

Présentation de quelques textures : fenêtres



Rédigé par Gerauld pour le site de la Confrérie des traducteurs

Outils nécessaires : Nifskope 1.1.1 et Gimp 2.6 avec plugin dds

Exemple: Data\Meshes\architecture\whiterun\wrinteriors\wrintwallstr01winb.nif

Ceci est une petite présentation des textures rencontrées en jouant avec les fenêtres. Un petit test visant à permettre de voir ce qui se trouve au-delà des fenêtres est aussi présent pour ceux qui veulent s'amuser ou parfaire leurs mods.

Présentation succincte des éléments sous Nifskope :

0 BSFadeNode	Txt	WRIntWallStr01WinB [0]
1 BSXFlags	Txt	BSX [1]
6 bhkCollisionObject		
7 NiTriShape	Txt	WRIntWallStr01WinB:0 [2]
11 NiTriShape	Txt	WRIntWallStr01WinB:1 [3]
15 NiTriShape	Txt	WRIntWallStr01WinB:2 [4]
19 NiTriShape	Txt	WRIntWallStr01WinB:3 [5]
20 NiTriShapeData		
21 BSLightingShaderProperty	Txt	
22 BSShaderTextureSet		

Le Nitrishape correspond à l'ensemble forme3D (sommets, faces), éléments de rendu, UV map et textures d'un élément du mesh.

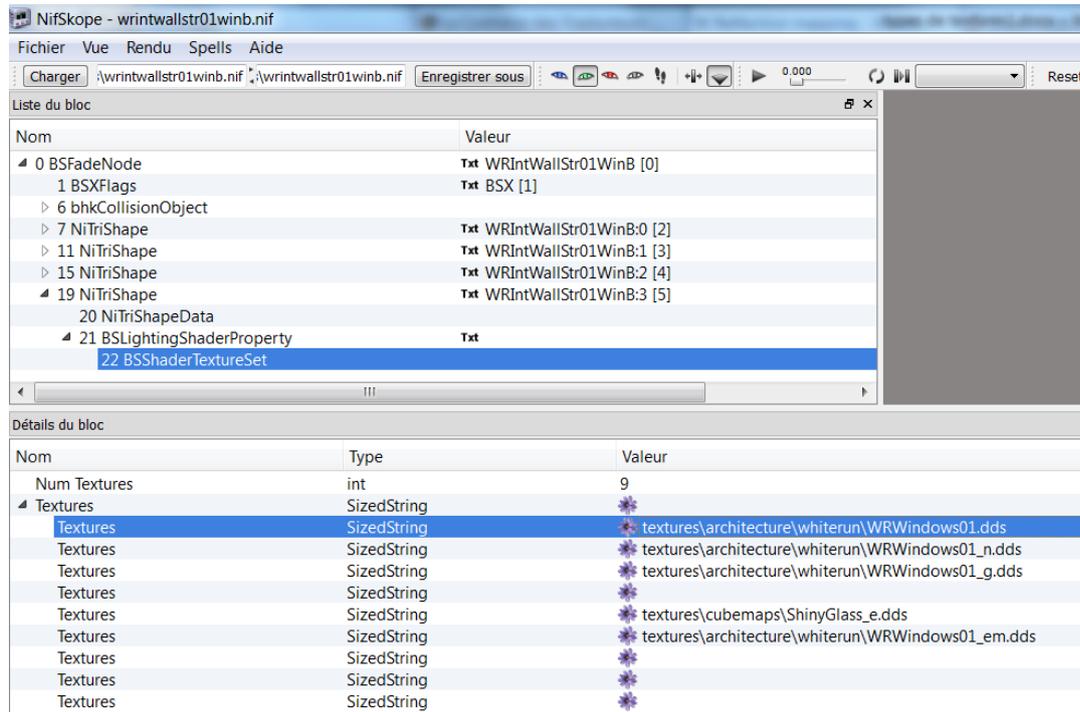
Dans notre cas, le Nitrishape 19 correspond à une petite fenêtre placée sur un mur.

Le NitrishapeData contient les informations 3D, les vertex colors et l'UVmap. Peu intéressant dans le cadre de cette présentation.

