



Blender - Lektion T3  
EINSTIEG IN CYCLES

Blender V2.70 - Skript V2.1

Autor: Uwe Gleiß, Franz-Ludwig-Gymnasium Bamberg, Computergrafikgruppe (CoGra-FLG) • Kontakt über: [cogra-flg@web.de](mailto:cogra-flg@web.de)  
Dieses Werk steht unter einer Creative Commons Lizenz (Details durch Klick auf diesen Text).

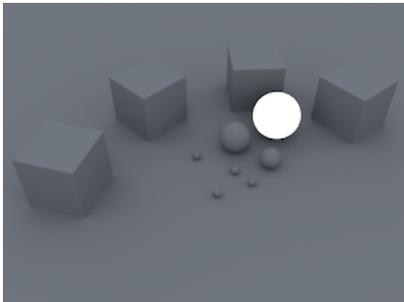


## WARUM CYCLES

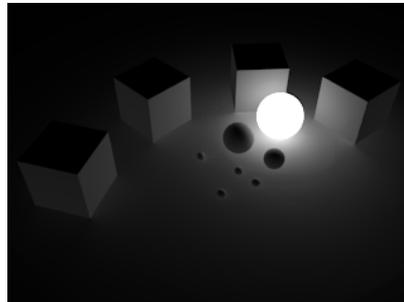
### Nachteile von klassischem Raytracing

Will man reale Beleuchtung nachempfinden, dann müsste man für die Farbe und Helligkeit eines Punktes die Farbe und Helligkeit aller anderen Punkte kennen. Denn von ihnen allen fällt Licht auf den gerade betrachteten Punkt. Da das leider für jeden Punkt gilt führt das zu einer horrenden Menge von Gleichungen. Wem es noch nicht genug gruselt: Nehmen wir eine simple Szene an, in der die Helligkeit von 1000 Oberflächenpunkten (auch solchen außerhalb des Bildes) bestimmt werden muss. Mathematisch ist dafür ein Gleichungssystem mit 1000 Gleichungen und 1000 Unbekannten zu lösen - das könnte ein klein wenig dauern (und 1000 ist wirklich niedrig angesetzt).

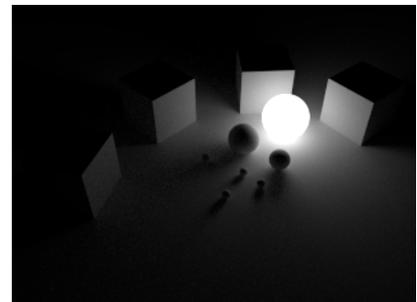
Ein kleines Beispiel soll das illustrieren: Eine leuchtende Kugel (Emit größer Null) in einer mit Blender Render berechneten Szene ist selbst zwar hell, beleuchtet aber nichts um sich herum wie im Bild unten links. Ohne das Umgebungslicht in diesem Bild wäre die Szene schwarz mit der Kugel als weißem Fleck. Mit Indirect Lighting (mittleres Bild) beleuchtet die Kugel zwar auch im Alleingang, aber von realistischen Schatten keine Spur. Erst Cycles (rechtes Bild) bekommt das alles hin (für den Preis höherer Rechenzeiten).



Blender Render - Leuchte ohne Beleuchtung



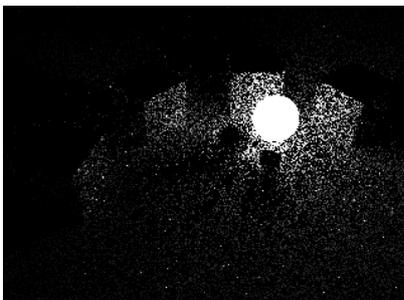
Indirect Lighting, keine realistischen Schatten



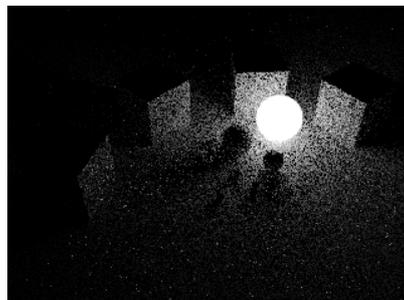
Cycles

### Wie funktioniert Cycles?

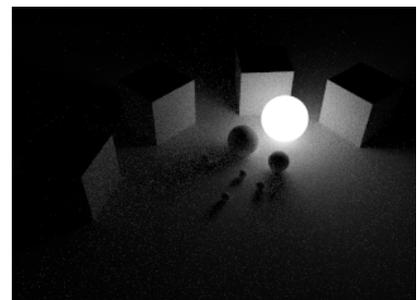
Jedes Bildberechnungsverfahren versucht sich irgendwie darum herum zu mogeln, das angesprochene Gleichungssystem (für Liebhaber von Theorie: die diskrete Form der Rendergleichung) komplett zu lösen. Man nähert sich der Lösung stattdessen mit verschiedenen Tricks an. Cycles ist da keine Ausnahme: Die Beleuchtung wird in mehreren Zyklen (der Name ist Programm) immer weiter verfeinert. Die Helligkeit im ersten Zyklus entsteht durch direkte Beleuchtung und wird nur für einen Teil der Oberflächenpunkte durchgeführt. Diese Helligkeitsverteilung wird gespeichert und im nächsten Zyklus als Beleuchtung verwendet, wobei neue Oberflächenpunkte ausgewählt werden. So bleibt der Aufwand pro Schritt im Rahmen und mit jedem Zyklus nähert sich die Beleuchtung realistischen Verhältnissen weiter an, erreicht diese aber nie ganz. Es bleibt ein Restrauschen, das manchmal stört, manchmal auch erwünscht ist (das Bild wirkt nicht so klinisch sauber wie bei anderen Berechnungsmethoden).



Nach einem Zyklus



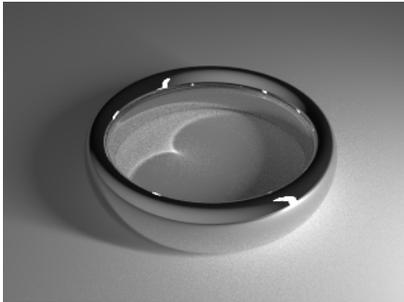
Nach fünf Zyklen



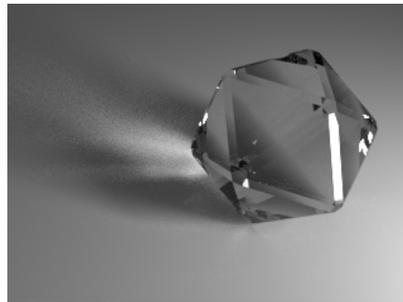
Nach 100 Zyklen

## Weitere Vorteile

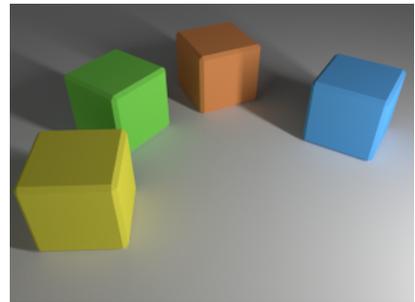
Neben der Vereinfachung der Beleuchtung (das mühsame Nachahmen realer Lichtverhältnisse durch diverse zusätzliche Lichtquellen - siehe Lektion R1) bietet eine solche Methode noch mehr, was mit einem klassischen Raytracing-Verfahren nicht denkbar ist. Wenn Licht durch spiegelnde oder durchsichtige Materialien gebündelt wird, entstehen Lichtfiguren (sog. Kaustiken). Auch das scharfe Bild bei Abbildung durch eine Linse ist nichts anderes als eine Kaustik. Ein Raytracer produziert hier nur verwaschene Flecken oder gar keinen solchen Effekt.



