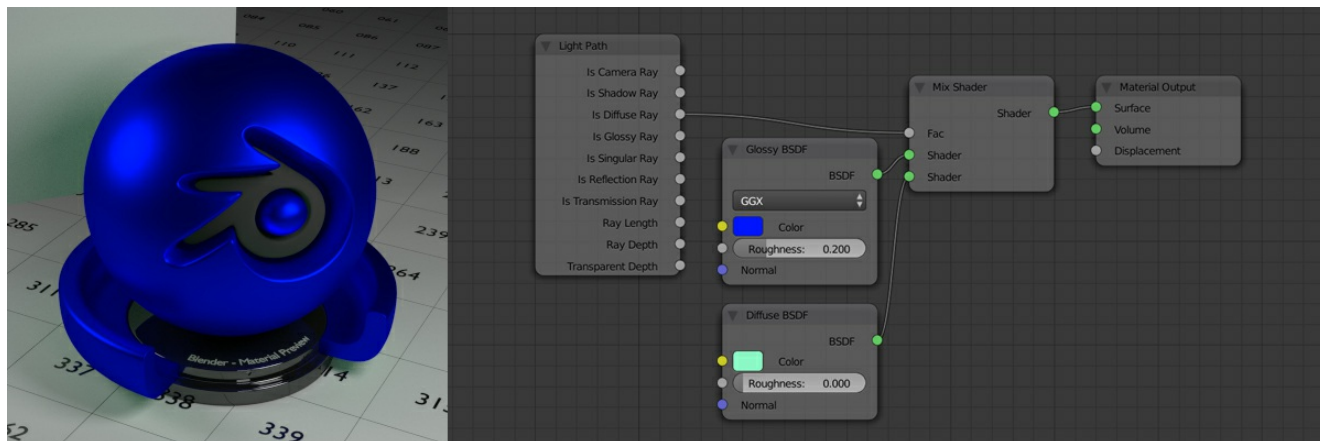


Рис 4.13) Позади сферы расположена плоскость (шейдер Diffuse). Присмотревшись, вы можете увидеть, что она немного окрашена в зеленый цвет. С точки зрения физики, подобное должно было произойти, если бы сфера была зеленой, а не синей, но с помощью данного нода мы можем это изменить.

Is Glossy Ray

Лучи, которые попали на глянцевую поверхность. С помощью данного выхода можно изменить внешний вид объекта в отражениях.

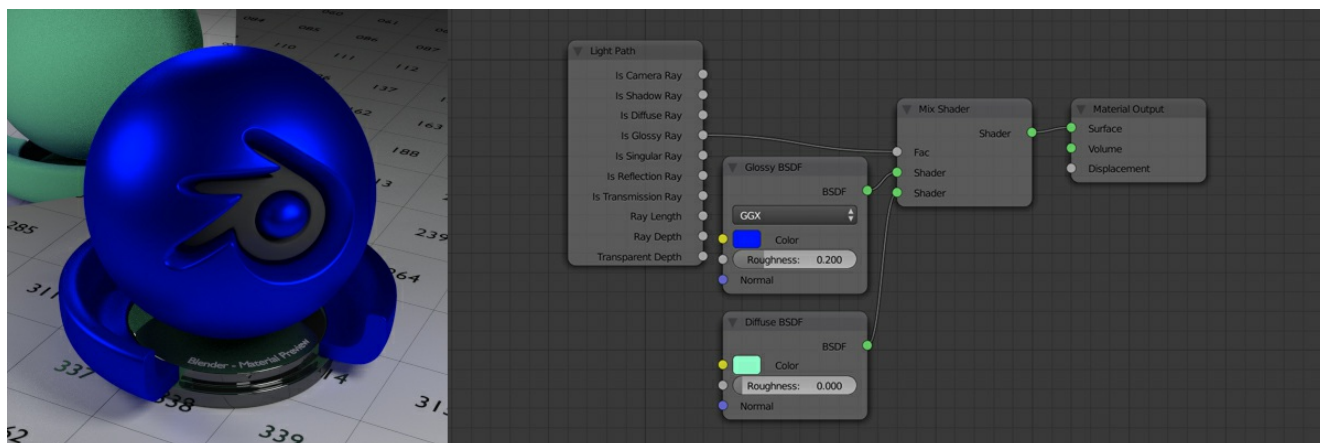


Рис 4.14) Позади сферы расположена плоскость (шейдер Glossy). С помощью данного типа лучей мы можем менять вид объектов в отражениях.

