

Traduire un jeu Wii de A à Z

Voilà je fais ce (premier) tuto dans le but d'aider des fans à éventuellement traduire des jeux ou continuer des patches de traduction déjà existants. Cependant cela demande un minimum d'expérience et de connaissance. Notamment en ce qui concerne la langue « source » que vous voulez traduire ainsi que le hack en règle générale. Il est **TRÈS** fortement conseillé de savoir comment fonctionne un éditeur hexadécimal.

Introduction

Tout d'abord, il faut savoir que traduire un jeu n'ai pas chose aisée. De plus, la façon de traduire un jeu n'est pas forcément la même d'un jeu à l'autre. Que ce soit à cause de l'encodage du texte, de fichiers compressés/chiffrés ou que tout le texte se situe dans le main.dol, dans une image ou ailleurs.

Cependant le principe reste relativement le même pour chaque jeu :

1. Trouver les données à traduire.
2. Les extraire, puis si nécessaire les modifier.
3. Les réinjecter (et prier pour que ça marche).

Comme les façons de traduire peuvent différer d'un jeu à l'autre. Nous allons procéder en différentes étapes :

- B.A.-BA
- La modification des textes
- La modification des images/textures
- La traduction et ses outils
- La modification de fichiers audio
- La modification de fichiers vidéo

Prérequis

En général

- ISO à traduire x2 (l'original, et celui à traduire)
- Wiiscrubber 1.4 (voire Wiiscrubber 1.3 MOD)
- Outils de décompression Wii
- Éditeur Hexadécimal (MadEdit est fortement conseillé !)
- JWPce (Pour traduire du Japonais)
- Pack d'exemple (textes)

Pour les images

- BrawlBox **ET/OU**
- SZS Modifier **ET/OU**
- CTools (SZS Tool/TPL Editor)
- Logiciel de retouche d'images (Photoshop, Gimp)

Pour l'audio

- BrawlBox
- BRSAR Patcher by Hamradio
- BRSTM Converter
- BRSTM Maker (SP)
- Logiciel de retouche audio (Audacity)

Pour la vidéo

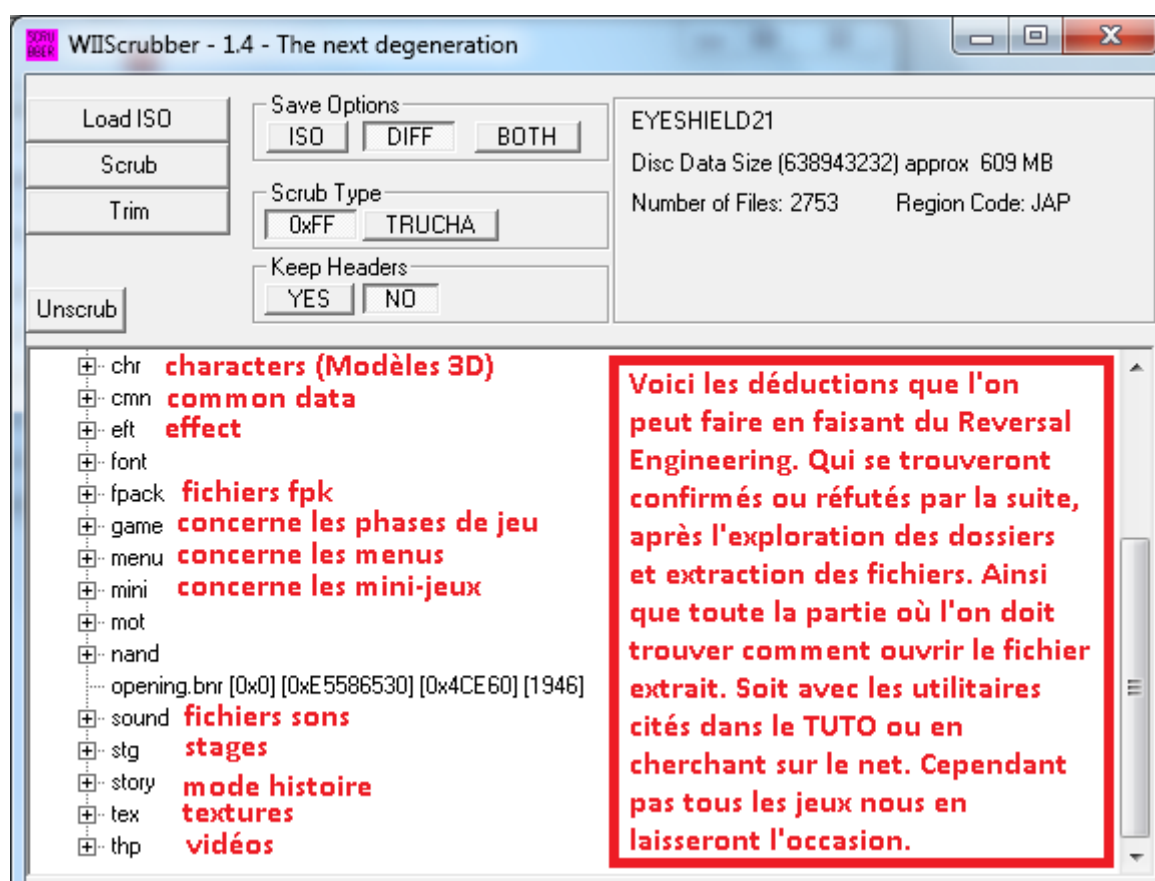
- SUPER (c)
- THP GUI
- Logiciel de retouche vidéo (Adobe Premiere Pro, Sony Vegas, Windows Movie Maker)

B.A.-BA

Ceci sera la base de la base et pourra éventuellement servir à traduire autre chose que des jeux Wii si vous avez les outils nécessaires.

a) Trouver des données à traduire

Pour commencer, le mieux est de faire de l'ingénierie inverse. C'est à dire vous mettre à la place des développeurs : « Si j'avais à mettre des images/textes quelque part, où les mettrais-je ? ». Là c'est pareil, donc il suffit de prendre WiiScrubber et de parcourir l'ISO à la recherche de fichiers texte, image ou audio. Puis de chercher des mots-clés (en général en anglais) dans le nom des fichiers et dossiers du genre :

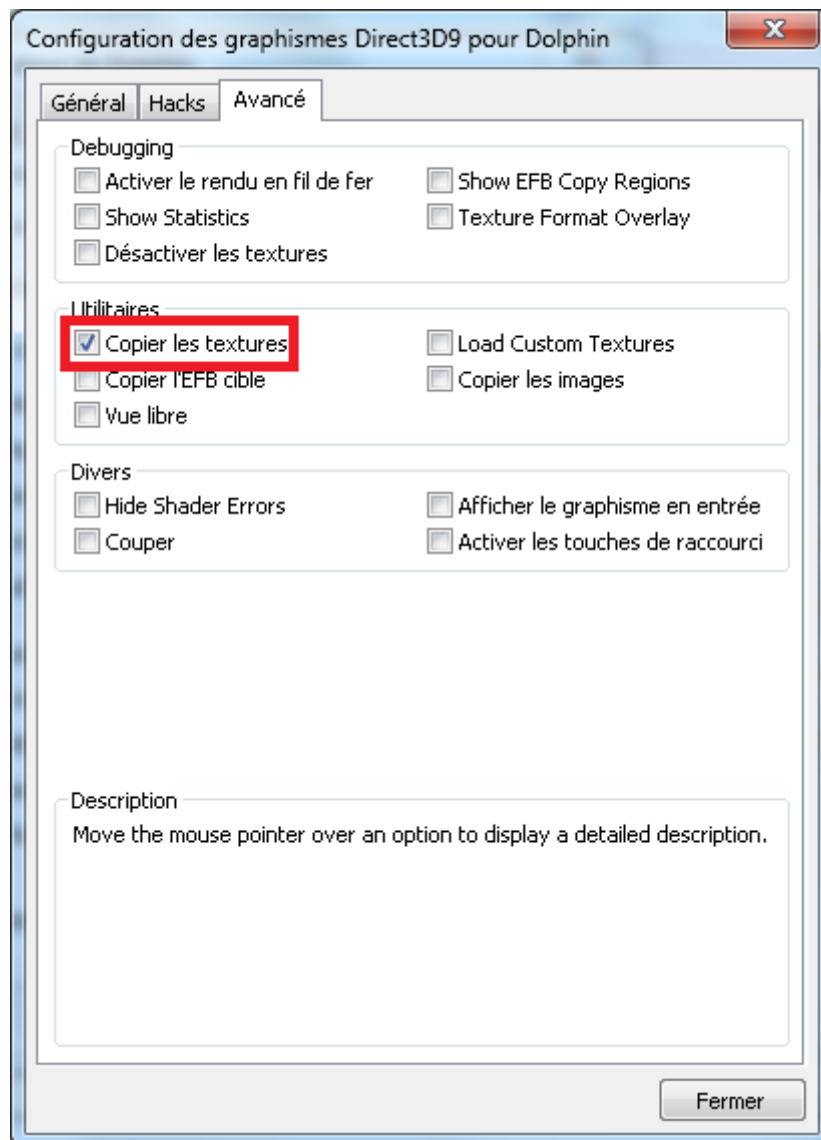


En général, les fichiers et dossiers auront des noms relatifs à leurs fonctions dans le jeu, comme cité plus haut. Cependant, il arrive aussi que les développeurs soient plus malins et qu'ils préfèrent les nommer sous forme de nombre comme XXXX.brres (*XXXX une suite de nombre, on ne peut rien en déduire*) voire sous d'autres formes toutes aussi cryptiques.

Ce qui ne nous arrange pas sauf s'il on connaît la nature du fichier et que l'on puisse l'examiner comme par exemple : un fichier image, audio, texte, etc.

Une fois ces données trouvées, référez-vous aux méthodes pour les extraire, les modifier puis les réinjecter (*voir plus bas*).

NOTE : Vous pouvez vous aider de l'émulateur Dolphin pour savoir si les textes affichés sont des textes contenus dans un fichier ou alors des textures. Il suffit pour cela d'autoriser le "Dump des textures" dans les paramètres de Dolphin. Si une image contenant le texte affiché est présente dans le répertoire BINGO c'est une image.

