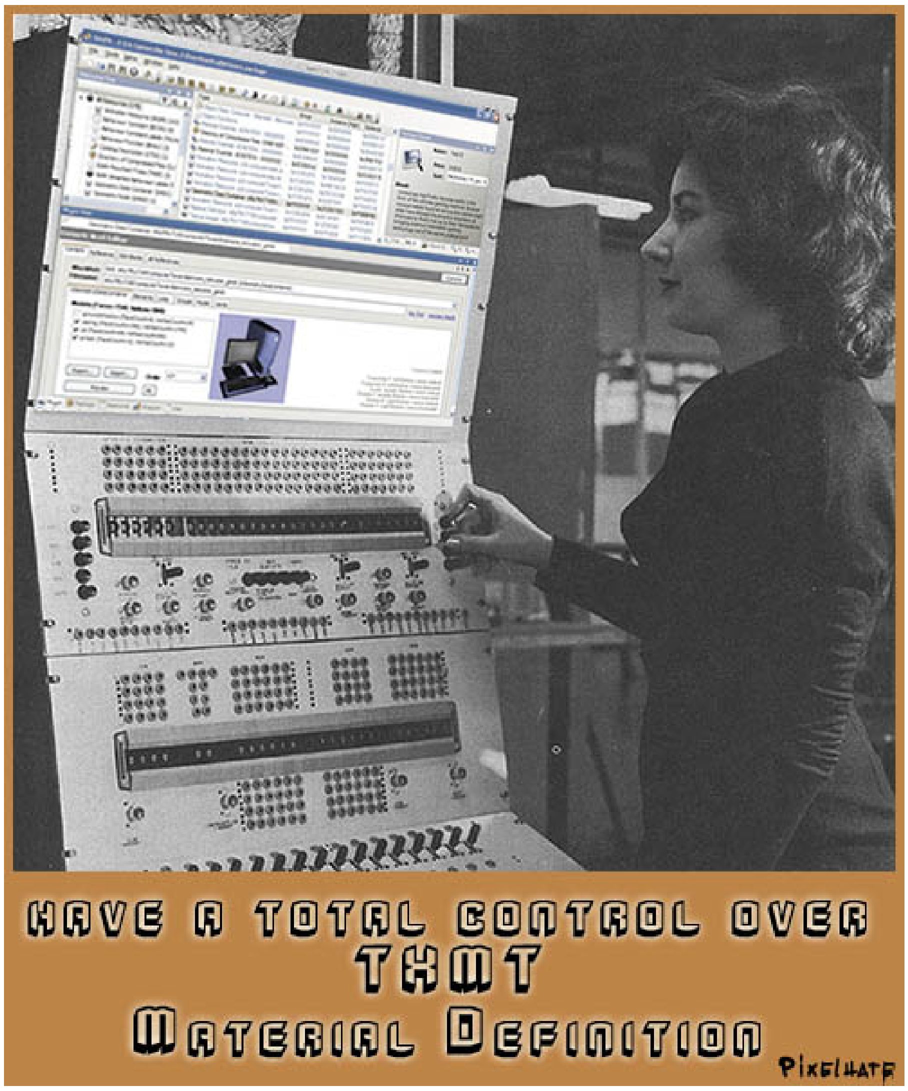
Exploring the Material Definition – TXMT aka MATD



Exploring the TXMT parameters: introduction

In my need to understand the texture section of an object and the different ways to influence it, I came across the idea of an illustrated systematic testing of all the TXMT parameters.

So, this isn't a guide or a tutorial but more a compilation of notes and experiments with comparative pictures, a reference table, made in hope that it will be useful for others.

My main sources of data where:

- The TXMT thread: http://www.modthesims2.com/showthread.php?t=35769

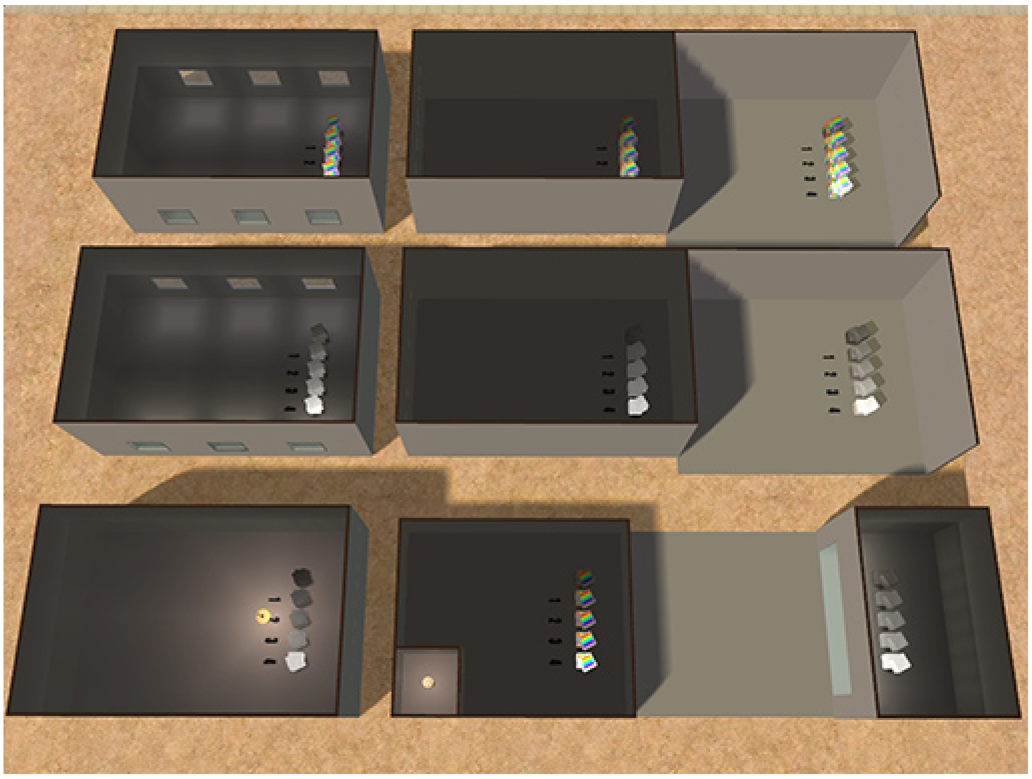
Anyone interested in this matter should read it entirely.

- The Wiki section about TXMT: http://www.sims2wiki.info/wiki.php?title=TXMT

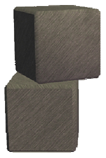
Moreover, bits of knowledge have been collected from the entire Help section at MTS2 Information from theses sources where copied/pasted directly.

My everlasting Gratitude to pioneers like: Numenor, Delphy, RGiles, Niol, Inge Jones, Xanathon and too numerous others contributors to name…

The tests where made on 2 x 5 recolours of a decorative object. Tested outside, inside without light, inside with light coming from lateral windows, inside with lamp above, inside with enclosed lamp, inside with full wall-windows on front.
Games settings are maxed out. No lighting hacks



The mesh used as test object is courtesy by Little_Tx_Mama. Thank you, Lil'Mama.



Pixelhate. 2009

Generality

The Material - Definition (TXMT) manages in single Resources the complete behaviour of the appearance of every kind of texture applied to the mesh or subset.

In other words, with the TXMT, you can decide how your object or parts of it will look in the game. Reflective or not, transparent or opaque, mat or shiny, dark or bright, coloured or not, bump enabled, alpha enabled, etc.

Each object has, *at least*, one Material – Definition (TXMT).
Object can have subsets. For each subset there is *at least* one Material – Definition (TXMT), but often we can find multiple Material – Definition (TXMT) for each subsets, because the subsets may be multistate (clean/dirty, lit/unlit...) and/or they may have more than 1 colour.

This is how you can have parts of an object transparent, other parts metal shiny while the rest is mat and opaque.

Some parameters of the Material – Definition (TXMT) can be used *in conjunction* or *alternatively* to the texture.
In other words, sometimes the texture is referenced into the Material – Definition (TXMT) and the special effects can be added to it.
On the contrary, sometimes the rendering of the object isn't based on a texture, but relies only on those special effects.
The typical example is the glass, which can be rendered without the use of a texture: the only things we need to specify are: its transparency, and its colour (if any) and eventually its reflection.

There are many parameters in TXMT and some are specific to certain type of object (Wall, floor, skin ...).
Some parameters are dependent or interrelated to others.

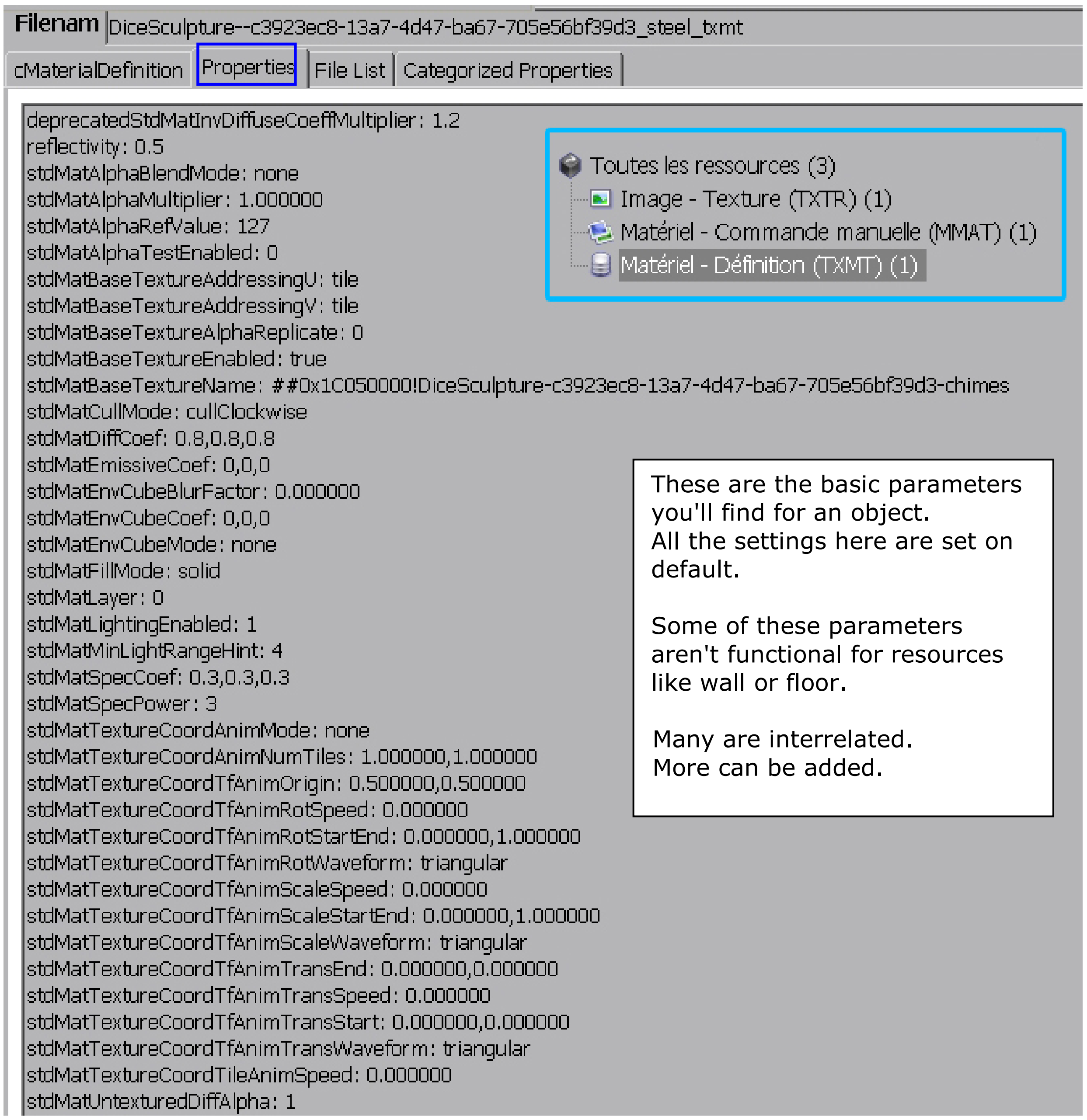
We are going to study the **basic parameters** you'll find in almost every object. SimPE will be our Control Console for that purpose.

We can categorize parameters in many ways; I propose to follow the categories offered in SimPE.

- **Part 1:** Basics settings (texture enabling, mesh test settings, lightning, colouring, glow effect)
  Default Textures in SimPE categorized properties.
- **Part 2:** Reflection settings (enabling, EnvCube map names & look, adding a custom reflection)
  Default Environment Map in SimPE categorized properties.
- **Part 3:** Blend settings (transparency and alpha, blending, glass effect with/without texture)
  Default Textures Blending in SimPE categorized properties.
- **Part 4:** Animation settings
  Texture Coord Animation in SimPE categorized properties.

For a complete list of all the known parameters:
http://www.sims2wiki.info/TXMT/Parameters/AlphabeticallySequencedList

This is how our Control Console looks in SimPE.
In Material - Definition (TXMT) properties tab:



Filenam DiceSculpture--c3923ec8-13a7-4d47-ba67-705e56bf39d3_steel_txmt

```
deprecatedStdMatInvDiffuseCoeffMultiplier: 1.2
reflectivity: 0.5
stdMatAlphaBlendMode: none
stdMatAlphaMultiplier: 1.000000
stdMatAlphaRefValue: 127
stdMatAlphaTestEnabled: 0
stdMatBaseTextureAddressingU: tile
stdMatBaseTextureAddressingV: tile
stdMatBaseTextureAlphaReplicate: 0
stdMatBaseTextureEnabled: true
stdMatBaseTextureName: ##0x1C050000!DiceSculpture-c3923ec8-13a7-4d47-ba67-705e56bf39d3-chimes
stdMatCullMode: cullClockwise
stdMatDiffCoef: 0.8,0.8,0.8
stdMatEmissiveCoef: 0,0,0
stdMatEnvCubeBlurFactor: 0.000000
stdMatEnvCubeCoef: 0,0,0
stdMatEnvCubeMode: none
stdMatFillMode: solid
stdMatLayer: 0
stdMatLightingEnabled: 1
stdMatMinLightRangeHint: 4
stdMatSpecCoef: 0.3,0.3,0.3
stdMatSpecPower: 3
stdMatTextureCoordAnimMode: none
stdMatTextureCoordAnimNumTiles: 1.000000,1.000000
stdMatTextureCoordTfAnimOrigin: 0.500000,0.500000
stdMatTextureCoordTfAnimRotSpeed: 0.000000
stdMatTextureCoordTfAnimRotStartEnd: 0.000000,1.000000
stdMatTextureCoordTfAnimRotWaveform: triangular
stdMatTextureCoordTfAnimScaleSpeed: 0.000000
stdMatTextureCoordTfAnimScaleStartEnd: 0.000000,1.000000
stdMatTextureCoordTfAnimScaleWaveform: triangular
stdMatTextureCoordTfAnimTransEnd: 0.000000,0.000000
stdMatTextureCoordTfAnimTransSpeed: 0.000000
stdMatTextureCoordTfAnimTransStart: 0.000000,0.000000
stdMatTextureCoordTfAnimTransWaveform: triangular
stdMatTextureCoordTileAnimSpeed: 0.000000
stdMatUntexturedDiffAlpha: 1
```

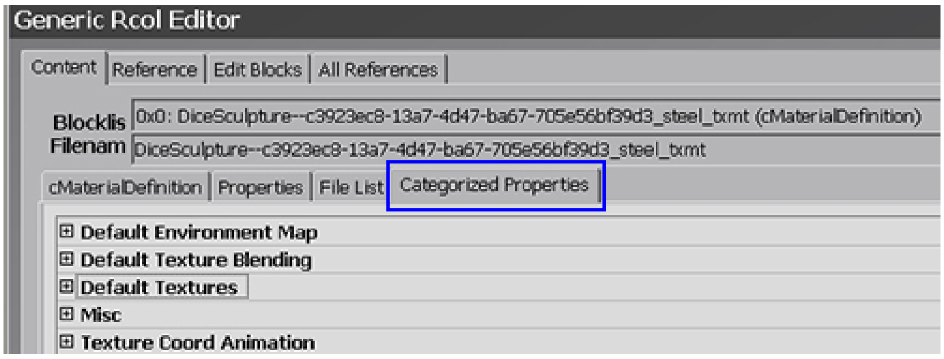
Toutes les ressources (3)
Image - Texture (TXTR) (1)
Matériel - Commande manuelle (MMAT) (1)
Matériel - Définition (TXMT) (1)

These are the basic parameters you'll find for an object.
All the settings here are set on default.

Some of these parameters aren't functional for resources like wall or floor.

Many are interrelated.
More can be added.

Categorized in SimPE:

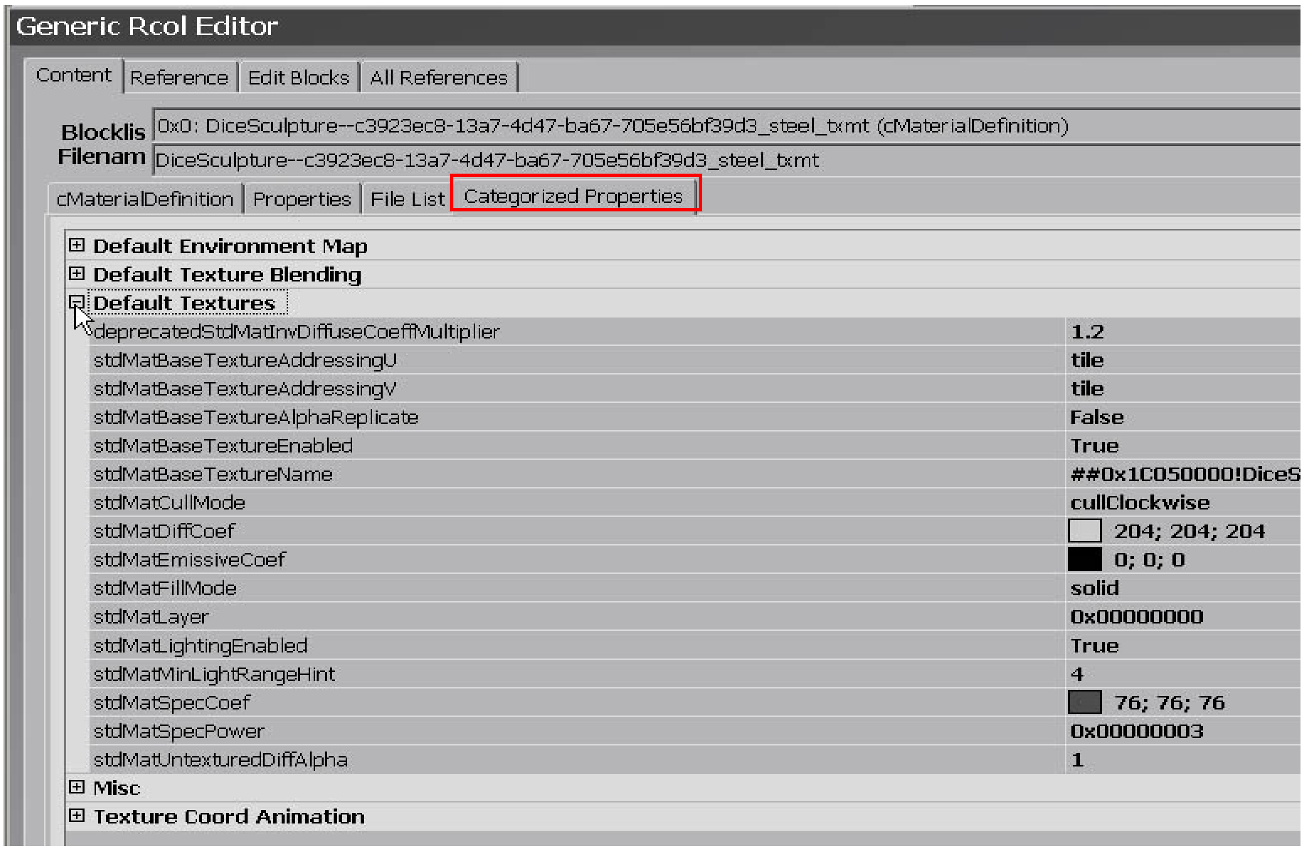


Blocklis 0x0: DiceSculpture--c3923ec8-13a7-4d47-ba67-705e56bf39d3_steel_txmt (cMaterialDefinition)
Filenam DiceSculpture--c3923ec8-13a7-4d47-ba67-705e56bf39d3_steel_txmt

- ⊞ **Default Environment Map**
- ⊞ **Default Texture Blending**
- ⊞ **Default Textures**
- ⊞ **Misc**
- ⊞ **Texture Coord Animation**

Exploring the TXMT parameters part 1

This part covers the Default Textures parameters section in the Categorized Properties in TXMT.

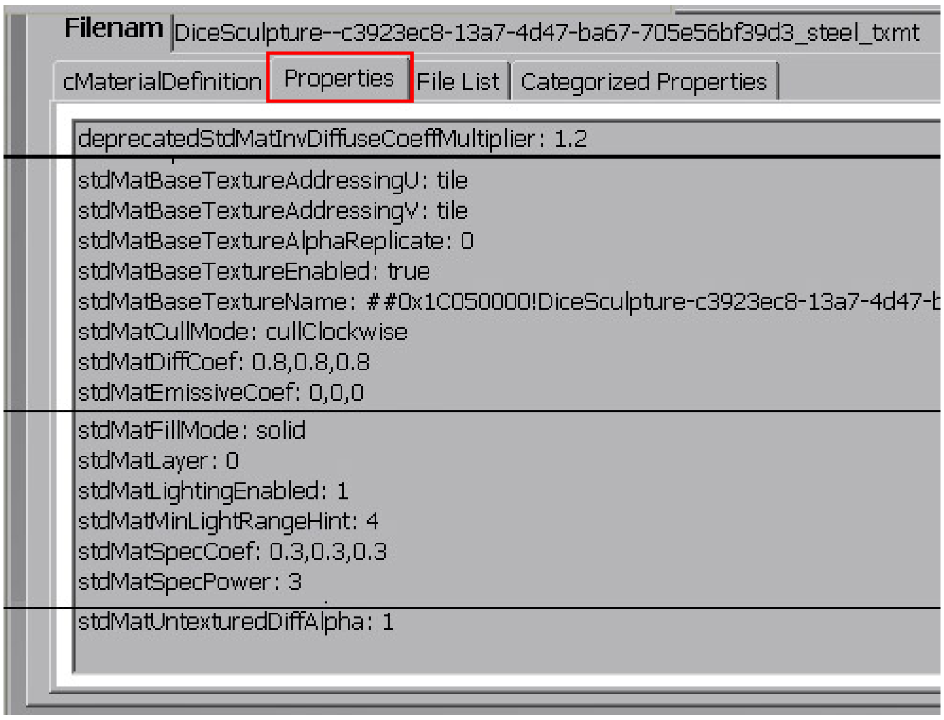


Blocklis 0x0: DiceSculpture--c3923ec8-13a7-4d47-ba67-705e56bf39d3_steel_txmt (cMaterialDefinition)
Filenam DiceSculpture--c3923ec8-13a7-4d47-ba67-705e56bf39d3_steel_txmt

deprecatedStdMatInvDiffuseCoeffMultiplier	1.2
stdMatBaseTextureAddressingU	tile
stdMatBaseTextureAddressingV	tile
stdMatBaseTextureAlphaReplicate	False
stdMatBaseTextureEnabled	True
stdMatBaseTextureName	##0x1C050000!DiceS
stdMatCullMode	cullClockwise
stdMatDiffCoef	⬜ 204; 204; 204
stdMatEmissiveCoef	⬛ 0; 0; 0
stdMatFillMode	solid
stdMatLayer	0x00000000
stdMatLightingEnabled	True
stdMatMinLightRangeHint	4
stdMatSpecCoef	▨ 76; 76; 76
stdMatSpecPower	0x00000003
stdMatUntexturedDiffAlpha	1

16 parameters.

Note how values are displayed in different formats on Properties tab and in Categorized Properties. You may modify settings in both tabs but you must respect their original format.

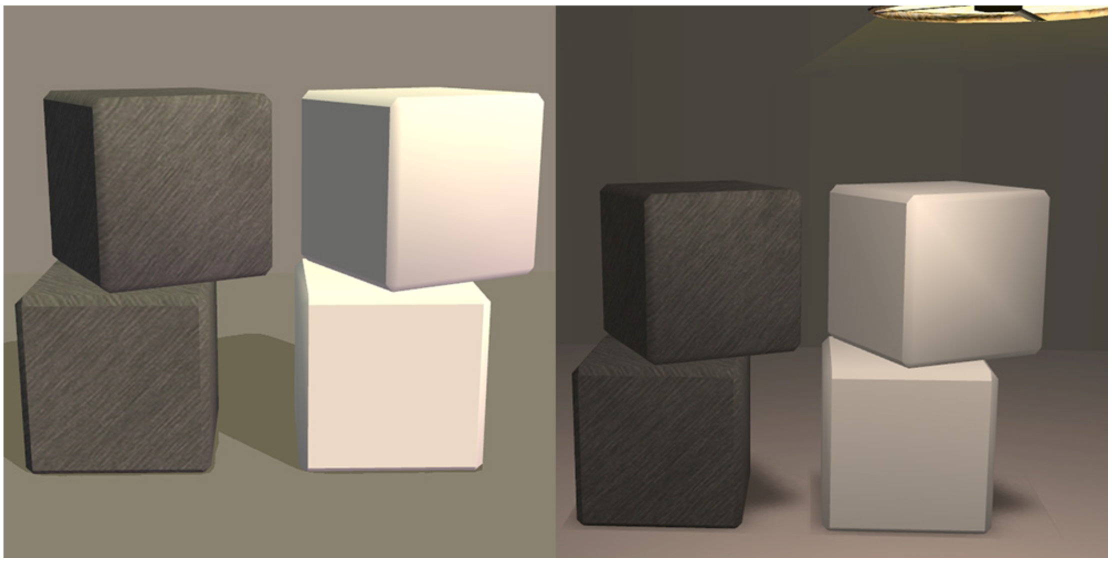


DiceSculpture--c3923ec8-13a7-4d47-ba67-705e56bf39d3_steel_txmt

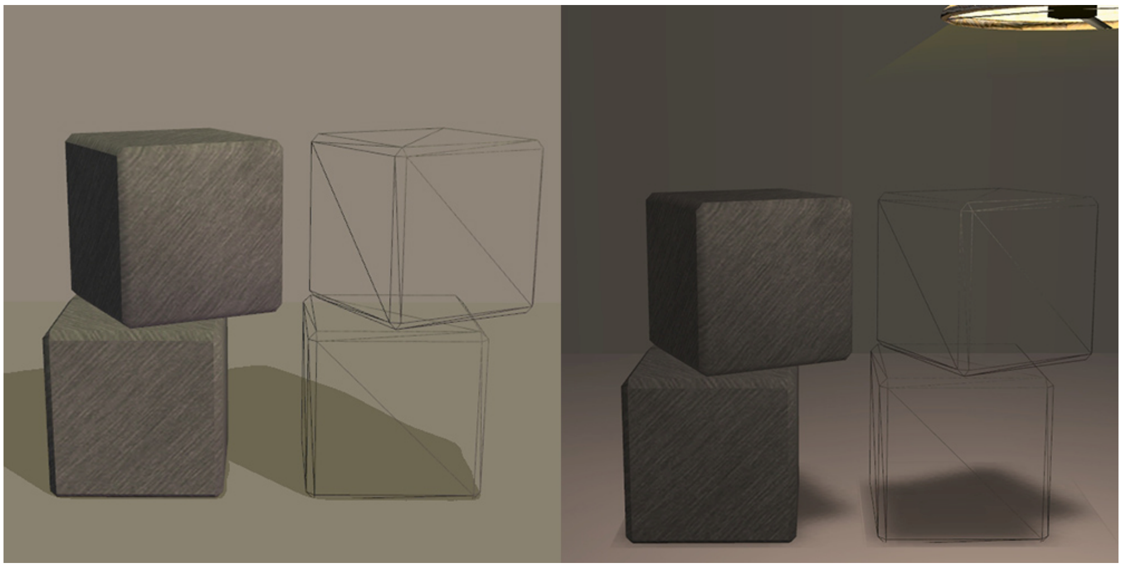
```
deprecatedStdMatInvDiffuseCoeffMultiplier: 1.2
stdMatBaseTextureAddressingU: tile
stdMatBaseTextureAddressingV: tile
stdMatBaseTextureAlphaReplicate: 0
stdMatBaseTextureEnabled: true
stdMatBaseTextureName: ##0x1C050000!DiceSculpture-c3923ec8-13a7-4d47-b
stdMatCullMode: cullClockwise
stdMatDiffCoef: 0.8,0.8,0.8
stdMatEmissiveCoef: 0,0,0
stdMatFillMode: solid
stdMatLayer: 0
stdMatLightingEnabled: 1
stdMatMinLightRangeHint: 4
stdMatSpecCoef: 0.3,0.3,0.3
stdMatSpecPower: 3
stdMatUntexturedDiffAlpha: 1
```

Default textures parameters

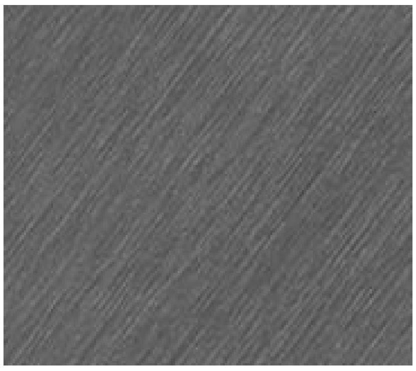
1. The first of interest is **stdMatBaseTextureEnabled.** It's a "main switch".
Its value must be set to **true**, in order to let the game display the texture. If set to **false**, the entire stdMatBaseTexture section is disabled. But it might be very convenient for checking only the mesh in game.