Is Singular Ray

Это те лучи, которые имеют лишь один возможный путь. Все лучи, исходящие из камеры являются Singular до тех пор, пока не попадут на

поверхность, за исключением, если тип поверхности не использует метод Sharp. В таком случае луч и дальше будет Singular. Данный тип вы можете установить для шейдеров Glossy, Glass и Refraction. При использовании других методов (Beckmann, GGX или Ashikmin-Shirley) с уровнем шероховатости равным 0, луч не будет Singular. Только при использовании типа Sharp.

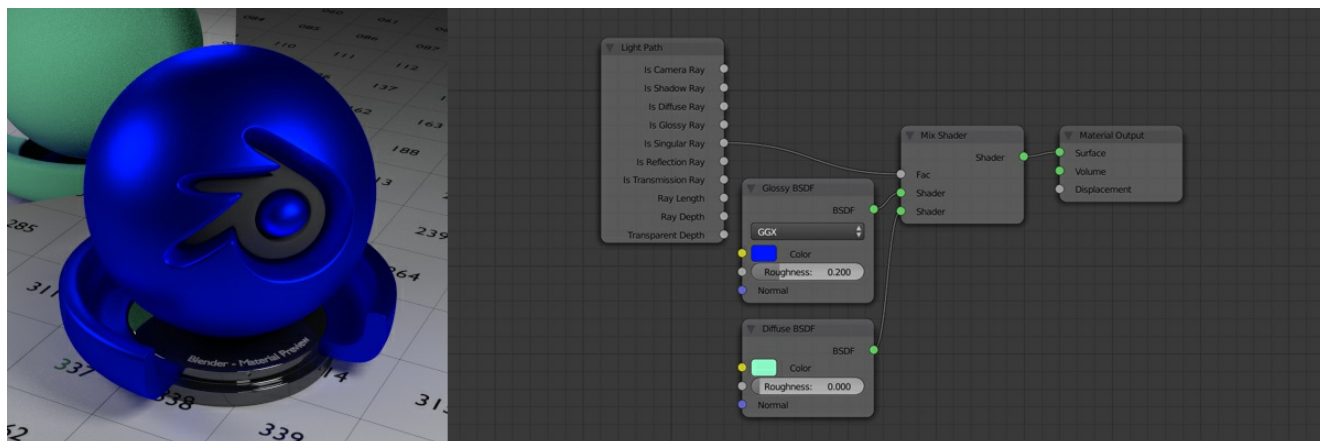


Рис 4.15) Позади сферы расположена плоскость (шейдер Glossy (тип Sharp)). С помощью данного типа лучей мы можем менять вид объектов на поверхностях с типом Sharp.

Is Reflection Ray

Практически все поверхности отражают свет. Попадая хоть на диффузную, хоть на глянцевую поверхность, луч отражается и становится Reflection. Таким образом, мы можем получить доступ ко всем отраженным лучам.