Spiegelungskauistik (2000 Zyklen)



Kauistik hinter Glas (5000 Zyklen)



Die Farbe der Würfel tönt den nahen Boden

Einen viel häufigeren Effekt zeigt das dritte Bild. Farbige Objekte strahlen farbiges Licht in ihre Umgebung ab. Decken Sie mal den orangen Würfel ab und betrachten Sie den Boden davor.

## Nachteile

Die Nachteile werden schon an den obigen Beispielen erkennbar: Die Rechenzeit für ein Bild mit ausreichend niedrigem Rauschen geht enorm in die Höhe. Hier kommt man schnell an die Grenze des Erträglichen, vor allem wenn Lichtbrechung, Spiegelung und damit verbundene Kaustiken realistisch dargestellt werden sollen.

# EINSTIEG

## Cycles aktivieren

Normalerweise ist Cycles bereits aktiviert, aber es könnte einmal vorkommen, dass das zugehörige Plugin abgeschaltet ist. Dies kann man in den Einstellungen (**Ctrl+Alt+U**) bei den Addons anpassen. Cycles ist in der Rubrik Render zu finden. Ebenfalls in den Einstellungen unter System kann man links unten bei Compute Device eventuell etwas anderes als CPU (Hauptprozessor des Computers) wählen. Manche moderne Grafikkarte können Cycles deutlich beschleunigen. Die Antwort auf die Frage, was hier mehr bringt, findet man nur durch Ausprobieren.

Für Blender Render erstellte Materialien ergeben keinen Sinn für Cycles. Man sollte früh entscheiden, welchen Renderer man nutzen will.

Damit Cycles auch wirklich für die Bildberechnung genutzt wird, muss am oberen Rand der Blenderoberfläche die Renderengine von Blender Render auf Cycles umgestellt werden . Das hat unter anderem zur Folge, dass die Einstellungen für Materialien sich verändern.

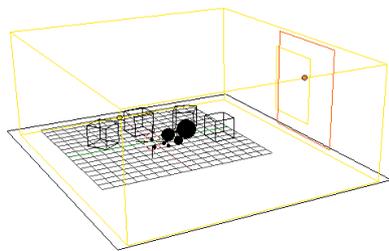
## Eine einfache Szene

Cycles kann mit den üblichen Lichtquellen von Blender arbeiten, aber auch hier verändern sich die Einstellungen. Zumindest für leuchtende Flächen wie Leuchtstoffröhren, Monitore oder auch ein Fenster empfiehlt es sich als Ersatz für Flächenlichtquellen ein leuchtendes Objekt zu verwenden.

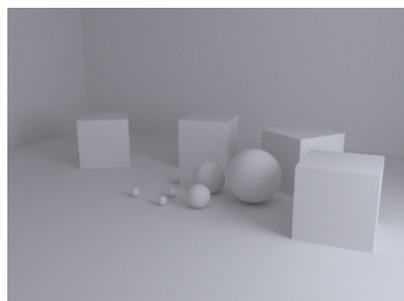
Dazu platziert man ein Objekt passender Form und verleiht diesem dann ein neues Material (so wie gewohnt, siehe Lektion T1). In diesem muss unter Surface von Diffuse BSDF auf Emission gewechselt werden. Bei der Helligkeit (Strength) ist zu beachten, dass es sich um die Helligkeit pro Fläche handelt. Vergrößert man das Lichtobjekt, dann nimmt die Helligkeit in der Szene zu.

In Cycles hängt die Lichtmenge eines leuchtenden Objekts auch von dessen Größe ab.

Wie auch in den Lektionen R1 und R2 empfiehlt sich die Erstellung einer kleinen Szene mit mehreren Objekten, die man zunächst alle mit dem gleichen Basismaterial versehen kann. So man die Szene zu den genannten Lektionen noch griffbereit hat, kann diese problemlos umgebaut werden (vorhandene Lichtquellen sollte man löschen). Um die Auswirkung der Beleuchtung genauer beurteilen zu können empfiehlt es sich, den Himmel zunächst schwarz einzufärben, denn seine Farbe hellt die Szene ebenfalls auf. Alternativ oder zusätzlich kann die Szene in einen simplen Raum eingebaut werden mit einem Fenster und dahinter der Lichtquelle.



Raum (gelb) und Lichtobjekt (orange)



Gerendert (Lichtfläche Strength 8, 1,5m x 1,5m)

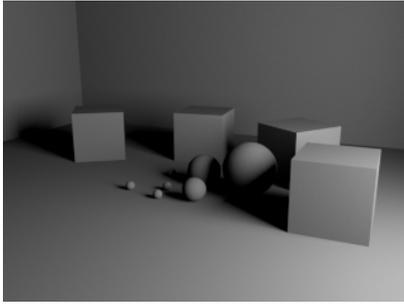
Die Zentrale Steuerung von Cycles ist in den Rendereinstellungen  zu finden. Damit beim Rendern mit **F12** ein halbwegs gutes Ergebnis erzielt wird muss unter Sampling - Samples bei Render die Anzahl auf einige Hundert hochgesetzt werden. Sonst bricht die Berechnung viel zu früh ab.

An gleicher Stelle kann mit Preview eingestellt werden, wie viele Zyklen für die Echtzeitberechnung im Editor verwendet werden sollen. Wie auch Blender Render kann Cycles direkt in einem 3D Bereich ein berechnetes Bild anzeigen, wenn der 3D Bereich von der Darstellung Solid  auf Rendered  umgestellt wird. Da das sehr rechenintensiv ist empfiehlt sich eine Beschränkung des berechneten Bereichs auf ein Rechteck mittels **Ctrl+B**.

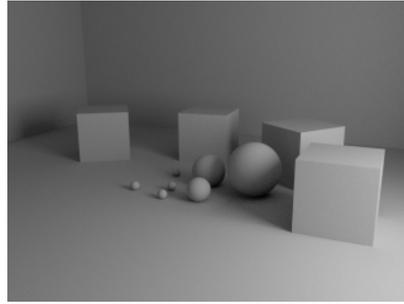
## LICHTSPIELE

### Berechnungsmethoden

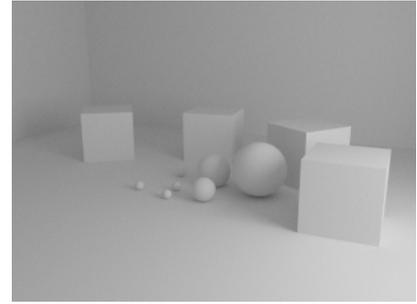
In den Rendereinstellungen unter Light Paths kann zwischen verschiedenen Voreinstellungen (Presets) gewählt werden. Direct Light berechnet das Licht fast genauso wie ein klassischer Raytracer. Die oben vorgestellten Effekte fehlen ganz. Limited Global Illumination verzichtet auf Kaustiken, ist relativ flott und liefert für viele Szenen ein vernünftiges Ergebnis. Full Global Illumination dürfte auch anspruchsvollen Bedürfnissen gewachsen sein, es empfiehlt sich aber, die Einstellungen den eigenen Bedürfnissen anzupassen (eigene Voreinstellungen können mit + als Presets in der Blenderdatei gespeichert werden).