2. **stdMatFillMode** is the next step and another way of testing the mesh without texture.
The default setting is **solid** and is needed if you want to see the texture.
But if set to **wireframe** it will look like this:



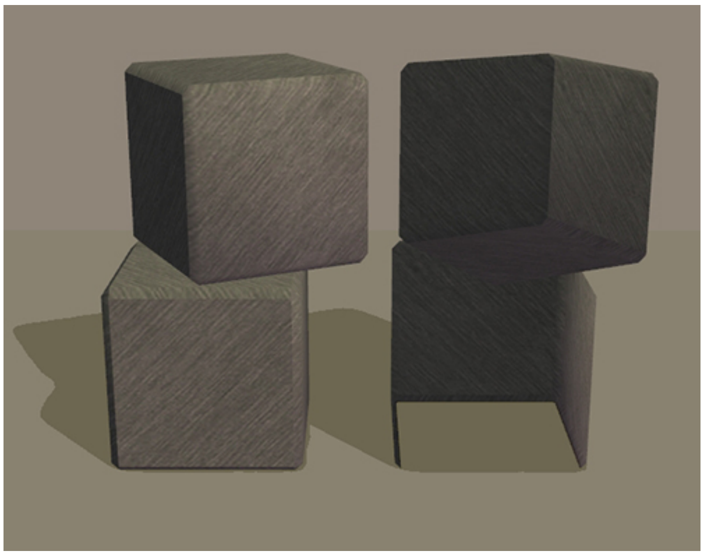
3. Once enabled, you'll need a texture to play with. The texture reference can be found in the **stdMatBaseTextureName** string field.



A sample of the .png texture used.
The full seamless texture is 512x512

4. Now, on which side of the mesh are you going to apply the texture: outside, inside, both sides?

The **stdMatCullMode** will let you decide. **cullClockwise**, **cullCounterClockwise**, **none,** are the possible values. Basically always **cullClockwise.** If you need both side of the mesh to be textured **none** can be interesting, among other things, for bottomless floor.
But it'll make your object twice heavier for the computer to render.



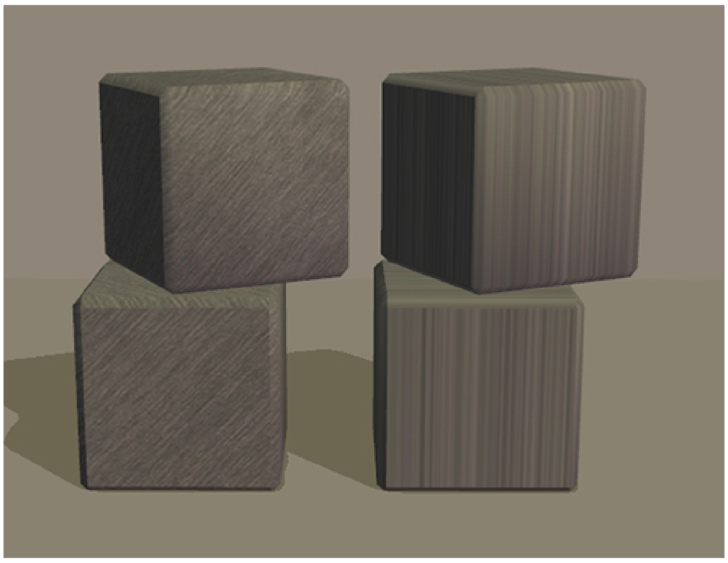
Left: **cullClockwise**
Right: **cullCounterClockwise**

Both side (**none**) not depicted here because it look the same as default.

Most useful for an open mesh (vase, barrel...)

5. The graphic engine must know how to apply the texture over the 3D mesh: the texture might be tiled over the surface, or stretched across it.

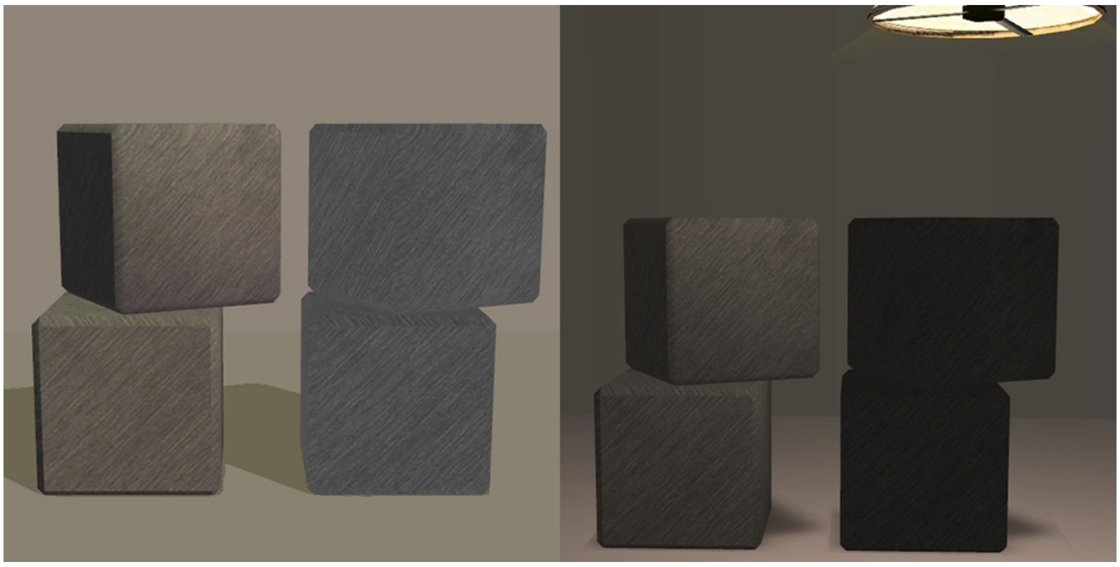
The **stdMatBaseTextureAddressingU** and **stdMatBaseTextureAddressingV** fields are suitable for the purpose.
The basic value is **tile.** Another possible value is **clamp.**



Clamp applied here on stdMatBaseTextureAddressingV

(If one is interested in knowing more about Clamp, one can Google "Clamp texture". It might have a relation with texture animation.)

6. **The next one could be stdMatLightingEnabled,** it determines if light is affecting the texture or not**.** Values can be **1** or **0**

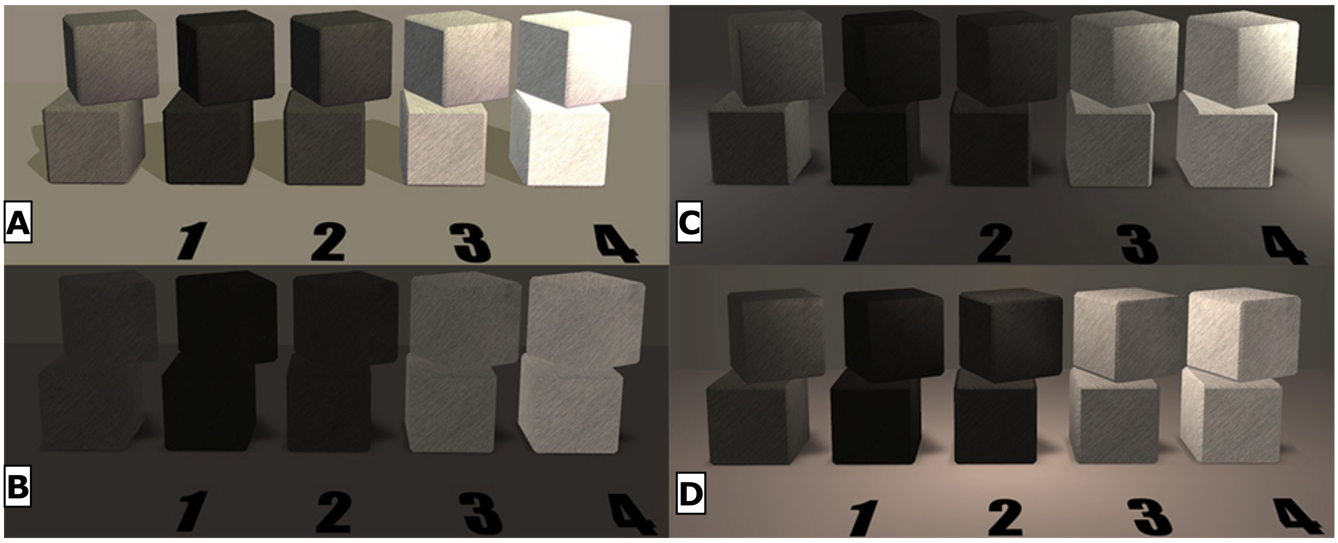


7. Then would logically be **stdMatMinLightRangeHint** with **1** / **2** / **4** as possible values when **4** is the default.
   But the different testing on that parameter alone didn't show any significant difference.

8. **stdMatDiffCoef** determines the diffusion of the texture colour. How much ambient light is affecting it.
   Dark or light? Change all the parameters in parallel.

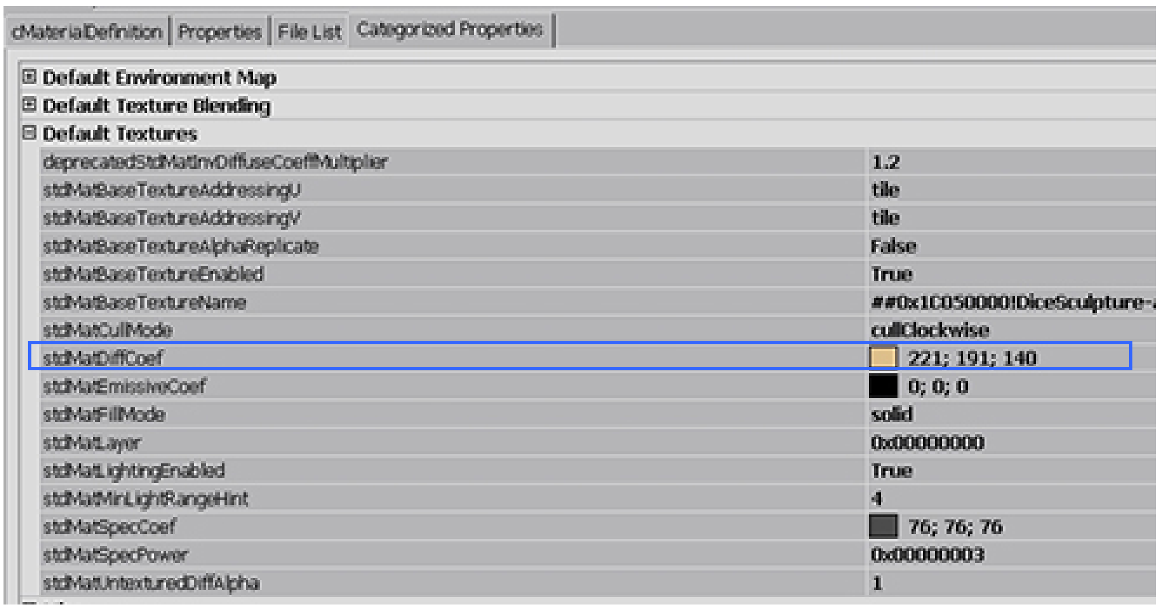
	BASE	1	2	3	4
StdMatDiffCoef	**0.8, 0.8, 0.8**	**0.3, 0.3, 0.3**	**0.5, 0.5, 0.5**	**1.6, 1.6, 1.6**	**2.4, 2.4, 2.4**

A: **Outside**, **B**: **Inside, C: Inside with window on sides, D: inside with lamp above**



stdMatDiffCoef have three values settings (0.8, 0.8, 0.8) that correspond in RGB (red, green, blue)
By modifying these separately you'll influence the tint of the diffusion.

In categorised properties tab, information are displayed in a convenient format.
A colour picker is even available.

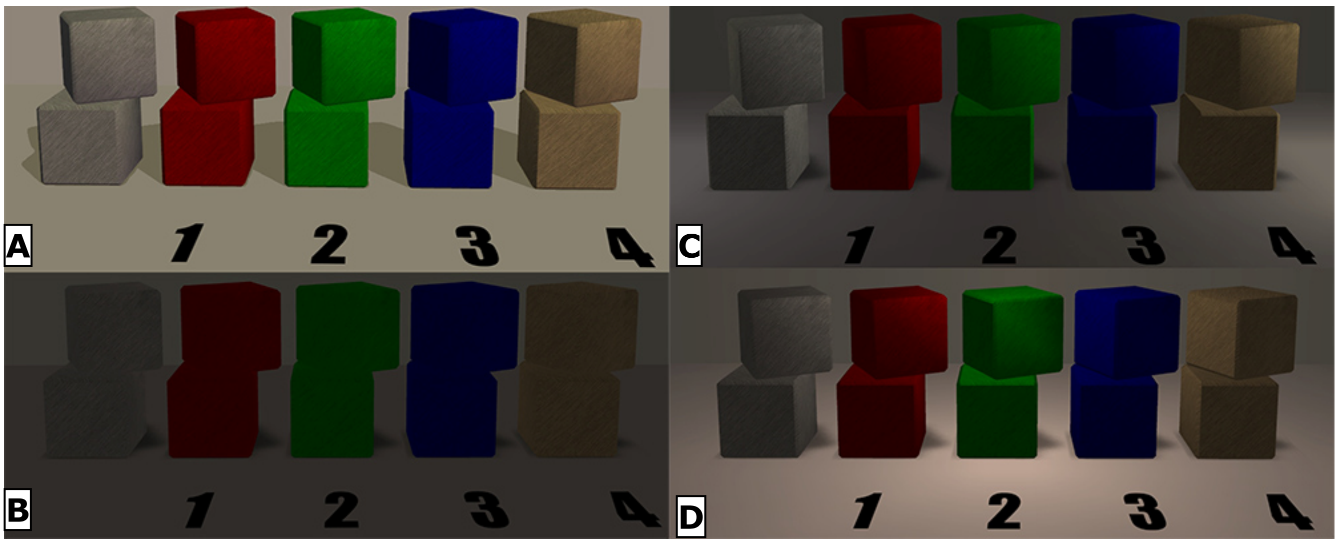


⊞ Default Environment Map
⊞ Default Texture Blending
⊟ Default Textures

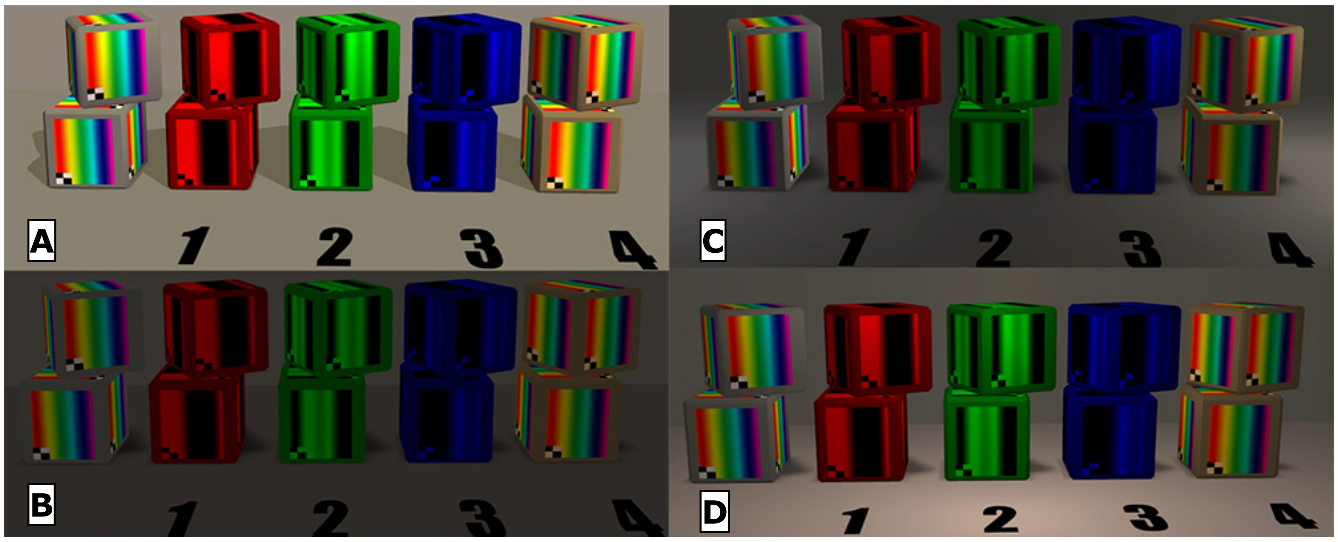
deprecatedStdMatInvDiffuseCoeffMultiplier	1.2
stdMatBaseTextureAddressingU	tile
stdMatBaseTextureAddressingV	tile
stdMatBaseTextureAlphaReplicate	False
stdMatBaseTextureEnabled	True
stdMatBaseTextureName	##0x1C050000!DiceSculpture-
stdMatCullMode	cullClockwise
stdMatDiffCoef	🟫 221; 191; 140
stdMatEmissiveCoef	⬛ 0; 0; 0
stdMatFillMode	solid
stdMatLayer	0x00000000
stdMatLightingEnabled	True
stdMatMinLightRangeHint	4
stdMatSpecCoef	⬛ 76; 76; 76
stdMatSpecPower	0x00000003
stdMatUntexturedDiffAlpha	1

	BASE	1	2	3	4
StdMatDiffCoef	0.8, 0.8, 0.8	0.8, 0, 0	0, 0.8, 0	0, 0, 0.8	0.87, 0.75, 0.55
	Balanced 204, 204,2 04	All red 204, 0, 0	All green 0, 204, 0	All blue 0, 0, 204	RGB= 221, 191, 140

A: Outside, B: Inside, C: Inside with window on sides, D: inside with lamp above



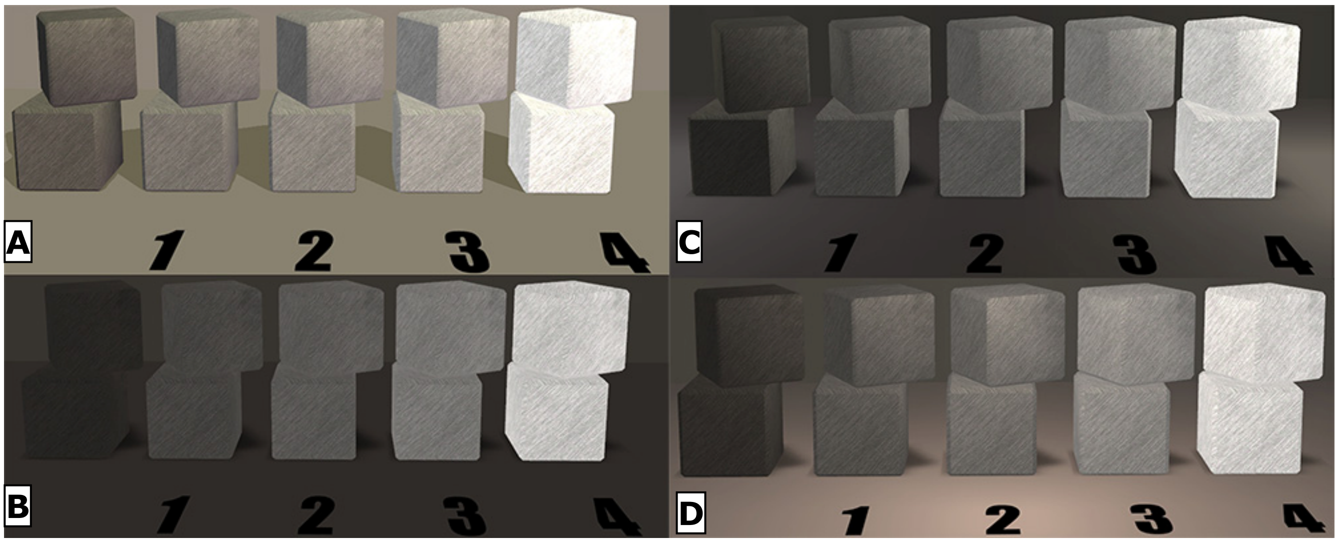
Same with coloured gradient texture



9. **stdMatEmissiveCoef** determines the emission of the texture colour.
This is the Glow-in-the-Dark effect, <u>not</u> for lighting abilities such as lamp.
All the parameters changed in parallel:

	BASE	1	2	3	4
StdMatEmissiveCoef	0, 0, 0	0.3, 0.3, 0.3	0.5, 0.5, 0.5	0.8, 0.8, 0.8	1.6, 1.6, 1.6

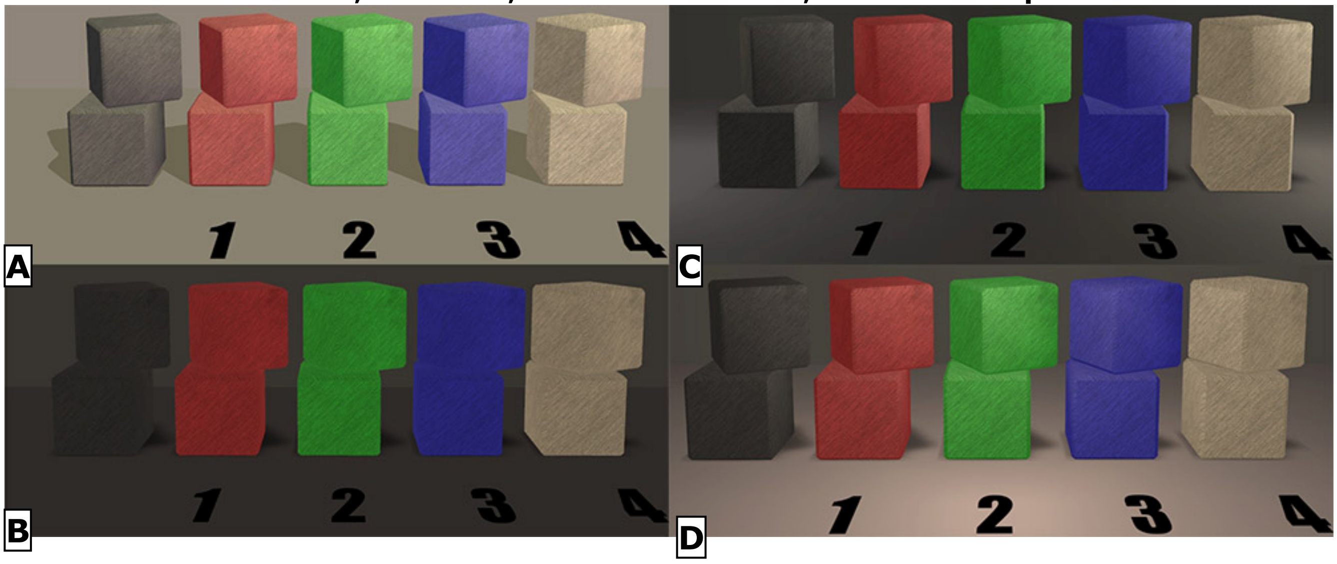
A: **Outside**, **B**: **Inside, C: Inside with window on sides, D: inside with lamp above**



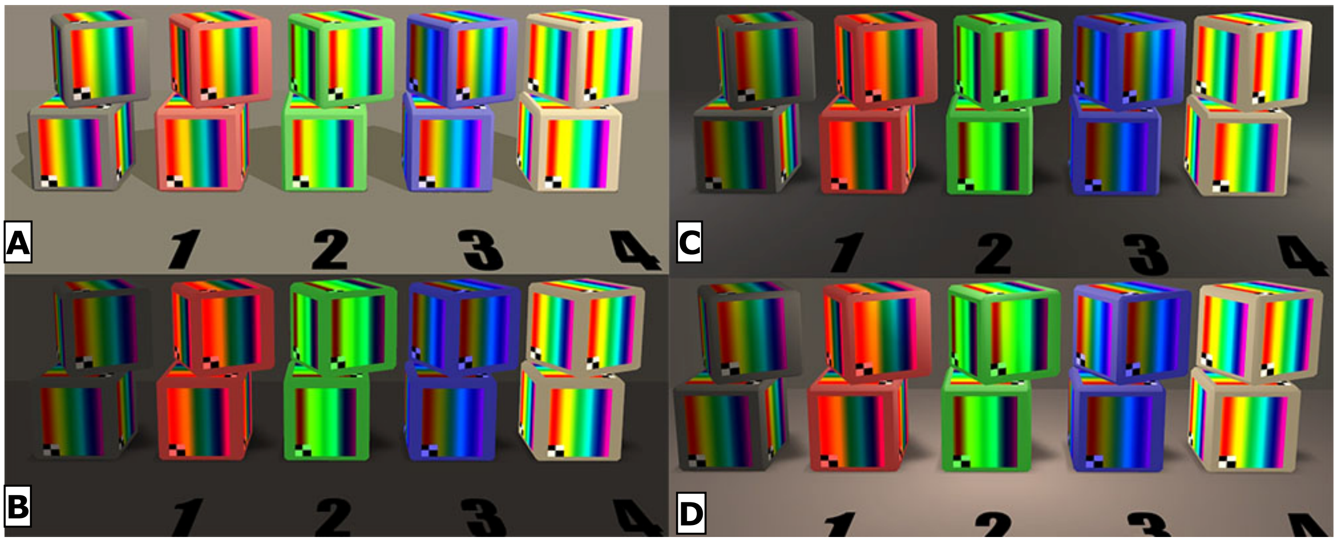
With the RGB values changed separately:

	BASE	1	2	3	4
StdMatEmissiveCoef	0.8, 0.8, 0.8	0.8, 0, 0	0, 0.8, 0	0, 0, 0.8	0.87, 0.75, 0.55
	Balanced 204, 204, 204	All red 204, 0, 0	All green 0, 204, 0	All blue 0, 0, 204	RGB= 221, 191, 140

A: Outside, B: Inside, C: Inside windows, D: Inside lamp above



Same with coloured gradient texture



By testing, the 6 next parameters don't seem to have an effect on their own.

10. **stdMatSpecPower** Values tested: 1, 2, 6, 12, 2 up to 1000. The default value is 3?
11. **stdMatSpecCoef** Either alone or in conjunction with the previous : no difference
12. **stdMatUntexturedDiffAlpha** *Overall Alpha transparency (0=invisible; 1=solid).*
13. **stdMatBaseTextureAlphaReplicate** *Whether to expect a greyscale "-alpha" texture and use that one value for all three colour channels.*

By different readings, it's suspected that they work with transparency or alpha or in conjunction with other parameters. We'll come back on these later...

14. **deprecatedStdMatInvDiffuseCoeffMultiplier** is still a mystery to me.
15. **stdMatLayer** seems to be useless, I've read somewhere that layer abilities have been abandoned by Maxis.

Let's summarize:

Default Texture parameter in Categorized properties

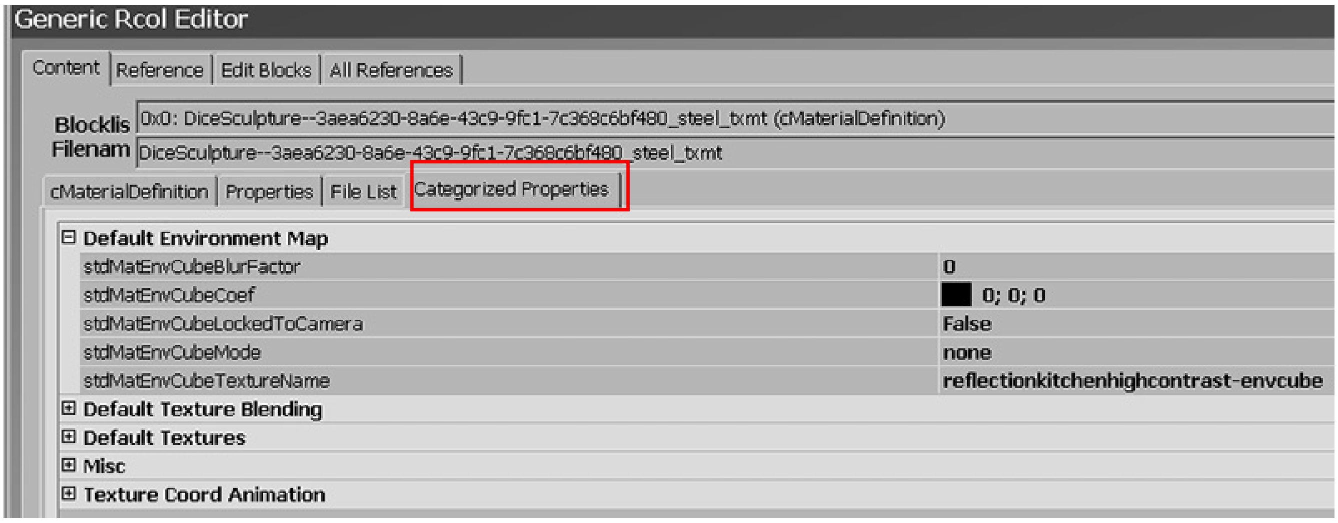
Test	Parameter	Default values	Other possible values	Notes (*in Italic notes from Wiki*)
14	deprecatedStdMatInvDiffuseCoeffMultiplier	1.2	0.0 |1.0| 1.1	?
5	stdMatBaseTextureAddressingU	tile	clamp	Tile or stretch the texture over the mesh
5	stdMatBaseTextureAddressingV	tile	clamp	Tile or stretch the texture over the mesh
13	stdMatBaseTextureAlphaReplicate	0	1	? *Whether to expect a greyscale "-alpha" texture and use that one value for all three colour channels.*
1	stdMatBaseTextureEnabled	true	false	Allows the texture to be visible
3	stdMatBaseTextureName	text	-	Refer to the base texture name
4	stdMatCullMode	cullClockwise	cullCounterClockwise, none	Apply the texture outside, inside or on both sides of the mesh
8	StdMatDiffCoef	0.8,0.8,0.8	0 =< n =< infinity	Diffusion = plain colour RGB
9	StdMatEmissiveCoef	0,0,0	0 =< r =< 1	Emissive = "inner light" RGB. Glows in the dark Fx
2	stdMatFillMode	solid	wireframe	For mesh testing
15	stdMatLayer	0	?	? Useless ?
6	stdMatLightingEnabled	1	0	Allow the light to affect the texture
7	stdMatMinLightRangeHint	4	1 | 2	? *The minimum required light intensity range this material must support*
11	stdMatSpecCoef	0.3,0.3,0.3	0 =< r =< 1	? *Specularity RGB*
10	stdMatSpecPower	3	0 =< n < 5000	? *Specularity effect strength*
12	stdMatUntexturedDiffAlpha	1	0	? *Overall Alpha transparency (0=invisible;1=solid)*

<u>**Exploring the TXMT parameters part 2**</u>

This part will study the reflection EnvCube ability. Default Environment Map section in Categorized Properties tab in SimPE

Note the reflection section is independent of the Default Texture section. <u>It means that if no texture is showed /enabled, the reflection will still be visible.</u>

5 parameters named: **stdMatEnvCubeBlurFactor, stdMatEnvCubeCoef, stdMatEnvCubeLockedToCamera, stdMatEnvCubeMode, stdMatEnvCubeTextureName**



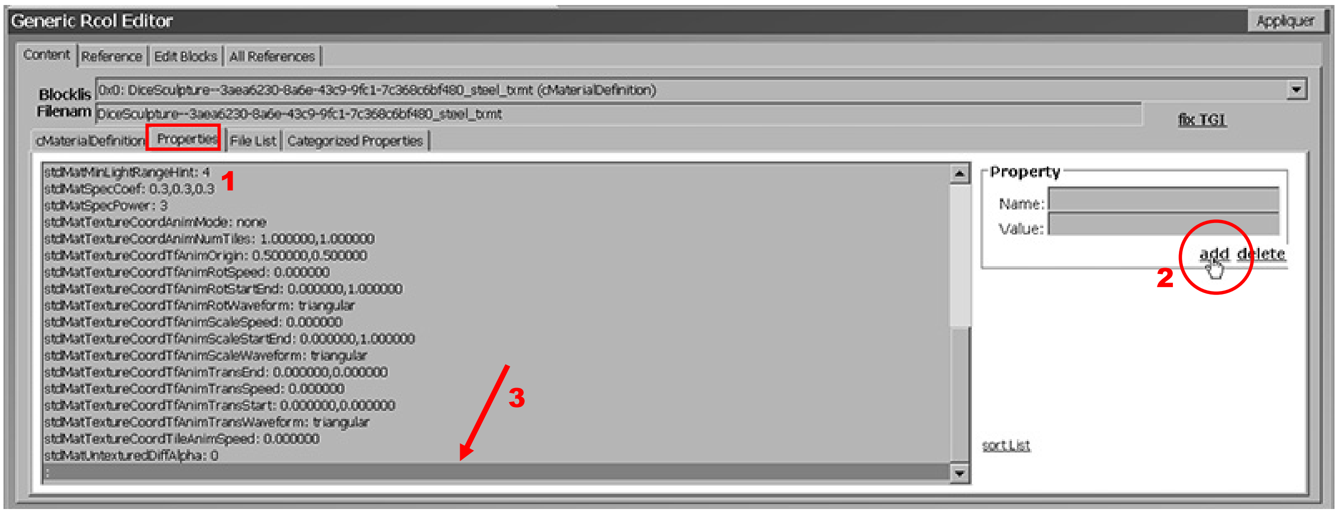
Blocklis: 0x0: DiceSculpture--3aea6230-8a6e-43c9-9fc1-7c368c6bf480_steel_txmt (cMaterialDefinition)
Filenam: DiceSculpture--3aea6230-8a6e-43c9-9fc1-7c368c6bf480_steel_txmt

cMaterialDefinition | Properties | File List | Categorized Properties

⊟ Default Environment Map	
stdMatEnvCubeBlurFactor	0
stdMatEnvCubeCoef	■ 0; 0; 0
stdMatEnvCubeLockedToCamera	False
stdMatEnvCubeMode	none
stdMatEnvCubeTextureName	reflectionkitchenhighcontrast-envcube
⊞ Default Texture Blending	
⊞ Default Textures	
⊞ Misc	
⊞ Texture Coord Animation	

If one or more parameters are missing, you'll need to add them.