Le BSLightingShaderProperty contient toutes les informations de rendu ainsi qu'un bloc BSShaderTextureSet contenant les chemins vers les textures.

Diffuse map (.dds)

Cette texture est celle de l'objet à proprement parler avec ses couleurs telles que vues par le joueur.



Dans notre cas, elle ressemble à ça :



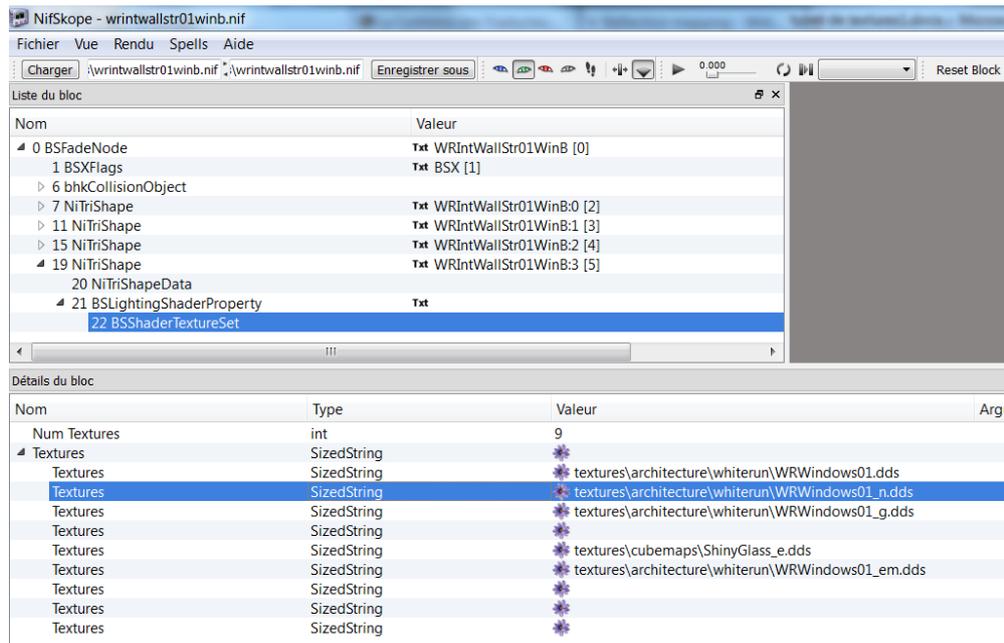
On remarquera que cette texture est partagée par les meshes de fenêtres de Blancherive. On y retrouve tous les modèles de fenêtres.

Normal map (.n.dds)

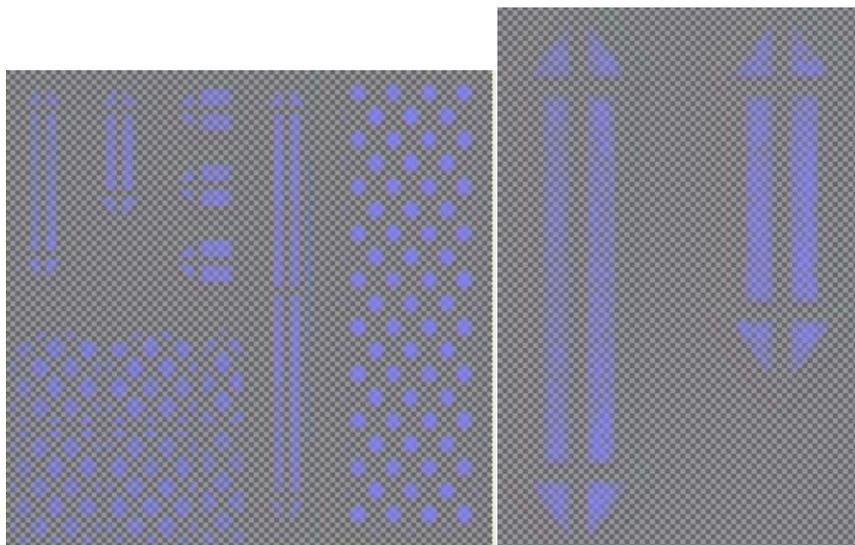
Cette texture sert à simuler un niveau de détails plus élevé des modèles 3D. Pour le même nombre de polygones, un objet utilisant une normal map semblera plus détaillé et sera globalement mieux rendu.