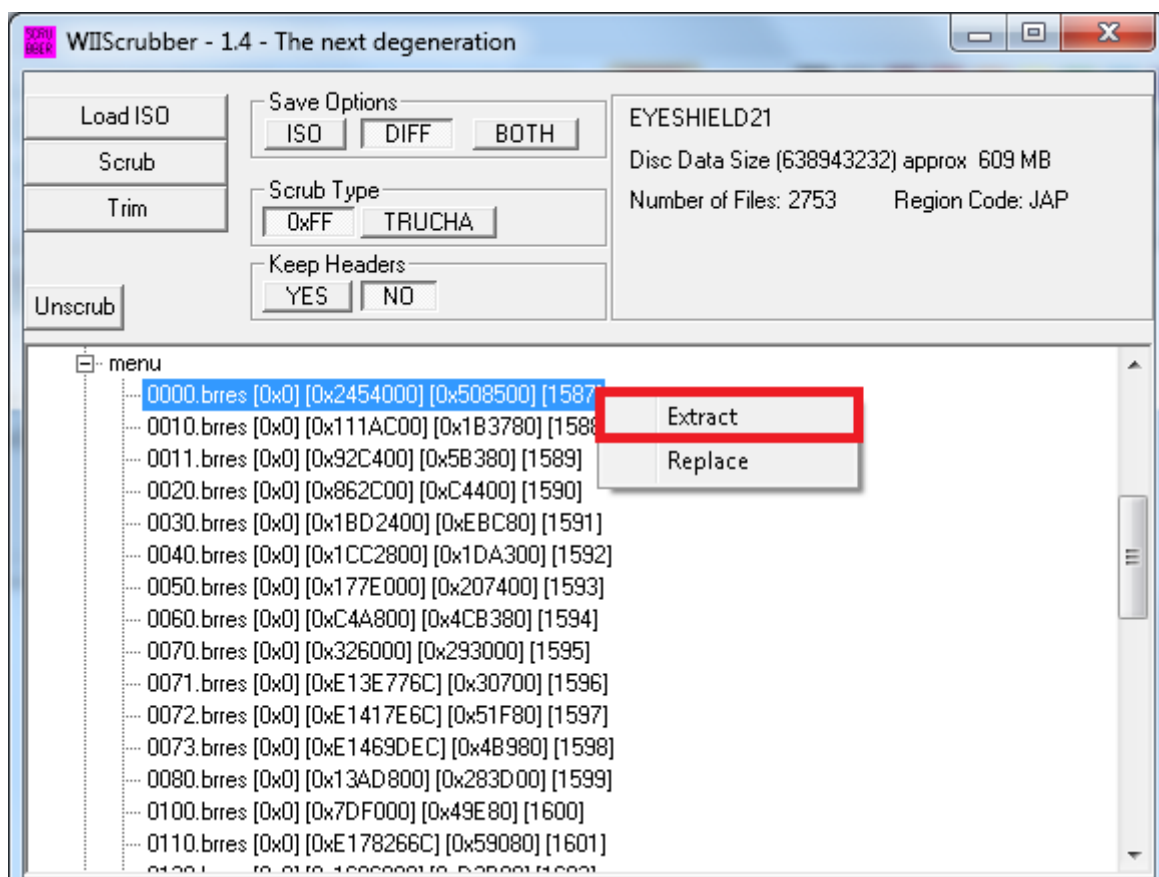


b) Extraire/Modifier/Réinjecter

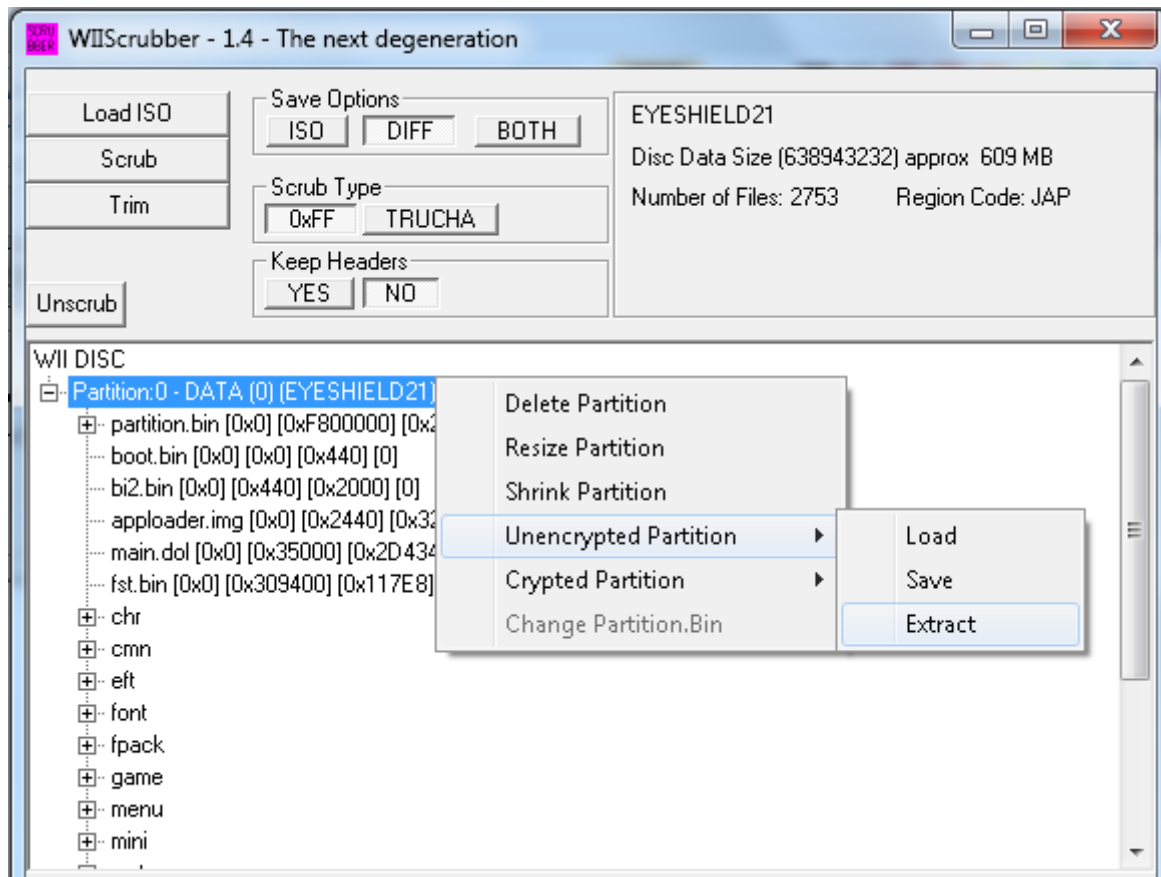
Pour extraire un fichier d'un ISO Wii, il faut :

1. Ouvrir WiiScrubber.
2. Charger son ISO.
3. Dérouler la partition DATA,
4. Faire clic droit sur un fichier puis « Extract » (*WiiScrubber mod v1.3 peut extraire des dossiers*).

Dès l'extraction terminée vous pourrez modifier le fichier extrait.



Vous pouvez également extraire tout le contenu de l'ISO, ce qui peut être plus pratique lorsque vous avez beaucoup de modifications à faire.

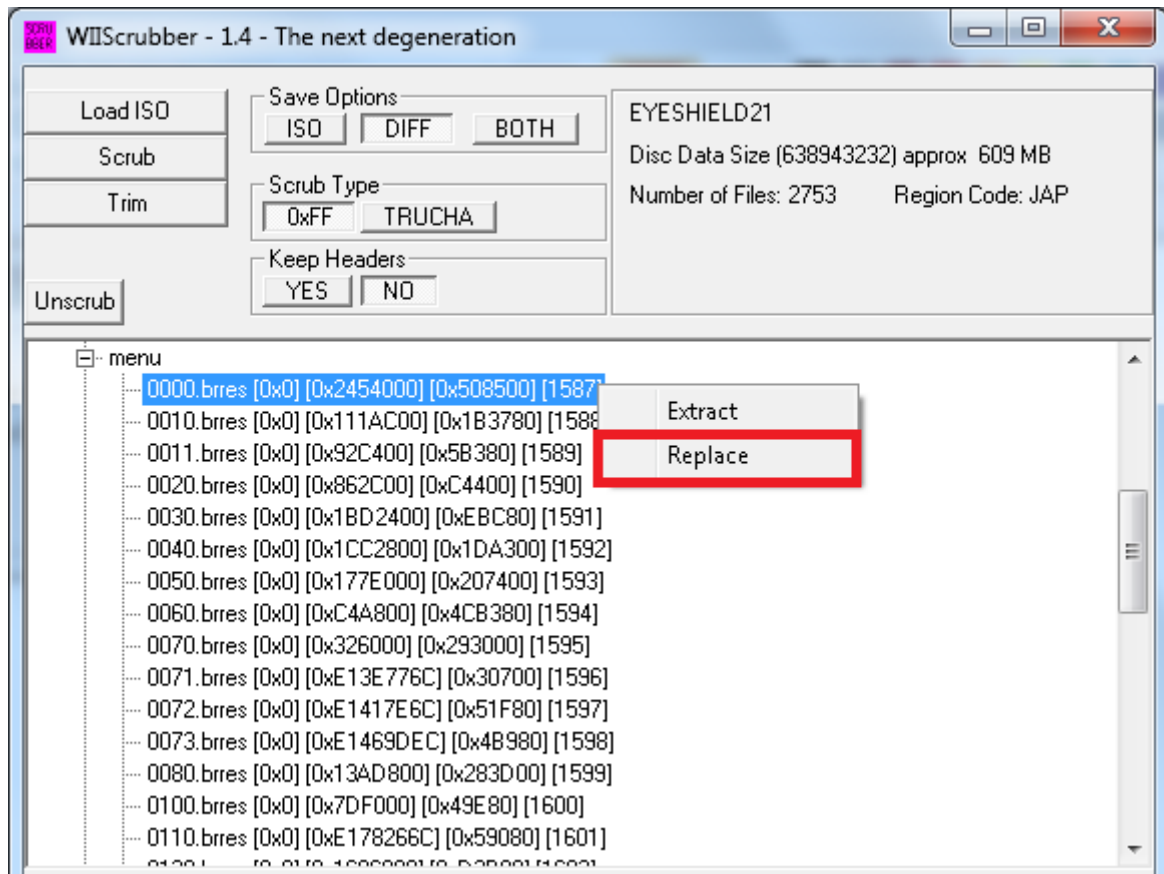


S'il s'agit :