Direct Light (Zeit: 0:31)



Limited Global Illumination (Zeit: 0:52)

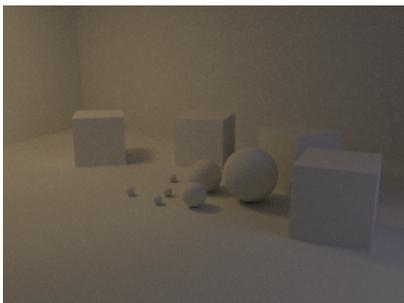


Full Global Illumination (Zeit: 2:15)

Was in den obigen Bildern auffällt ist die zunehmende Helligkeit bei gleicher Lichtquelle. Die drei Methoden lassen mehr und mehr Reflexionen zu. Das sorgt dafür, dass die gegenseitige Beleuchtung der einzelnen Oberflächenpunkte zunimmt. Das Bild wirkt vielleicht realistischer, aber der Kontrast leidet. In Cycles bedeuten höhere Einstellungen nicht unbedingt ein „besseres“ Ergebnis. Für das Blitzbild unten rechts wurde die Anzahl der diffusen Reflexionen (Bounces) bewusst auf 1 gesetzt. Auch für die anderen Bilder in diesem Skript werden die Einstellungen immer wieder angepasst.

## Variationen in Licht

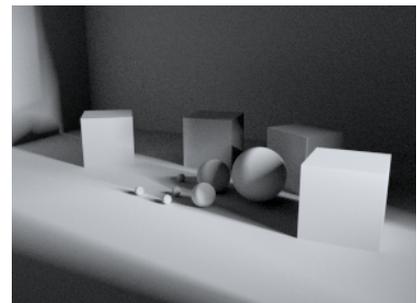
Allein durch die Verwendung verschiedene Lichtquellen lässt sich die kleine Szene grundlegend verändern.



Mondlicht von draußen, Deckenlicht rechts



ein lauschiger Teppichbrand



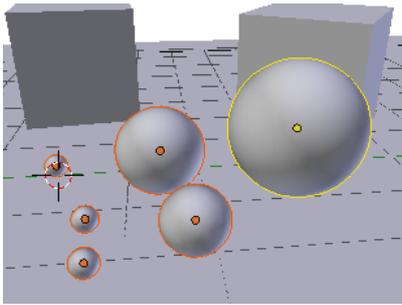
Mami ich hab Angst bei Gewitter

## MATERIALIEN

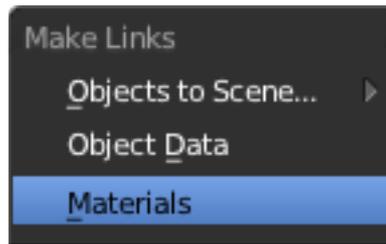
### Ein neues Material

Der Hauptteil der Szene behält zunächst sein Basismaterial, nur einige Auserwählte werden neu eingekleidet. Einem Objekt verhilft man zu einem neuen Material durch Klick auf + neben der Materialanzeige  CMatt 15 F  (die Nummer daneben gibt an, wie viele Objekte das Material im Moment verwenden). Wenn mehrere Objekte das neue Material erhalten sollen, dann wählt man all diese Objekte aus und sorgt dafür, dass ein Objekt, das das Material bereits besitzt aktives Objekt ist. **Ctrl+L** ruft ein Menü auf, mit dessen Hilfe Daten vom aktiven Objekt auf alle anderen selektierten Objekte übertragen werden können (in diesem Fall das Material).

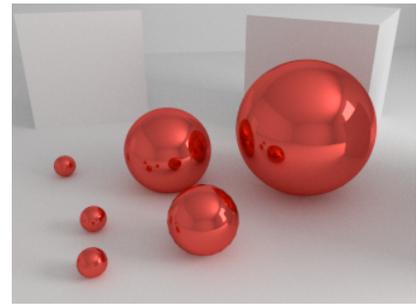
Verschiedene Grundmodelle für das Verhalten von Licht an Oberflächen stehen als sog. Shader zur Verfügung. Schalten Sie doch mal den Shader (neben Surface) im neuen Material von Diffuse auf Glossy und verändern Sie auch noch die Farbe. Erhöht man den Wert Roughness ein wenig, dann erhält man matte Spiegelungen. Auf der nachfolgenden Seite sehen Sie, wie das Ergebnis mit roter Spiegelung ohne Roughness aussieht.



Die große Kugel besitzt das neue Material



Übertragung der Materialdaten

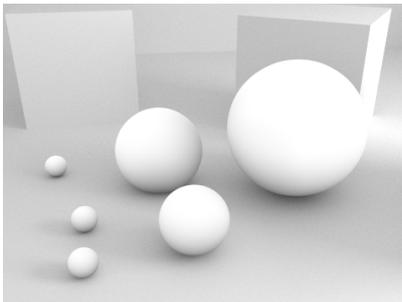


Ja ist denn schon Weihnachten?

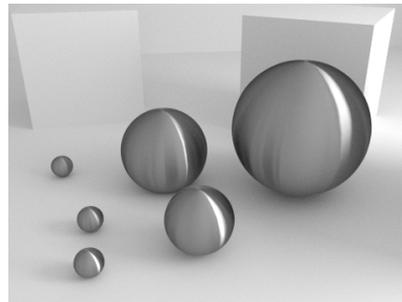
## Shadertypen

Nachfolgend werden die verschiedenen Shader von Cycles kurz vorgestellt. An dieser Stelle ist Spieltrieb von Vorteil. Probieren Sie alles mal aus, drehen Sie an Reglern und sehen Sie zu, was passiert. Die nachfolgende Liste ist alphabetisch sortiert, nicht wie im Menü von Blender. Ausgenommen sind die Shader Add und Mix, die mehrere Effekte kombinieren können und daher erst am Ende der Liste stehen.

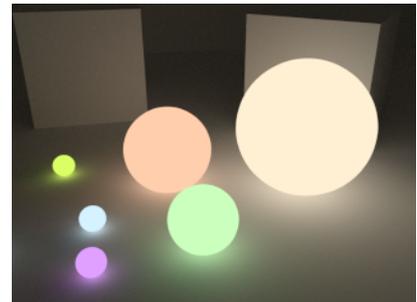
- **Ambient Occlusion** (siehe Lektion R2) sorgt für Schatten überall dort, wo andere Oberflächen in der Nähe sind und ist damit z.B. geeignet, um Kerben oder Löcher deutlicher abzusetzen bzw. um Dreckablagerungen zu simulieren. Meist wird man den Shader mit anderen kombinieren.
- **Anisotropic** simuliert spiegelnde Oberflächen, die das Licht abhängig von der Einfallrichtung unterschiedlich reflektieren. Damit lässt sich gebürstetes Metall simulieren oder auch das Glanzverhalten von Seide.
- **Background** sollte nicht für Objekte verwendet werden. Setzt man diesen Shader für den Hintergrund ein (bei den Welteinstellungen), dann dient die zugehörige Farbe oder Textur auch für eine Rundumbeleuchtung.
- **Diffuse** erzeugt eine Oberfläche die ausschließlich diffus reflektiert. Glanzlichter und Spiegelungen fehlen vollständig. Roughness steuert die Helligkeitsverteilung des Shaders, je mehr desto rauer wirkt das Material.
- **Emission** lässt wie bereits angesprochen eine Oberfläche zur Lichtquelle werden.