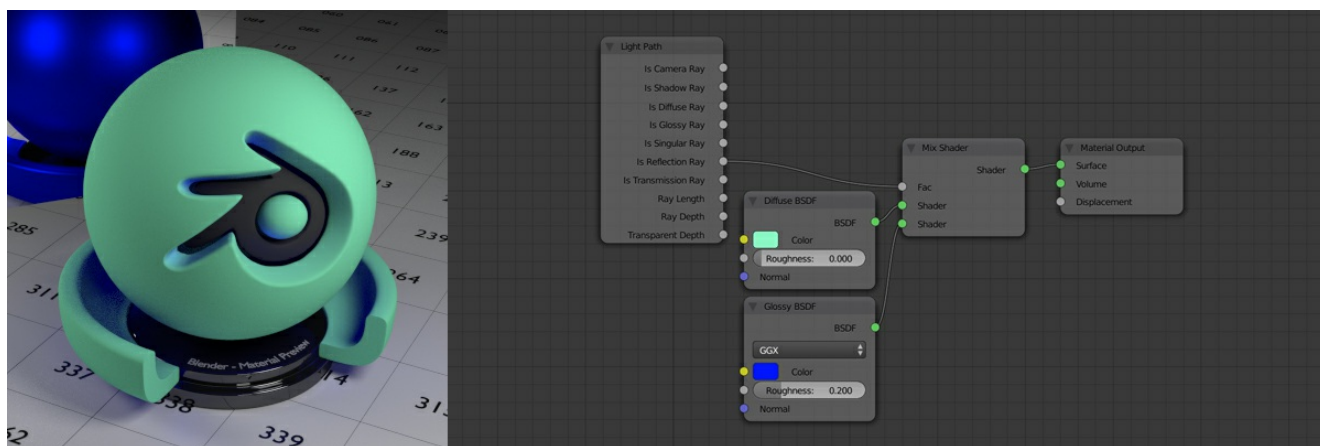


Рис 4.16) Позади сферы расположена плоскость (шейдер Glossy). С помощью лучей отражения мы изменили материал не только в зеркале, но и во всех остальных участках диффузной сферы.

Is Transmission Ray

Когда лучи проходят сквозь объект, они становятся Transmission. Это могут быть такие шейдеры, как: Glass, Refraction или Translucent. Таким образом, с помощью данного выхода мы можем получить доступ к лучам, прошедшим сквозь объект. Стоит отметить, что при прохождении через шейдер Transparent, луч не становится Transmission. Данный шейдер производит лучи с типом Shadow.

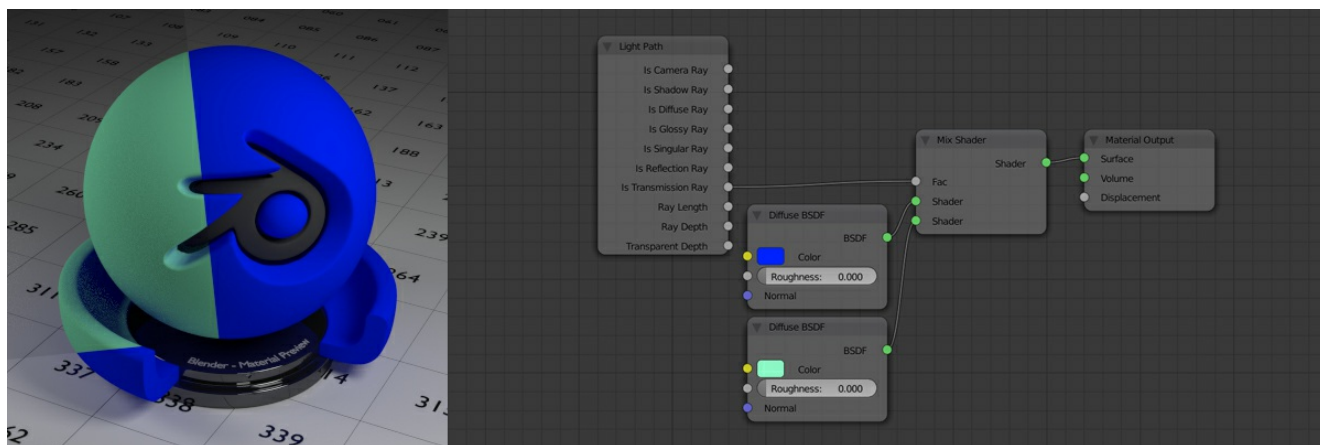


Рис 4.17) Перед сферой расположена плоскость (шейдер Glass с IOR равным 1). Все лучи, изменившие свой тип на Transmission (прошедшие сквозь плоскость), при попадании на сферу окрашивают ее в зеленый цвет.

