

Materialtypen

Inhaltsverzeichnis

- [1 Materialnamen](#)
- [2 Materialtypen](#)
 - [2.1 m.pbrmetal](#)
 - [2.2 m.pbrmetalmasked](#)
 - [2.3 m.pbrmetaldetail](#)
- [3 Alte Materialien](#)
 - [3.1 m.notex](#)
 - [3.2 m.onetex](#)
 - [3.3 m.reflect](#)
 - [3.4 m.gloss](#)
 - [3.5 m.tbumpdex](#)
 - [3.6 m.tbumpgloss](#)
 - [3.7 m.tbumpenv](#)
- [4 Material Beispiele](#)
- [5 Bodentextur Material](#)

Materialien beschreiben wie eine modellierte Oberfläche in der Simulation dargestellt werden. Sie spezifizieren eine Reihe and Konfigurationen und Ressourcen wie klassische Materialeigenschaften, Texturen und Shader.

Diese Definition ist eine Übersetzung aus dem [TrainzDevWiki](#)

Materialien beschreiben wie eine modellierte Oberfläche in der Simulation dargestellt werden. Sie spezifizieren eine Reihe and Konfigurationen und Ressourcen wie klassische Materialeigenschaften, Texturen und Shader. Jedes Material ist eine Instanz eines bestimmten vordefinierten Materialtyps, enthalten in der unten beschriebenen Auswahl.

Diese Seite beschreibt das [Content Format v4.6](#) und setzt die Nutzung des [FBX Dateiformates](#) für alle Meshes voraus.

1 Materialnamen

Ein Materialname besteht aus einem durch den Contentersteller definierten Namen und einem Material-Suffix. Der Materialtyp entspricht einem der unten definierten Typen, z.B. "m.onetex". Alle Materialtypen beginnen mit "m.". Der Materialtyp wird manuell einem Material in einem 3D-Programm zugewiesen, z.B. Autodesk 3ds Max. Es wird nicht durch die Konfiguration des Materials impliziert. Materialtypen in Trainz unterliegen nicht der Groß- und Kleinschreibung (case sensitive).

Für mehr Informationen zur Konfiguration von Materialien, siehe [Exportieren mit 3ds Max](#).

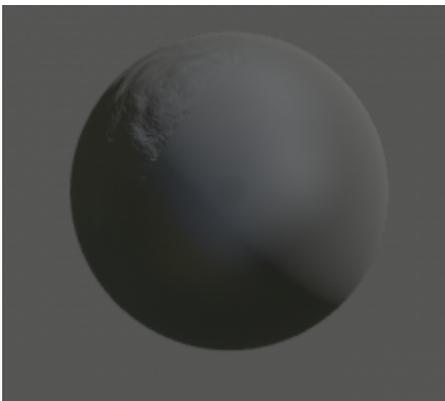
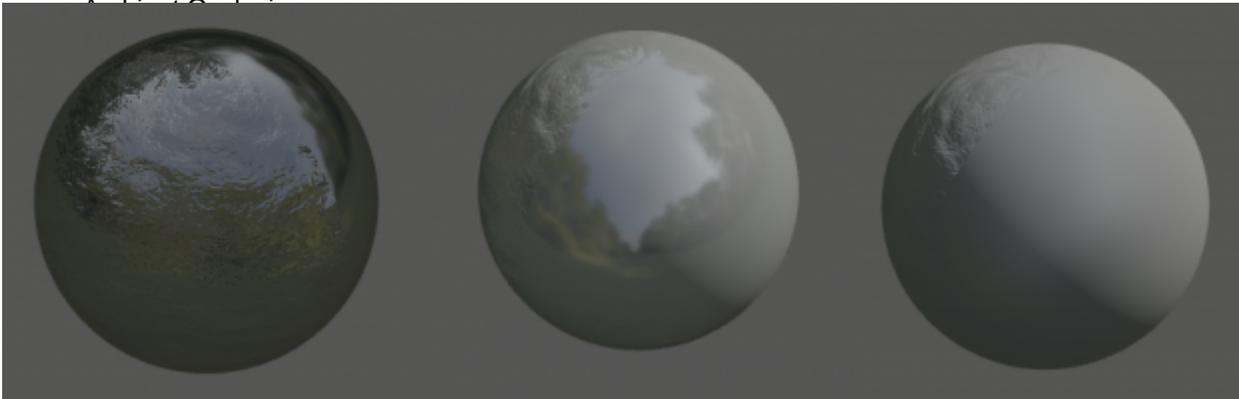
2 Materialtypen

Trainz nutzt Materialien die über den Grad der Rauheit und wie metallisch eine Oberfläche (Metallic/Roughness) aussehen soll definiert werden ([PBR](#)). Alle Materialien folgen diesem Format, oder (im Falle von alten Materialtypen) werden so gut wie möglich in dieses Konvertiert. PBR und die Metallic/Roughness-Arbeitsweise werden sehr ausführlich online in einer Vielzahl von frei verfügbaren Präsentationen und Demonstrationen vom einfachen Material bis zur hohen Mathematik behandelt. Es wird empfohlen, dass Content-Ersteller sich über das Thema im Vorfeld informieren, bevor sie beginnen Trainzmaterialien zu konfigurieren.