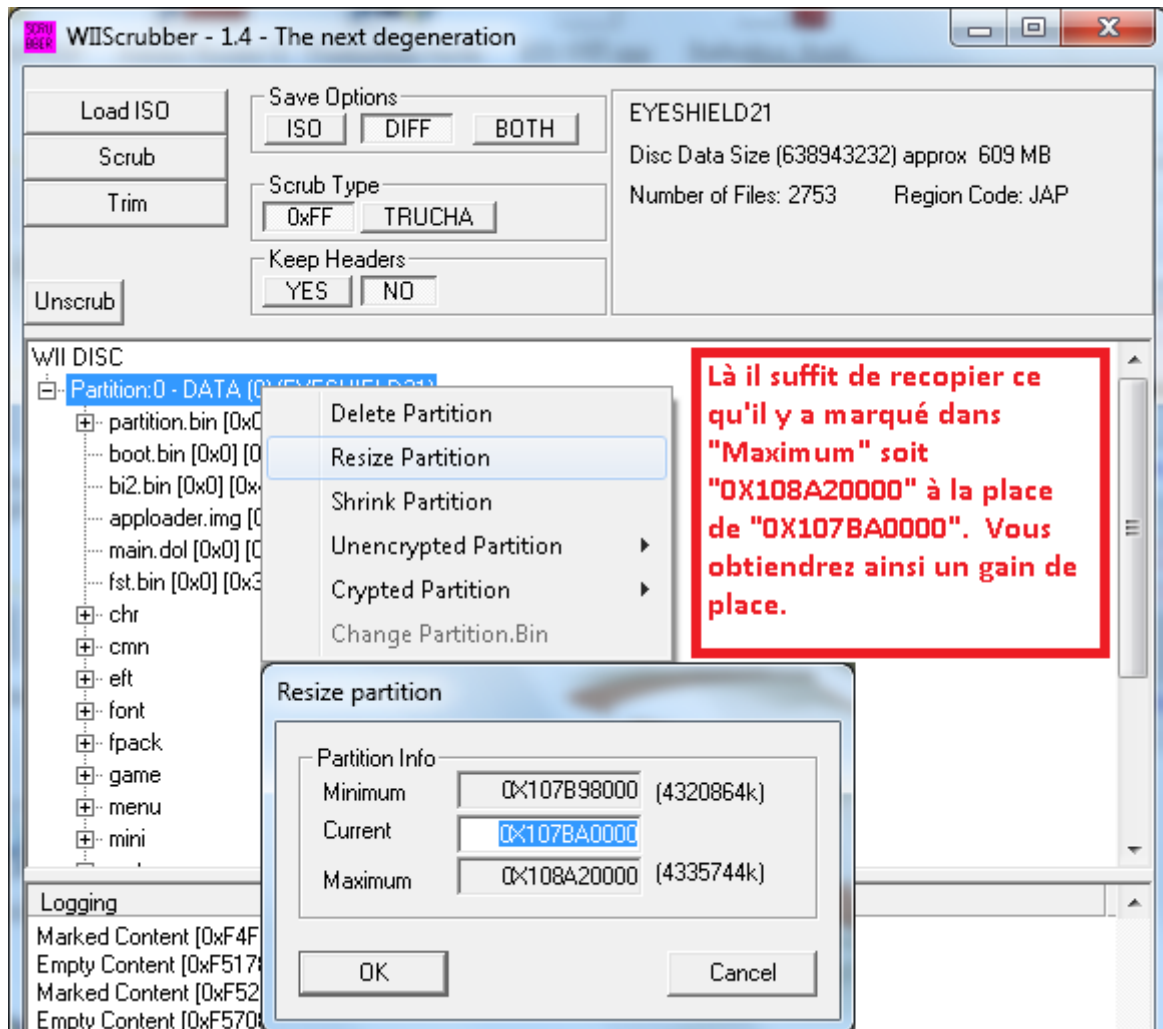
- d'un fichier contenant du texte : utiliser un éditeur hexadécimal (cf. « La modification des textes »)
- d'un fichier contenant des images/textures : extraire/convertir le fichier en un fichier image compatible à l'aide des outils ci-dessus et le modifier avec un logiciel de retouche d'images. Une fois l'image modifiée, la reconverter au format d'origine ou la réinjecter s'il s'agit d'un fichier présent dans une archive (*contenant des images/textures comme les fichiers BRSAR, par exemple*) avec les mêmes outils. (cf. « La modification des images/textures »)
- d'un fichier audio : même principe que pour les images. (cf. « La modification de fichiers audio »)
- d'un fichier vidéo : idem. (cf. « La modification de fichiers vidéo »)

Ensuite, réinjecter les fichiers modifiés dans l'ISO à l'aide de Wiiscrubber :

- Sélectionner le fichier à remplacer puis : « Clic droit -> Replace »



S'il vous manque de la place, supprimer la **partition UPDATE** dédiée aux mises à jour de la Wii. Pour ce faire, « Clic Droit -> Resize partition » sur la **partition DATA** et choisir la taille maximale.



Si ça ne fonctionne toujours pas, vérifier que votre fichier modifié ne dépasse pas trop la taille du fichier original sinon ça peut ne pas marcher.

NOTE : Il est conseillé de travailler sur un ISO Wii **NON-trimé NI compressé** sinon vous serez à coup sûr en manque de place.

c) Compression/Chiffrement

Il existe différents moyens pour compresser/chiffrer un fichier. Dans le pack d'outils de décompression vous trouverez certainement votre bonheur. Mais il est possible que certains fichiers soient chiffrés/compressés dans un format inconnu par les outils cités plus haut. Si tel est le cas, chercher des informations relatives à l'extension du fichier (Ex: « .mrg » s'il s'appelle « 0001.mrg ») et au jeu Wii en question (sur des sites comme gbatemp par exemple). Si votre recherche n'a pas porté ses fruits... Je n'ai qu'un seul outil ([QuickBMS](#)). Je ne vous apprendrais pas à vous en servir et de toute façon les instructions sont sur le site officiel du programme. Il permet de tester une multitude d'algorithmes de format de compression. Sont même présents, des scripts sur cette page pour des formats de fichier propres à certains jeux.

La modification des textes

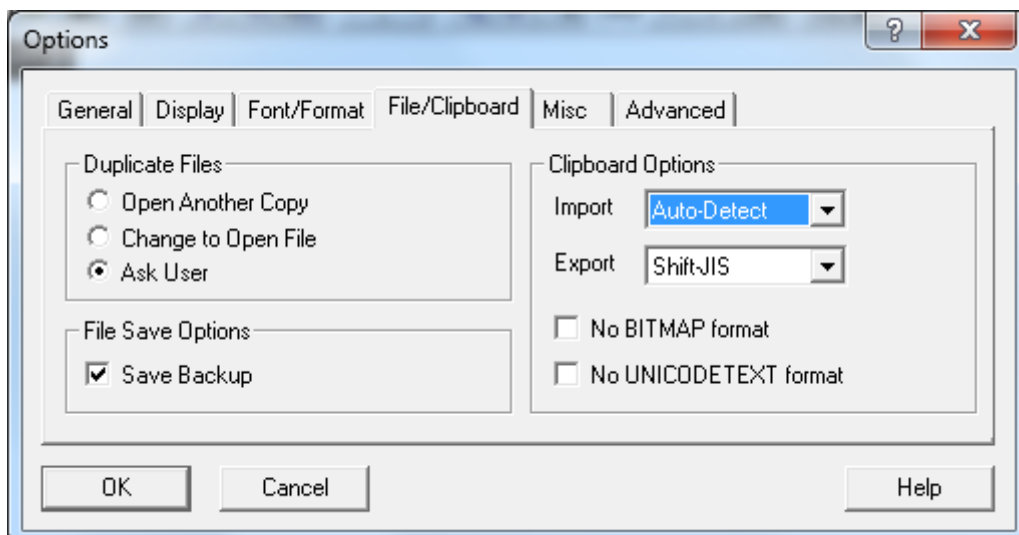
a) Encodage du texte

En ce qui concerne les textes, en fonction des langues l'encodage peut changer d'où le fait que les logiciels MadEdit et JWPce sont conseillés.

D'une part MadEdit car il possède un **mode hexadécimal** (Menu : View -> Hex Mode ou Alt+3) et que l'on peut régler l'encodage du texte (Menu : View -> Encoding). C'est particulièrement utile lorsque l'on doit traduire du texte d'une langue orientale comme le Japonais, le Chinois, le Coréen... Il affichera le contenu en hexadécimal à gauche et les caractères correspondants à l'encodage sélectionné à droite. On peut même télécharger des dictionnaires, notamment des dictionnaires japonais-français.

Puis JWPce qui permet de « taper » des caractères tel que des Hiragana, Katakana, Kanji en accédant à la table des caractères (CTRL+T - Menu: Kanji -> JIS Table). Très utile lorsque l'on doit traduire une image en japonais. On peut aussi modifier dans le menu : « Utilities -> Option -> File/Clipboard ». Vous pourrez alors choisir l'encodage d'import/export d'un fichier et du presse-papier. Exemple de ma configuration JWPce (pour traduire la JAP) :

Menu < Utilities => Options (CTRL + O) => File/Clipboard> et de mettre <Import : Auto-Detect> et <Export : Shift-JS>



b) Modifier les textes

Une fois le texte à modifier trouvé et le bon encodage sélectionné avec MadEdit il ne vous reste plus qu'à le traduire soit à la volée via l'éditeur hexadécimal ou via des patches grâce à des logiciels divers. Personnellement je ne m'attarderais pas non plus sur les moyens de patcher un fichier (certains éditeurs le font). D'habitude je traduis à la volée c'est à dire que je remplace (**ET PAS SUPPRIMER**, la taille du fichier texte **ne doit PAS BOUGER**) chaque caractère par ce qui constituera ma traduction à l'aide du clavier.

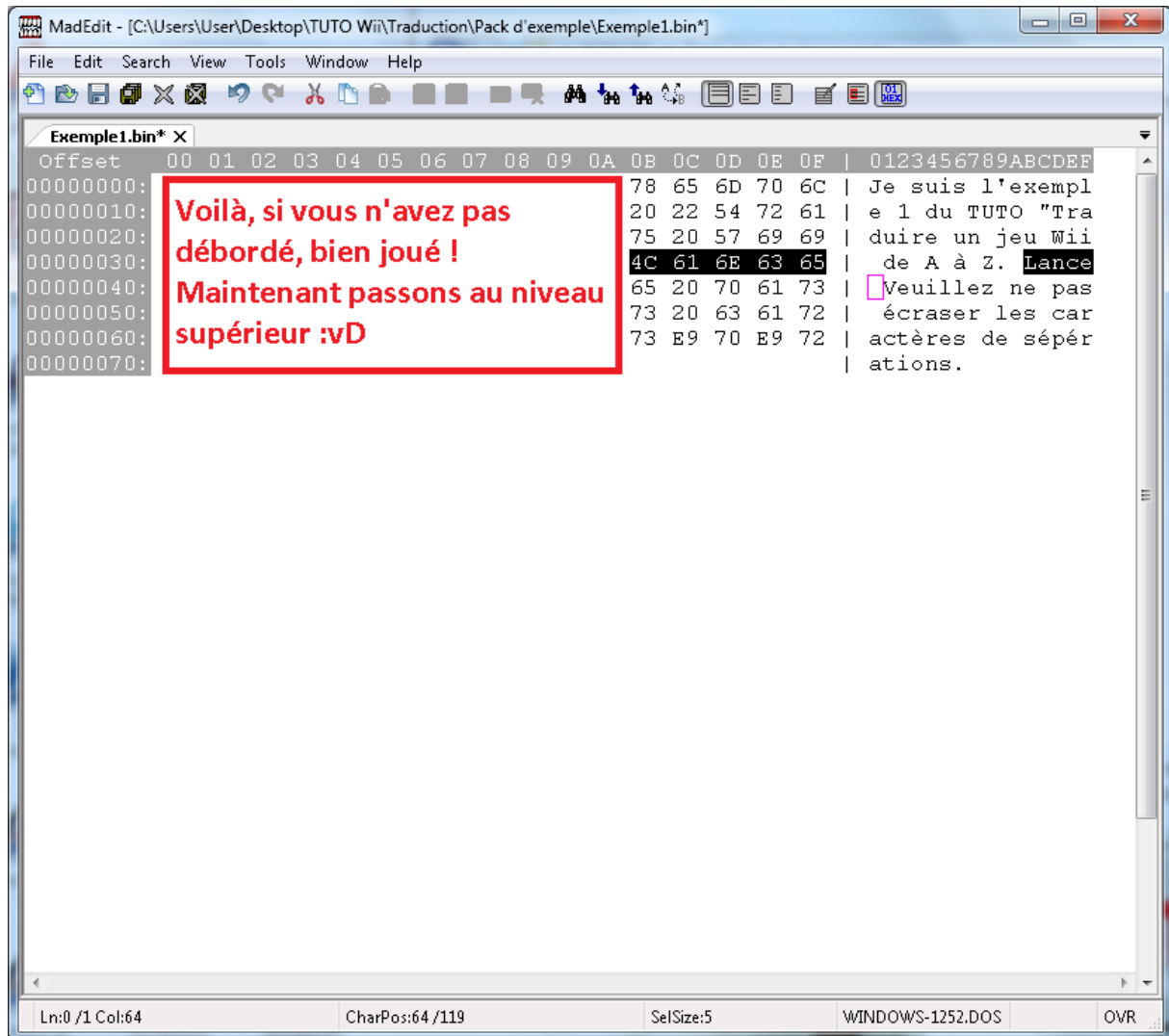
NOTE : Il est impératif de se référer à la partie « c) Limites ».