Ambient Occlusion im Alleingang

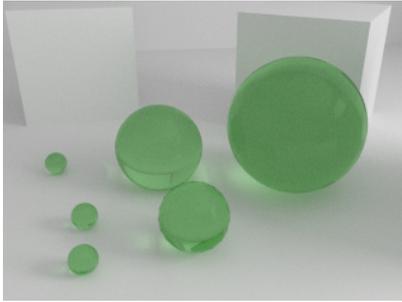


Gebürstetes Metall (Anisotropic)

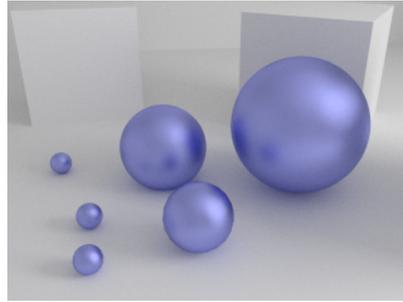


Verschiedene Leuchtkugeln (Emission)

- **Glass** ist (fast) reine Transparenz. Licht wird bei der Brechung nie vollständig gebrochen sondern je nach Auftreffwinkel auch reflektiert. Cycles berücksichtigt das zwar, evtl. will man dem Glass trotzdem noch etwas gerichtete Reflexion zumischen (wie gesagt: dazu weiter unten). Mit Roughness über 0 entsteht Milchglas o.ä.
- **Glossy** ist das Gegenteil zu Diffuse, zumindest wenn Roughness auf 0 gesetzt ist. Hiermit werden alle Arten von Spiegelungen erzeugt. Dazu gehören auch Glanzlichter, die ja nichts anderes als Spiegelungen der Lichtquelle sind. Reale Materialien zeigen fast immer eine Mischung aus diffuser und gerichteter Reflexion. Wie man Diffuse und Glossy mixt, dazu weiter unten.
- **Hair** ist darauf spezialisiert, die beiden verschiedenen Glanzlichter bei Haaren zu simulieren. Das eine entsteht durch Reflexion an der Oberfläche, das andere durch Brechung und Reflexion im Haar selbst. Um beide zusammen zu nutzen müssen zwei Haarshader z.B. mit Add (s.u.) kombiniert werden.



Ein paar Murmeln (Glass)

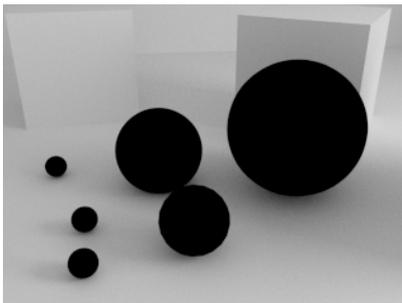


Mattblau ist total in (Glossy mit Roughness)

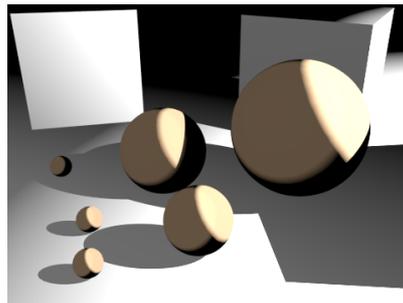


Suzanne als Hippie

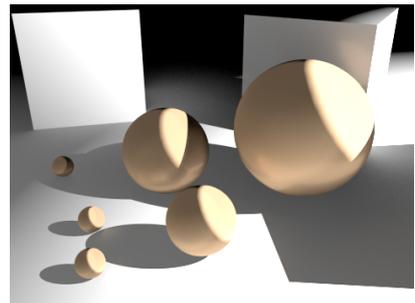
- **Holdout** lässt die zugehörigen Objekte schlicht schwarz, was beim Compositing genutzt werden kann.
- **Refraction** ist eine Alternative zu Glass mit einem etwas anderem Verhalten bei der Berechnung des durchgelassenen Lichtes. Laut Blender Wiki sollte man diesen Shader besser mit anderen mischen.
- **Subsurface Scattering** simuliert Licht, dass in Material wie Haut, Wachs o.ä. eindringt und unter der Oberfläche in verschiedene Richtungen gestreut wird. Wie weit es dabei kommt kann für die drei Farbanteile getrennt eingestellt werden.
- **Toon** erzeugt relativ scharf abgegrenzte Helligkeitsstufen in der Oberfläche, ähnlich wie in manchen Zeichentrickfilmen. Der Effekt wird bei flächigen Lichtquellen allerdings total verwaschen, man sollte bei Verwendung von Toon Punktlichtquellen der Größe Null benutzen. Zusätzlich empfiehlt sich, unter Light Paths auf Direct Light umzustellen, da sonst ebenfalls der Toon Effekt verwaschen wird.



Holdout - schwarze Löcher

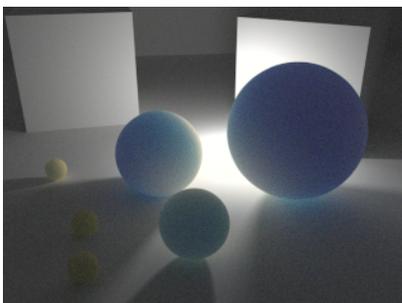


Toon mit zwei Punktlichtern, Direct Light

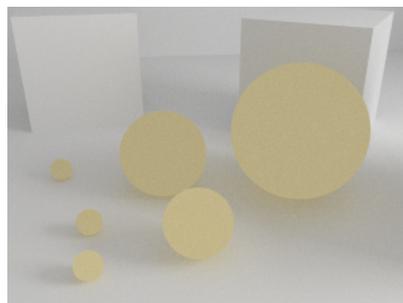


Wie links mit Limited Global Illumination

- **Translucent** kommt zum Einsatz, wenn ein Material zwar nicht durchsichtig ist, aber dennoch genügend Licht durchlässt, so dass seine Rückseite von hinten her aufgehellt wird. Blätter in Sonnenlicht oder Tischtennisbälle zeigen diesen Effekt. Als alleinstehender Shader macht Translucent meist wenig Sinn.
- **Transparent** ist für Durchsichtigkeit ohne Brechung zuständig. Das kann ein hauchfeiner Stoff sein, Transparentpapier oder mit Textur eine Nebelschwade. Auch dieser Shader macht allein meist wenig Sinn.



Durchscheinend dank Subsurface Scattering



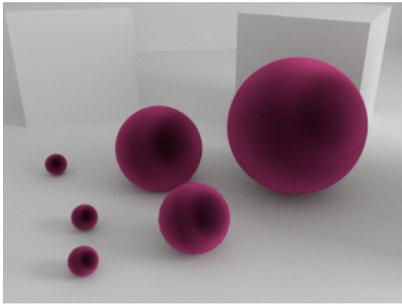
Bälle aus mattem ... Dings (Translucent)



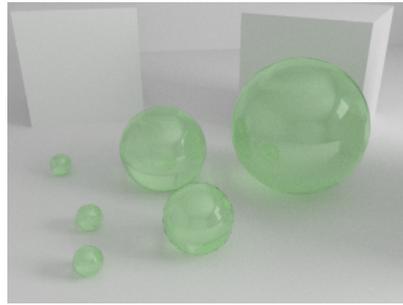
Transparent - platzt bei erster Berührung!

- **Velvet** sorgt für einen samtweichen Effekt, der bei diversen Stoffen oder sehr kurzem Fell für die richtige Helligkeitsverteilung sorgen kann. Unter einem flachen Winkel erscheint die Oberfläche heller, bei direkter Drauf-

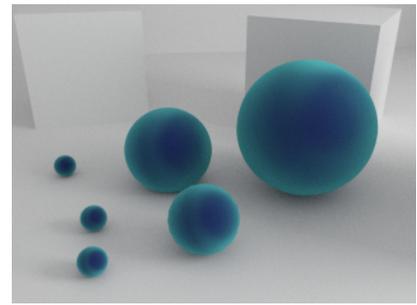
sicht dunkler oder sogar schwarz (was durch Mischung mit anderen Shadern behoben werden kann).  
Der Wert Sigma verstärkt den Effekt, je kleiner er ist; Werte über 1 machen hier keinen Sinn.