2.1 [m.pbrmetal](#)

[m.pbrmetal](#) ist ein generisches undurchsichtiges Material. Es sollte als Einstiegspunkt jeder Contenterstellung ohne spezielle Anforderungen verwendet werden.

- Albedo (RGB)
- Normal (XYZ)
- Height
- Emissive
- Roughness



2.2 m.pbrmetalmasked

m.pbrmetalmasked ergibt die gleichen Ergebnisse wie [m.pbrmetal](#), ergänzt dieses jedoch noch um die Möglichkeit maskierter Transparenz.

- Albedo (RGB)
- Masked Opacity (A)
- Normal (XYZ)
- Height
- Emissive
- Roughness
- Ambient Occlusion
- Metallicity

2.3 m.pbrmetaldetail

m.pbrmetaldetail ergibt die gleichen Ergebnisse wie [m.pbrmetal](#), ergänzt dieses jedoch noch um eine [Detailmap](#).

- Albedo (RGB)
- Normal (XYZ)
- Height
- Emissive
- Roughness
- Ambient Occlusion
- Metallicity

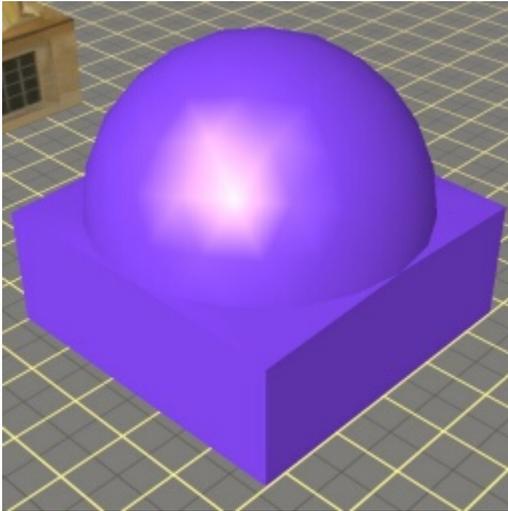
3 Alte Materialien

Diese alten Materialien sollten nur benutzt werden, wenn Kompatibilität gewährleistet werden soll oder für alte Trainzversionen gebaut wird

3.1 m.notex

[m.notex](#) ist ein untexturiertes Material

- Opacity = Gleichmäßige Transparenz kann über die Materialeinstellungen justiert werden



3.2 m.onetex

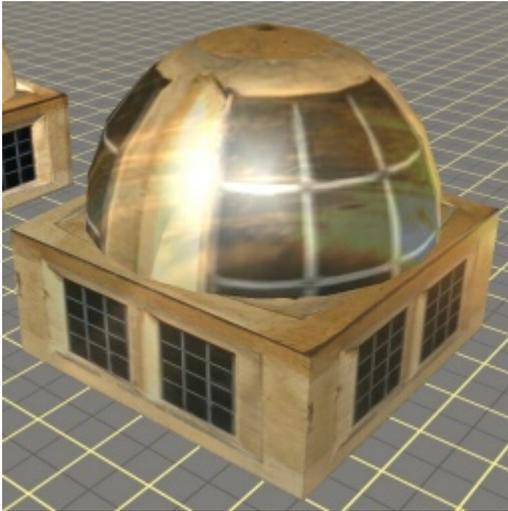
[m.onetex](#) ist ein Material mit Diffuse Map

- Opacity = Per-Texel-Transparenz kann über den Alphakanal der Diffusemap justiert werden



3.3 m.reflect

[m.reflect](#) ist ein reflektierendes Material



3.4 m.gloss

[m.gloss](#) ist ein glänzendes reflektierendes Material

- Opacity = Einheitliche Transparenz kann über Materialeinstellungen justiert werden