Exemples:

Ex1 – S'il y a écrit en anglais « Spear » je le traduis par : « Lance »

```
MadEdit - [C:\Users\User\Desktop\TUTO Wii\Traduction\Pack d'exemple\Exemple1.bin]
File Edit Search View Tools Window Help
Exemple1.bin X
Offset 00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F | 0123456789ABCDEF
00000000: Traduisez le texte en surbrillance ;v) | Je suis l'exempl
00000010: | e 1 du TUTO "Tra
00000020: | duire un jeu Wii
00000030: 20 64 65 20 41 20 E0 20 5A 2E 00 53 70 65 61 72 | de A à Z. Spear
00000040: 00 56 65 75 69 6C 6C 65 7A 20 6E 65 20 70 61 73 | Veuillez ne pas
00000050: 20 E9 63 72 61 73 65 72 20 6C 65 73 20 63 61 72 | écraser les car
00000060: 61 63 74 E8 72 65 73 20 64 65 20 73 E9 70 E9 72 | actères de sépér
00000070: 61 74 69 6F 6E 73 2E | ations.
Ln:0 /1 Col:64 CharPos:64 /119 SelSize:5 WINDOWS-1252.DOS OVR
```

Je ne dois pas supprimer des caractères **JUSTE les remplacer** (par du vide s'il faut, voir « c) Limites »)



The screenshot shows the MadEdit application window with the following content:

```
MadEdit - [C:\Users\User\Desktop\TUTO Wii\Traduction\Pack d'exemple\Exemple1.bin*]
File Edit Search View Tools Window Help
Exemple1.bin* X
Offset 00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F | 0123456789ABCDEF
00000000: 78 65 6D 70 6C | Je suis l'exempl
00000010: 20 22 54 72 61 | e 1 du TUTO "Tra
00000020: 75 20 57 69 69 | duire un jeu Wii
00000030: 4C 61 6E 63 65 | de A à Z. Lance
00000040: 65 20 70 61 73 | Veuillez ne pas
00000050: 73 20 63 61 72 | écraser les car
00000060: 73 E9 70 E9 72 | actères de séper
00000070: | ations.
```

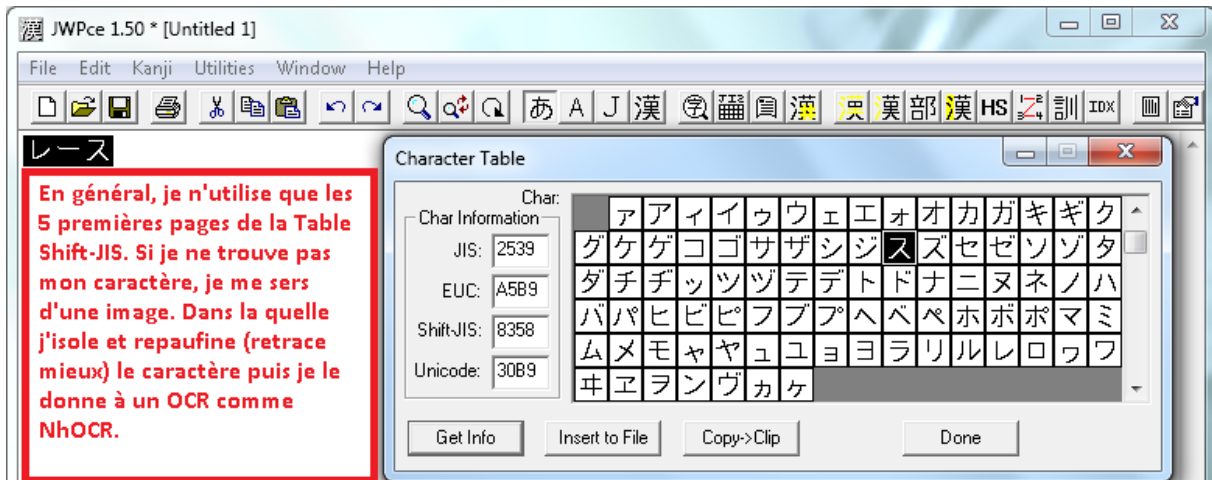
A red box highlights the hex values 4C 61 6E 63 65 on line 00000030, with the following red text inside:

**Voilà, si vous n'avez pas débordé, bien joué !
Maintenant passons au niveau supérieur :vD**

At the bottom of the window, the status bar shows: Ln:0 /1 Col:64 CharPos:64 /119 SelSize:5 WINDOWS-1252.DOS OVR

Ex2 – « レース » (en Japonais) => « Course »

Vous pouvez aussi vous entrainez à le taper sur JWPce :vP



NOTE : Un caractère Japonais est codé sur 2 octets (donc 2 caractères sur lesquelles on peut écrire. C'est pour ça qu'il est important de se renseigner sur l'encodage du texte) tandis que les caractères ASCII standard sont codés sur 1 octet (soit l'alphabet occidental, caractères spéciaux exclu). D'où le fait que j'ai le droit d'écrire « Course » qui est 2 fois plus grand.

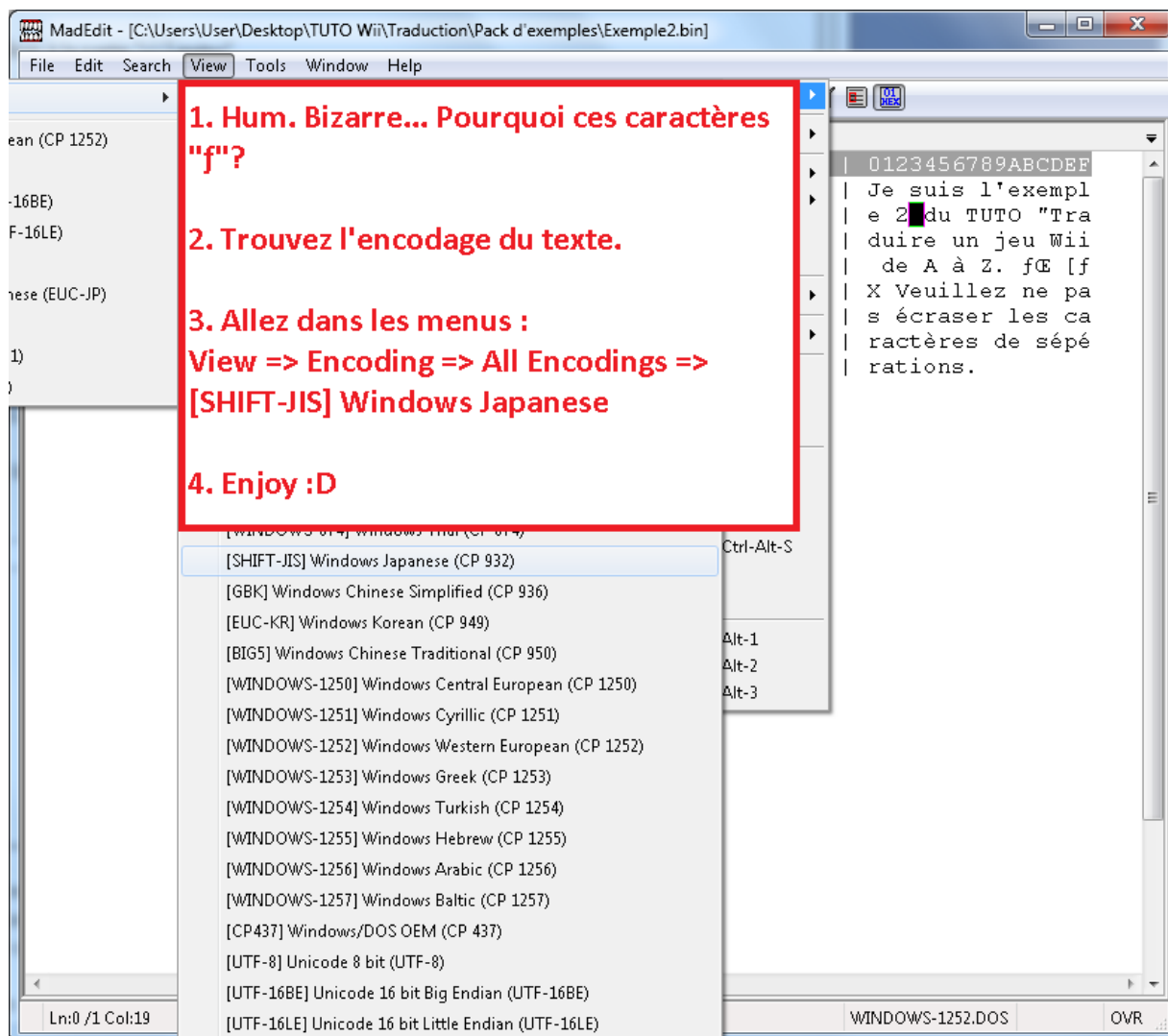
Donc à gauche on a $3 \times 2 = 6$, puis à droite $6 \times 1 : 6 = 6$ donc on a rien dépassé, tout est en ordre. Néanmoins ceci est valable en Shift-JIS mais pas forcément dans les autres encodages.

Ex3 – Lancer MadEdit en mode hexadécimal. Dans la colonne de gauche (soit celle hexadécimal) créer un nouveau fichier et entrer:

Code :

`83 8C 81 5B 83 58`

Puis trouver l'encodage =vP



Solution :

Shift-JS ; View -> Encoding -> [SHIFT-JS] Windows Japanese (CP932)

C'est l'encodage le plus utilisé en ce qui concerne les textes Japonais que j'ai eu l'occasion de traduire. C'est également le même mot que dans l'exemple 2.