Cette technique permet de proposer des objets visuellement détaillés sans pour autant augmenter le nombre de polygones du modèle, ce qui aurait tendance à peser sur le moteur 3D.

Ce type de texture se reconnaît à sa teinte violacée. Le canal ALPHA (transparence) est utilisé dans cette texture et doit être sauvegardé si on en génère une.



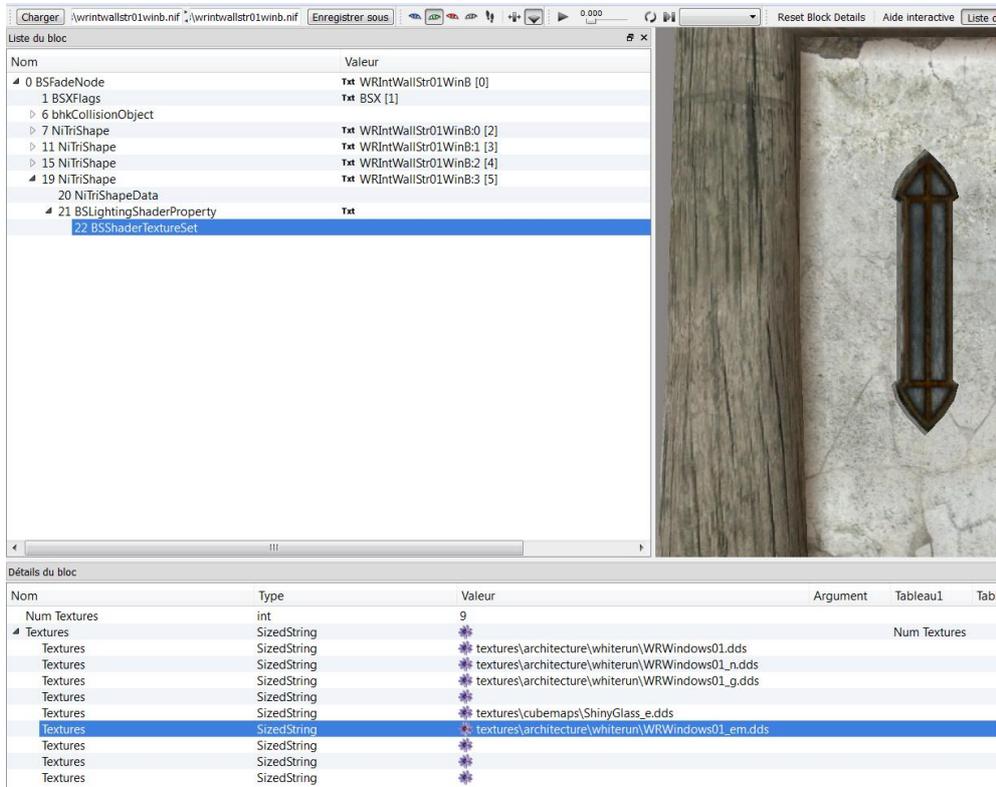
Voici à quoi ressemble notre normal map :



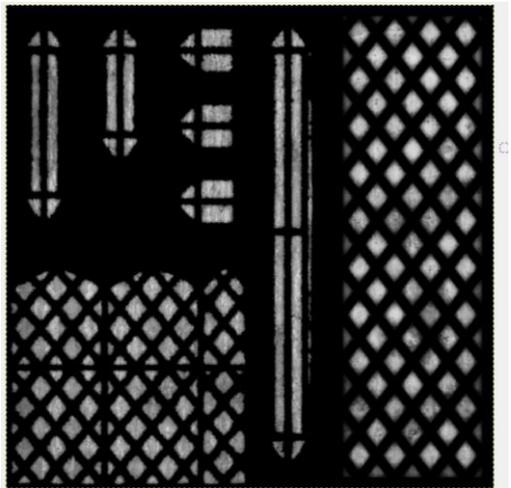
La mosaïque en arrière-plan permet de visualiser la transparence et ne fait pas partie de la texture en elle-même.