Here's the procedure:
Go in Properties tab, click on add, scroll down to find the line you've just added and select it.

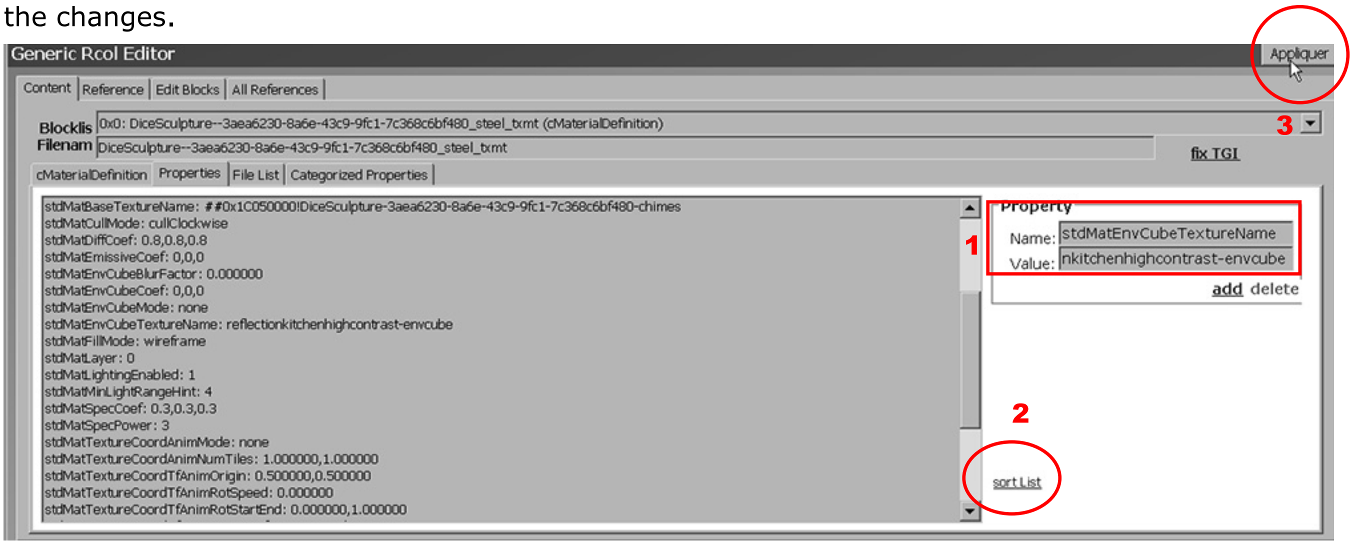


Blocklis: 0x0: DiceSculpture--3aea6230-8a6e-43c9-9fc1-7c368c6bf480_steel_txmt (cMaterialDefinition)
Filenam: DiceSculpture--3aea6230-8a6e-43c9-9fc1-7c368c6bf480_steel_txmt

cMaterialDefinition | Properties | File List | Categorized Properties

```
stdMatMinLightRangeHint: 4
stdMatSpecCoef: 0.3,0.3,0.3
stdMatSpecPower: 3
stdMatTextureCoordAnimMode: none
stdMatTextureCoordAnimNumTiles: 1.000000,1.000000
stdMatTextureCoordTfAnimOrigin: 0.500000,0.500000
stdMatTextureCoordTfAnimRotSpeed: 0.000000
stdMatTextureCoordTfAnimRotStartEnd: 0.000000,1.000000
stdMatTextureCoordTfAnimRotWaveform: triangular
stdMatTextureCoordTfAnimScaleSpeed: 0.000000
stdMatTextureCoordTfAnimScaleStartEnd: 0.000000,1.000000
stdMatTextureCoordTfAnimScaleWaveform: triangular
stdMatTextureCoordTfAnimTransEnd: 0.000000,0.000000
stdMatTextureCoordTfAnimTransSpeed: 0.000000
stdMatTextureCoordTfAnimTransStart: 0.000000,0.000000
stdMatTextureCoordTfAnimTransWaveform: triangular
stdMatTextureCoordTileAnimSpeed: 0.000000
stdMatUntexturedDiffAlpha: 0
:
```

Type the name of the missing parameter and its value. Sort the list for clear viewing and apply the changes.



Blocklis 0x0: DiceSculpture--3aea6230-8a6e-43c9-9fc1-7c368c6bf480_steel_txmt (cMaterialDefinition)
Filenam DiceSculpture--3aea6230-8a6e-43c9-9fc1-7c368c6bf480_steel_txmt

```
stdMatBaseTextureName: ##0x1C050000!DiceSculpture-3aea6230-8a6e-43c9-9fc1-7c368c6bf480-chimes
stdMatCullMode: cullClockwise
stdMatDiffCoef: 0.8,0.8,0.8
stdMatEmissiveCoef: 0,0,0
stdMatEnvCubeBlurFactor: 0.000000
stdMatEnvCubeCoef: 0,0,0
stdMatEnvCubeMode: none
stdMatEnvCubeTextureName: reflectionkitchenhighcontrast-envcube
stdMatFillMode: wireframe
stdMatLayer: 0
stdMatLightingEnabled: 1
stdMatMinLightRangeHint: 4
stdMatSpecCoef: 0.3,0.3,0.3
stdMatSpecPower: 3
stdMatTextureCoordAnimMode: none
stdMatTextureCoordAnimNumTiles: 1.000000,1.000000
stdMatTextureCoordTfAnimOrigin: 0.500000,0.500000
stdMatTextureCoordTfAnimRotSpeed: 0.000000
stdMatTextureCoordTfAnimRotStartEnd: 0.000000,1.000000
```

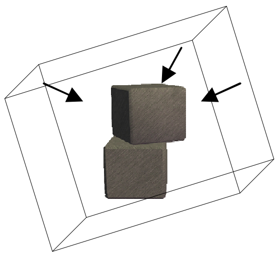
Name: stdMatEnvCubeTextureName
Value: nkitchenhighcontrast-envcube

Envcube reflection is a feature visible from Neighbourhood view.

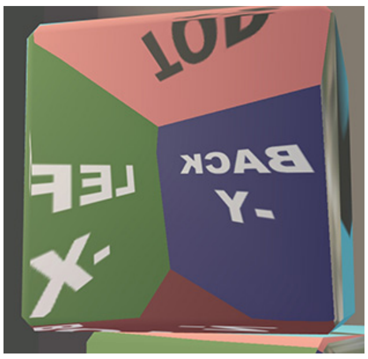
First we need to switch on the EnvCube section.

1. **stdMatEnvCubeMode** needs to be set on **reflection**.

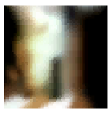
Once enabled, a virtual cube is drawn all around the object.
The texture warping this cube will be projected on the object as a reflection.



Playing with reflectioncubetemplate-envcube can give a good idea of how EnvCube is working.



For the following tests reflectionkitchenhighcontrast-envcube will be used.



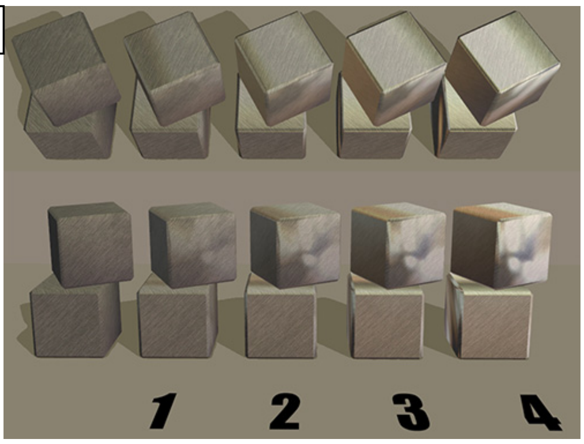
reflectionkitchenhighcontrast-envcube
reflection texture 64x64

2. stdMatEnvCubeCoef

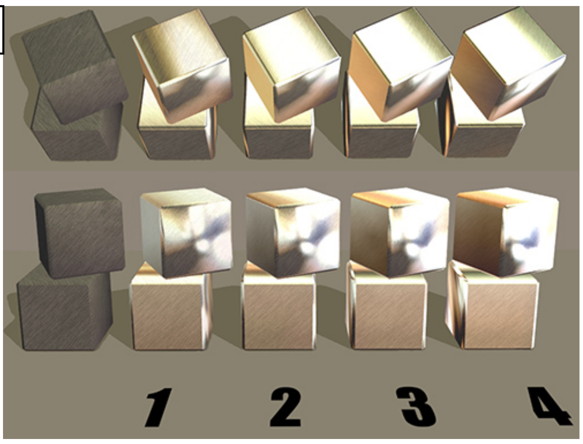
When the 3 values are changed in parallel; more or less reflection is added.

A	BASE	1	2	3	4
stdMatEnvCubeCoef	0, 0, 0	0.1, 0.1, 0.1	0.2, 0.2, 0.2	0.3, 0.3, 0.3	0.4, 0.4, 0.4

A



B

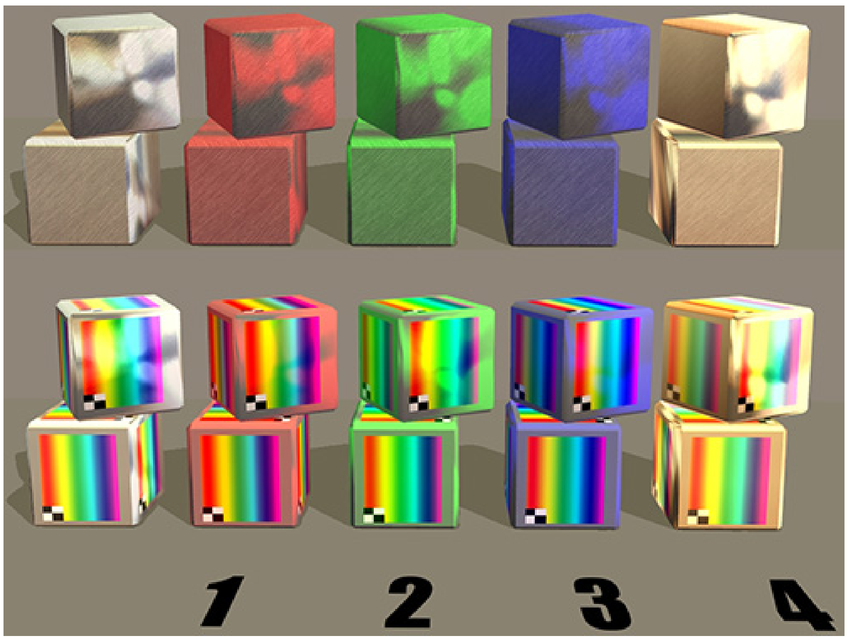


B	BASE	1	2	3	4
stdMatEnvCubeCoef	0, 0, 0	0.6, 0.6, 0.6	0.7, 0.7, 0.7	0.8, 0.8, 0.8	1, 1, 1

Of course, it's possible to change the 3 values separately to tint the reflection.
Again the 3 values correspond to RGB (red, Green, Blue)

reflectionkitchenhighcontrast-envcube

	BASE	1	2	3	4
stdMatEnvCubeCoef	0.5, 0.5, 0.5	0.5, 0, 0	0, 0.5, 0	0, 0, 0.5	0.87, 0.75, 0.55
RGB=	neutral 76, 76, 76	red 76, 0, 0	green 0, 76, 0	blue 0, 0, 76	Gold 221, 191, 140



3. stdMatEnvCubeBlurFactor

Add some blur to the reflection. At high setting it will also blur and "eat" the texture of the object.

The known range value is: 0 =< r < 5

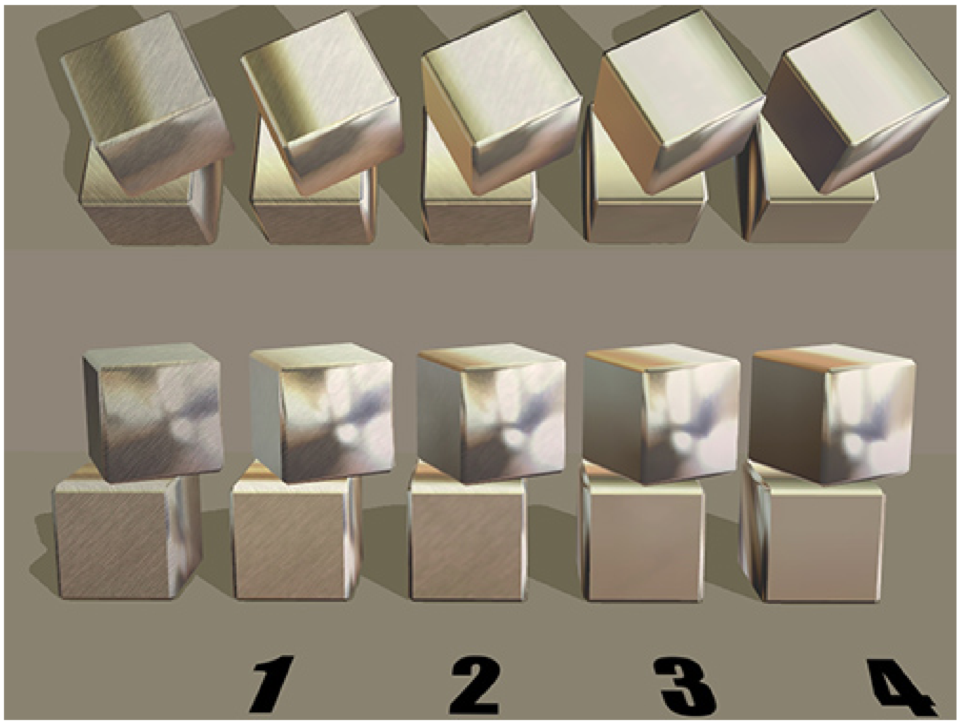
The more blur, the more material will appear polished.

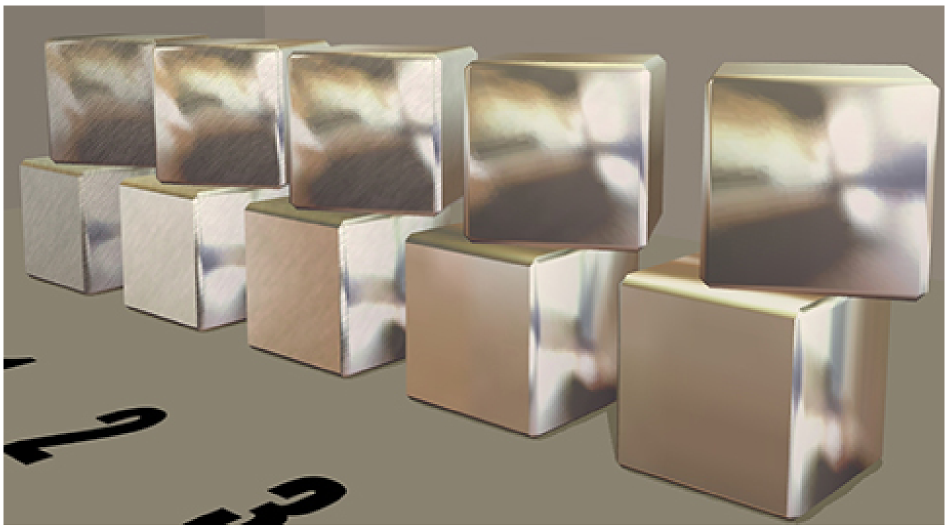
For demonstration purpose, values have been tested quite high.

Very fine values are allowed (e.g. 0.0385)

reflectionkitchenhighcontrast-envcube

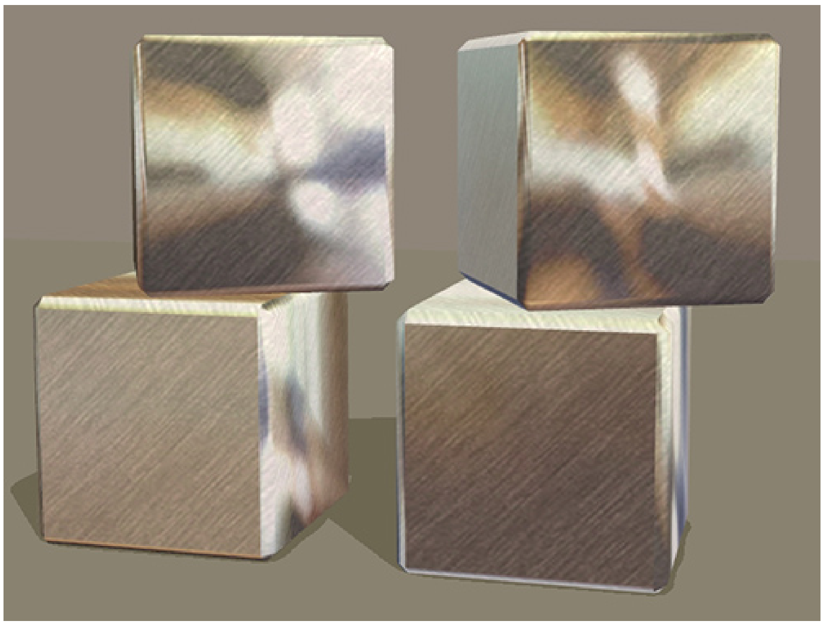
	BASE	1	2	3	4
stdMatEnvCubeCoef	0.5, 0.5, 0.5	0.5, 0.5, 0.5	0.5, 0.5, 0.5	0.5, 0.5, 0.5	0.5, 0.5, 0.5
stdMatEnvCubeBlurCoef	**0.000000**	**0.5**	**1.0**	**3.0**	**5**





4. stdMatEnvCubeLockedToCamera

Although the name is self-explanatory, it's difficult to describe the difference when you lock the EnvCube to the camera or not. The warping is differently oriented. The reflection is static in camera mode when the mouse used for moving if set on **0**.
Both possibilities are to be tested, results depending also from the mesh.



Left : **0**
Right :**1**

5. stdMatEnvCubeTextureName

Till now, the **reflectionkitchenhighcontrast-envcube** was used as reflection map as it is one of the most commonly used. But there are more, quite some in fact.
Any game texture can be used as reflection map: Textures.package in (C:\Program Files\EA GAMES\The Sims 2\TSData\Res\Sims3D)

You can even make your own.

The ones used in the game are mostly 64 x 64 in size, although the "outdoor" one used on the cars is 128 x 128, and the Maxis template is 256 x 256.
The game may be set to only use a square image of the standard image sizes (64 x 64, 128 x 128, 256 x 256, 512 x 512).

The image itself is rotated 90 degrees to the left on the Maxis ones.

Keep in mind that EnvCube reflection effect is fully revealed when the camera is moving.

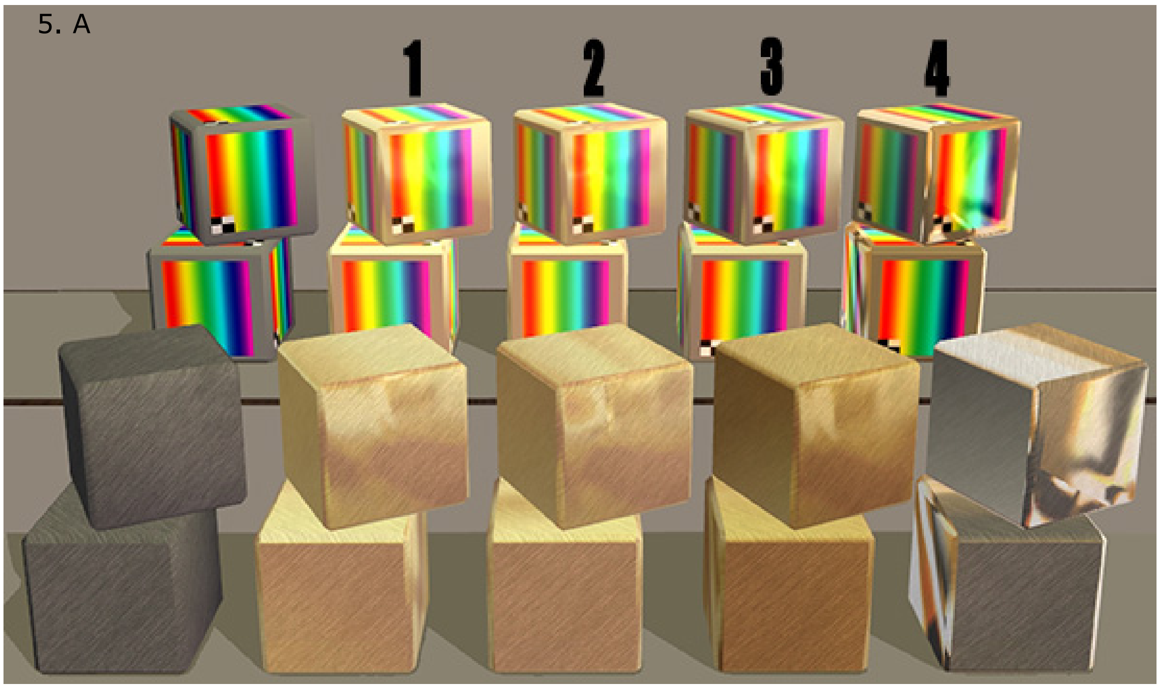
A discreet effect with low values gives often a better result.

For testing the different **stdMatEnvCubeTextureName,** these settings were constant through the experiment

stdMatEnvCubeCoef	0.5, 0.5, 0.5
stdMatEnvCubeBlurCoef	0.000000
stdMatEnvCubeLockedToCamera	1

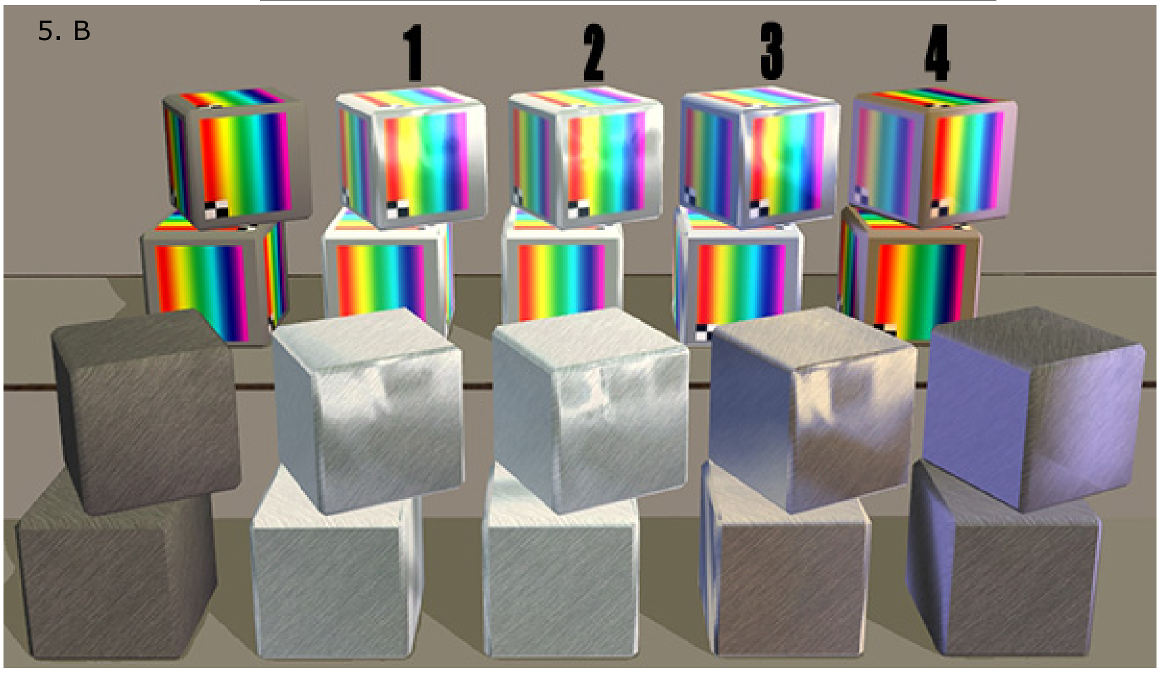
5. A

1	reflectiongold-envcube
2	reflectiongoldnonlit-envcube
3	reflectiondarkgold-envcube
4	fencemoroccangate-envcube



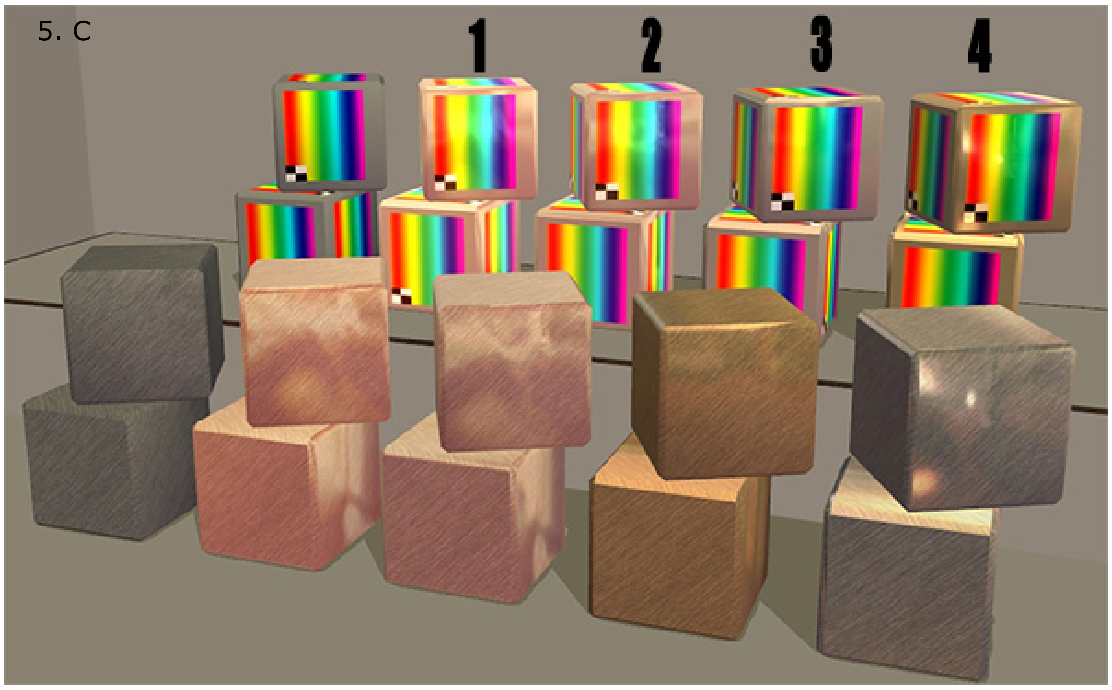
5. B

1	reflectionsilver-envcube
2	reflectionsilvernonlit-envcube
3	reflectionkitchenhighcontrastonyx-envcube
4	sidebluelights-envcube