Ray Length

После того, как луч попал на прозрачный объект, его длина может быть измерена. С использованием данного выхода вы можете указать, как должен вести себя луч на определенной глубине. Например, вы можете создать эффект подповерхностного рассеивания, делая тонкие части объекта более светлыми. Работает со следующими шейдерами: Translucent, Glass и Refraction.

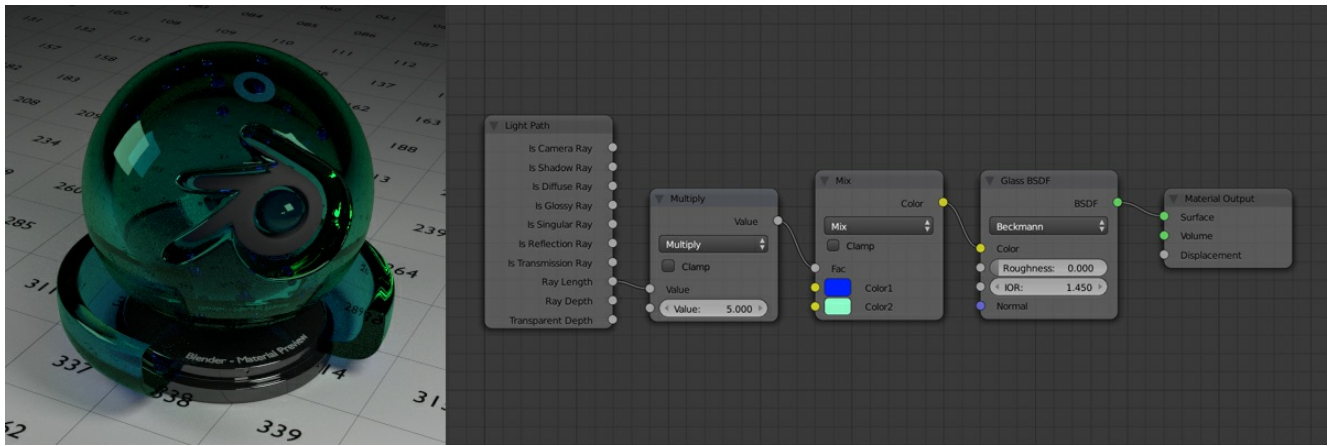


Рис 4.18) На тонких участках сферы преобладает синий цвет.

Ray Depth

Данный параметр отвечает за количество отскоков луча света. Изначально все лучи, исходящие из камеры, имеют число отскоков равное 0. Далее, с каждым отскоком данное значение увеличивается на единицу. Таким образом, вы можете изменить поведение луча при определенном количестве отскоков. Например, вместо того, чтобы увеличивать до бесконечности количество отскоков света, вы можете сделать объекты в вашей сцене прозрачными после двух отскоков. Это не только сэкономит время рендеринга, но и поможет избавиться от черных пятен на стекле.