Exemple2.bin X

Offset	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	0123456789ABCDEF
00000000:																	Je suis l'exempl
00000010:																	e 2 du TUTO "Tra
00000020:																	duire un jeu Wii
00000030:	20	64	65	20	41	20	E0	20	5A	2E	00	83	8C	81	5B	83	de A à Z. ㄣ
00000040:	58	00	56	65	75	69	6C	6C	65	7A	20	6E	65	20	70	61	Veillez ne pa
00000050:	73	20	E9	63	72	61	73	65	72	20	6C	65	73	20	63	61	s ㄣ raser les ca
00000060:	72	61	63	74	E8	72	65	73	20	64	65	20	73	E9	70	E9	ract ㄣ es de ㄣ ㄣ
00000070:	72	61	74	69	6E	6E	73	2E									ations.

Exemple2.bin* X

Offset	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	0123456789ABCDEF
00000000:	4A	65	20	73	75	69	73	20	6C	65	78	65	6D	70	6C		Je suis l'exempl
00000010:	65	20	32	20	64	75	20	54	55	4F	20	22	54	72	61		e 2 du TUTO "Tra
00000020:	64	75	69	72	65	20	75	6E	20	6A	65	75	20	57	69	69	duire un jeu Wii
00000030:	20	64	65	20	41	20	E0	20	5A	2E	00	43	6E	75	72	73	de A à Z. Cours
00000040:	65	00	56	65	75	69	6C	6C	65	7A	20	6E	65	20	70	61	e Veuillez ne pa
00000050:	73	20	E9	63	72	61	73	65	72	20	6C	65	73	20	63	61	s ㄣ raser les ca
00000060:	72	61	63	74	E8	72	65	73	20	64	65	20	73	E9	70	E9	ract ㄣ es de ㄣ ㄣ
00000070:	72	61	74	69	6E	6E	73	2E									ations.

NOTE: Là on voit bien qu'un caractère ASCII prend un octet et les caractères japonais 2. Cependant pas tous les encodages font cela. C'est le cas de l'Unicode UTF-16, les caractères ACSII prendront 2 octets comme ceux japonais. Là ça pourrait éventuellement poser problème pour la traduction.

ASTUCE: Si le caractère "f" apparait massivement (et que vous utilisez un encodage de base "occidentale". Exemple: Windows Western European), il y a 99% de chance que ce soit du SHIFT-JIS et donc du texte à traduire... Ou que ce soit crypter ><'

Ln:0 /1 Col:65 CharPos:65 /120 SelSize:6 SHIFT-JIS.DOS OVR

c) Limites

Lorsque vous traduisez du texte vous devez garder deux choses à l'esprit :

- Ne pas supprimer d'octet (parce que ça décalerait tous les textes et fouetterais le bord*!)
- Ne pas dépasser le nombre de caractère (en hexadécimal) du texte original

Ces deux points sont très importants. Car en général le texte est stocké dans une sorte de tableau. Dans ce tableau sont stocké des pointeurs qui mémorisent les adresses des différents textes. Et donc s'il on supprime un octet les adresses dans les pointeurs n'auront pas changé mais celle des textes oui. Les jeux (en général) fonctionne de la façon suivante : Il demande d'afficher du texte, ensuite il regarde le pointeur et enfin il copie tout ce qu'il y a du pointeur au caractère de séparation (qui est 00 en hexadécimal, cf. exemple). Si le texte est décalé, les caractères de séparation aussi. Donc ne jamais supprimer d'octet ! Mais en sachant cela on peut utiliser les caractères de séparation pour raccourcir du texte et donc éviter de mettre des espaces quand on a des caractères en trop.

Exemple :

Même principe que l'exemple précédent : Lancer MadEdit en mode hexadécimal. Dans la colonne de gauche (soit celle hexadécimal) créer un nouveau fichier et entrer :

Code :

```
74 78 74 31 00 74 65 78 74 65 32 00 4C 4F 4C
```

Là l'encodage ne devrait pas poser de problème =vP.

Dans ce fichier on distingue trois textes : « txt1 », « texte2 » et « LOL ».

NOTE : Lorsque je devrais parler d'un caractère en hexadécimal je le noterais \$XX. Exemple : Le caractère de séparation \$00, le caractère en hexadécimal "AD", soit \$AD.

Ce qu'il est permis de faire :

Remplacer « texte2 » par « txt2 » et utiliser l'astuce du caractère de séparation. S'il y a trop de caractère restant même après la traduction, voilà comment l'on peut procéder :

Admettons que « texte2 » soit du texte à traduire et que ma traduction soit :

Code :

```
74 78 74 31 00 74 78 74 32 20 20 00 4C 4F 4C
```

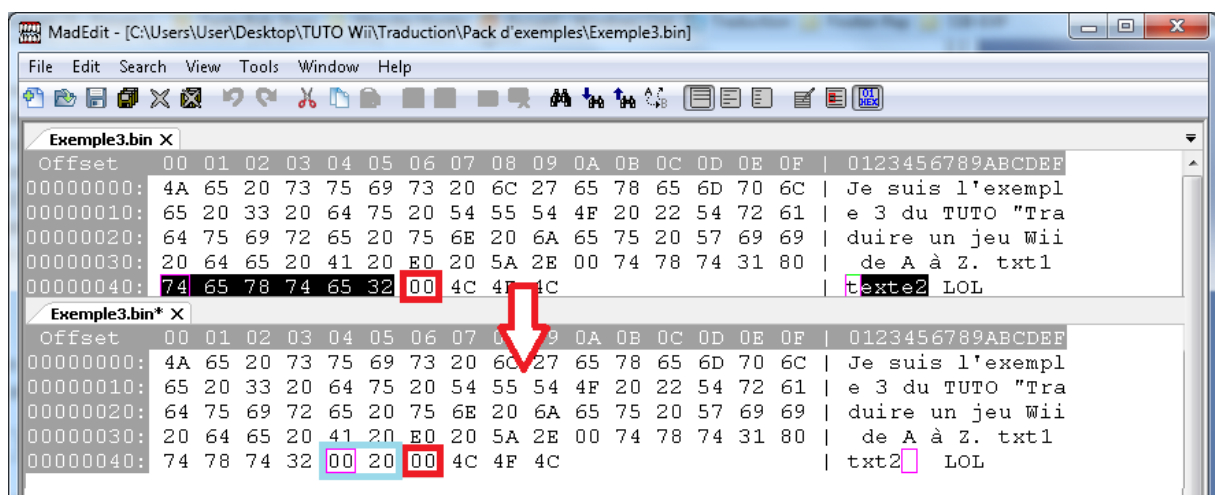
\$20 - Est le caractère hexadécimal qui représente un espace. Si le jeu demande d'afficher le second texte il va alors nous afficher « txt2 » avec 3 espaces. C'est un peu gênant surtout si le texte est centré. Mais voici la solution :

Code :

```
74 78 74 31 00 74 78 74 32 00 20 00 4C 4F 4C
```

Le jeu va parcourir le texte et comme il aura trouvé un caractère de séparation, il ne va pas se poser de question et va s'arrêter là. En admettant que les trois pointeurs se trouvaient sur le 1er octet, 6e octet et 13e octet, les pointeurs ne poseront pas de problème car la position du texte n'a pas changé. En fait, on a juste « feinté » le jeu s'il on peut dire.

HS : Même si pour ceux qui connaissent la programmation, ils savent que \$00 est le caractère qui désigne la fin d'un string (= chaîne de caractère).



d) Mise en pratique

Ici je vais appliquer tous ce que j'ai appris, dans le fichier com_data.rso du jeu Monster Hunter 3 (~Tri) (en japonais bien sûr).

1. Chercher les caractères de séparation qui délimite un texte à traduire. (Ceux encadrés)

2. Traduire le texte (Vous pouvez le sélectionner puis le copier-coller dans un document texte ou un traducteur)

3. Le modifier (Pensez impérativement à sélectionner UN octet et non plusieurs. Car sinon ça supprimera des octets !)

Offset 00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F | 0123456789ABCDEE

0004AAB0: CC 8F D8 96 BE 81 42 00 83 43 83 93 83 65 83 8A | 討伐した証。天にかざすことで天候が狂いだす不思議な指輪。捕獲したモンスターから判明した生態研究のレポート。モンスターを多種捕獲した証。全モンスターの生態と調合の極意を記録した書。ハンターノートを極めた証。伝説の竜職人が使用した道具。汚れているのにどこか神々しい。装備コレクターの証明。インテリ

0004ABD0: 83 41 83 7D 83 58 83 5E 81 5B 82 CC 8A B4 8E D3 | アマスターの感謝の印。多くのイン

0004ABE0: 82 CC 0A 88 F3 81 42 91 BD 82 AD 82 CC 83 43 83 | テリアを開発し

0004ABF0: 93 83 65 83 8A 83 41 82 F0 0A 8A 4A 94 AD 82 B5 | た証。巨大な敵を

0004AC00: 82 BD 8F D8 81 42 00 8B 90 91 E5 82 C8 93 47 82 | 狩り続けた証。

0004AC10: F0 8E EB 82 E8 91 B1 82 AF 82 BD 8F D8 81 42 0A | 金色の冠に何人も

0004AC20: 8B E0 90 46 82 CC 8A A5 82 C9 89 BD 90 6C 82 E0 | のハンターが挑み

0004AC30: 82 CC 83 6E 83 93 83 5E 81 5B 0A 82 AA 92 A7 82 | 挫折してきた幻の

0004AC40: DD 8D C1 90 DC 82 B5 82 C4 82 AB 82 BD 8C B6 82 | 勲章。ちっこい

0004AC50: CC 8C 4D 8F CD 81 42 00 82 BF 82 C1 82 B1 82 A2 | 敵を狩り続けた証

0004AC60: 93 47 82 F0 8E EB 82 E8 91 B1 82 AF 82 BD 8F D8 | 。実力と運を兼ね

0004AC70: 81 42 0A 8E C0 97 CD 82 C6 89 5E 82 F0 8C 93 82 | 備えたハンター

0004AC80: CB 94 F5 82 A6 82 BD 83 6E 83 93 83 5E 81 5B 0A | にか手に入れ

0004AC90: 82 CB 82 8E 82 82 8E 82 82 CB 82 82 82 82 82 82

Ln:0 /1 Col:306 136 CharPos:306 136 /750 976 SelSize:62 SHIFT-JIS.DOS OVR

En général, je sélectionne le texte sur MadEdit. Puis je le copie-colle sur JWPce ou un traducteur style Google. Je vais aussi sur dictionnaire japonais qui propose également la recherche de Romaji : pour faire simple c'est la traduction littérale en caractère ASCII. Histoire de ne pas traduire ça comme un porc.

4. Voici le rendu final. J'ai utilisé la technique du caractère de séparation pour remplir le vide. Et pour que le texte s'affiche correctement, sans cette succession d'espace.