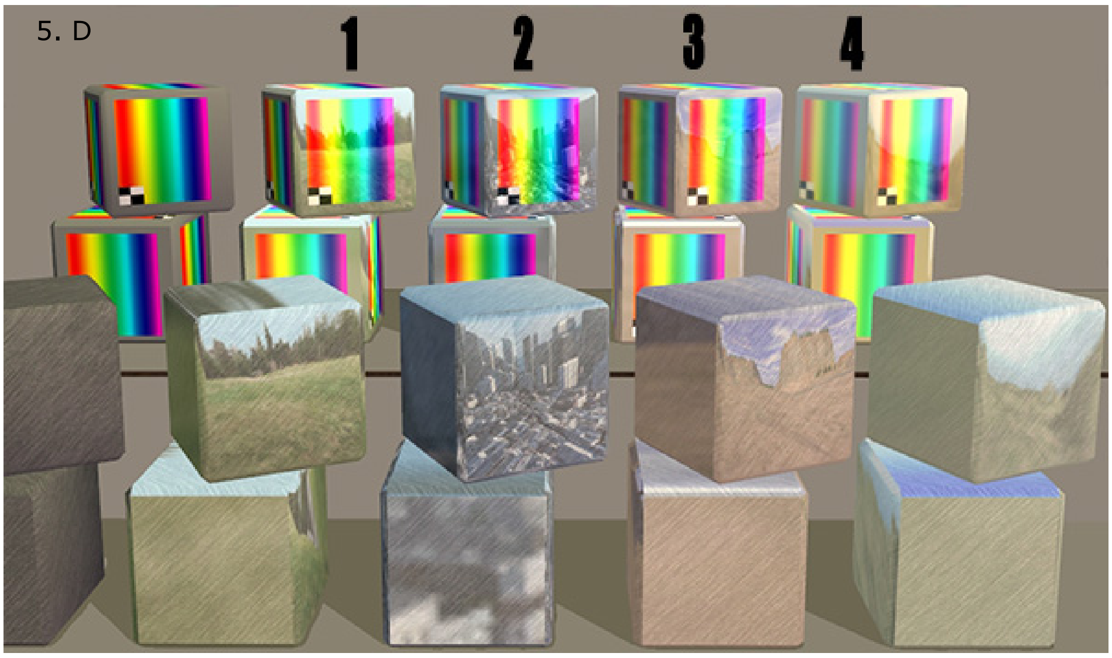
5. C

1	reflectionbronze-envcube
2	reflectionbronzenonlit-envcube
3	gothlivingroom_01-envcube
4	reflectionsparking-envcube



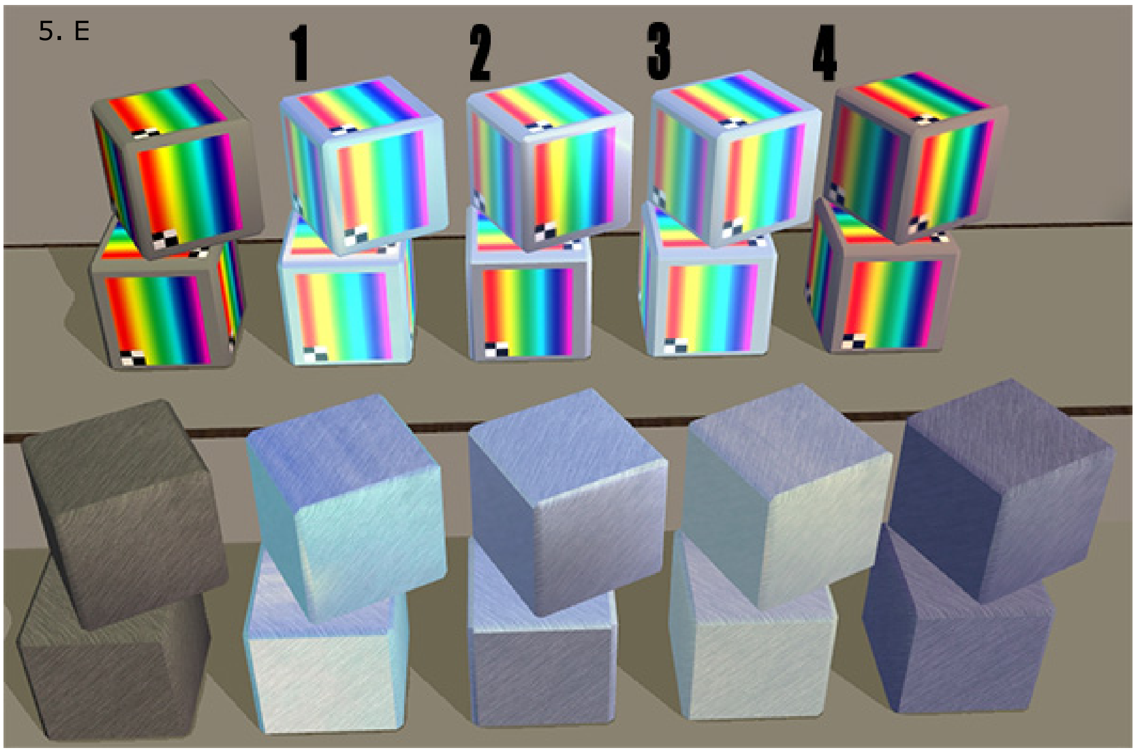
5. D

1	outdoordaytime-envcube
2	reflectioncubecityscape-envcube
3	neighborhooddesert-envcube
4	reflectionoutdoorwater-envcube



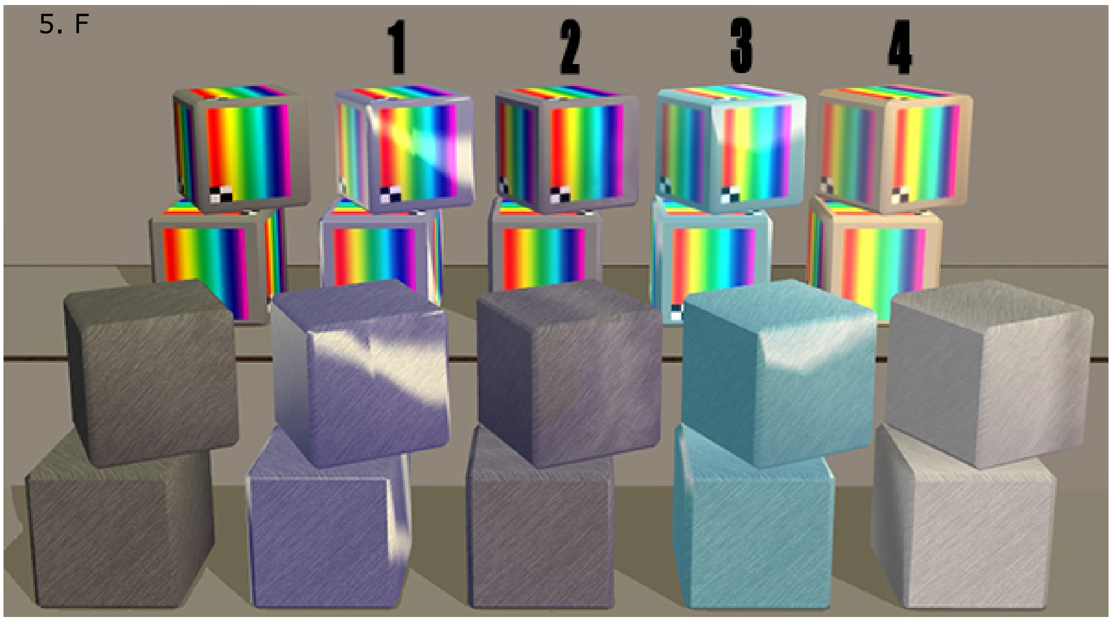
5. E

1	neighborhood-sky2-envcube
2	skymap-generic-day-envcube
3	skymap-generic-day-highres-envcube
4	skymap-generic-night-envcube



5. F

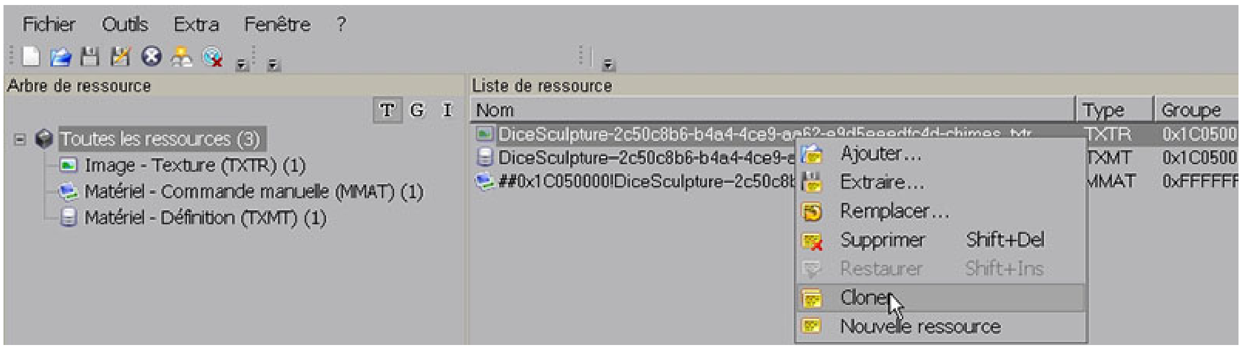
1	outdoornighttime-envcube
2	nightreflection2-envcube
3	swimming_pool-envcube
4	shineymetal1 *(game texture)*



6. Adding a custom Envcube

While it's quite easy to create and add custom an Envcube, predict and control the result isn't.

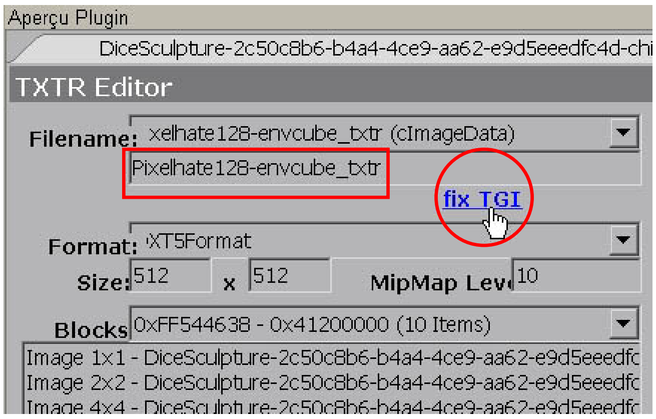
1. In your package, select the Image-Texture (TXTR) and clone it.



Toutes les ressources (3)
- Image - Texture (TXTR) (1)
- Matériel - Commande manuelle (MMAT) (1)
- Matériel - Définition (TXMT) (1)

Nom	Type	Groupe
DiceSculpture-2c50c8b6-b4a4-4ce9-aa62-e9d5eeedfc4d-chimes_txtr	TXTR	0x1C0500
DiceSculpture–2c50c8b6-b4a4-4ce9-a	TXMT	0x1C0500
##0x1C050000!DiceSculpture–2c50c8	MMAT	0xFFFFFF

2. Select the clone, rename it ***with* _txtr *suffix***,
Fix TGI and commit to apply the change.

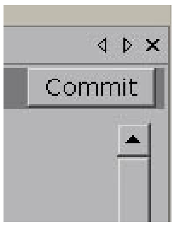


DiceSculpture-2c50c8b6-b4a4-4ce9-aa62-e9d5eeedfc4d-chi

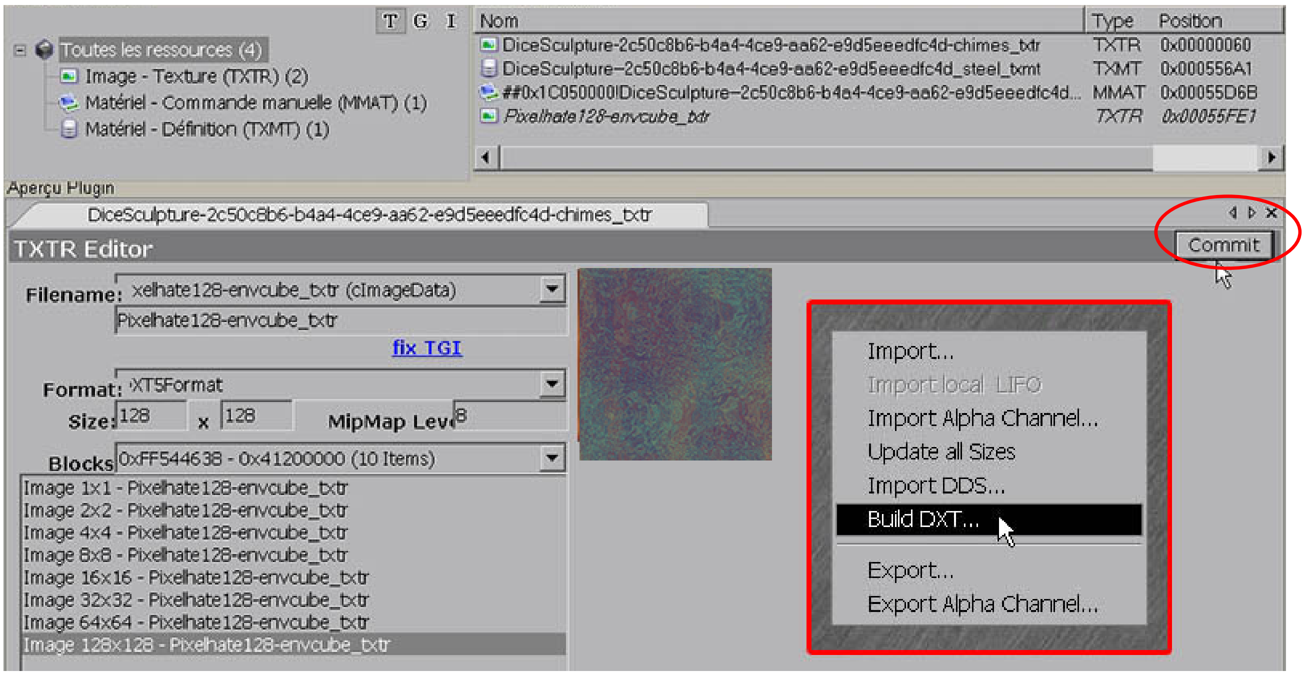
Filename: xelhate128-envcube_txtr (cImageData)
Pixelhate128-envcube_txtr

Format: XT5Format
Size: 512 x 512 MipMap Lev 10
Blocks 0xFF544638 - 0x41200000 (10 Items)

Image 1x1 - DiceSculpture-2c50c8b6-b4a4-4ce9-aa62-e9d5eeedfc
Image 2x2 - DiceSculpture-2c50c8b6-b4a4-4ce9-aa62-e9d5eeedfc
Image 4x4 - DiceSculpture-2c50c8b6-b4a4-4ce9-aa62-e9d5eeedfc



3. Use the Build DXT function to bring your custom image, Commit to apply the change.



Toutes les ressources (4)
- Image - Texture (TXTR) (2)
- Matériel - Commande manuelle (MMAT) (1)
- Matériel - Définition (TXMT) (1)

Nom	Type	Position
DiceSculpture-2c50c8b6-b4a4-4ce9-aa62-e9d5eeedfc4d-chimes_txtr	TXTR	0x00000060
DiceSculpture–2c50c8b6-b4a4-4ce9-aa62-e9d5eeedfc4d_steel_txmt	TXMT	0x000556A1
##0x1C050000!DiceSculpture–2c50c8b6-b4a4-4ce9-aa62-e9d5eeedfc4d...	MMAT	0x00055D6B
Pixelhate128-envcube_txtr	*TXTR*	*0x00055FE1*

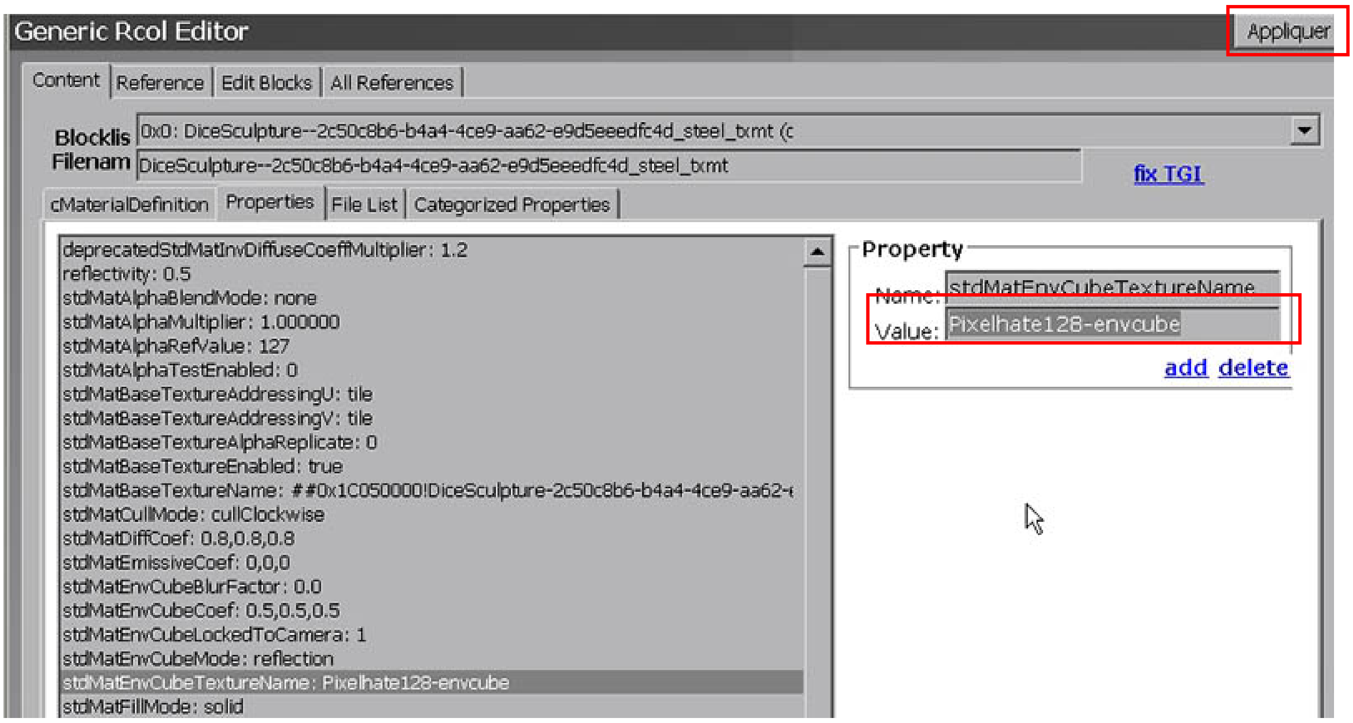
DiceSculpture-2c50c8b6-b4a4-4ce9-aa62-e9d5eeedfc4d-chimes_txtr

Filename: xelhate128-envcube_txtr (cImageData)
Pixelhate128-envcube_txtr

Format: XT5Format
Size: 128 x 128 MipMap Lev 8
Blocks 0xFF544638 - 0x41200000 (10 Items)

Image 1x1 - Pixelhate128-envcube_txtr
Image 2x2 - Pixelhate128-envcube_txtr
Image 4x4 - Pixelhate128-envcube_txtr
Image 8x8 - Pixelhate128-envcube_txtr
Image 16x16 - Pixelhate128-envcube_txtr
Image 32x32 - Pixelhate128-envcube_txtr
Image 64x64 - Pixelhate128-envcube_txtr
Image 128x128 - Pixelhate128-envcube_txtr

4. In Material Definition (TXMT) **stdMatEnvCubeTextureName,** change the value with your custom one ***without* _txtr *suffix*.** Apply the change and save.



Generic Rcol Editor

Appliquer

Content | Reference | Edit Blocks | All References

Blocklis 0x0: DiceSculpture--2c50c8b6-b4a4-4ce9-aa62-e9d5eeedfc4d_steel_txmt (c

Filenam DiceSculpture--2c50c8b6-b4a4-4ce9-aa62-e9d5eeedfc4d_steel_txmt

fix TGI

cMaterialDefinition | Properties | File List | Categorized Properties

```
deprecatedStdMatInvDiffuseCoeffMultiplier: 1.2
reflectivity: 0.5
stdMatAlphaBlendMode: none
stdMatAlphaMultiplier: 1.000000
stdMatAlphaRefValue: 127
stdMatAlphaTestEnabled: 0
stdMatBaseTextureAddressingU: tile
stdMatBaseTextureAddressingV: tile
stdMatBaseTextureAlphaReplicate: 0
stdMatBaseTextureEnabled: true
stdMatBaseTextureName: ##0x1C050000!DiceSculpture-2c50c8b6-b4a4-4ce9-aa62-
stdMatCullMode: cullClockwise
stdMatDiffCoef: 0.8,0.8,0.8
stdMatEmissiveCoef: 0,0,0
stdMatEnvCubeBlurFactor: 0.0
stdMatEnvCubeCoef: 0.5,0.5,0.5
stdMatEnvCubeLockedToCamera: 1
stdMatEnvCubeMode: reflection
stdMatEnvCubeTextureName: Pixelhate128-envcube
stdMatFillMode: solid
```

Property
Name: stdMatEnvCubeTextureName
Value: Pixelhate128-envcube

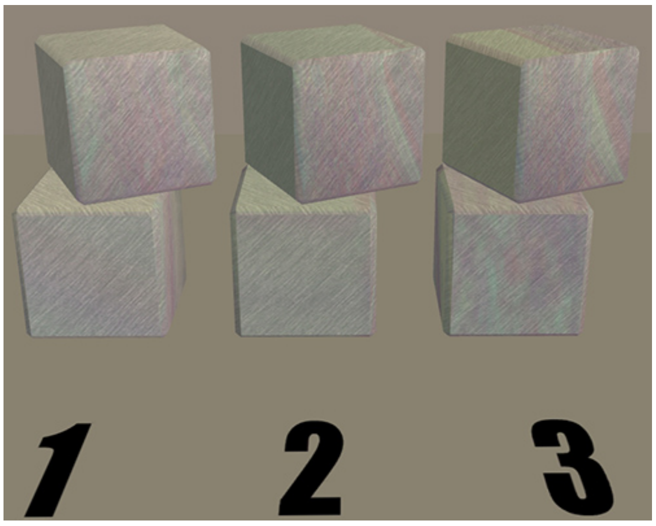
add delete

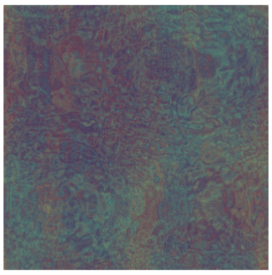
Note that, once a custom EnvCube is attached to a file and loaded in game, it can be called for any other object, just by its name.

Quick pseudo repository system
Create a package with only a texture, drop it in your download folder, replace EnvCube name in any object with the new one. Et voilà !

Custom EnvCube sizes

	1	**2**	**3**
Custom EnvCube	**64 x 64**	**128 x 128**	**256 x256**





With glass transparency (See part 3)

Default Environment Map in Categorized properties

Test	Parameter	Default values	Other possible values	Notes (*in Italic notes from Wiki*)
3	stdMatEnvCubeBlurFactor	0.000000	0 =< r < 5	*Blurriness of environment cube reflection*
2	stdMatEnvCubeCoef	0.0, 0.0, 0.0	0 =< r =< 1	*Reflection RGB (in conjunction with Reflection texture)*
4	stdMatEnvCubeLockedToCamera	0	1	*1 = the Reflection texture moves along with the camera*
1	stdMatEnvCubeMode	reflection	none	Enables the Reflection section
5	stdMatEnvCubeTextureName	-	texture filename without "_txtr"	See the list below

EnvCube TextureName

			EP
5A1	reflectiongold-envcube	Light gold	BG
5A2	reflectiongoldnonlit-envcube	Medium tone gold	BG
5A3	reflectiondarkgold-envcube	Darker gold	BG
5A4	fencemoroccangate-envcube	Orange	BG
5B1	reflectionsilver-envcube	Silver	BG
5B2	reflectionsilvernonlit-envcube	Darker silver	BG
1	reflectionkitchenhighcontrast-envcube	Glassy/metallic	BG
5B3	reflectionkitchenhighcontrastonyx-envcube	Glassy/metallic blue	NL
5B4	sidebluelights-envcube	Mauve/blue	BG
5C1	reflectionbronze-envcube	Bronze	BG
5C2	reflectionbronzenonlit-envcube	Darker bronze	BG
5C3	gothlivingroom_01-envcube	Copper	BG
5C4	reflectionsparking-envcube	Dark with orange spark	BG
5D1	outdoordaytime-envcube	Outdoor scape	BG
5D2	reflectioncubecityscape-envcube	City scape	NL
5D3	neighborhooddesert-envcube	Desert scape	BG
5D4	reflectionoutdoorwater-envcube	Outdoor with water	BG
5E1	neighborhood-sky2-envcube	Sky texture Blue	BG
5E2	skymap-generic-day-envcube	Blue	BG
5E3	skymap-generic-day-highres-envcube	Blue with shade	BG
5E4	skymap-generic-night-envcube	Dark blue	BG
5F1	outdoornighttime-envcube	Blue with white spark	BG
5F2	nightreflection2-envcube	Night blue	BG
5F3	swimming_pool-envcube	Water blue	BG
1	reflectioncubetemplate-envcube	Test purpose	BG

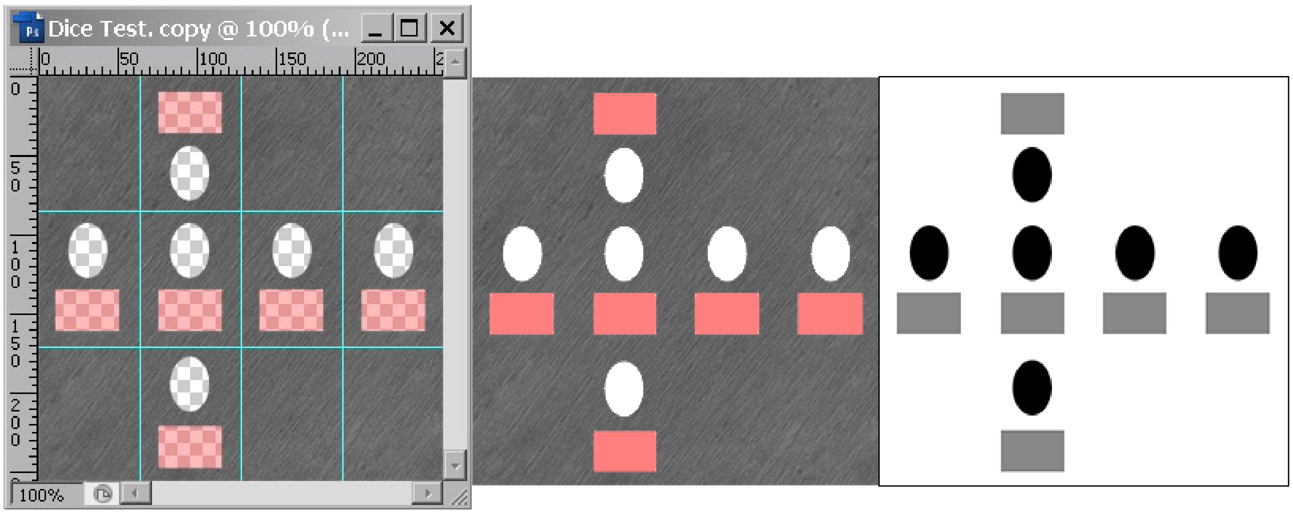
Exploring the TXMT parameters part 3

This part covers: transparency and alpha, glass effect and invisibility

Alpha channel and Transparency

A texture in .Png format is composed from 3 colour channel RGB (red, green, blue) and 1 alpha channel. The Alpha channel manages the transparency of the texture.
If the texture is opaque, with no transparency, the alpha channel will appear white.
If the texture is totally transparent, the alpha channel will appear black.
Semi transparent shades will be displayed in greyscale.

Here, ellipses are totally transparent, red rectangles are 50% opacity.

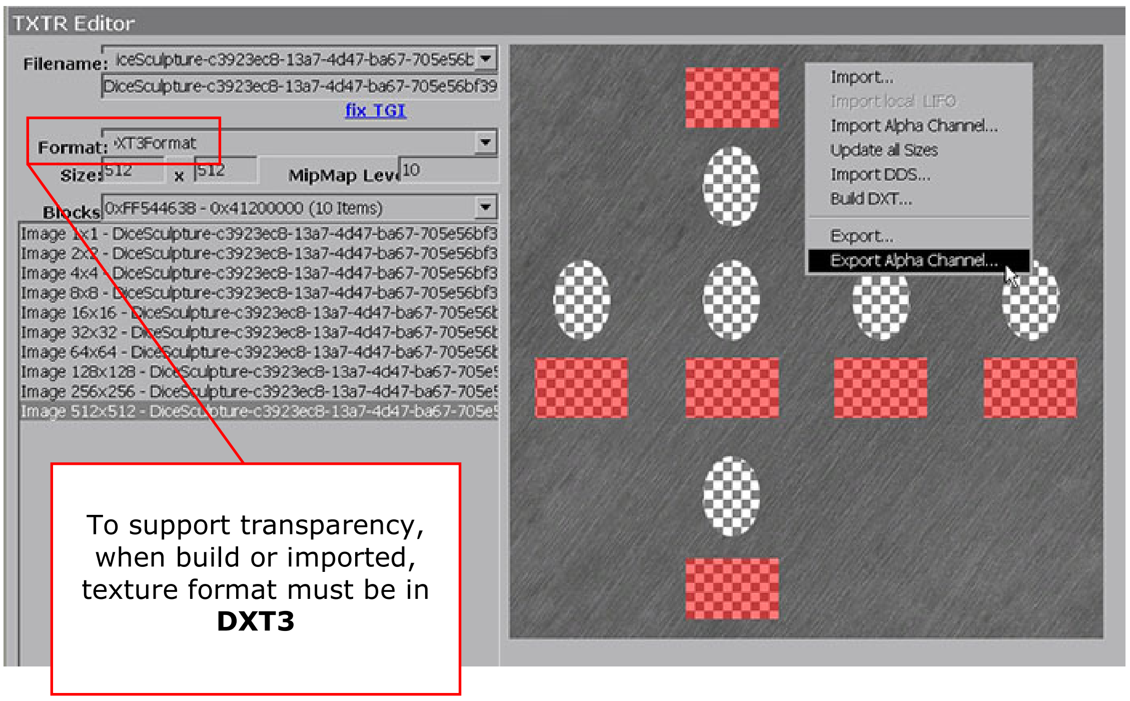


In Graphic program

In .png format

Alpha channel

By Right clicking on texture, you can export your alpha channel from SimPE, modify it in your graphic app. and import back in SimPE. (Always commit and save!)