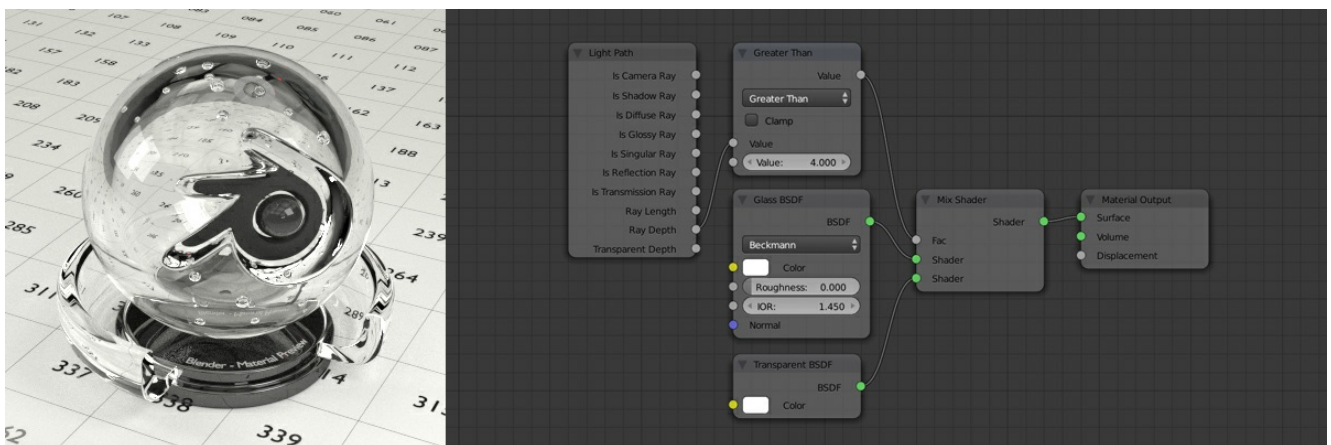


Рис 4.19) После 4-х отскоков луча сфера становится прозрачной.

Transparent Depth

Данный параметр отвечает за количество прозрачных поверхностей, через которые прошел луч. Как и в предыдущем случае, изначально он равен 0. Работает он лишь в качестве фактора смешивания для нода Mix с шейдером Transparent. Таким образом, вы можете заменить материал на другой после

определенного количества прохождений через прозрачный объект. Это очень полезно при создании растительности. Вы не можете видеть листок дерева, если перед ним находятся еще 2-3 таких же. В таком случае нет смысла рендерить его с прозрачностью, и можно просто сделать ваш материал непрозрачным, сохранив тем самым память без видимых визуальных потерь.

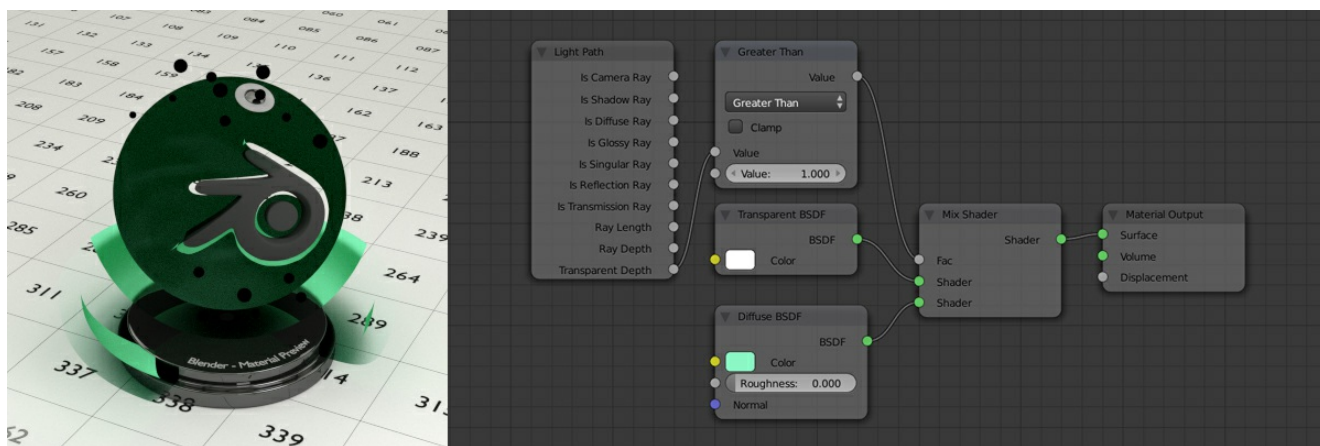


Рис 4.20) До первого прохождения луча сквозь поверхность сфера является прозрачной. Далее используется диффузный шейдер.