Plüsch in Altrosa (Velvet mit Sigma 0,7)



Glas wie oben aber mit addierter Spiegelung



Mix aus Diffuse (blau) und Velvet (türkis)

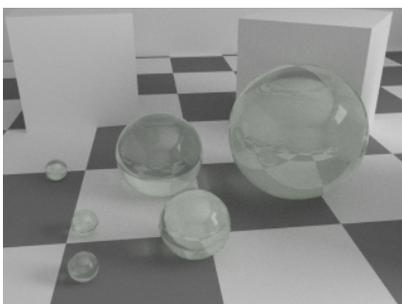
- **Add Shader** addiert die Effekte von zwei Shadern. Dabei sollte man in den meisten Fällen die Helligkeit der beiden Shader reduzieren, da es sonst zu Überbelichtung kommen kann.
- **Mix Shader** ermöglicht mittels des Reglers Fac die Mischung von zwei Shadern in beliebiger Stärke (0 entspricht dem oberen Shader, 1 dem unteren, alle Zahlen dazwischen entsprechenden Mischverhältnissen).

## NOCH MEHR MÖGLICHKEITEN

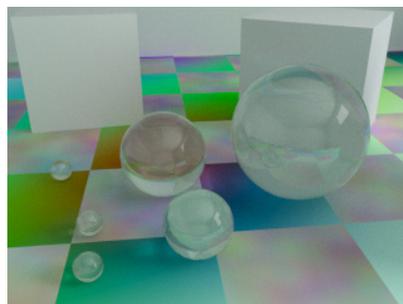
### Steuerung von Materialeigenschaften

Jede Einstellung in Cycles, die rechts einen kleinen Punkt  besitzt kann mit mehr als nur einem einzelnen Wert gefüttert werden. Fast alle Aspekte können so feingesteuert werden. Unter anderem kommen hier die vielleicht schon schmerzlich vermissen Texturen ins Spiel.

Gehen wir noch mal von einem einfachen Diffuse Shader aus, den diesmal der Boden erhält. Klickt man auf den angesprochenen Punkt neben der Farbe eröffnen sich jede Menge neue Möglichkeiten. Bevor wir uns eine davon schnappen ein wichtiger Hinweis: Einen Eintrag in einer Eigenschaft kann man mit dem Punkt Remove aus diesem neuen Menü wieder löschen. Zur Veranschaulichung verwenden wir zunächst Checker Texture. Deren Größe kann mittels Scale angepasst werden (zur Erinnerung: Scale entspricht der Anzahl der Wiederholungen der Textur - also je höher desto kleiner das Muster). Bei dieser Textur können die beiden Farben eines Schachbrettmusters gewählt werden. Aber wer sagt denn, dass das konstante Farben sein müssen? Erneuter Klick auf kleine Punkte und schon bestehen die einen Felder aus einer Noise Texture und die anderen aus einer Magic Texture. Beide kann man wieder mit Scale anpassen.



Ein Schachbrettboden



mit bunten Feldern



und auch noch verbogenen Grenzen

Farbe abwandeln ist eine Sache, aber wir können fast alles durch neue Elemente steuern. Durch Klick auf die Eigenschaft Scale der Checker Texture kann man dort Math eintragen (zu finden unter Converter). Setzt man dessen Wert

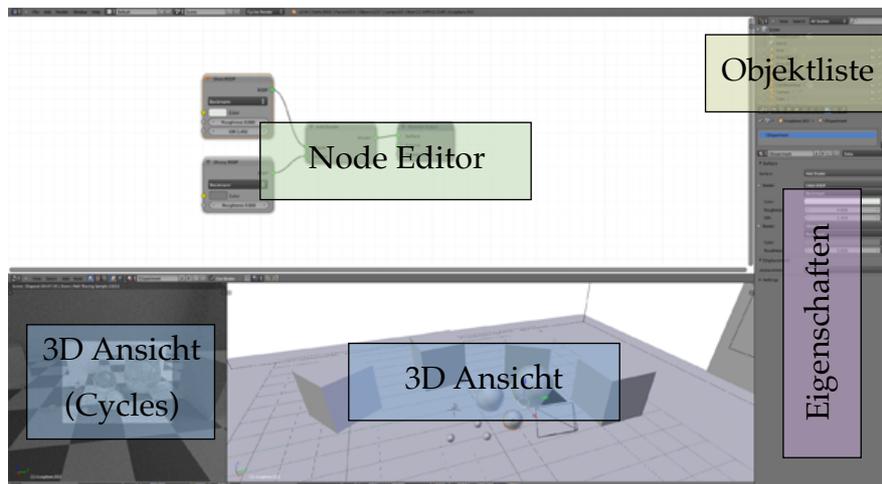
Value auf 10, trägt in den unteren eine Noise Textur ein und schaltet von Add auf Multiply, dann wird das Schachbrett in Abhängigkeit von der Noise Textur an verschiedenen Stellen unterschiedlich skaliert (Scale =1 in der Noise Textur ist zu empfehlen).

## Detailliertere Kontrolle - Noodles

Die obigen Schritte haben schon für eine respektable Liste an Einstellungen gesorgt. Darin den Überblick zu behalten wird zunehmend schwerer (selbst wenn man mit den kleinen Minuszeichen Teile der Liste verbirgt). Diverse Möglichkeiten von Cycles können auf diese Art auch gar nicht genutzt werden, man benötigt dafür Nodes. Zur Beruhigung: streng genommen hat man schon in allen vorangegangenen Beispielen mit Nodes hantiert, wie gleich zu sehen sein wird. Wenn man mit Nodes noch nie Kontakt hatte könnte ein Blick in Lektion R3 helfen.

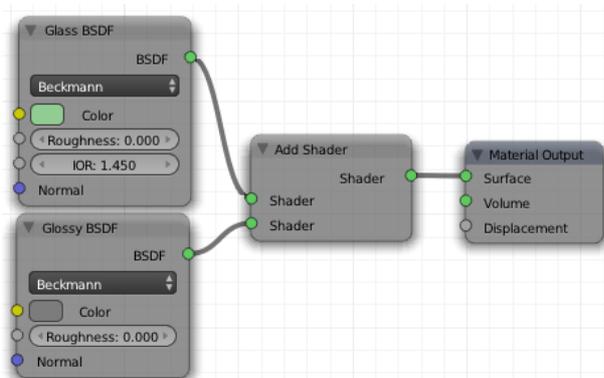
### Ein passendes Layout

Für vernünftiges Arbeiten mit Nodes empfiehlt sich ein anderes Bildschirmlayout. Ein guter Ausgangspunkt dafür ist das Layout Compositing (in der obersten Leiste von Blender unter Default zu finden), das man wie unten dargestellt anpassen kann. Im Node Editor sollte man auf die Bearbeitung von Materialien umschalten, eine 3D Ansicht mit aktiver Darstellung in Cycles ist hilfreich und ein weiteres 3D Fenster dient für Arbeiten an der Szene.

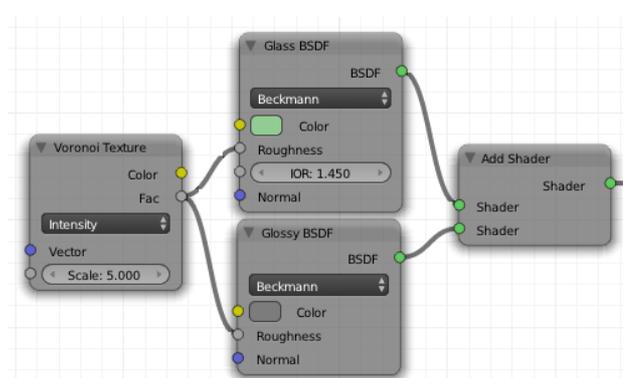


### Erste Noodles

Klickt man ein Objekt mit Cyclesmaterial an, dann erscheint die zugehörige Schaltung im Node Editor. Für die Glaskugeln (das Beispiel zu Add) von weiter oben sieht das wie unten links aus.