Filename: iceSculpture-c3923ec8-13a7-4d47-ba67-705e56b
DiceSculpture-c3923ec8-13a7-4d47-ba67-705e56bf39

Format: XT3Format

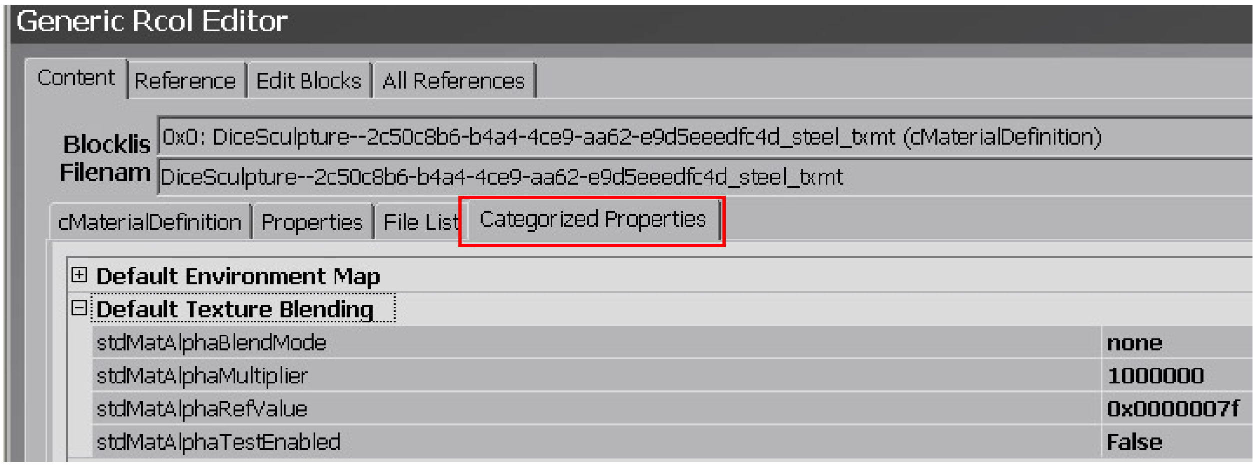
Size: 512 x 512 MipMap Leve 10

Blocks 0xFF54463B - 0x41200000 (10 Items)

```
Image 1x1 - DiceSculpture-c3923ec8-13a7-4d47-ba67-705e56bf3
Image 2x2 - DiceSculpture-c3923ec8-13a7-4d47-ba67-705e56bf3
Image 4x4 - DiceSculpture-c3923ec8-13a7-4d47-ba67-705e56bf3
Image 8x8 - DiceSculpture-c3923ec8-13a7-4d47-ba67-705e56bf3
Image 16x16 - DiceSculpture-c3923ec8-13a7-4d47-ba67-705e56b
Image 32x32 - DiceSculpture-c3923ec8-13a7-4d47-ba67-705e56b
Image 64x64 - DiceSculpture-c3923ec8-13a7-4d47-ba67-705e56b
Image 128x128 - DiceSculpture-c3923ec8-13a7-4d47-ba67-705e5
Image 256x256 - DiceSculpture-c3923ec8-13a7-4d47-ba67-705e5
Image 512x512 - DiceSculpture-c3923ec8-13a7-4d47-ba67-705e5
```

To support transparency, when build or imported, texture format must be in **DXT3**

The Default Texture blending parameters section in the Categorized Properties in TXMT.


Generic Rcol Editor

Content | Reference | Edit Blocks | All References

Blocklis 0x0: DiceSculpture--2c50c8b6-b4a4-4ce9-aa62-e9d5eeedfc4d_steel_txmt (cMaterialDefinition)

Filenam DiceSculpture--2c50c8b6-b4a4-4ce9-aa62-e9d5eeedfc4d_steel_txmt

cMaterialDefinition | Properties | File List | Categorized Properties

⊞ Default Environment Map

⊟ Default Texture Blending

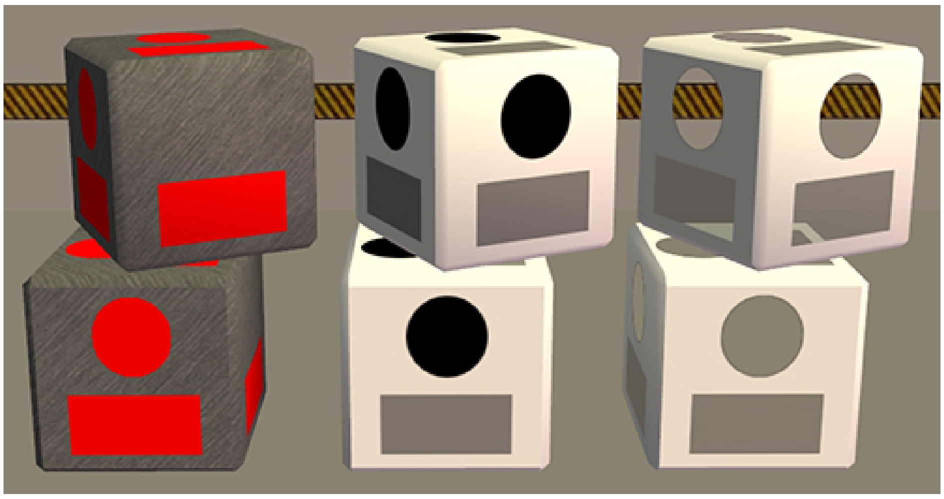
stdMatAlphaBlendMode	none
stdMatAlphaMultiplier	1000000
stdMatAlphaRefValue	0x0000007f
stdMatAlphaTestEnabled	False

4 parameters, to which I'm going to add 2 more from the Default Textures category, with Alpha in their names. (Tested in part 1 with no results).

stdMatBaseTextureAlphaReplicate
stdMatUntexturedDiffAlpha

For comfortable viewing, shadows have been turned off

1. **stdMatBaseTextureAlphaReplicate**
   Values can be **0** (default) or **1** (shows the alpha channel while hiding the texture).



In Mode Blend

2. **stdMatAlphaBlendMode**
   Decides how the Alpha channel (transparency) of the texture is managed.
   Values can be **none**, **blend**, **additive** or **addNoSrcAlphaScale**.

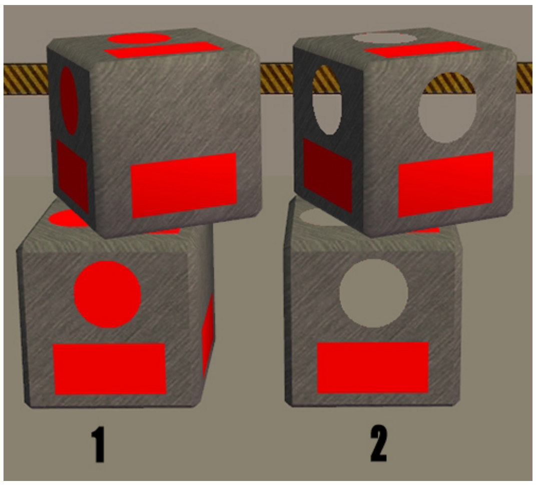
   Default value is **none** and is used when there's no transparency in texture.
   BUT none can also show binary transparency.

None

3. stdMatAlphaTestEnabled (in none mode)

Declares that the texture uses an Alpha channel. Values can **0** (default) or **1**

	1	**2**
stdMatAlphaBlendMode	**none**	**none**
stdMatAlphaTestEnabled	**0**	**1**



1. **0** No transparency is shown.

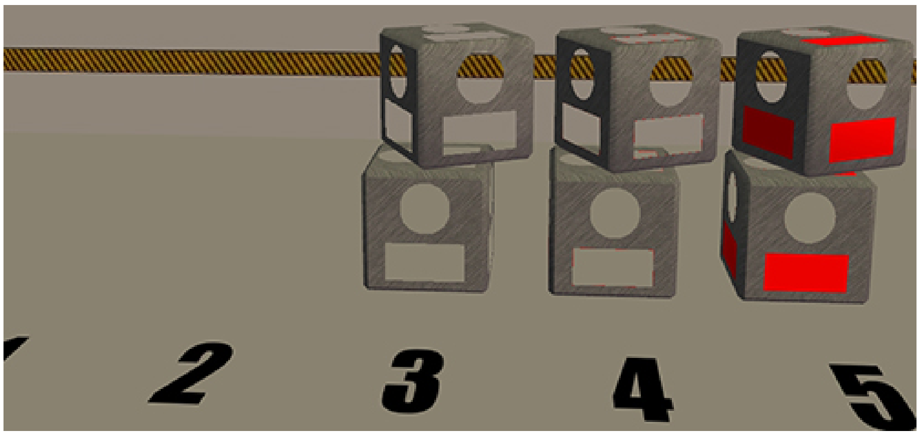
2. **1** Uses binary alpha.
(No shades of grey), texture is either there or isn't.

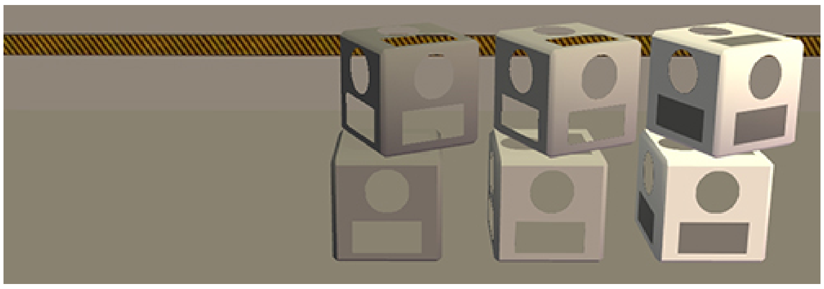
This setting doesn't allow any partial transparency.

4. stdMatAlphaMultiplier (in none mode)

Seems to multiply the strength of the Alpha channel.
Although you could make invisible object through these settings, it would force the game to draw the surface, and *then* make it so much transparent to make it invisible. This leads to useless strain on the graphic processor.

	1	**2**	**3**	**4**	**5**
stdMatAlphaBlendMode	none	none	none	none	none
stdMatAlphaTestEnabled	1	1	1	1	1
stdMatAlphaMultiplier	0	0.2	0.5	0.7	1



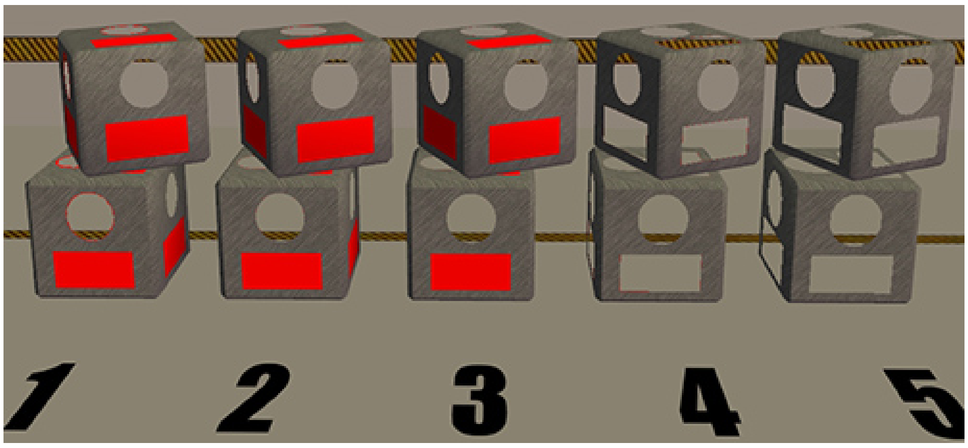


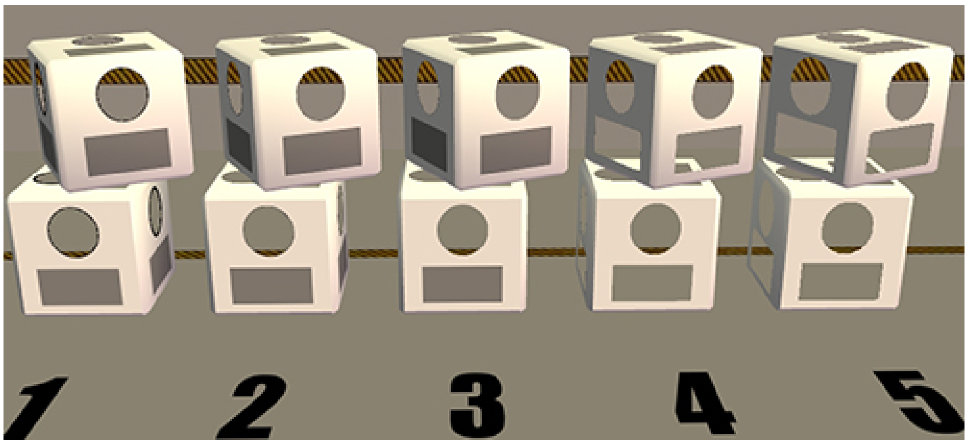
Alpha channel only

5. stdMatAlphaRefValue (in none mode)

Acts like a threshold for Alpha channel. Values go from **0** to **255**

	1	2	3	4	5
stdMatAlphaBlendMode	none	none	none	none	none
stdMatAlphaTestEnabled	1	1	1	1	1
stdMatAlphaRefValue	0	64	127	192	255



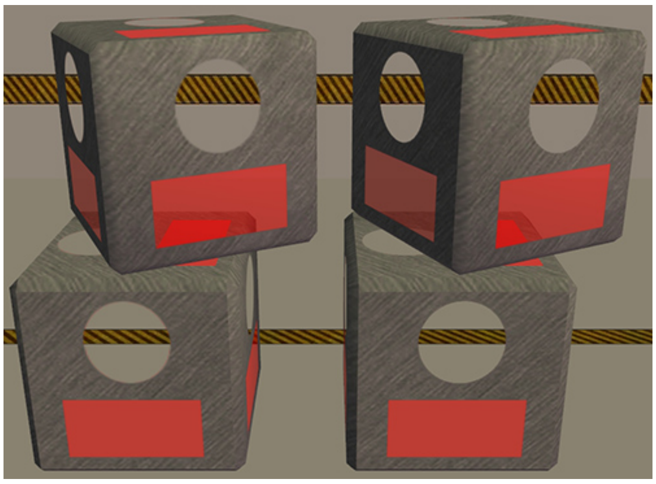


Alpha channel only

Blend

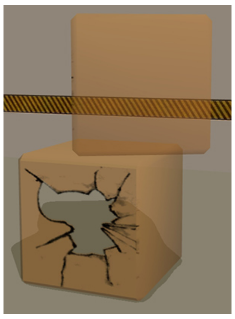
6. stdMatAlphaTestEnabled (in blend mode)

	1	2
stdMatAlphaBlendMode	**blend**	**blend**
stdMatAlphaTestEnabled	**0**	**1**



Both settings allow partial transparency.

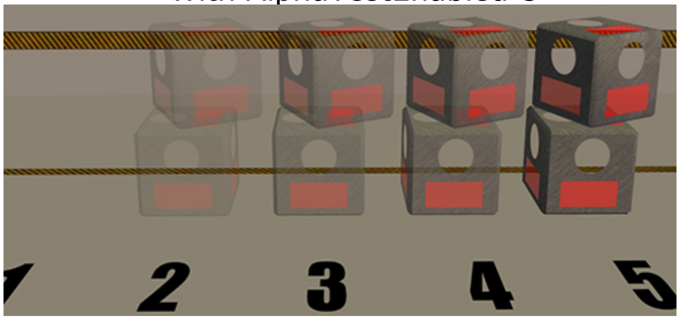
Convenient for textured glass (frosted, crackled)



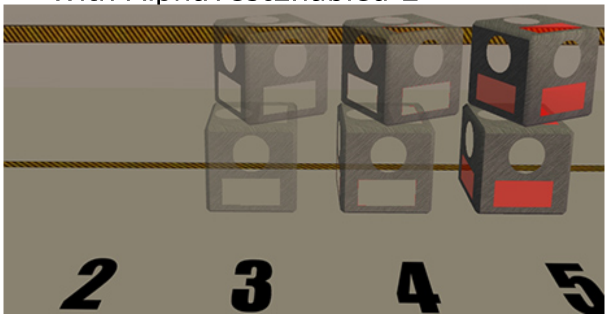
7. stdMatAlphaMultiplier (blend)

	1	2	3	4	5
stdMatAlphaBlendMode	blend	blend	blend	blend	blend
stdMatAlphaMultiplier	0	0.25	0.5	0.75	1

With AlphaTestEnabled **0**



With AlphaTestEnabled **1**



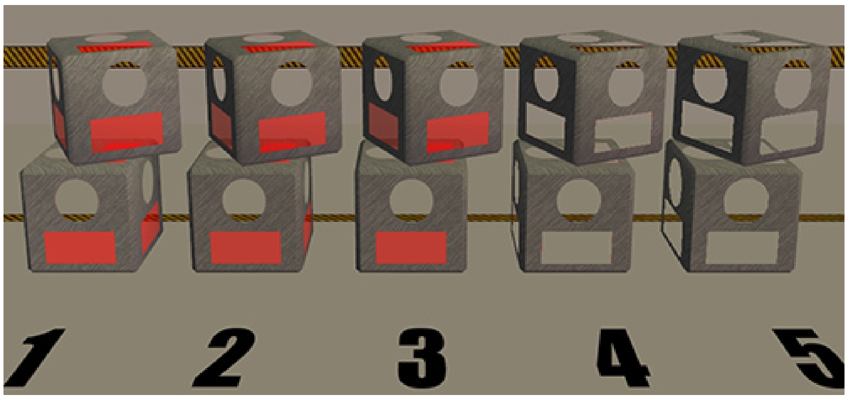
8. stdMatAlphaRefValue (blend)

	1	2	3	4	5
stdMatAlphaBlendMode	blend	blend	blend	blend	blend
stdMatAlphaRefValue	0	64	127	192	255

With AlphaTestEnabled **0**

Not active

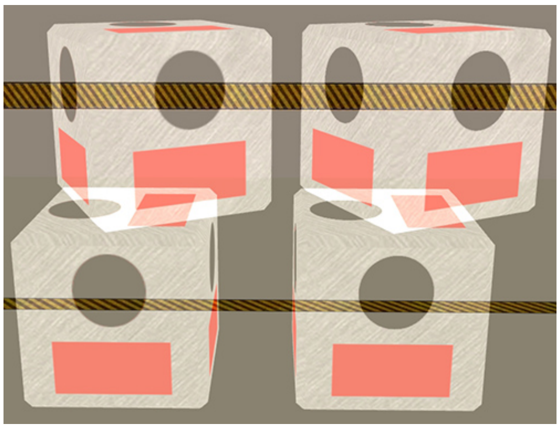
With AlphaTestEnabled **1**



additive

9. stdMatAlphaTestEnabled (in additive mode)

	1	2
stdMatAlphaBlendMode	none	none
stdMatAlphaTestEnabled	**0**	**1**



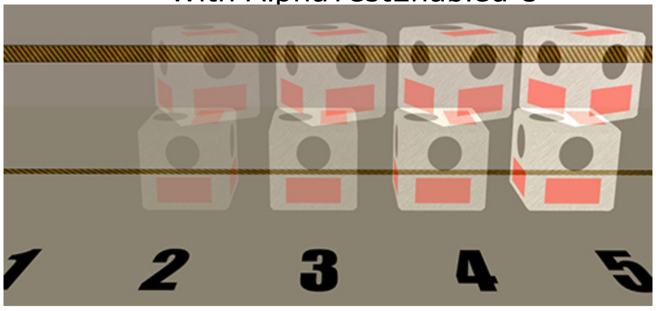
Light up the texture and allow transparency.

Object appears like when picked with the eye-dropper.

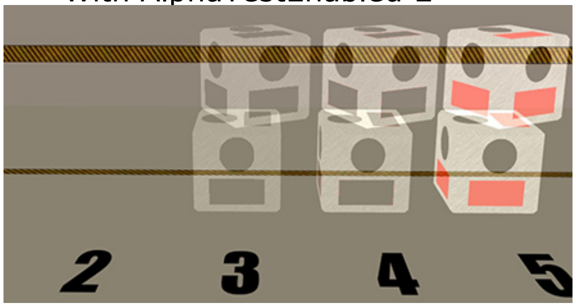
10. stdMatAlphaMultiplier (additive)

	1	2	3	4	5
stdMatAlphaBlendMode	additive	additive	additive	additive	additive
stdMatAlphaMultiplier	0	0.2	0.5	0.7	1

With AlphaTestEnabled **0**



With AlphaTestEnabled **1**



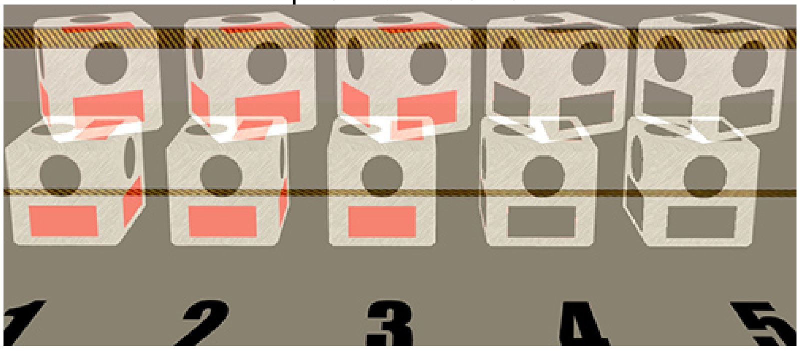
11. stdMatAlphaRefValue (additive)

	1	2	3	4	5
stdMatAlphaBlendMode	additive	additive	additive	additive	additive
stdMatAlphaRefValue	0	64	127	192	255

With AlphaTestEnabled **0**

Not active

With AlphaTestEnabled **1**



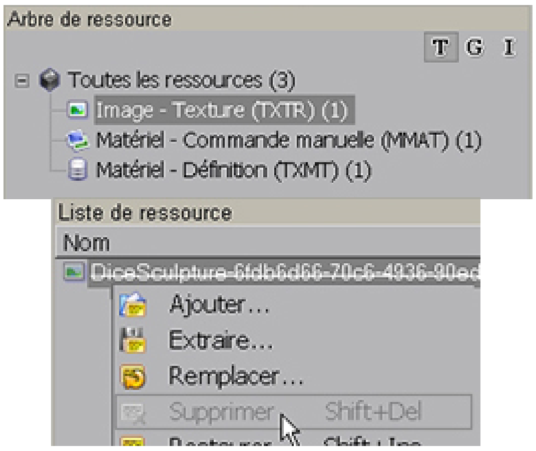
addNoSrcAlphaScale

Behaves almost like **additive**

12. stdMatUntexturedDiffAlpha

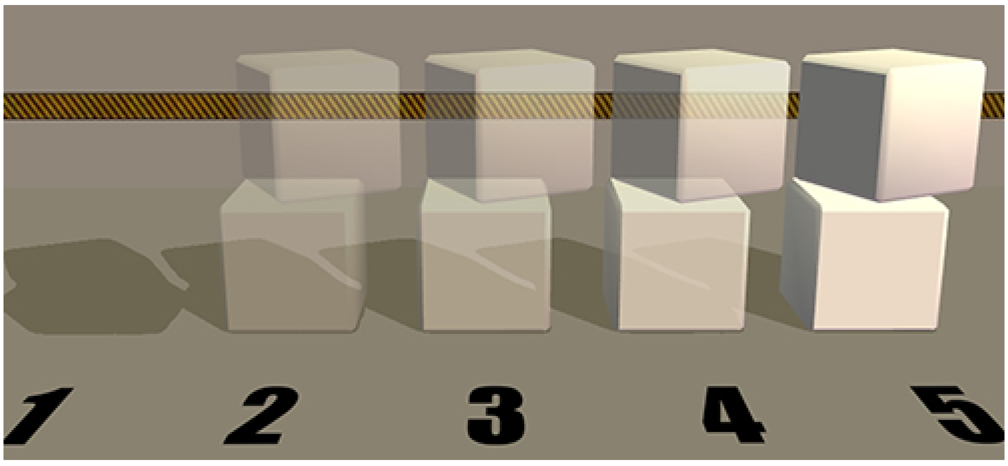
Determine the degree of transparency of an untextured object. (Glass)
From 0=invisible to 1=solid
That feature works only if there's no texture.

To have an object with no texture you have to disable it:
stdMatBaseTextureEnabled = **false (**see part 1)
and **delete it**. (Optional)

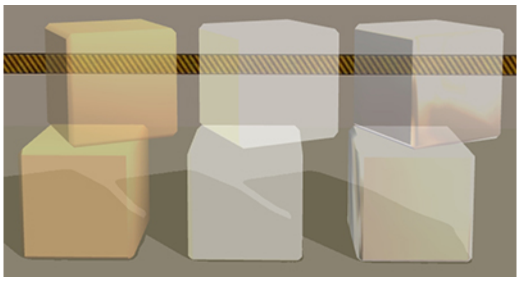

Arbre de ressource
T G I
Toutes les ressources (3)
Image - Texture (TXTR) (1)
Matériel - Commande manuelle (MMAT) (1)
Matériel - Définition (TXMT) (1)
Liste de ressource
Nom
DiceSculpture-6fdb6d66-70c6-4936-90ed
Ajouter…
Extraire…
Remplacer…
Supprimer Shift+Del

stdMatUntexturedDiffAlpha (for glass effect & invisibility)
Shadows will still be cast, even if the object is invisible.

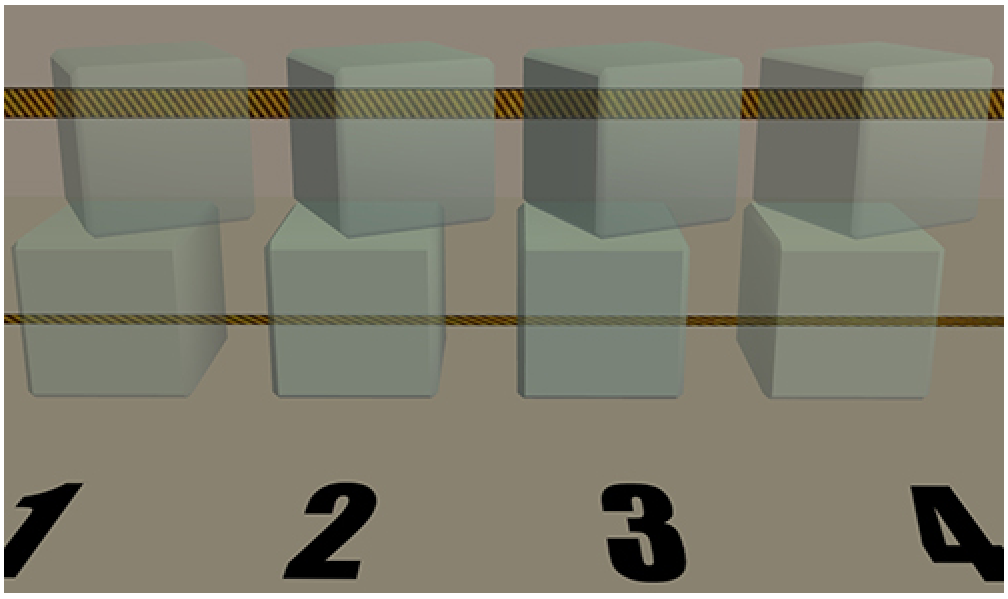
	1	2	3	4	5
stdMatAlphaBlendMode	blend	blend	blend	blend	blend
stdMatAlphaTestEnabled	0	0	0	0	0
stdMatUntexturedDiffAlpha	0	0.3	0.5	0.75	1



Of course, other parameters such as: Diffuse, Emissive, Reflection can be combined.



Comparison between the several glass settings options



	1	2	3	4
stdMatAlphaBlendMode	blend	blend	blend	blend
stdMatBaseTextureEnabled	true	true	true	false
stdMatAlphaTestEnabled	0	0	0	0
stdMatAlphaMultiplier	0.4	1	1	1
stdMatUntexturedDiffAlpha	1	1	1	0.4
stdMatDiffCoef	0.85, 0.85, 0.85	0.9, 0.9, 0.9	0.9, 0.9, 0.9	0.42353, 0.54118 ,0.54902
Texture size	512x512 plain	256x256 opacity 50%	4x4 Opacity 50%	-
File size	345 Ko	89 Ko	3 Ko	3 Ko

The first option is the worst: a big plain texture, a computer that makes double work, a large sized file.

Second option is slightly better: mid sized semi-transparent texture, mid sized file, still double work.

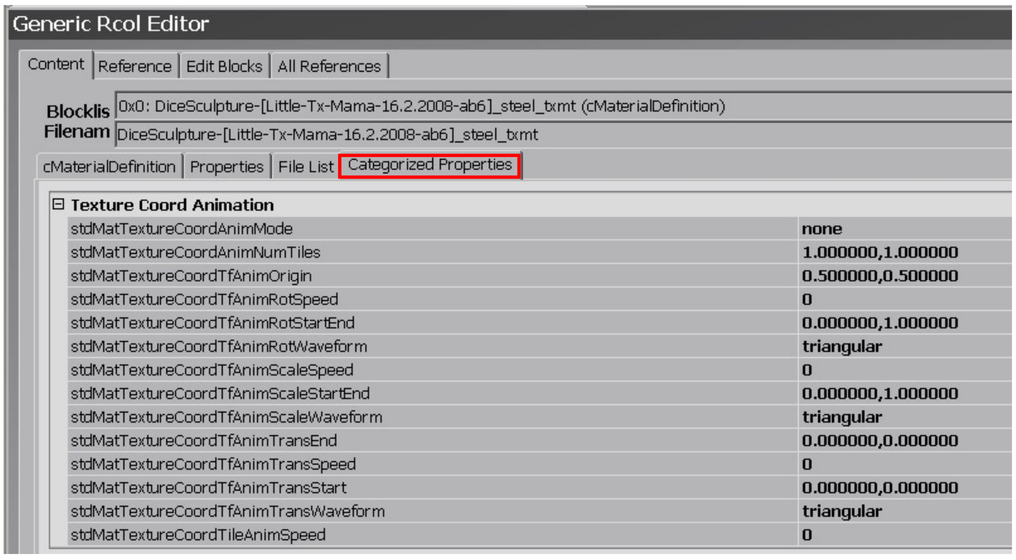
Making the semi-transparent texture very small (4x4 pixels) improves the file size, but still forces the game to draw the surface, and *then* make it transparent.