NOTE: Si vous ne pouvez pas entrer des caractères spéciaux (é;è;à;etc...) c'est que l'encodage du jeu ne les supporte pas ! (Ce qui explique que je n'ai pas mis les "e" accents aiguë)

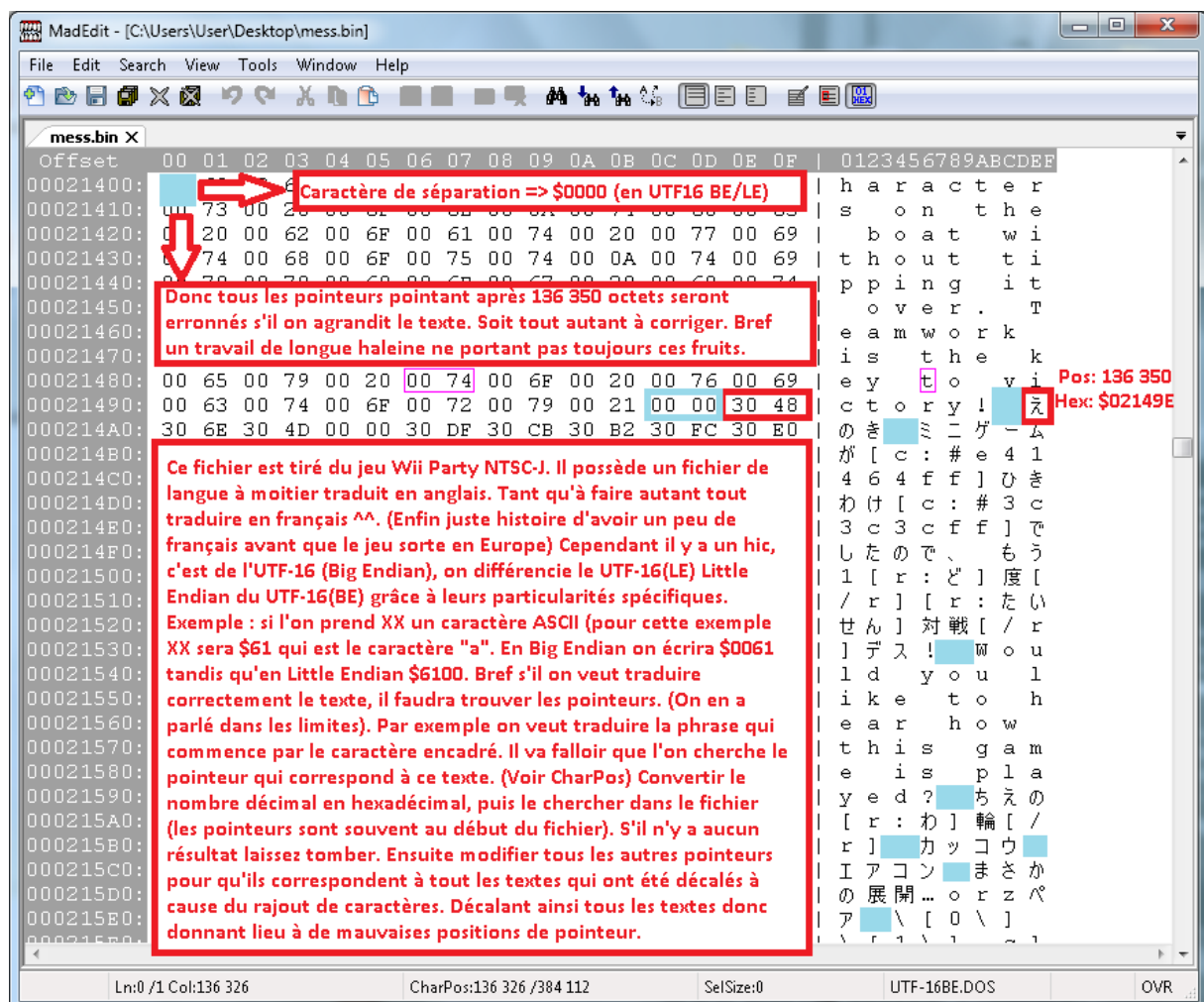
討伐した証。天にかざすことで天候が狂いだす不思議な指輪。捕獲したモンスターから判明した生態研究のレポート。モンスターを多種捕獲した証。全モンスターの生態と調合の極意を記録した書。ハンターノートを極めた証。伝説の竜職人が使用した道具。汚れているのどこか神々しい。装備コレクターの証明。 Merci ma
titre! Votre chambre a été redécorée.
■ 巨大な敵を狩り続けた証。金色の冠に何人もハンターが挑み挫折してきた幻の勲章。ちっこい敵を狩り続けた証。実力と運を兼ね備えたハンター

Voilà en ce qui concerne les textes. On a fait quasiment le tour.

e) Avancé (Pointeurs)

NOTE : Cette partie est **EXTRÊMEMENT DÉCONSEILLÉE** aux personnes n'ayant aucune base en hexadécimal. Et la programmation (même de "bas" niveau, peu importe le langage) peut être d'une grande aide.

On a parlé d'un problème dans les limites : celui des pointeurs qui délimite les textes. Or ce sont ses pointeurs qui nous restreignent à conserver le même nombre de caractère pour chaque texte. Tout ça car il pointe sur un offset (= position en hexadécimal du caractère dans le fichier) qui ne doit pas bouger.

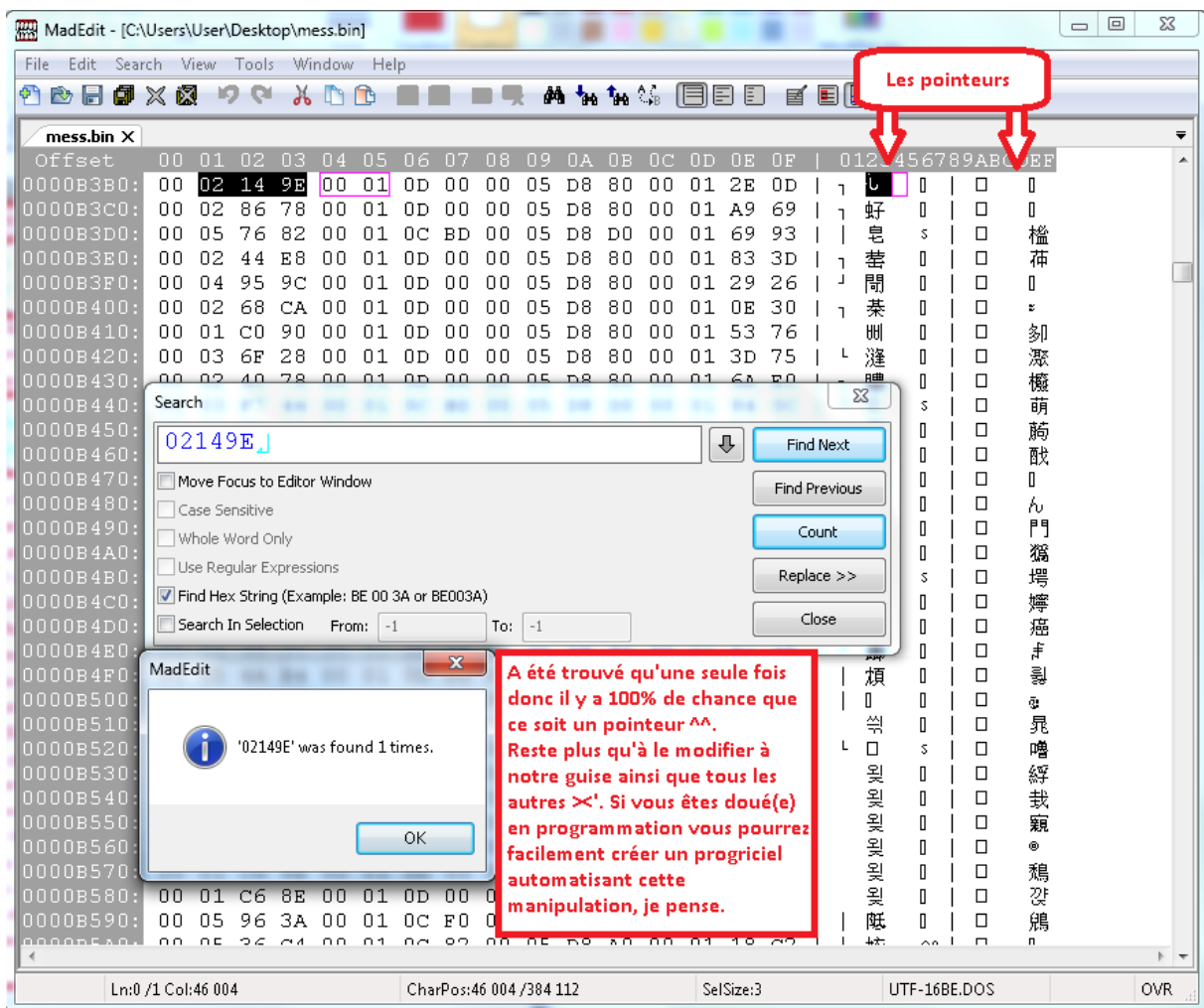


The screenshot shows the MadEdit application window with a binary file named 'mess.bin' open. The main window displays a hex dump of the file's contents, with columns for offsets (00 to 0F) and characters (01 to 0F). The text is displayed in a mix of English and Japanese characters. Several red annotations are present:

- A red arrow points to the hex value '00 00' at offset 00021436, with the text: "Caractère de séparation => \$0000 (en UTF16 BE/LE)".
- A red box contains the text: "Donc tous les pointeurs pointant après 136 350 octets seront erronés s'il on agrandit le texte. Soit tout autant à corriger. Bref un travail de longue haleine ne portant pas toujours ces fruits."
- Another red box contains the text: "Ce fichier est tiré du jeu Wii Party NTSC-J. Il possède un fichier de langue à moitié traduit en anglais. Tant qu'à faire autant tout traduire en français ^^." (Enfin juste histoire d'avoir un peu de français avant que le jeu sorte en Europe) Cependant il y a un hic, c'est de l'UTF-16 (Big Endian), on différencie le UTF-16(LE) Little Endian du UTF-16(BE) grâce à leurs particularités spécifiques. Exemple : si l'on prend XX un caractère ASCII (pour cette exemple XX sera \$61 qui est le caractère "a". En Big Endian on écrira \$0061 tandis qu'en Little Endian \$6100. Bref s'il on veut traduire correctement le texte, il faudra trouver les pointeurs. (On en a parlé dans les limites). Par exemple on veut traduire la phrase qui commence par le caractère encadré. Il va falloir que l'on cherche le pointeur qui correspond à ce texte. (Voir CharPos) Convertir le nombre décimal en hexadécimal, puis le chercher dans le fichier (les pointeurs sont souvent au début du fichier). S'il n'y a aucun résultat laissez tomber. Ensuite modifier tous les autres pointeurs pour qu'ils correspondent à tout les textes qui ont été décalés à cause du rajout de caractères. Décalant ainsi tous les textes donc donnant lieu à de mauvaises positions de pointeur.
- On the right side, a red box highlights the hex value '30 48' at offset 000214A0, with the text: "Pos: 136 350 Hex: \$02149E".