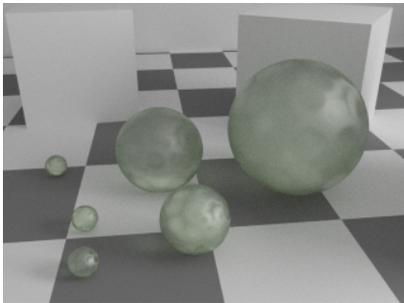
Zwei Shader (Glass und Glossy) werden mittels Add verknüpft



Eine Textur steuert die Rauigkeit des Glass Shaders

Verbindungen zwischen Nodes trennt man z.B. durch **Ctrl**, Linksklick und ziehen wie mit einem Messer.

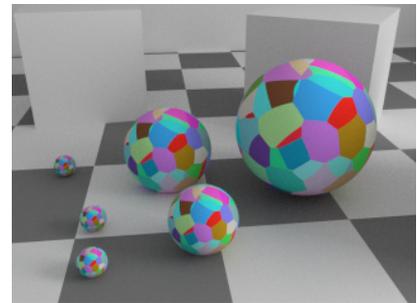
Für Unregelmäßigkeit ist ein Textur Node notwendig. Neue Nodes werden wie neue Objekte mit **Shift+A** aus einer Liste ausgewählt. Für einen ersten Versuch eignet sich z.B. Voronoi Texture, dessen Ausgang Fac mit dem Parameter Roughness von Glass und Glossy Shader verbunden wird. Ein Node kann problemlos mehrfach verwendet werden (was auch Rechenzeit spart).



Voronoi beeinflusst Glossy und Glass Shader



Die nächste Schaltung



und ihr Effekt

Ein neues Beispiel: Glass, Glossy und Add werden entfernt und durch einen Diffuse Shader ersetzt. Dessen Eigenschaft Color soll vom Voronoi Node gespeist werden (gelbe Anschlüsse verwenden). Damit es lustig bunt wird kann man im Voronoi Node Cells statt Intensity verwenden.

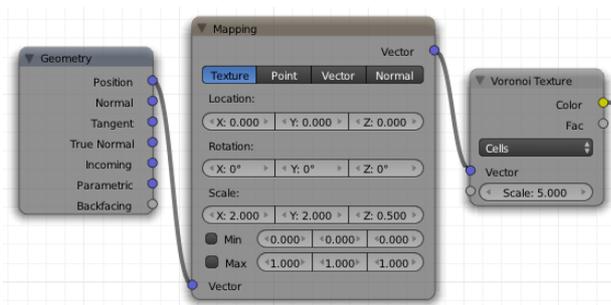
### Texturkoordinaten

Im Moment entscheidet Blender eigenmächtig, wie die Textur auf dem Objekt platziert wird. Hier kann eingegriffen werden, indem man passende Nodes einbaut. Geeignet ist ein Node aus der Rubrik Input, meist Texture Coordinate. Dieser liefert verschiedene Formen von Koordinaten (blaue Anschlüsse ●), die der Voronoi Node nutzen kann. Zu Demonstrationszwecken wird an dieser Stelle aber ein Node vom Typ Geometry verwendet und dessen Ausgang Position mit dem passenden Eingang des Voronoi Node verbunden. Jetzt orientiert sich die Textur an den Weltkoordinaten - die Kugeln sehen trotz gleichen Materials verschieden aus (und verändern sich bei Bewegung).

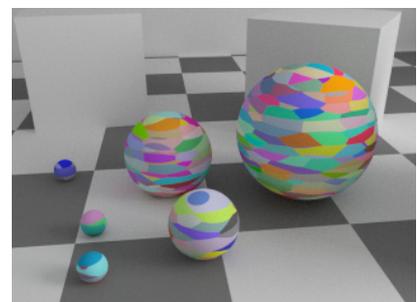
Als nächstes kommt ein Mapping Node hinzu (in der Kategorie Vector). Dieser wird zwischen Geometry und Voronoi installiert (zur Erinnerung: auf die Verbindung ziehen, bis diese aufleuchtet).

In dem neuen Node könnte man beispielsweise die Textur mit den Werten bei Scale verzerren. Schaltet man dazu den Node auf Texture, dann wirken die Werte bei Scale ausnahmsweise einmal wie erwartet (Werte über 1 vergrößern das Muster).

Auch Drehung und Verschiebung ist möglich. Dabei muss einem klar sein, dass die Transformationen wie abgebildet verarbeitet werden (erst Verschiebung, dann Drehung, dann Skalierung). Möchte man eine andere Reihenfolge erzwingen, dann muss man mehrere Mapping Nodes hintereinander schalten.



Ausschnitt aus der neuen Schaltung

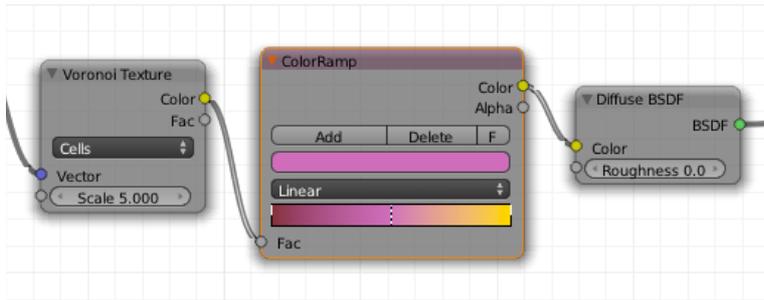


und die neu gestalteten Gummibälle

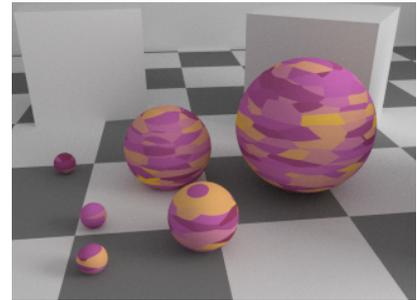
### Gezielt gesteuerte Farbe

Die Textur ist ja schön bunt, aber vielleicht sollen es andere Farben sein. Um die Farbpalette zu verändern muss man in Cycles ein klein wenig umdenken. Der Voronoi Node hat keinen Eingabekanal für Farbe. Deshalb wird nicht er

mit Farbe gefüttert, sondern er wird benutzt um einen anderen Node zu steuern. Dazu ist ein Node vom Typ Color Ramp notwendig (unter Converter) der in die Verbindung von Voronoi zu Diffuse Node eingebaut wird.



Farbe nach Wunsch durch eine Color Ramp

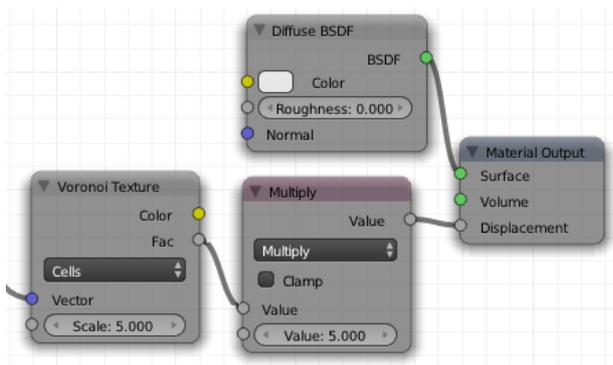


und wieder der Effekt der Schaltung

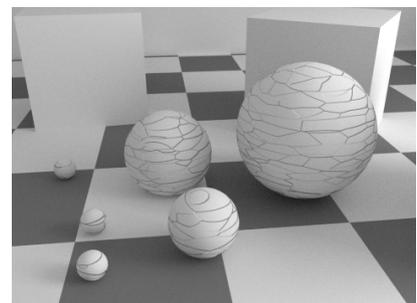
### Dann noch einige Beulen

Der Ausgabenode Material Output hat offensichtlich zwei weitere Eingänge. Der mittlere wird in späteren Versionen einmal dazu dienen, volumetrische Materialien wie z.B. Rauch zu verwirklichen. Im Moment hat ein angeschlossener Shader keinerlei Wirkung. Displacement kann man mit einem anderen Node verbinden, der ein Muster ausgibt. Dabei muss nicht ein numerischer Wert (grau ●) verwendet werden, ein Farbwert tut es auch (gelb ●, es wird dann nur die Helligkeit benutzt).