3.5 m.tbumpmap

[m.tbumpmap](#) ist ein Material mit Normalmap

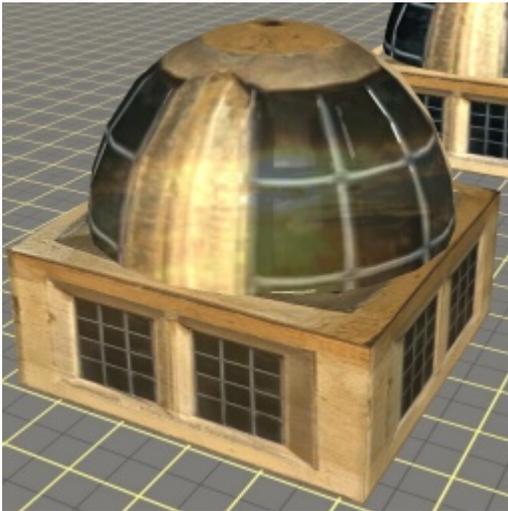
- Opacity = Per-Textel-Transparenz kann über den Alphakanal der Diffusemap justiert werden
- Specular = Per-Textel-Glanz kann über den Alphakanal der Normalmap justiert werden



3.6 m.tbumpgloss

[m.tbumpgloss](#) ist ein reflektierendes Material mit Normalmap

- Opacity = Einheitliche Transparenz kann über Materialeinstellungen justiert werden
- Specular = Per-Texel-Glanz kann über den Alphakanal der Normalmap justiert werden
- Reflection = Per-Texel-Reflektion kann über den Alphakanal der Diffusemap justiert werden



3.7 m.tbumpenv

[m.tbumpenv](#) ist ein die Umgebung reflektierendes Material mit Normalmap. In den derzeitigen Versionen von Trainz ist dieses das bevorzugte Material für die meisten Modelle, die durch die Szenenbeleuchtung beleuchtet werden - solange kein anderer spezieller Effekt erzielt werden soll. Ausnahmen bilden Tunnel und Innenleben/Führerstände.

- Opacity = Einheitliche Transparenz kann über Materialeinstellungen justiert werden
- Specular = Per-Texel-Glanz kann über den Alphakanal der Normalmap justiert werden

- Reflection = Per-Texel Reflektionsstärke der Umgebungsmap kann über den Alphakanal der Diffusemap justiert werden



4 Material Beispiele

Lade die Beispielmodelle unten herunter und platziere sie in Trainz um zu sehen welche Effekte mit den unten gelisteten Materialien erzielt werden können:

Diese Beispiele nutzen die neuen PBR Shader nicht.

Rost [Download Asset Tutorial ansehen](#)

Farbe [Download Asset Tutorial ansehen](#)

Erstellung von Specularmaps [Download Asset Tutorial ansehen](#)

5 Bodentextur Material

Inhalte vom Typ [KIND Groundtexture](#) nutzen ein besonderes Material welches die gleichen Funktionen bietet wie [m.pbrmetal](#). Es ist nicht möglich das Bodentexturmaterial zu überschreiben.

Im Editor können Nutzer mehrere Ebenen von [Bodentexturen](#) mit Überlagerungseffekten malen. Historisch bedingt wurde diese Technologie oft verwendet, um niedrig aufgelöste und häufig gekachelte Texturen zu verschleiern. Die Zuvielverwendung dieses Verfahrens kann zu wesentlicher Unschärfe führen und die Performance reduzieren. Aktuelle Bodentextur-Materialien sind hoch aufgelöst und sollten keine Kachelung erkennen lassen. Für die beste Performance und Optik sollten Überlagerungen nur für tatsächliche Änderungen des Oberflächenmaterials benutzt werden (z.B. Gras zu Stein zu Matsch), anstelle Texturdetails zu simulieren.

Ein Beispiel dazu hat N3V bereits als Vorschau auf [TrainzPortal](#) veröffentlicht.

Externer Inhalt [youtu.be](#)

Inhalte von externen Seiten werden ohne deine Zustimmung nicht automatisch geladen und angezeigt.

Alle externen Inhalte anzeigen

Durch die Aktivierung der externen Inhalte erklärst du dich damit einverstanden, dass personenbezogene Daten an Drittplattformen übermittelt werden. Mehr Informationen dazu haben wir in unserer Datenschutzerklärung zur Verfügung gestellt.

Externer Inhalt [youtu.be](#)

Inhalte von externen Seiten werden ohne deine Zustimmung nicht automatisch geladen und angezeigt.

Alle externen Inhalte anzeigen

Durch die Aktivierung der externen Inhalte erklärst du dich damit einverstanden, dass personenbezogene Daten an Drittplattformen übermittelt werden. Mehr Informationen dazu haben wir in unserer Datenschutzerklärung zur Verfügung gestellt.

Hinweis: Die PBR-Materialien sind erst ab einer der kommenden Trainzversionen (Arbeitstitel [TRS2018](#)) verfügbar und sind nicht mit T:ANE oder tiefer kompatibel. Für diese Trainzversionen sollen noch die Materialien m.tbumpenv etc. benutzt werden.