The status bar at the bottom shows: Ln:0 /1 Col:136 326 CharPos:136 326 /384 112 SelSize:0 UTF-16BE.DOS OVR

Honnêtement je n'ai pas eu l'occasion de modifier les pointeurs d'un quelconque fichier par manque de temps et d'expérience (à l'époque). Sachant que le mieux est d'utiliser un programme qui corrige au fur et à mesure la position des pointeurs en fonction de la taille du texte et des modifications qu'on y apporte. De plus certains jeux ne supportent pas que l'on modifie les pointeurs comme ça. Pourquoi ? Car certains jeux : se vérifient eux-mêmes pour éviter des erreurs ou qu'il y ai plusieurs pointeurs, que les pointeurs seuls ne servent à rien car le pointeur est lié à une information (un argument qui donnerait une certaine valeur à une partie du texte ce qui expliquerait les "%quelquechose" ou ce genre d'annotation n'ayant aucun lien avec le texte. Surement une variable), etc.



Bref, c'est loin d'être une obligation (sauf dans certains cas). Si jamais vous vous lancez, faites attention de bien faire une **LONGUE** phase de test pour être **SÛR à 200%** que **TOUT** fonctionne ! Si j'ai un conseil à vous donner pour faire un programme ce serait de :

1 – Balayer le début du fichier (l'entête contenant les pointeurs).

Sachant que les pointeurs sont à intervalle régulier.

2 – De charger dans un tableau à l'aide des pointeurs les textes.

3 – De créer une Interface graphique permettant de modifier les textes.

Avec par exemple deux colonnes. Une affichant le texte original et l'autre le texte que l'on veut rentrer. Avec le support de l'encodage du texte ainsi qu'une fonction copier-coller tant qu'à faire.

4 – Un bouton Sauvegarder. Et qui balayera le tableau de Haut en bas et qui modifiera la position des pointeurs texte par texte.

II - La modification des images/textures

a) Encodage des images

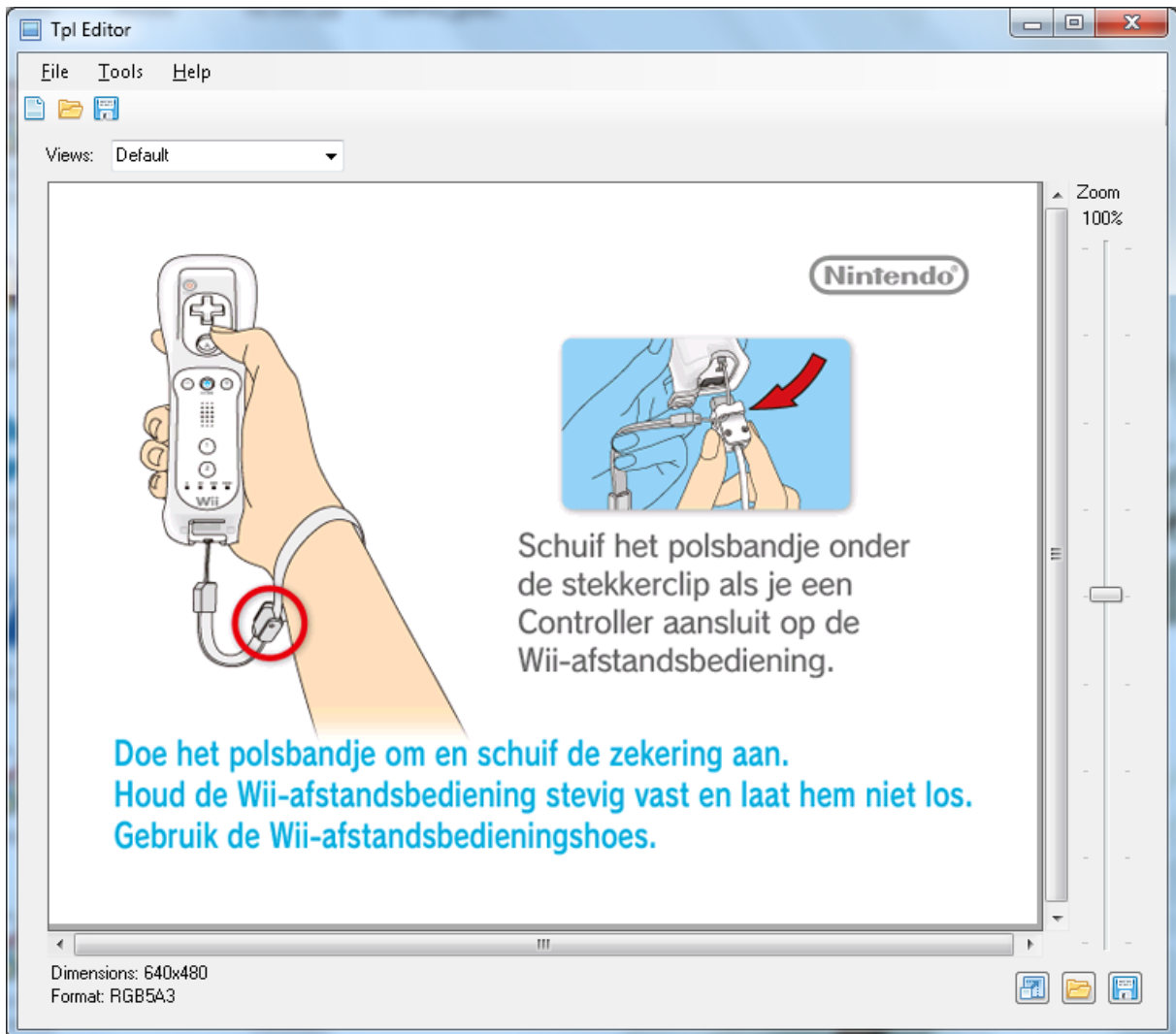
Les images, textures, modèles d'un jeu sont en général compressées. Ils peuvent aussi être dans une archive où tout ce beau monde est réuni. En règle générale ils sont dans des fichiers BRRES, TPL, MRG et autres. Cependant le principe reste le même que pour les textes, une fois que vous aurez trouvé les fichiers à modifier il ne restera plus qu'à trouver le logiciel pour les ouvrir (cf. [Wii File Type Database](#)). Là encore Dolphin peut vous être utile pour vous aider à trouver des images et des textures.

b) Modifier les images

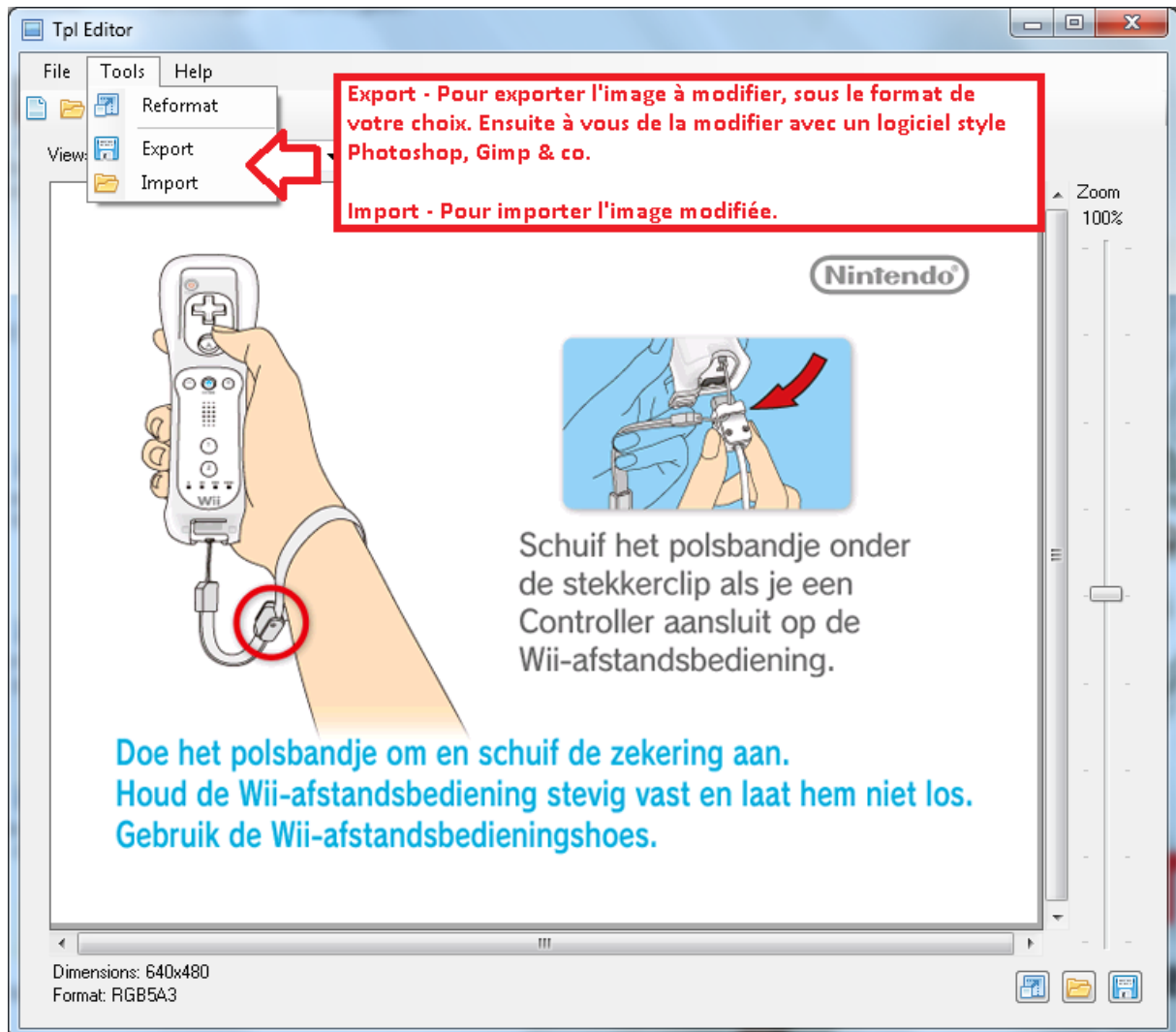
Cette fois en revanche ce n'est pas comme les textes. On ne peut pas patcher le fichier à coup d'éditeur hexadécimal à moins d'être un génie et savoir comment l'algorithme code chaque pixel de l'image. Bref pour nous le commun des mortels, on va tout simplement se servir d'outils. Ces outils comme BrawlBox, SZS Modifier ou le CTool TPL Editor vont juste nous permettre dans un premier temps d'extraire l'image. Puis viendra à nous de la modifier à notre guise. Ensuite nous referons appel à ces outils pour réinjecter les images modifiées.