The fourth option is the best: no texture, very small file.

Test	Parameter	Default values	Other possible values	Notes (*in Italic notes from Wiki*)
2	stdMatAlphaBlendMode	none	Blend, additive, addNoSrcAlphaScale	*Manages the Alpha channel of the texture*
4, 7, 10	stdMatAlphaMultiplier	1.000000	-5 =< r =< 5	multiply the strength of the Alpha channel
5, 8, 11	stdMatAlphaRefValue	127	0 =< n =< 255	*threshold value for alpha*
3, 6	stdMatAlphaTestEnabled	0	1	Declares that the texture uses an Alpha channel
1	stdMatBaseTextureAlphaReplicate	0	1	Shows the alpha channel-Hides the texture
12	stdMatUntexturedDiffAlpha	1	0 <= n <= 1	transparency for textureless materials (0=invisible;1=solid)

Exploring the TXMT parameters part 4

This part is about Texture Coord Animation parameters section in the Categorized Properties.
It was possible thanks to research made by **Niol**.



Generic Rcol Editor

Content | Reference | Edit Blocks | All References

Blocklis 0x0: DiceSculpture-[Little-Tx-Mama-16.2.2008-ab6]_steel_txmt (cMaterialDefinition)
Filenam DiceSculpture-[Little-Tx-Mama-16.2.2008-ab6]_steel_txmt

cMaterialDefinition | Properties | File List | Categorized Properties

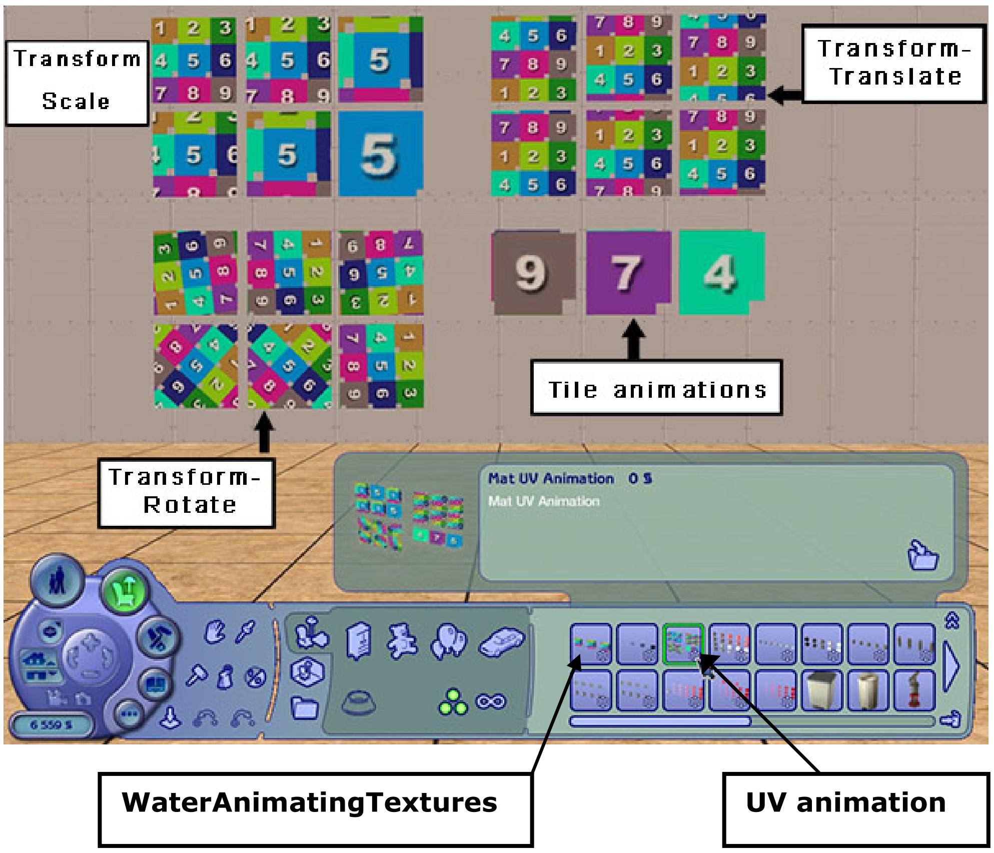
⊟ Texture Coord Animation

stdMatTextureCoordAnimMode	none
stdMatTextureCoordAnimNumTiles	1.000000,1.000000
stdMatTextureCoordTfAnimOrigin	0.500000,0.500000
stdMatTextureCoordTfAnimRotSpeed	0
stdMatTextureCoordTfAnimRotStartEnd	0.000000,1.000000
stdMatTextureCoordTfAnimRotWaveform	triangular
stdMatTextureCoordTfAnimScaleSpeed	0
stdMatTextureCoordTfAnimScaleStartEnd	0.000000,1.000000
stdMatTextureCoordTfAnimScaleWaveform	triangular
stdMatTextureCoordTfAnimTransEnd	0.000000,0.000000
stdMatTextureCoordTfAnimTransSpeed	0
stdMatTextureCoordTfAnimTransStart	0.000000,0.000000
stdMatTextureCoordTfAnimTransWaveform	triangular
stdMatTextureCoordTileAnimSpeed	0

It will require an effort of imagination. Static pictures are not the best way to illustrate animation. And I'm not good enough in English to make imaged description…

For an in game animated visualisation, I recommand to download the Lost&Found#6 objects from **Numenor**. Settings described further will refer to these.

http://www.modthesims2.com/download.php?t=50669



Animation can enrich greatly the look of an object, but
it will bring extra-strain on the computer.

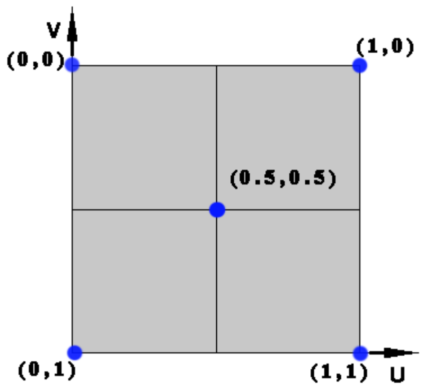
1. Enabling animation

1.1.stdMatTextureCoordAnimMode is a main switch, it will activate/deactivate animation but it will also control the kind of animation when activated.
Value can be : **none, tile, transform** or **video.**

none: will disabled the animation options.
tile: will allow strip cell animation.
transform: will give access to 3 types of animation : rotation, scaling and translation.
video: have an unknow usage till now.

2. Animation : coordonates

Textural animations work with the UV coordonates of the texture. UV coordonates are independent of the size of the texture map but animation results are related to how the object is mapped.



UV (1, 1) means
100% of texture width, 100% of texture height.

UV (0.5, 2) means
50% of texture width, 200% of texture height.

2.1.stdMatTextureCoordTfAnimOrigin

Determines the origin of **tile**, **translation**, **rotation** and **scaling** animation.

Set by default at **0.5,0.5,** which centers the relative location on the texture.

Disclaimer

This is where things become weird.

While I'm pretty confident about the theory given here, ***I can't make the texture rotate on a centered axis*** (like the Lost&Found#6 test objects found by **Numenor**), no matter what settings I choose.

I've made a lot of tests with different settings for CoordOrigin for rotation alone or combined with the different other transform animation CoordOrigin, I've tried many combinaison with translation and scaling settings.

I've tried different sort of object or surface (I even received some specialy made for me, thanks to **Buggybooz**), I've tested different size of textures and offset tricks.

Out of logic, axis of rotation in my tests are <u>always</u> on the border or off the texture.
It might have something to do with UV mapping, which is beyond my competence.

I've been methodic, systematic and persistent.
If you find out the solution, please contact me via MTS2 and deliver me from ignorance.

I only can say : "This is how it should work. But it doesn't. I don't know why."

Pixelhate. 2009

3. Tile animation

This is the most used kind of animation, you'll find numerous examples in game objects as in custom ones as well. (From candles to animated flags)
The complete creation of a cell (tile) animation is described in an early JWood's tutorial.

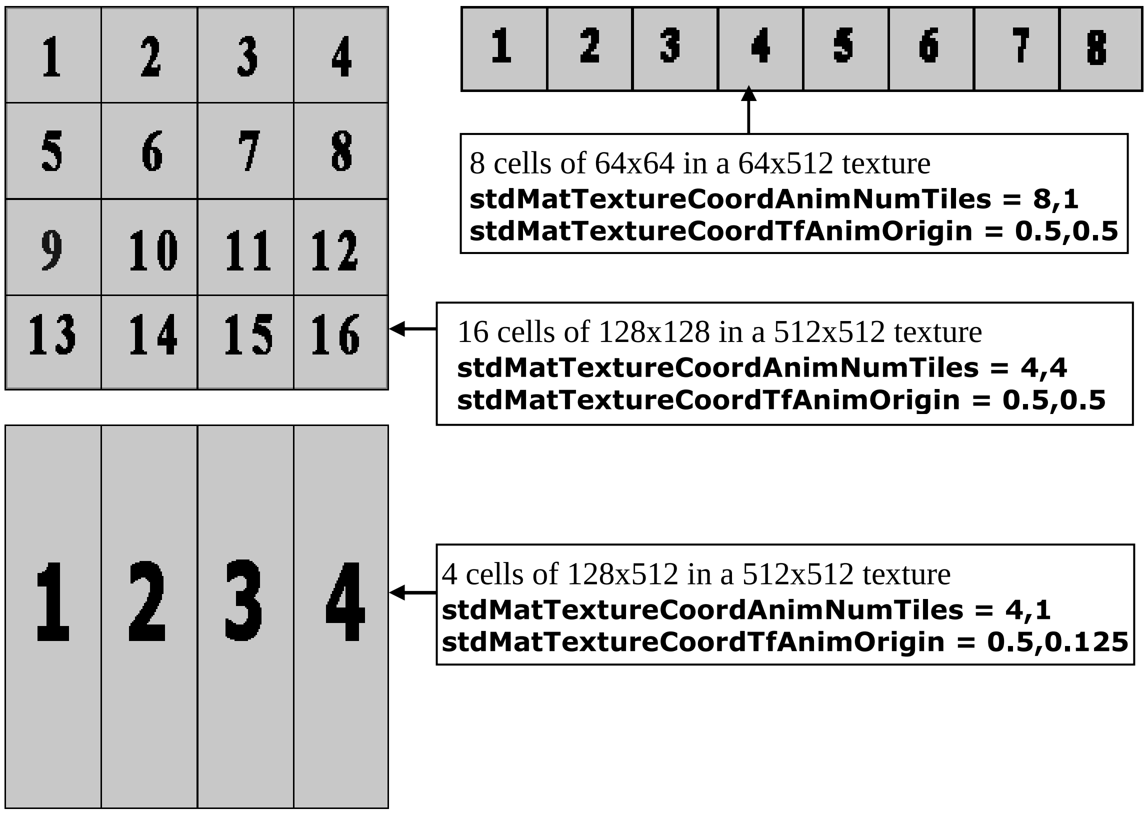
http://www.modthesims2.com/showthread.php?t=82395

The principle is the one used by frame-by-frame animation, like Gif.
The texture is divided in cells, the cells are displayed one by one at a desired speed to create illusion of mouvement. The fundamentals of cinematograph...

Textures are often square (128x128, 256x256, etc) but not inevitably (256x512, 1024x256).
Whatever size you give, it must **always** be of power of two.

3.1.stdMatTextureCoordAnimNumTiles

The parameter to determine the number of cells. Default value = **1,1**
The first value indicates the number of columns.
The second one, the number of lines.
Parameter related to **stdMatTextureCoordTfAnimOrigin.**



8 cells of 64x64 in a 64x512 texture
stdMatTextureCoordAnimNumTiles = 8,1
stdMatTextureCoordTfAnimOrigin = 0.5,0.5

16 cells of 128x128 in a 512x512 texture
stdMatTextureCoordAnimNumTiles = 4,4
stdMatTextureCoordTfAnimOrigin = 0.5,0.5

4 cells of 128x512 in a 512x512 texture
stdMatTextureCoordAnimNumTiles = 4,1
stdMatTextureCoordTfAnimOrigin = 0.5,0.125

3.2.stdMatTextureCoordTileAnimSpeed

Allows you to change the speed of the displayed animation in cells per second.
The minimum value is **0** (animation disabled), maximum is **60**.

10-12 cells per second will trick the eyes most of the time for a fluid morphing/animation.

Cinema is 24/sec and video 25.

For references: Lost&Found#6 is: Fast = 16, Medium = 8, Slow = 2

4. Transform Animation : Rotation

The texture will rotate to the desired direction. (*see disclaimer*)

4.1. stdMatTextureCoordTfAnimRotSpeed

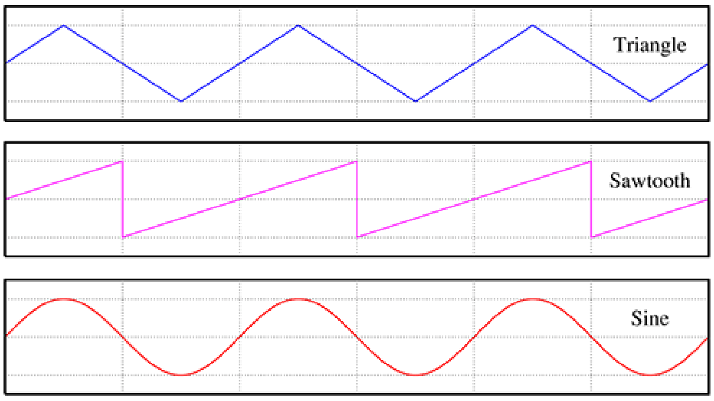
Controls the revolutions cycle per second in a range from **0.0** to **60**.
Default value = **0.000000** to disable animation.

4.2. stdMatTextureCoordTfAnimRotStartEnd

Rotation start/end position (1=360 degree)
Default value = **0.000000,1.00000** = Will make a complete counterclockwise rotation.
1.000000,0.000000 = clockwise

4.3. stdMatTextureCoordTfAnimTransWaveform

Transform Animation can be displayed according frequency patterns based on wave-forms.
Values can be : **triangular** (default value)**, sawtooth** or **sine**



These patterns will influence the way the animation is looped (in relation with the previous parameters)

triangular: Texture starts to rotate in one direction then will reverse.
sawtooth: Continuous rotation in one direction.
sine: Same as triangular, with smoother transitions.

Lost&Found#6 references:

Rotation Upper serie	**Left** (fast)	**center**	**right**
stdMatTextureCoordTfAnimOrigin	0.5,0.5	0.5,0.5	0.5,0.5
stdMatTextureCoordTfAnimRotSpeed	1.5	0.75	0.2
stdMatTextureCoordTfAnimRotStartEnd	0.0,0.1	0.0,0.1	0.0,0.1
stdMatTextureCoordTfAnimTransWaveform	triangular	triangular	triangular

Rotation Down serie	**Left** (fast)	**center**	**right**
stdMatTextureCoordTfAnimOrigin	0.5,0.5	0.5,0.5	0.5,0.5
stdMatTextureCoordTfAnimRotSpeed	1.5	0.75	0.2
stdMatTextureCoordTfAnimRotStartEnd	0.0,0.1	0.0,0.1	0.0,0.1
stdMatTextureCoordTfAnimTransWaveform	sawtooth	sawtooth	sawtooth

5. Transform Animation : Scale

Scale will give an in and/or out zoom effect by scaling up and/or down the texture.

5.1.stdMatTextureCoordTfAnimScaleSpeed

Scale cycles per second (max : **60**)
Default value = **0.000000** (disable translation animation)

High value for speed will result in flashing effect.

5.2.stdMatTextureCoordTfAnimScaleStartEnd

Decide which size to start and to end the scaling out of the texture.
The former value is the beginning scale, the latter the ending scale.

Default = **0.000000,1.000000**

Min:**0.001** max:**100**

100 means 100 x the base texture!

Assigning extreme size values will result in a unique flat color and moving moiré effect.
Giving the same values to start and end will keep the texture motionless.

5.3.stdMatTextureCoordTfAnimScaleWaveform

triangular: zoom in and out
sawtooth: flashing bouncing-loop
sine: Same as triangular with smoother transitions.

The use of sine can give the illusion of a breathing texture.

Lost&Found#6 references:

Scale Upper serie	**Left** (fast)	**center**	**right**
stdMatTextureCoordTfAnimOrigin	0.5,0.5	0.5,0.5	0.5,0.5
stdMatTextureCoordTfAnimScaleSpeed	1.5	0.2	0.2
stdMatTextureCoordTfAnimScaleStartEnd	1.0,20.0	1.0,7.0	1.0,2.0
stdMatTextureCoordTfAnimScaleWaveform	triangular	triangular	triangular

Scale Down serie	**Left** (fast)	**center**	**right**
stdMatTextureCoordTfAnimOrigin	0.5,0.5	0.5,0.5	0.5,0.5
stdMatTextureCoordTfAnimScaleSpeed	1.5	0.8	0.2
stdMatTextureCoordTfAnimScaleStartEnd	1.0,20.0	1.0,7.0	1.0,2.0
stdMatTextureCoordTfAnimScaleWaveform	sawtooth	sawtooth	sawtooth

6. <u>Transform Animation : Translate</u>

Texture will be moving to any desired direction.

6.1.stdMatTextureCoordTfAnimTransSpeed

Translation cycles per second (max : **60**) or **0** to disable translation animation
Default value = **0.000000**
For references: Lost&Found#6 is: Fast = 1.5, Medium = 0.75, Slow = 0.2

6.2.stdMatTextureCoordTfAnimTransStart

Use it to control from which coordonates to start the translation animation.
Default value **0.000000,0.00000**

6.3.stdMatTextureCoordTfAnimTransEnd

Default value **0.000000,0.000000**
Use it to control from which coordonates to end the translation animation

Relation between start/end and direction of the translation

Start	1,0	End	0,0	→
Start	0,0	End	1,0	←
Start	0,1	End	0,0	↓
Start	0,0	End	0,1	↑
Start	1,1	End	0,0	↘
Start	0,0	End	1,1	↖
Start	1,0	End	0,1	↗
Start	0,1	End	1,0	↙

Giving the same values to start and end will keep the texture motionless.

Values can be larger than 1: range from **0.0** to **∞** (infinity).
Ex: Start = 5,0 End = 0,0 will mean, in case of use of triangular or sine Waveform, displace 5x time to the right before reversing.

6.4.stdMatTextureCoordTfAnimTransWaveform

triangular: Animation will start moving in one direction then will reverse.
sawtooth: Continuous translation.
sine: Same as triangular with smoother transitions.

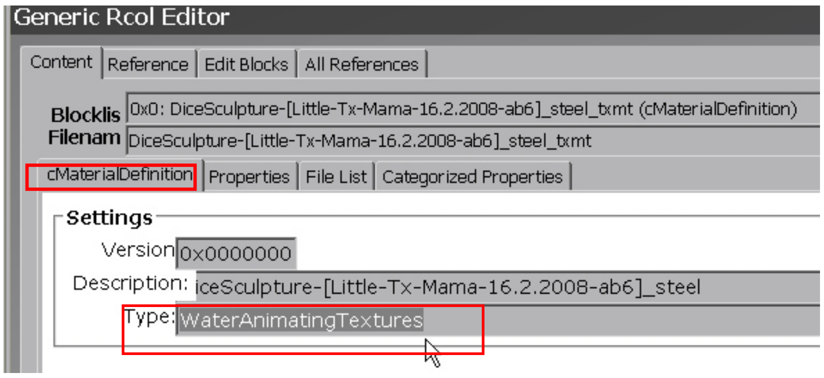
The three types of transform animations can be played separately
or they can be combined.

There're many possible combinaisons; rotating and scaling, rotating and translate, scaling and translate, the three togheter, changing this parameter here, that one there. I can't described them all.
So, what can be the use ?
Well, you'll find an answer by testing yourself, if you feel like, all I can say is that it gives often weird psychedelic results and playing with animation for hours can give dizzyness...

7. WaterAnimatingTextures

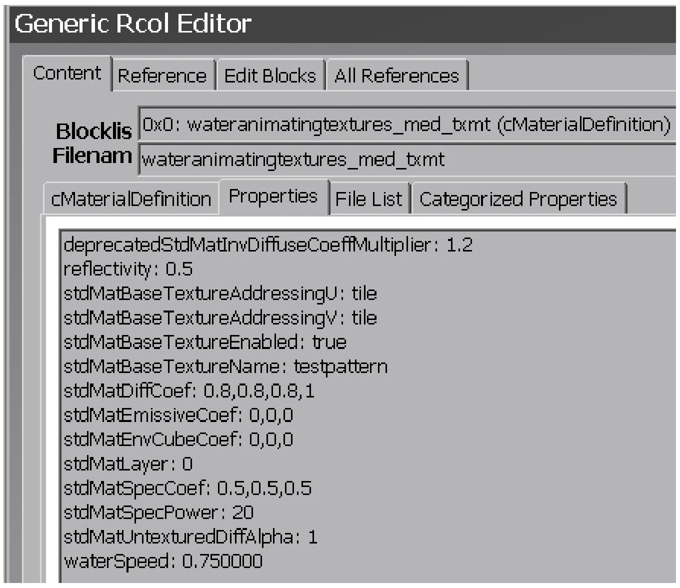
The previous described animation settings where related to the StandardMaterial categorie (most of buy object); which means that other parameters where available as well.
(Alpha-transparency, EnvCube-reflection, Emissive-glowing, etc.)

Changing material from **StandardMaterial** type to **WaterAnimatingTextures** type in **cMaterialDefinition** tab will enabled another sort of textural animation. Commit.



0x0: DiceSculpture-[Little-Tx-Mama-16.2.2008-ab6]_steel_txmt (cMaterialDefinition)
DiceSculpture-[Little-Tx-Mama-16.2.2008-ab6]_steel_txmt
Version 0x0000000
Description: iceSculpture-[Little-Tx-Mama-16.2.2008-ab6]_steel
Type: WaterAnimatingTextures

It will also disabled many of the other parameters (which can be deleted).
Here are the parameters that must remain for the texture animation to work.



0x0: wateranimatingtextures_med_txmt (cMaterialDefinition)
wateranimatingtextures_med_txmt

```
deprecatedStdMatInvDiffuseCoeffMultiplier: 1.2
reflectivity: 0.5
stdMatBaseTextureAddressingU: tile
stdMatBaseTextureAddressingV: tile
stdMatBaseTextureEnabled: true
stdMatBaseTextureName: testpattern
stdMatDiffCoef: 0.8,0.8,0.8,1
stdMatEmissiveCoef: 0,0,0
stdMatEnvCubeCoef: 0,0,0
stdMatLayer: 0
stdMatSpecCoef: 0.5,0.5,0.5
stdMatSpecPower: 20
stdMatUntexturedDiffAlpha: 1
waterSpeed: 0.750000
```

stdMatDiffCoef, stdMatEmissiveCoef are tweakable
stdMatEnvCubeCoef must be present but no refection can be added.

Attempt of description

A certain amount of transparency is automatically assigned to the texture. Two outward moving texture planes, alternately fading in and out are producing a constant mouvement.

7.1.waterSpeed (parameter must be added manually)

Determine the speed of the animation.

Default value : **0.305** Supposed range : from **0.0** to **60**

For references: Lost&Found#6 is: Fast = 1.5, Medium = 0.75, Slow = 0.3

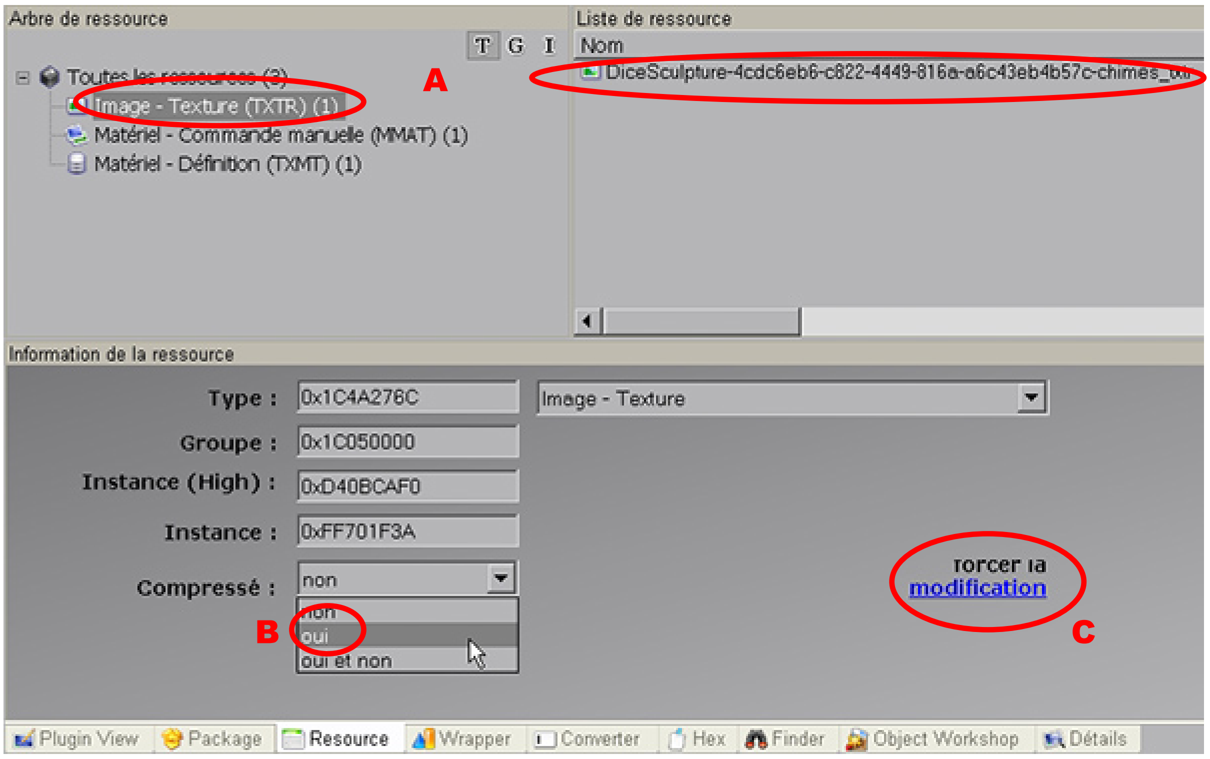
8. Compressing

Because of use of large texture size, you will end with a large and heavy package file.

I strongly urge you to take benefit of the compressing possibilities implemented in SimPE.
It can be performed on any file.
It will not decrease the quality of your texture.
It will not degrade the look of your object.
It's easy as a pair of clicks and every download folder in the World will bless you eternaly.

Here's the procedure:

A. Go to the Resource tab, check if the texture (TXTR) is selected.
B. Choose Compressed : yes
C. Commit (force modification).
Once your file is saved, a Directory of Compressed Files will appear.



Arbre de ressource

Toutes les ressources (3)
Image - Texture (TXTR) (1)
Matériel - Commande manuelle (MMAT) (1)
Matériel - Définition (TXMT) (1)

Liste de ressource

Nom
DiceSculpture-4cdc6eb6-c822-4449-816a-a6c43eb4b57c-chimes_

Information de la ressource

Type : 0x1C4A276C Image - Texture
Groupe : 0x1C050000
Instance (High) : 0xD40BCAF0
Instance : 0xFF701F3A
Compressé : non
non
oui
oui et non

forcer la modification

For a more drastic compression use the compressorizer made by Jfade and avaible at :
http://www.djssims.com/index.php?category=1&subcat=5

9. Other types of animations ?

The **video** value, in UV coordonate animation, have an unknow usage.
There're other types of textural animation, the field isn't totally explored.
AnimatedTextures type will basically do the same as tile animation.

Niol have made awe-inspiring researches and incredidle objects, involving caustics and shader tweaking. It worth the checking !

Wavy Sphere : http://www.modthesims2.com/download.php?t=131632
SwimPool Surface : http://www.modthesims2.com/download.php?t=187425
Caustic Wall : http://www.modthesims2.com/download.php?t=179725

There's a suspicion of multiple texture layers animation but info are very few and quite technical.

Textural Animation Parameters

Test	Parameter	Default values	Other possible values	Notes
1.1	stdMatTextureCoordAnimMode	none	tile, transform, video	Enable/disable animation and select the type that will displayed
2.1	stdMatTextureCoordTfAnimOrigin	0.500000,0.500000	0.0 =< r =< ∞ (infinity)	Determines the origin of tile, translation, rotation and scaling animation
3.1	stdMatTextureCoordAnimNumTiles	1.000000,1.000000	0.0 =< r =< 1	Determines the number of cells by Ligns, columns
3.2	stdMatTextureCoordTileAnimSpeed	0.000000	0 =< r =< 60	Determines the speed of the tile animation in cells per second
4.1	stdMatTextureCoordTfAnimRotSpeed	0.000000	0 =< r =< 60	Determines the speed of the rotation animation in cycle per second
4.2	stdMatTextureCoordTfAnimRotStartEnd	0.000000,1.000000	0.0 =< r =< ∞	Determines the rotation expanse 1 = 360 degree
4.3	stdMatTextureCoordTfAnimRotWaveform	triangular	sawtooth, sine	Determines the loop type of the rotation
5.1	stdMatTextureCoordTfAnimScaleSpeed	0.000000	0 =< r =< 60	Determines the speed of the scaling animation in cycle per second
5.2	stdMatTextureCoordTfAnimScaleStartEnd	0.000000,1.000000	0.0 =< r =< ∞	How large the scaling starts, how large it ends
5.3	stdMatTextureCoordTfAnimScaleWaveform	triangular	sawtooth, sine	Determines the loop type of the scaling
6.1	stdMatTextureCoordTfAnimTransSpeed	0.000000	0 =< r =< 60	Determines the speed of the translation animation in cycle per second
6.2	stdMatTextureCoordTfAnimTransStart	0.000000,0.000000	0.0 =< r =< ∞	From which coordonates to start the translation animation
6.3	stdMatTextureCoordTfAnimTransEnd	0.000000,0.000000	0.0 =< r =< ∞	From which coordonates to end the translation animation
6.4	stdMatTextureCoordTfAnimTransWaveform	triangular	sawtooth, sine	Determines the loop type of the translation
7.1	waterSpeed	0.305	**?** 0 =< r =< 60 **?**	Determines the movement speed of the water animation

Exploring the TXMT parameters part 5

Bump/Normal Mapping is a proccess giving illusion of relief without increasing geometry. It's a polys saviour technique.
Bump/Normal Mapping requires a graphic card that support it, not all do. It will also require extra calculation, putting more strain on the graphic card (but less than for high-poly objects).