Wenn der Effekt verstärkt oder abgeschwächt werden soll, dann empfiehlt sich ein zwischengeschalteter Math Node, der auf Multiply umgestellt wurde.



Voronoi, nur verbunden mit Displacement



Der Effekt: Brüche im Material

### Hinweis zu Blender Render

Vieles was in Cycles mit Nodes geht, kann man auch mit Materialien und Texturen in Blender Render verwirklichen (die exakte Technik variiert ein wenig, aber die Grundidee bleibt). Dort muss aber für Material bzw. Textur jeweils eine eigene Noodle gebastelt werden und der Umgang ist nicht ganz so intuitiv wie in Cycles. Um die Möglichkeit von nodebasierten Texturen oder Materialien zu haben muss man bei ausgewähltem Material (bzw. Textur) in einem Nodeeditor in der Einstellung Material  (oooder Textur ) Use Nodes aktivieren (alles im Header des Node Editors).

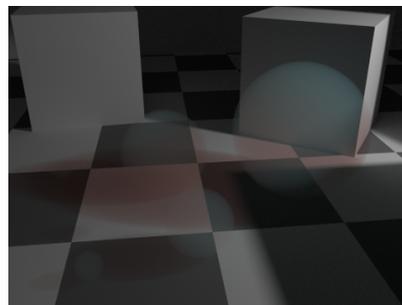
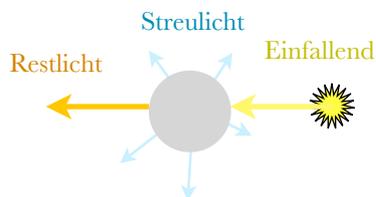
# INNERE WERTE

## Volume Shader

alle bisher vorgestellten Möglichkeiten bleiben buchstäblich an der Oberfläche. Selbst für die Brechung in Glas sind nur Berechnungen beim Übergang von Luft ins Glas notwendig - zumindest prinzipiell. Die Surface Shader nutzen Informationen über den Weg des Lichts innerhalb eines Objekts nicht. Wenn man aber einen passenden Shader in den Eingang Volume des Material Output einsteckt, dann sind ganz neue Dinge möglich. Für ein erstes Kennenlernen sollte dazu zunächst der Eingang Surface ohne Verbindung bleiben.

### Volume Scatter

Wandert Licht durch Dunst oder Gas, dann wird ein Teil davon an den kleinen Teilchen gestreut und gelangt so auch zu einem seitlich stehenden Betrachter. Das Licht, das den streuenden Bereich durchdringt ist dadurch abgeschwächt.



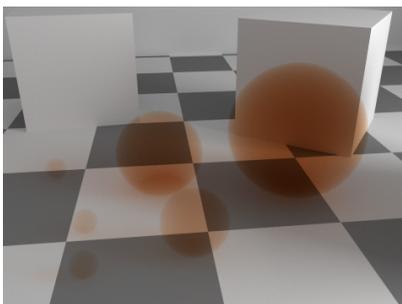
Licht wird in einem Gas teils gestreut

Volume Scatter in Blender

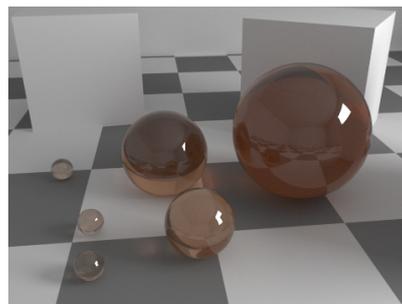
Ist die Farbe eines Volume Scatter Materials nicht weiß, dann wird gestreutes Licht entsprechend gefärbt. Da dieser Farbanteil dem durchdringenden Licht fehlt ist dieses genau komplementär zum gestreuten Licht gefärbt. Diesem Effekt verdanken wir auch den blauen Himmel (Licht der Sonne, das seitlich zu uns gestreut wird) und den roten Sonnenuntergang (die Farbe die beim Durchgang durch die Luft bleibt, wenn Blau und Grün seitlich gestreut wurden). Dieser Effekt ist oben erkennbar: Die direkt beschienenen Bereiche der Kugeln erscheinen in dem eingestellten Hellblau, dort wo Licht der hellen Bodenbereiche in Richtung Kamera durch die Kugeln geht ist es rötlich getönt.

### Volume Absorption

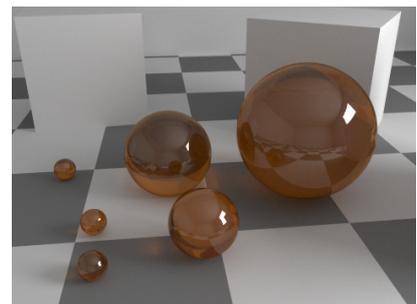
Vor allem beim Durchgang durch festes transparentes Material wird Licht immer zum Teil absorbiert. Diesen Effekt kann der Oberflächenshader Glas nicht simulieren, da er die Dicke des Materials nicht berücksichtigt. Volume Absorption macht genau das und nur das. Im Alleingang macht dieser Shader damit meist wenig Sinn, in Kombination mit Glass dafür um so mehr. Vergleicht man das mittlere mit dem rechten Bild unten, dann sieht man, dass mit Volume Absorption ein realistischeres Aussehen von gefärbtem Glas o.ä. möglich ist.