Environment map (_em.dds)

Les textures en _em.dds s'appellent environment map (ou reflection map). Ce sont elles qui rendent les usrfaces plus ou moins réfléchissantes.



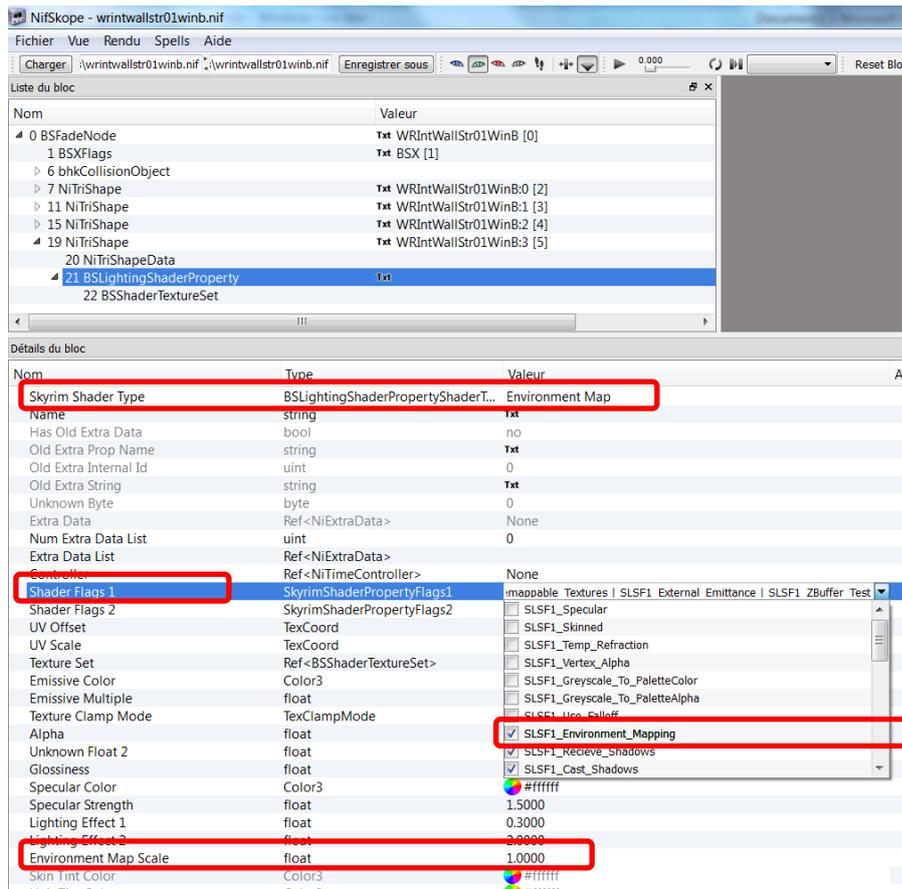
Cette texture ressemble à ça :



On remarquera que cette texture est partagée par les meshes de fenêtres de Blancherive.