A bit of theory to wipe away confusions

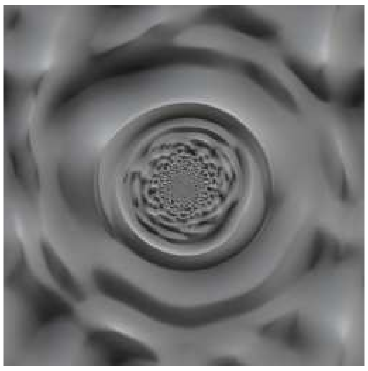
Bump and normal maps are like tables of data used by the computer for a per pixel rendering.

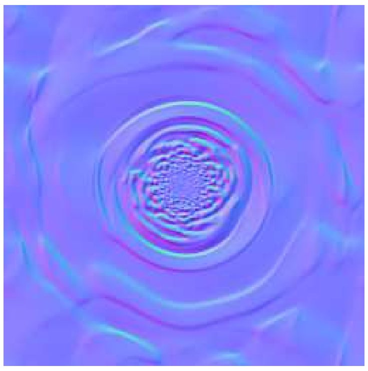
Bump maps are grey scale textures, with one value per pixel, which determine the "highness" of the corresponding base texture pixel.
The lighter areas are rendered as raised portions of the surface and darker areas are rendered as depressions. Gradient will be rendered as a slope.

Normal maps are colour textures, with three values per pixel, wich determine the "direction" and the "highness" of the corresponding base texture pixel. R,G,B = X,Y,Z.

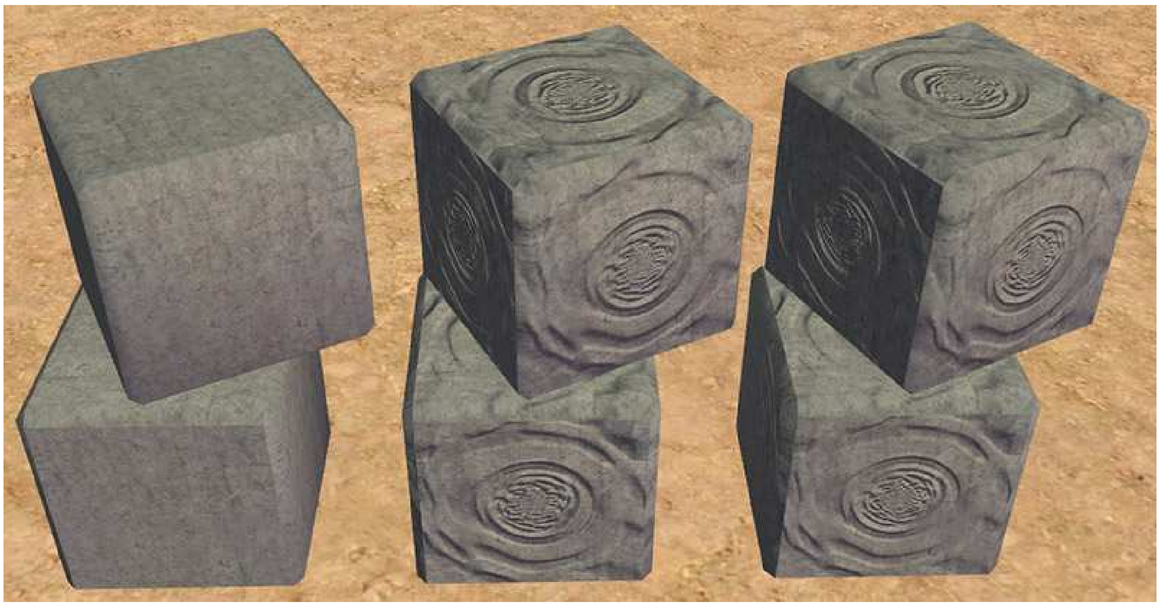
Both are working with Sims 2

Texture map


Bump map


Normal map

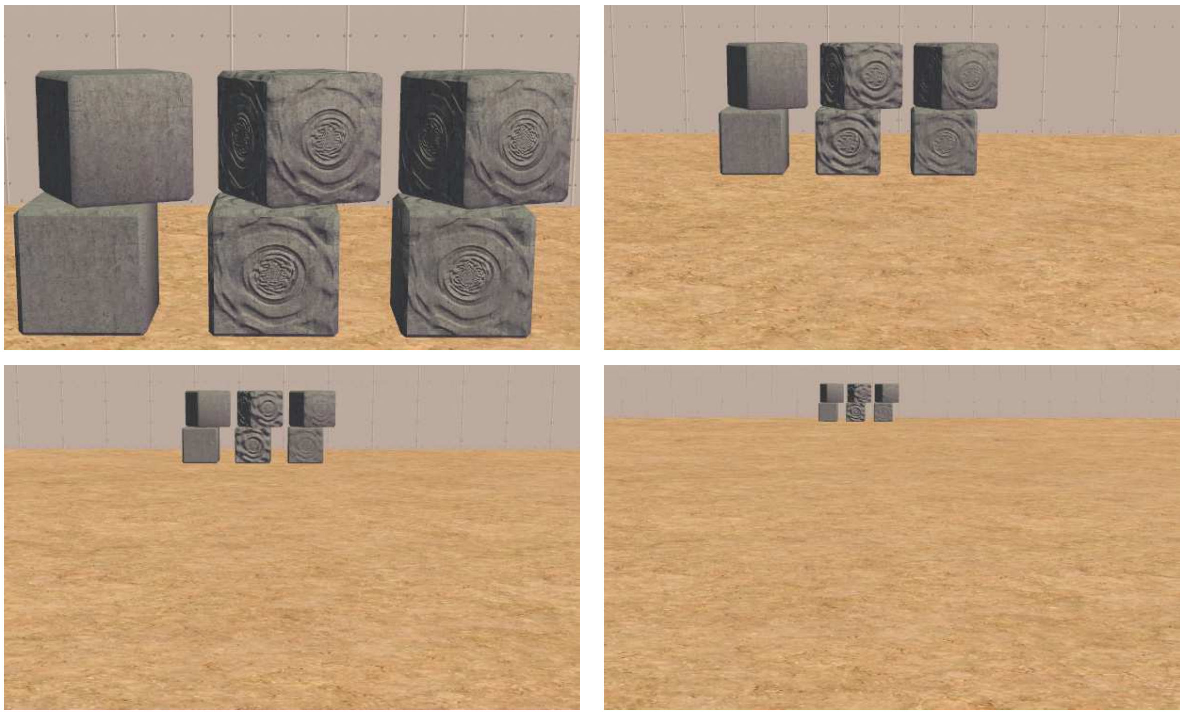
While it's relatively easy to visualize and draw a bump map by hand, it's almost impossible with normal maps, they need to be computer generated.


Texture only   Texture + Bump   Texture + Normal

Bump map	Normal map
Can be hand drawn/corrected	Must be computer generated/tweaked
Loose precision when zooming out, exagerated relief	Smooth transition when zooming out, realistic rendering
Important loss of quality when downsized	Acceptable loss of quality when downsized
256 shades for information – Less detail definition	256x256x256 = 16777216 shades – Higher detail definition
Raw8Bit format - heavy	DXT1 format - lighter

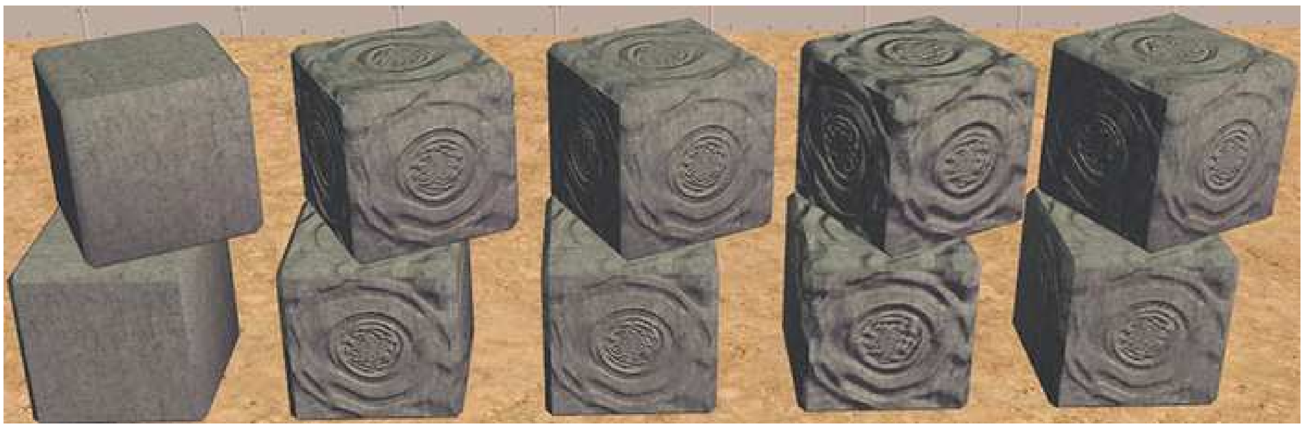
Zooming

Left : Texture only, Center : Bump, Right : Normal



Downsizing

Texture map=512x512 pixels.



Bump 512x512

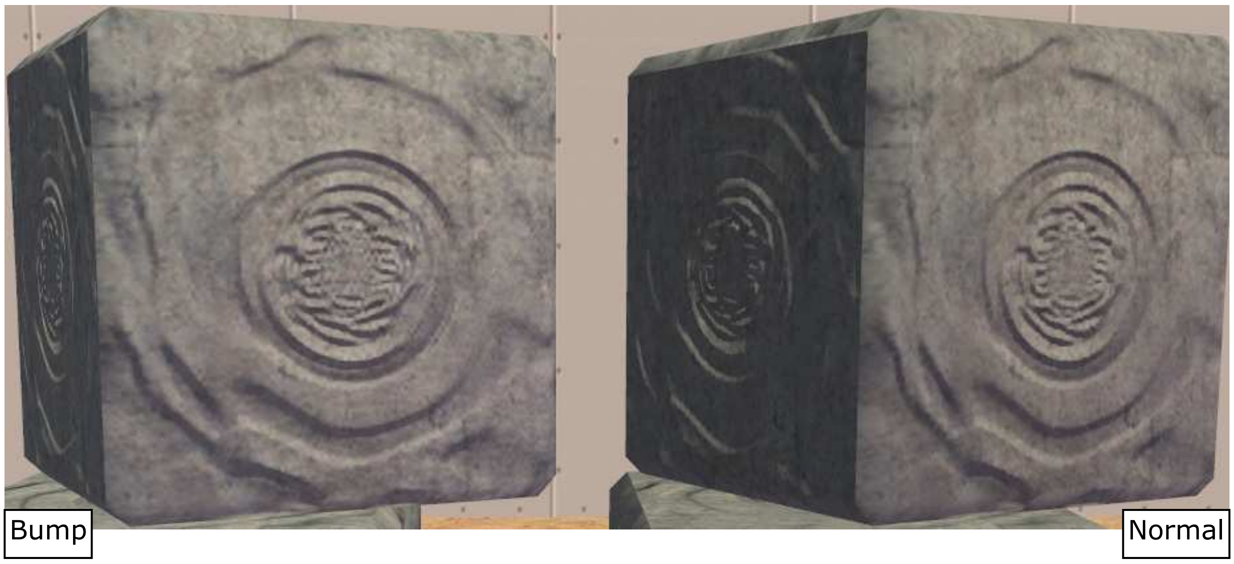
Normal 512 x512

Bump 256x256

Normal 256x256

<u>Details quality</u> (@512x512)

Fine details are slightly better defined with Normal



<u>Compression & file sizes</u>

Textures are build with DXT3. Bump with DXT5 changed into Raw8Bit. Normal with DXT1

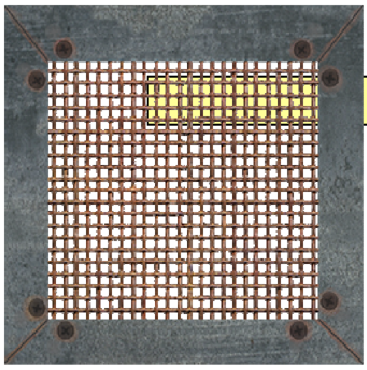
	Pixel size	Uncompressed	Compressed in SimPE	through Compressorizer
Texure only	512x512	344 Ko	52 Ko	49 Ko
Texture + Bump	512x512	685 Ko	122 ko	115 Ko
Texture + Normal	512x512	515 Ko	226 Ko	97 Ko

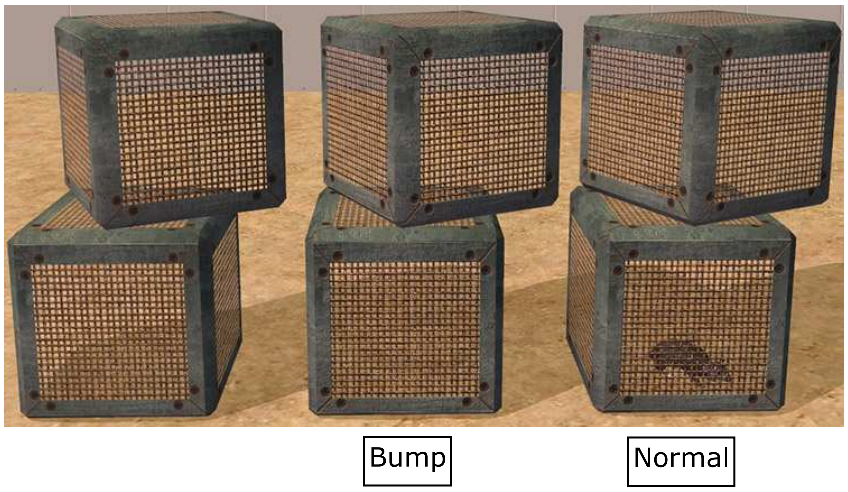
The compression feature in SimPE doesn't always work in a optimal way (like normal here)
The Compressorizer by **Jfade** treat files in a more drastic way. With no quality loss..
(http://www.djssims.com/index.php?category=1&subcat=5)

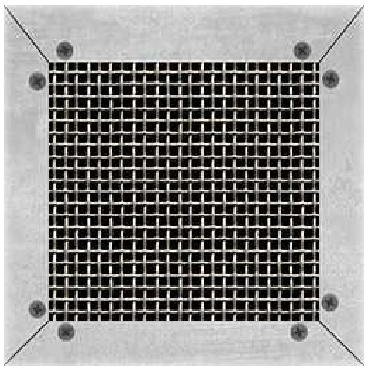
<u>BumpMap and transparency</u>

Transparency in texture is preserved when used in conjunction with Bump or Normal Map.

Transparent parts should be filled with the lowest value (black).





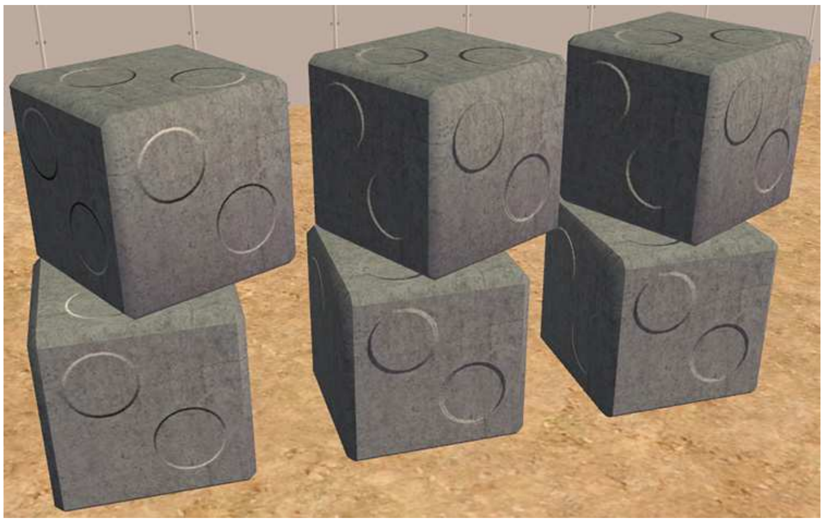


Light, shader and specularity

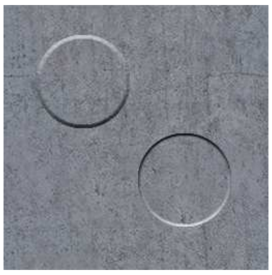
The effect rendered is related to the shaders, the lightning system used in game.
It hit results impossible to obtain only by texturing. Bump and Normal maps will simulate relief according to the light source (sun or lamp), its intensity and its direction, while shadow painted on texture will remain static and look flat and unrealistic in many angles.

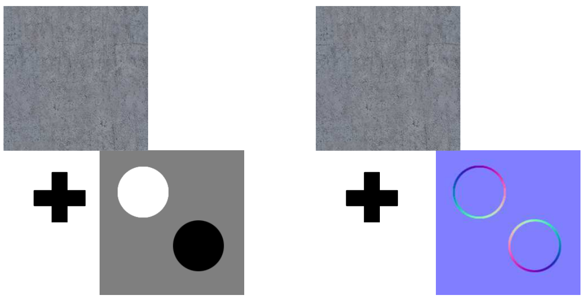
Results may differ according the type of shader used, like the GunMod lightning system (not tested) or the graphic card.

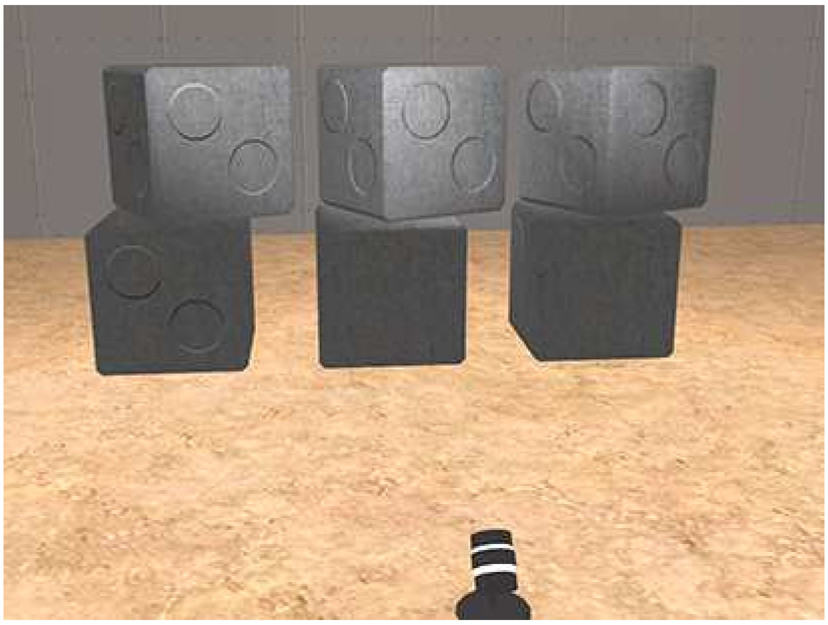
Give particular attention to the shadowed side :



Painted texture







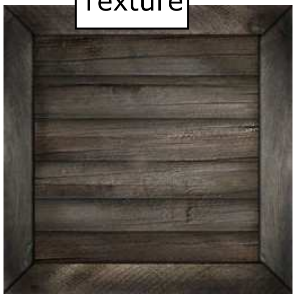
Relying on Bump/Normal Map only to create relief may lead to graphical oddities.
People with non bump capable graphic card will loose every details..
A neat Bump Map combined with a proprely shaded texture will offer the best result.

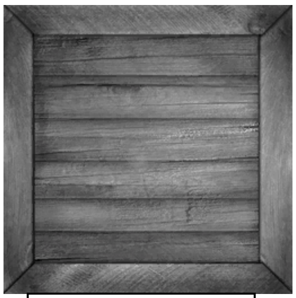
1. **Creating bump map** (in Photoshop)

- **Creating a bump map from a texture**

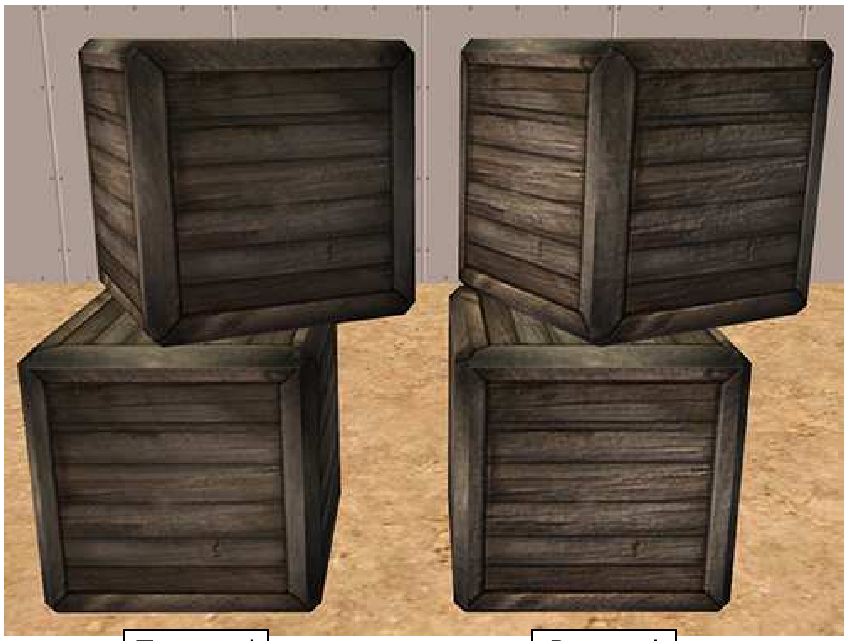
The quick and rough way to create bump map is to desaturate a copy of the texture, and play with brightness/contrast to adjust.

Texture





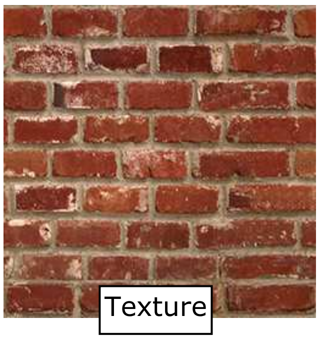
Desat+ contrast
= Final Bump



Textured

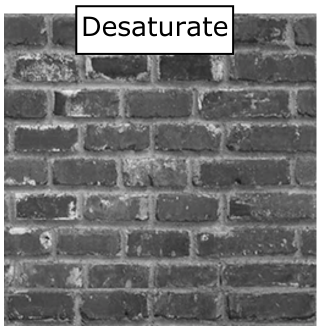
Bumped

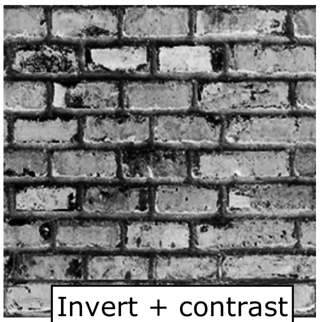
Eventually, use invert to respect the direction of the relief (remember white is up , black is down).



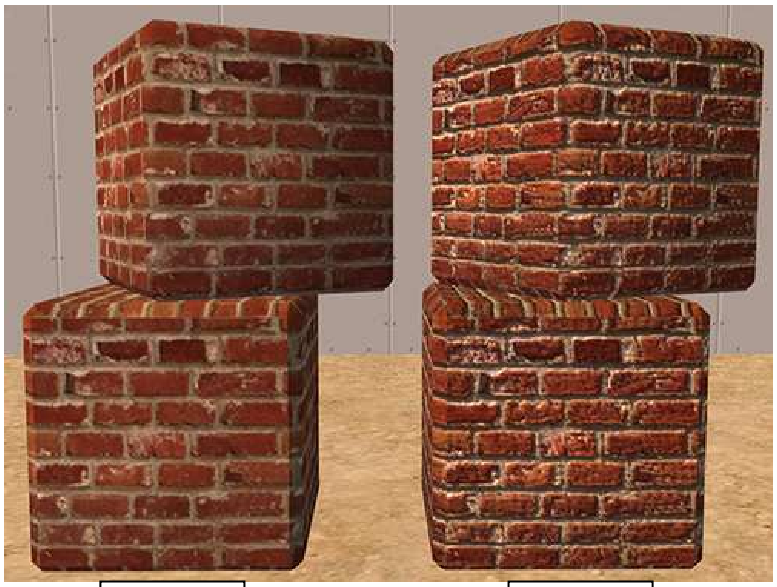
Texture

Desaturate





Invert + contrast
= Final Bump



Textured

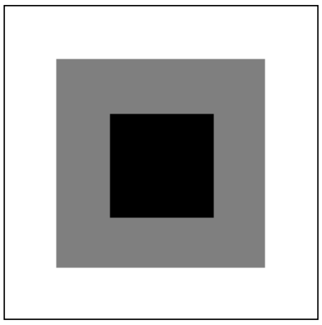
Bumped

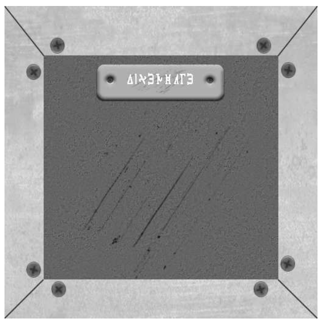
The contrast determmine the depth scaling.

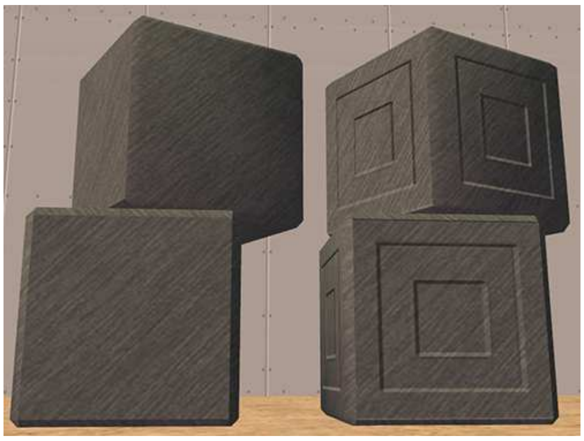
Exagerated here, for demonstration purpose.

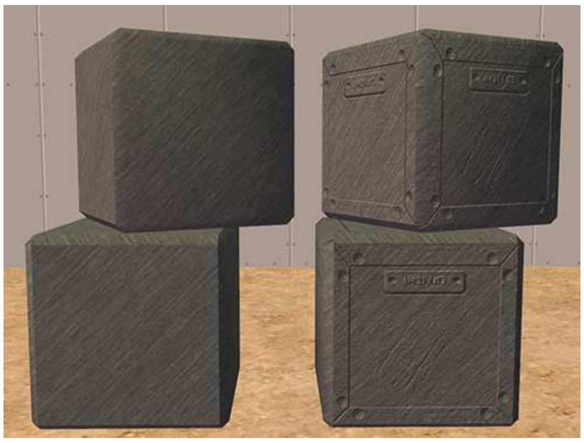
- **Creating a bump map from scratch**

The Bump Map is in grey scale and the middle (flat) value is RGB : 127,127,127.
– Fill background layer with flat grey.
– Draw in darker value to create depression, in lighter value to create protusion.







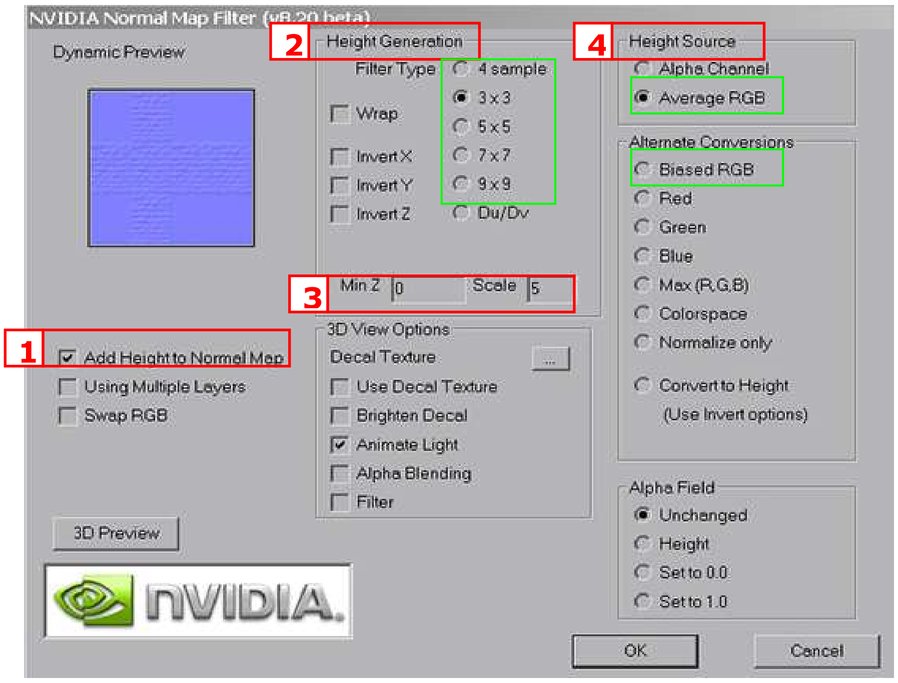


2. **Creating Normal Map** (in Photoshop)

You'll need the free NVIDIa plugin :
http://developer.nvidia.com/object/photoshop_dds_plugins.html
(**Gimp** normal plugin http://nifelheim.dyndns.org/~cocidius/normalmap/)

In PS, it avaible under : Filters > NVIDIA Tools > NormalMapFilter..

You can create a Normal map directly from the original texture or out of a Bump map. Several adjustments can be made in the filter window. Only the basic are given here, for more info, consult Nvidia documentation. Or make experiments..