Volume Absorption im Alleingang



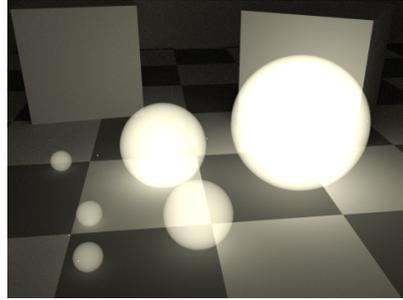
Kombiniert mit farblosem Glas



Gefärbtes Glas ohne Volume Absorption

## Emission

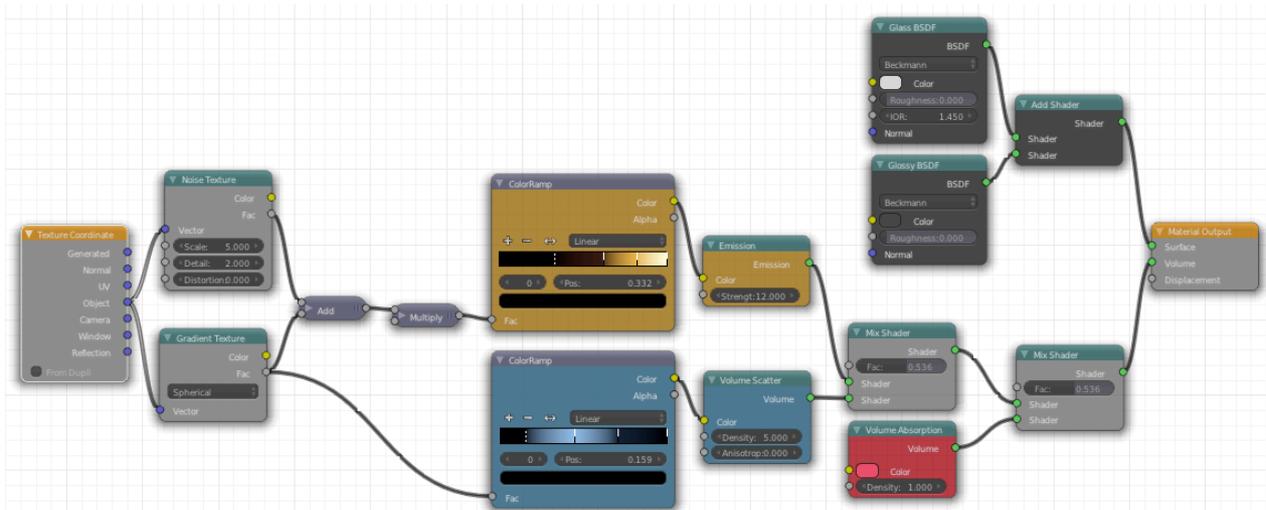
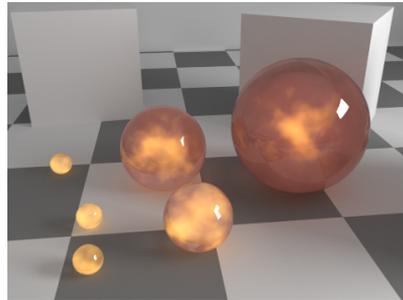
Der oben schon vorgestellte Emission Shader kann auch volumetrisch verwendet werden. Das Ergebnis ist eine leuchtende Gaswolke und die Option für jede Menge Spezialeffekte. Da volumetrische Materialien rechenintensiv sind sollte man sich aber gut überlegen, ob es vielleicht auch anders geht.



Volumetrisches Licht

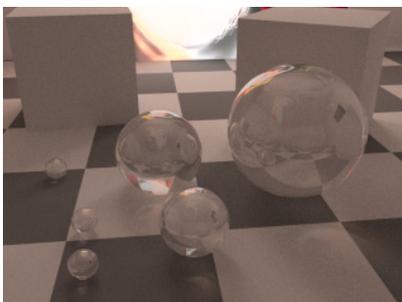
## Und noch viel mehr

Dieses Skript ist ausreichend lang, darum hier am Ende nur noch der Hinweis: Natürlich lassen sich auch die Volumenshader mit Texturen und anderen Mitteln ebenfalls beeinflussen. Und nichts spricht dagegen, sie miteinander zu kombinieren. Ja, oft ist das sogar anzuraten. Beispielsweise leuchtet praktisch kein reales Material nur, sondern absorbiert auch immer Licht. Das Beispiel unten kombiniert so ziemlich alles, was dort oben an Möglichkeiten aufgetaucht ist. Das innere der Opalkugeln besteht aus Emission (gelb), Scattering (blau) und Absorption (rot). Dazu kommt Glas und Glossy (dunkelgrau).

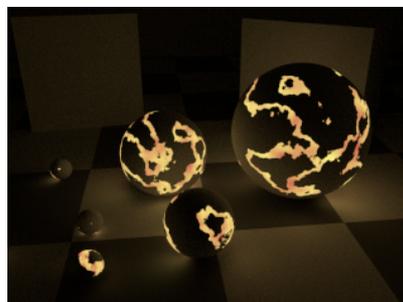


## ÜBUNGS AUFGABEN

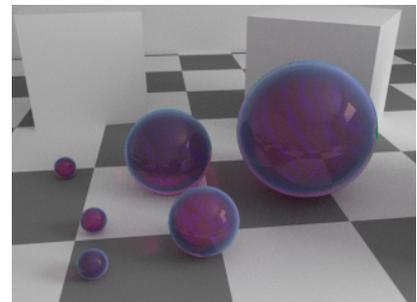
1. Stellen Sie eine reale Beleuchtungssituationen in einer einfachen Szene nach. Im Gegensatz zu Blender Render können Sie dabei in weiten Teilen von realen Beleuchtungsverhältnissen ausgehen.
2. Wiederholen Sie Aufgabe 1 ein paar mal und verpassen Sie Ihrer Szene einige der folgenden Lichtstimmungen: Düstere Vorratskeller; Großraumbüro; Spot auf der Bühne; Mondlicht; Morgenstimmung im Frühling; Sommer in der Sahara; unter schattigen Bäumen; Tauchen in der Karibik (Schwierigkeit ist ansteigend)
3. Bilden Sie durch Verknüpfung eines Diffuse Shaders mit einem Glossy Shader verschiedene Oberflächen nach (Roughness nicht vergessen). Vielleicht wollen Sie die Materialien aus Aufgabe 1 in Lektion T1 noch mal umsetzen: Eierschale, Gummiball und Christbaumkugel (heut mal golden?).
4. Bleiben wir doch beim Thema Gummiball. Die gibt es auch leicht transparent mit rauer Oberfläche. Beleuchtet man so einen Ball von hinten mit einer Taschenlampe, dann glüht er schick von innen.
5. Verpassen Sie einer Kugel ein Schachbrettmuster und füttern Sie die zugehörige Textur mal mit allem, was Nodes vom Typ Geometry oder Texture Coordinates so bieten. Einfach um gesehen zu haben was passiert.
6. Eine spiegelnde Diskokugel wirft kleine Lichtpunkte in den umgebenden Raum - das müsste doch mit Cycles gehen (Vorsicht, nicht so einfach, wie man meinen mag und vor allem rechenintensiv).
7. Falls Sie noch nicht ausgelastet sind können Sie mal einer Lichtquelle ein Cycles Material verpassen und damit experimentieren.
8. Und wenn Sie wirklich Spaß haben wollen, dann bekommt auch die Welt ein Material vom Typ Background, dessen Farbe mittels Sky Texture beeinflusst wird.
9. Wenn Sie immer noch nicht Lust auf kreative Eigenkreationen verspüren, dann sehen Sie sich mal um - da gibt es haufenweise Materialien, die man nachbilden könnte. Oder nehmen Sie eines der nachfolgenden Bilder als Herausforderung an.



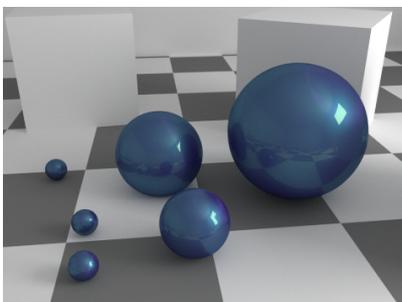
Lebendige Beleuchtung mit Charakter



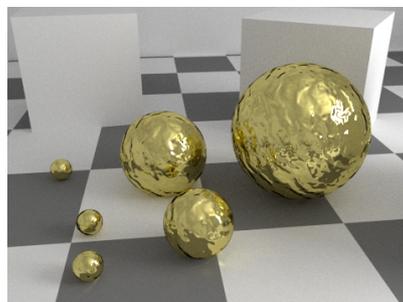
Höllenkugeln



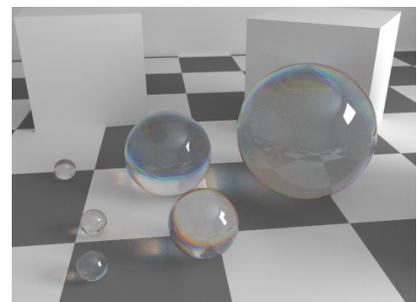
Luxusmurmeln



Mehrschichtiger Autolack (Diffuse + 3x Glossy + Anisotropic, versch. Roughness und Farben)



Gold!?

Lichtbrechung mit Farbzerlegung (Dispersion)  
Mit 3 Glass Shadern geht das!