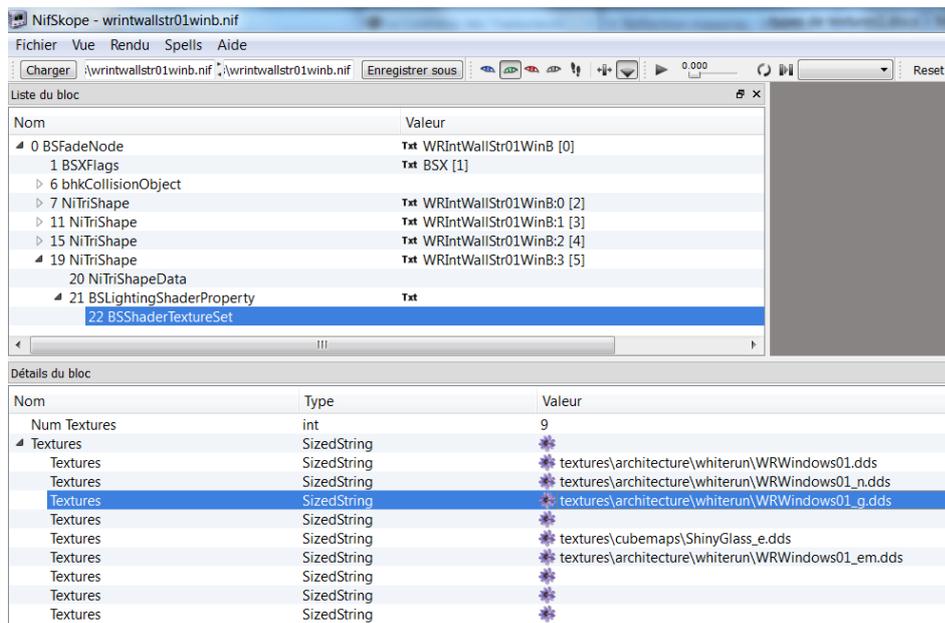
Plus on se rapproche du blanc, plus la surface est réfléchissante.

Pour qu'elle soit prise en compte, il faut que le BSLightingShader Property soit correctement configuré :

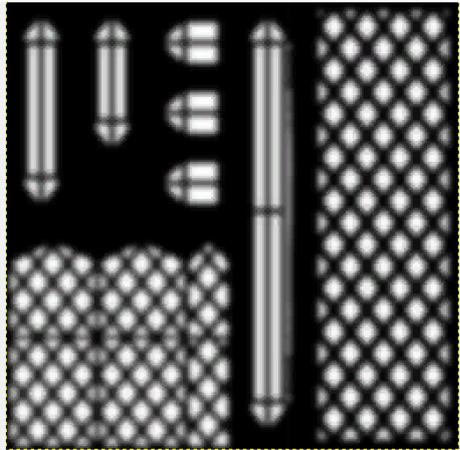


Glow map (_g.dds)

Cette texture ajoute des parties lumineuses à l'objet, indépendamment de ce qui se passe autour de l'objet (jour, nuit...). En gros, il suffit de penser aux bandes lumineuses des effets de Tron pour comprendre.



La texture glow map :



Le sujet est abordé [ICI](#) et un tuto est disponible sur le [NEXUS](#).

Cubemap (_e.dds)

Ce sont des textures qui sont placées en plus des autres sur le modèle servant souvent à représenter une matière ou quelque chose se trouvant derrière dans le cas d'une fenêtre.

La texture utilisée pour simuler les motifs du verre par Bethesda :



Petite expérience amusante : rendre les fenêtres transparentes

Tant qu'à jouer avec un mesh de fenêtre et parler de textures, nous allons faire un test.

J'ai pris ce mesh :

Data\Meshes\architecture\whiterun\wrinteriors\wrshackinterior\wrshintalcove01win

Si je prends une cubemap représentant un lieu et que je mette provisoirement des textures neutres aux autres emplacements.

La cubemap (elle est dans le bon sens) :



En regardant dans le CK, on voit la porte :



Si on se déplace par rapport à cette fenêtre, le reste de la cubemap apparaît comme si on voyait à travers la fenêtre :



Si on se rapproche, on voit plus d'éléments de la cubemaps :



La cubemap peut donc servir à simuler le fait que l'on puisse voir à travers les fenêtres.

Ce test a été effectué avec une cubemap de mauvaise qualité et des textures quasiment vierges. Il est normal de que cela soit moche. En jouant sur les textures, il y a moyen d'obtenir facilement des fenêtres transparentes de qualité.

Il ne s'agit que d'une présentation succincte destinée à ceux qui font leurs premiers pas dans le monde des textures de Skyrim.

Avis aux moddeurs, des captures d'écrans effectuées à l'extérieur des fenêtres et retravaillées permettent de faire les cubemaps nécessaires à un mod de fenêtres transparentes.

Je ne le fais pas par manque de temps mais si un moddeur a besoin d'aide pour rendre nos fenêtres transparentes, la Forge reste ouverte.

Bon modding.

Gérauld.