Height Generation
Filter Type: 4 sample [✓ 3 x 3] 5 x 5 7 x 7 9 x 9 Du/Dv
☐ Wrap
☐ Invert X
☐ Invert Y
☐ Invert Z
Min Z 0 Scale 5

Height Source
Alpha Channel [✓ Average RGB]

Alternate Conversions
Biased RGB
Red
Green
Blue
Max (R,G,B)
Colorspace
Normalize only
Convert to Height (Use Invert options)

Alpha Field
[✓ Unchanged] Height Set to 0.0 Set to 1.0

☑ Add Height to Normal Map
☐ Using Multiple Layers
☐ Swap RGB

3D View Options
Decal Texture ...
☐ Use Decal Texture
☐ Brighten Decal
☑ Animate Light
☐ Alpha Blending
☐ Filter

1. Need to be checked for our purpose.

**2. Sharpness / bluriness.
4 sample = very detailed.
9x9 = very smooth.**

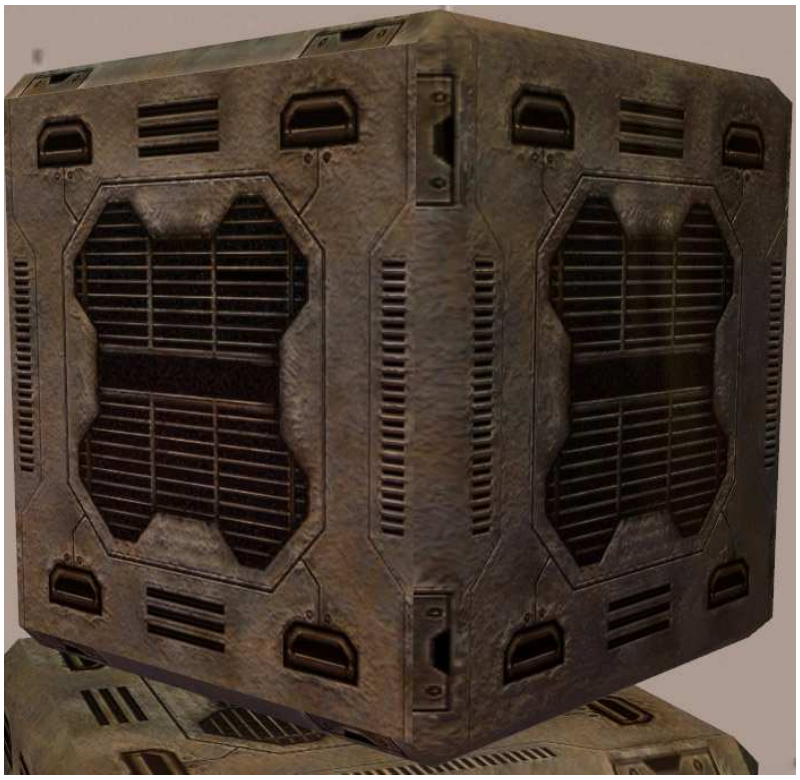
3. Act on the depth ratio, like contrast.

**4. Select the source used for conversion:
Average RGB, in general.**

***Biased RGB* is equivalent to invert, when Bumpmapping.**

***Normalize only* is for passing an existing Normal map through the filter.**

Note : Filtering a Bump or normal out of an existing texture will certainly enhance the look of the object. But making a Bump map from scratch, converting it into Normal and drawing a texture *after*, to give consitency to the relief created, can lead to nice results, even with low polys object.



Another technique for perfectionist freaks, is to make a High-poly version of an object, make very detailled maps (UV & Bump), then, make a low-poly version of the object and apply the detailled maps.
No illustration here, but with some search you'll find image of 5000 polys objects, lowered to 500. It's hard to see the difference once in game.

This section was possible only thanks to the erudition and kindness of **Jasana Bugbreeder**. It has only a theoretical value if your graphic doesn't support Bump Map. A recolour of a base game object is provided as a tester, see end of document for details.

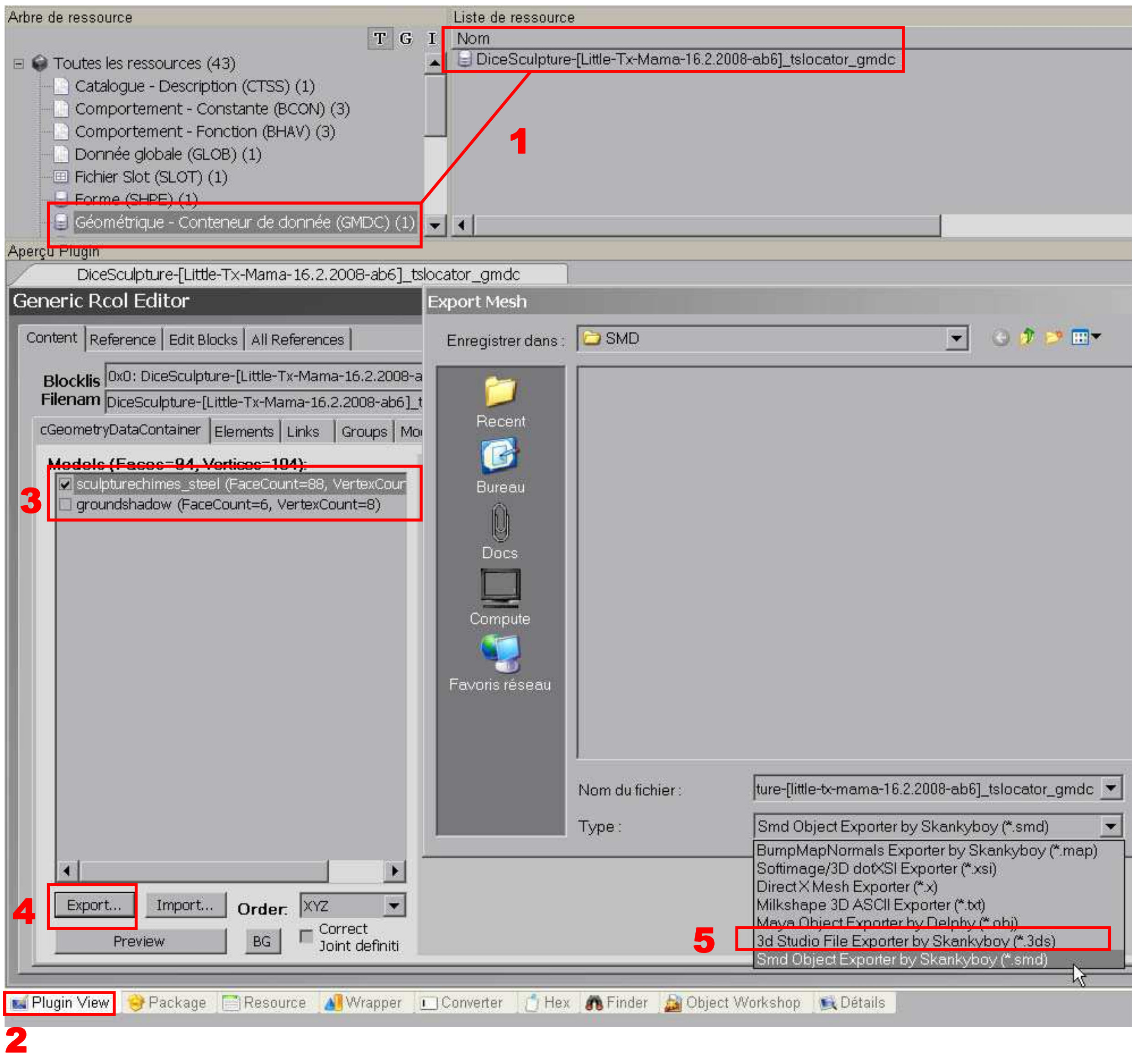
The process of enabling an exististing object is possible thanks to the work of **Skankyboy**. His plugin is integrated in SimPE.

Will be considered here only decorative boneless object, like sculpture, without mesh animation, in standard material TXMT.

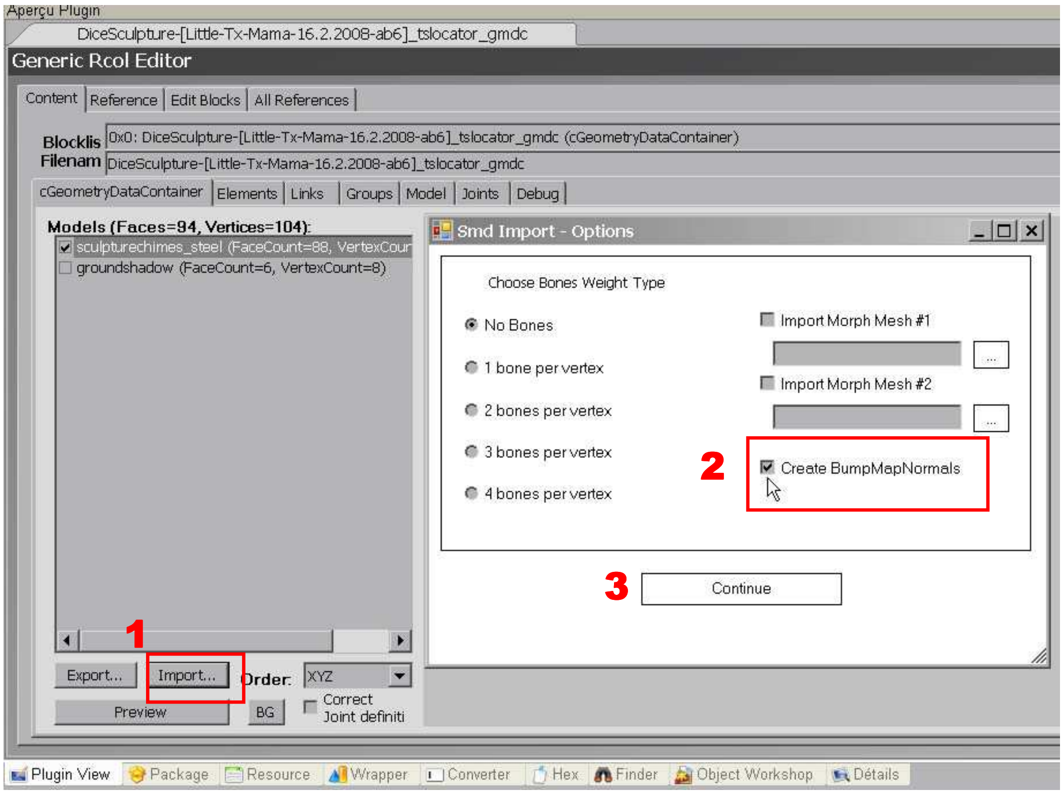
Always make back-up or work on copies, I will NOT be responsible for messed up packages.

3. Bump enabling an existing objects

Open your desired mesh in SimPE,
1. select the geometic data container GMDC
2. choose plug in view.
3. Make sure that only the mesh is selected or only one of the subset, if several.
4. click **Export**, **5.** select .smd format and choose the folder where to save it. Save.

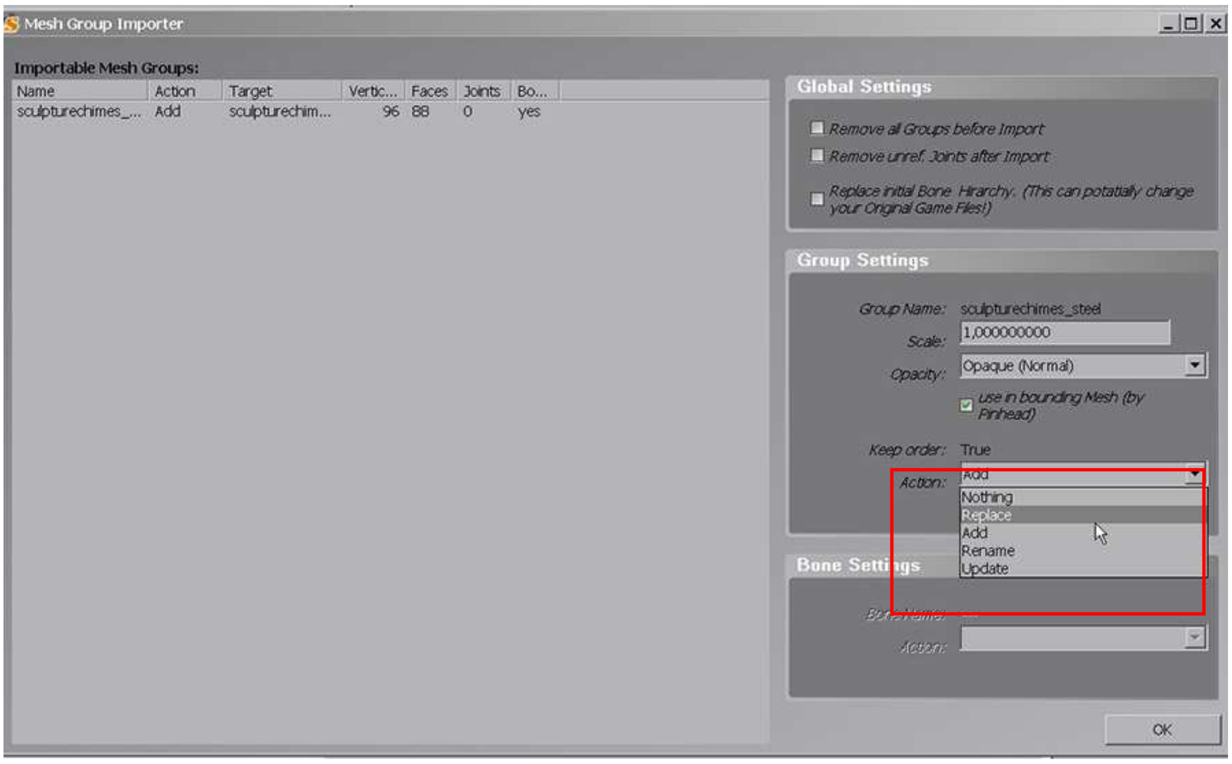

Arbre de ressource
T G I
Toutes les ressources (43)
Catalogue - Description (CTSS) (1)
Comportement - Constante (BCON) (3)
Comportement - Fonction (BHAV) (3)
Donnée globale (GLOB) (1)
Fichier Slot (SLOT) (1)
Forme (SHPE) (1)
Géométrique - Conteneur de donnée (GMDC) (1)
Liste de ressource
Nom
DiceSculpture-[Little-Tx-Mama-16.2.2008-ab6]_tslocator_gmdc
1
Aperçu Plugin
DiceSculpture-[Little-Tx-Mama-16.2.2008-ab6]_tslocator_gmdc
Generic Rcol Editor
Content
Reference
Edit Blocks
All References
Blocklis
0x0: DiceSculpture-[Little-Tx-Mama-16.2.2008-a
Filenam
DiceSculpture-[Little-Tx-Mama-16.2.2008-ab6]_t
cGeometryDataContainer
Elements
Links
Groups
Mo
Models (Faces=94, Vertices=104):
sculpturechimes_steel (FaceCount=88, VertexCour
groundshadow (FaceCount=6, VertexCount=8)
3
4
Export...
Import...
Order:
XYZ
Preview
BG
Correct Joint definiti
Export Mesh
Enregistrer dans :
SMD
Recent
Bureau
Docs
Compute
Favoris réseau
Nom du fichier :
ture-[little-tx-mama-16.2.2008-ab6]_tslocator_gmdc
Type :
Smd Object Exporter by Skankyboy (*.smd)
BumpMapNormals Exporter by Skankyboy (*.map)
Softimage/3D dotXSI Exporter (*.xsi)
DirectX Mesh Exporter (*.x)
Milkshape 3D ASCII Exporter (*.txt)
Maya Object Exporter by Delphy (*.obj)
3d Studio File Exporter by Skankyboy (*.3ds)
Smd Object Exporter by Skankyboy (*.smd)
5
Plugin View
Package
Resource
Wrapper
Converter
Hex
Finder
Object Workshop
Détails
2

Then,
1.Click **Import** button, find the smd file you've just saved and import it.
In the first pop up window, **2.** check **Create BumpMapsNormals** **3. continue**.



Aperçu Plugin
DiceSculpture-[Little-Tx-Mama-16.2.2008-ab6]_tslocator_gmdc
Generic Rcol Editor
Content | Reference | Edit Blocks | All References
Blocklis 0x0: DiceSculpture-[Little-Tx-Mama-16.2.2008-ab6]_tslocator_gmdc (cGeometryDataContainer)
Filenam DiceSculpture-[Little-Tx-Mama-16.2.2008-ab6]_tslocator_gmdc
cGeometryDataContainer | Elements | Links | Groups | Model | Joints | Debug
Models (Faces=94, Vertices=104):
sculpturechimes_steel (FaceCount=88, VertexCour
groundshadow (FaceCount=6, VertexCount=8)
Smd Import - Options
Choose Bones Weight Type
No Bones
1 bone per vertex
2 bones per vertex
3 bones per vertex
4 bones per vertex
Import Morph Mesh #1
Import Morph Mesh #2
2
Create BumpMapNormals
3
Continue
1
Export...
Import...
Order: XYZ
Correct Joint definiti
Preview
BG
Plugin View | Package | Resource | Wrapper | Converter | Hex | Finder | Object Workshop | Détails

Next window,
In action, select **Replace** then **ok**.



Mesh Group Importer
Importable Mesh Groups:
Name | Action | Target | Vertic... | Faces | Joints | Bo...
sculpturechimes_... | Add | sculpturechim... | 96 | 88 | 0 | yes
Global Settings
Remove all Groups before Import
Remove unref. Joints after Import
Replace initial Bone Hirarchy. (This can potatially change your Original Game Files!)
Group Settings
Group Name: sculpturechimes_steel
Scale: 1,000000000
Opacity: Opaque (Normal)
use in bounding Mesh (by Pinhead)
Keep order: True
Action: Add
Nothing
Replace
Add
Rename
Update
Bone Settings
Bone Name:
Action:
OK

Commit to apply the changes.
Save.

Repeat operation for each subset, if needed.

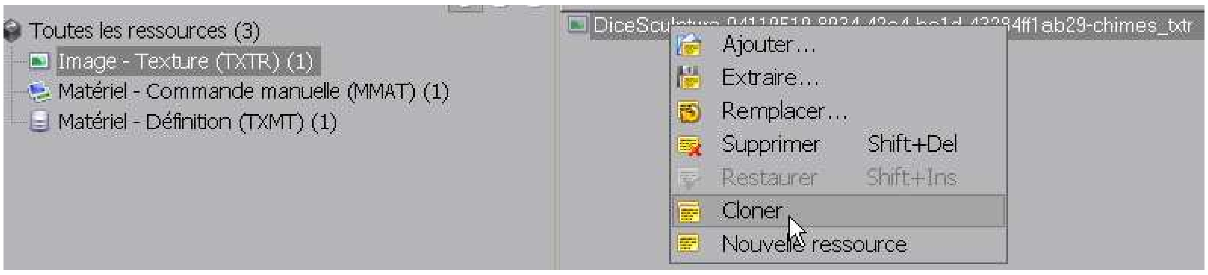
The mesh is now bump enabled. Recolours of this mesh will be bump enabled also. Parameters related to bump mapping sometimes need to be added manualy.

Recolours made before the mesh was bump enabled will show and work normaly.
Recolours created with the bump enabled mesh will work with the non-enabled mesh (bump won't be cast in that case, of course).

4. Adding a Bump map *OR* a Normal map texture

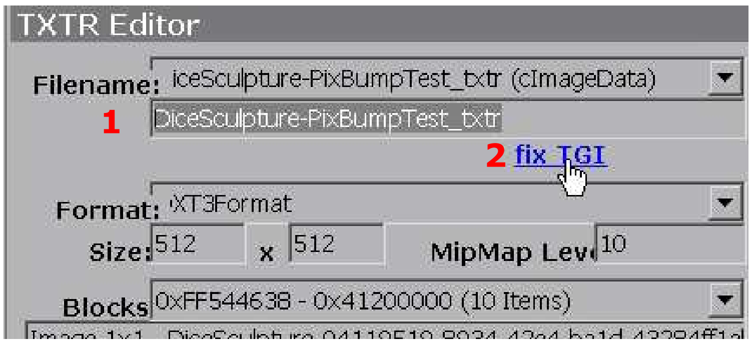
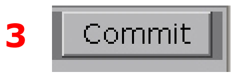
Till now, there's no bump/normal map attached to the package. We'll need to add one.

Select the Texture - TXTR ressource and clone it (rightclick).



Toutes les ressources (3)
Image - Texture (TXTR) (1)
Matériel - Commande manuelle (MMAT) (1)
Matériel - Définition (TXMT) (1)
Ajouter...
Extraire...
Remplacer...
Supprimer Shift+Del
Restaurer Shift+Ins
Cloner
Nouvelle ressource

1.Change the name of the cloned texture to a convenient and distinctive one, ***keeping the _txtr suffixe.***
2.Fix TGI. **3**.Commit to apply changes.

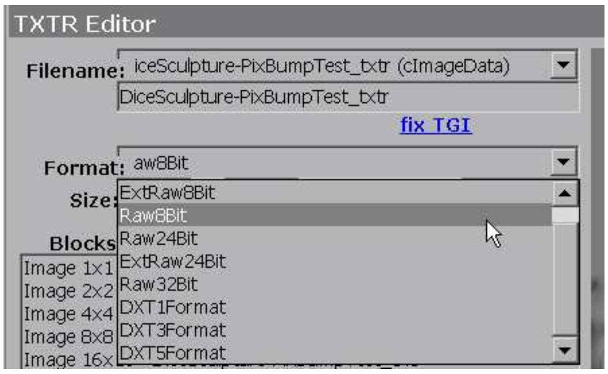



TXTR Editor
Filename: iceSculpture-PixBumpTest_txtr (cImageData)
1
DiceSculpture-PixBumpTest_txtr
2 fix TGI
Format: XT3Format
Size: 512 x 512
MipMap Lev 10
Blocks 0xFF544638 - 0x41200000 (10 Items)
3
Commit

Then, depending to the kind of Map, either build:

4.1. Bump Map only

Rightclick on the cloned texture image,
Use **Build DXT** to bring your bump map in.
Make sure **DXT5** option is selected and build.
Change the texture format in **Raw8Bit**.
Commit. Save.



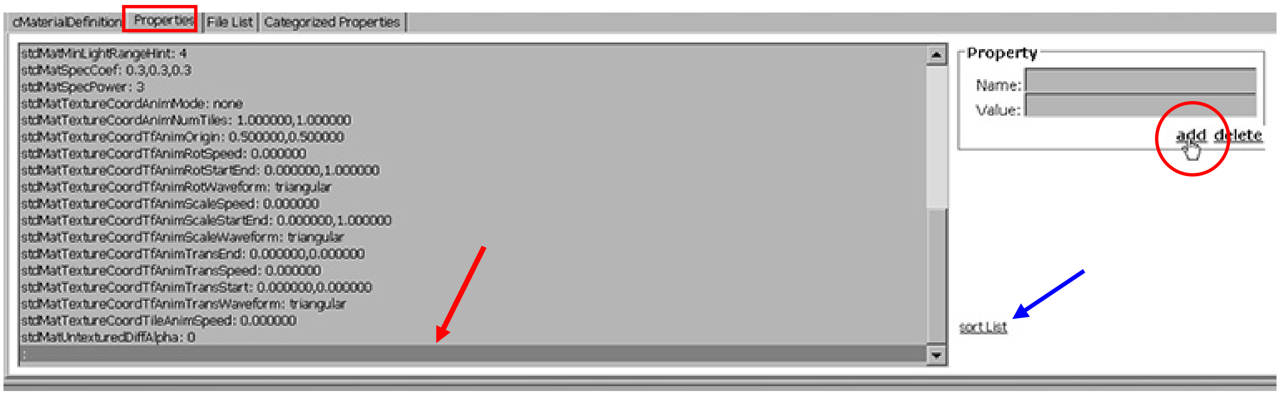
4.2. Normal Map only

Rightclick on the cloned texture image,
Use **Build DXT** to bring your normal map in.
Make sure **DXT1** option is selected and build.
Commit. Save

5. <u>Adding the relevant parameters in Material definition – TXMT.</u>

Sometimes, Bump related parameters needs to be added manualy.

In **propreties tab**, click **add** parameters 4 times, **scroll down** to find them and declare their name and values.



```
stdMatMinLightRangeHint: 4
stdMatSpecCoef: 0.3,0.3,0.3
stdMatSpecPower: 3
stdMatTextureCoordAnimMode: none
stdMatTextureCoordAnimNumTiles: 1.000000,1.000000
stdMatTextureCoordTfAnimOrigin: 0.500000,0.500000
stdMatTextureCoordTfAnimRotSpeed: 0.000000
stdMatTextureCoordTfAnimRotStartEnd: 0.000000,1.000000
stdMatTextureCoordTfAnimRotWaveform: triangular
stdMatTextureCoordTfAnimScaleSpeed: 0.000000
stdMatTextureCoordTfAnimScaleStartEnd: 0.000000,1.000000
stdMatTextureCoordTfAnimScaleWaveform: triangular
stdMatTextureCoordTfAnimTransEnd: 0.000000,0.000000
stdMatTextureCoordTfAnimTransSpeed: 0.000000
stdMatTextureCoordTfAnimTransStart: 0.000000,0.000000
stdMatTextureCoordTfAnimTransWaveform: triangular
stdMatTextureCoordTileAnimSpeed: 0.000000
stdMatUntexturedDiffAlpha: 0
:
```

Name	Value
stdMatNormalMapTextureEnabled	true
stdMatNormalMapTextureAddressingU	tile
stdMatNormalMapTextureAddressingV	tile
stdMatNormalMapTextureName	Name of the bump/normal you've build ***without*** _txtr ***suffxe***

Don't forget to sort list. New added parameters need to be alphabeticaly organised to show proprely.

Commit and save.

6. <u>One Bump fits all – Pseudo repository</u>

If you plan to make several recolours with the same Bump mapping, you don't need to add the Bump Map texture to each package.
Just add a Bump/Normal map to the mesh. It will act as a "Master"

***The others recolours only need the relevant TXMT parameters added, with the same* stdMatNormalMapTextureName *to act like* "Slaves".**

Of course, the "Master" needs to be present in the Donwloads folder for the "Slaves" to be showned.

<u>Bump and animation</u>

Bump/Normal are compatible with textural animation. It means that the animated texture will display Bump/Normal.
It means also that the computer will have to caculate everything twice.

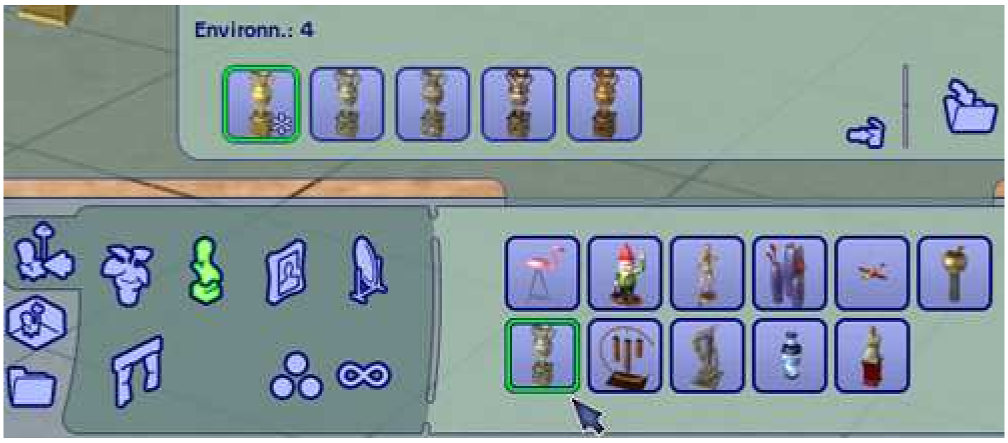
Take advantage of the compression possibilities of SimPE..

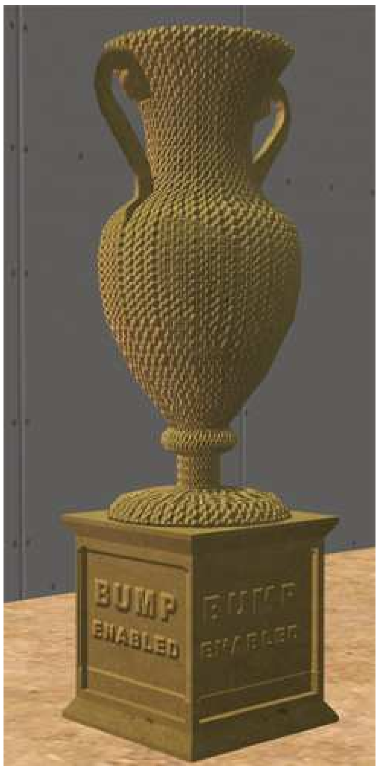
Bump/Normal enabling is something that can be done when creator build their packages, at mesh import stage (don't ask me how). But most of the time isn't.
As a result that most of objects (Custom as Maxis as well) aren't bump enabled.
Some objects don't accept bump mapping, like beds, while other work with bump but not with normal. I've no clue why.

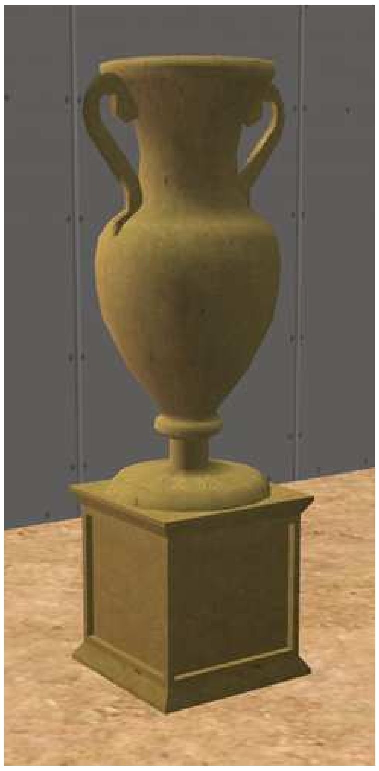
Animated/boned objects must be tested case by case.

<u>**Bump tester**</u>

Following the excellent suggestion of **BlueTexasBonnie** and thanks to her, a recolour of a base game object is provided as a Bump tester. It will indicate if your game is Bump enabled or not. Just drop it in your Downloads folder, and find the following in game :







Bump enabled

Not enabled

(Tutorial for wall Bump mapping : http://www.sims2wiki.info/wiki.php?title=Wall_bump_mapping)