1 - Avec TPL Editor

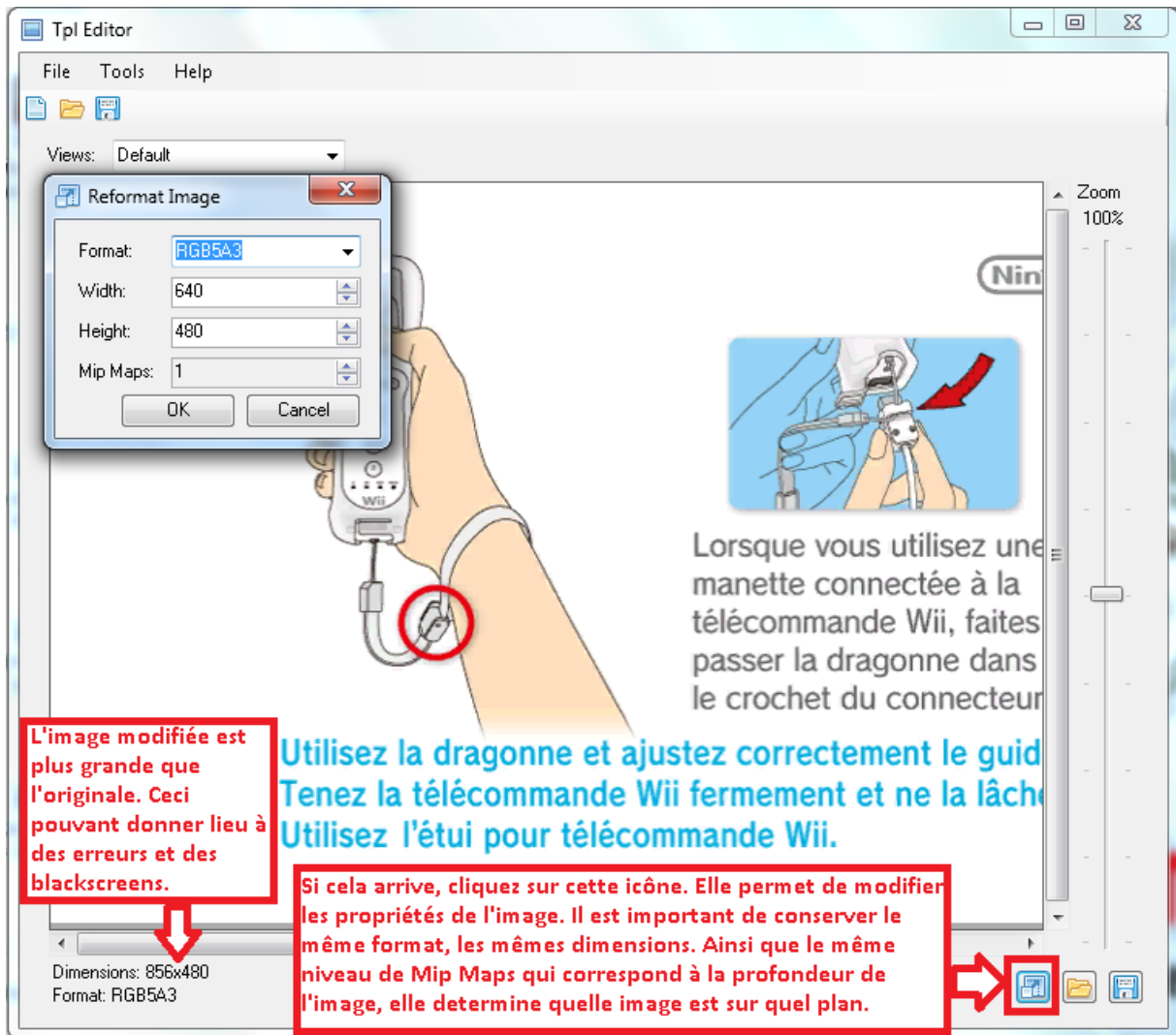
Ouvrer votre image avec TPL Editor depuis le menu *File : Open*.



Exporter l'image sous un format compatible avec votre logiciel de retouche photo. Puis une fois l'image modifiée, l'importer. Si vous n'arrivez pas à la traduire, allez à la partie « III - La traduction et ses outils ».

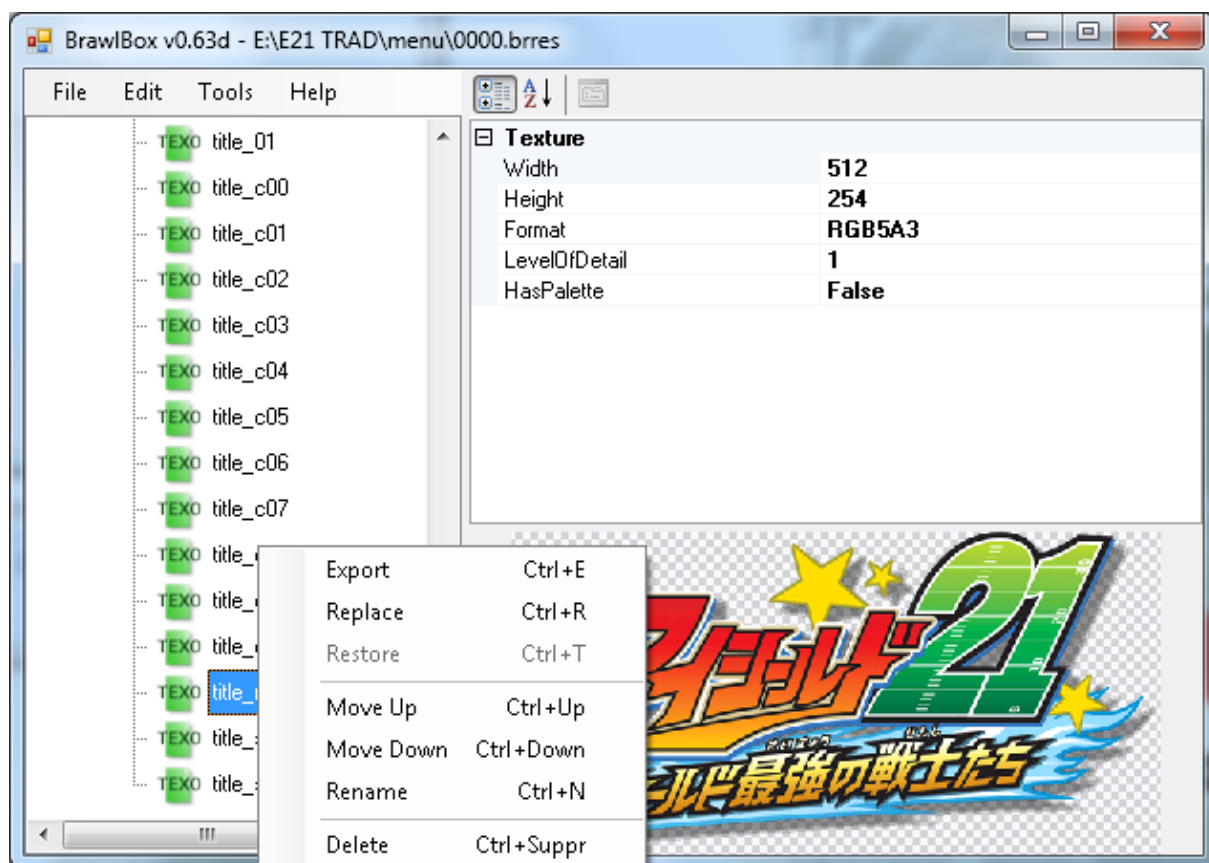


Vérifier que l'image possède les mêmes propriétés que l'original : soit les mêmes dimensions, le même format. Sachant que contrairement à Dolphin qui lui supporte ce genre d'erreur d'image, la Wii elle ne le peut pas aussi bien. (Je me suis fait avoir en croyant que ma Wii avait un problème car l'image modifiée sous Dolphin passait nickel)

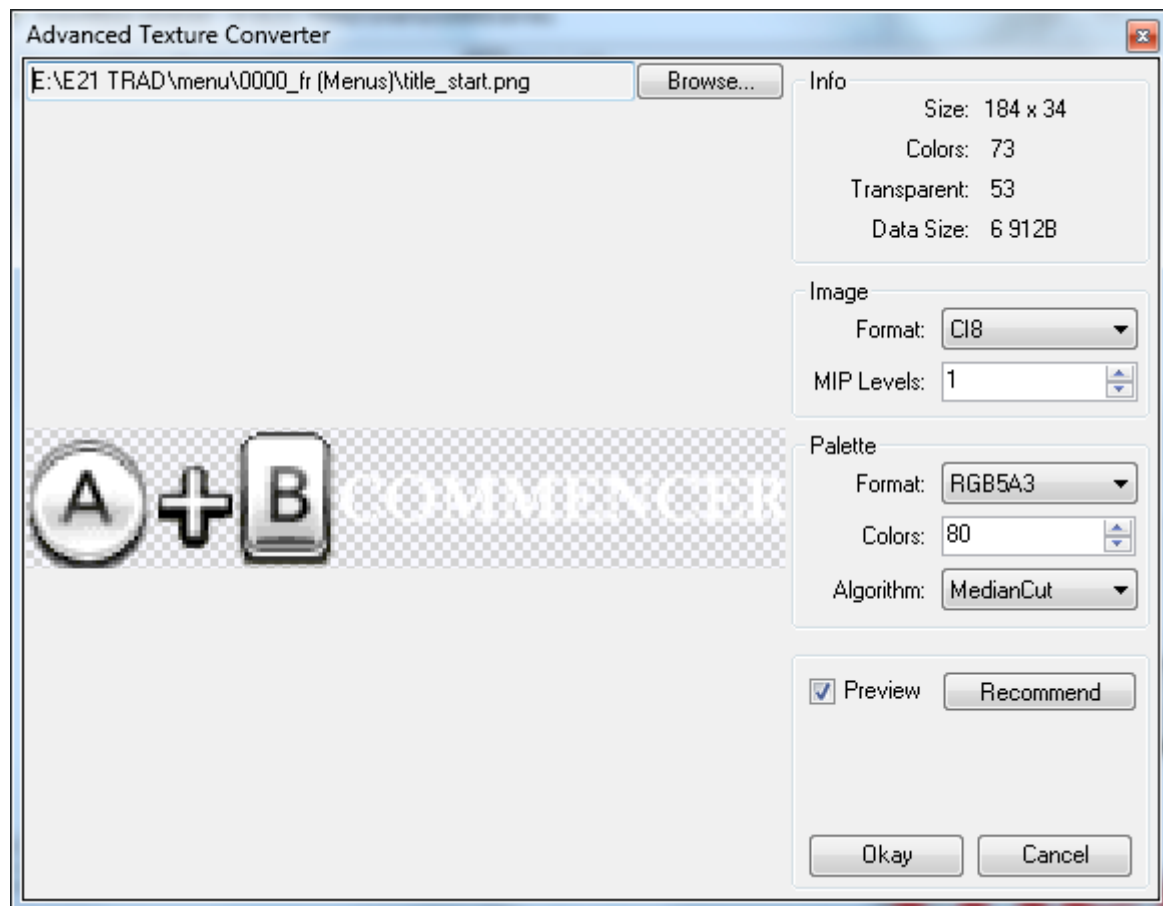


2 - Avec BrawlBox

Ouvrir votre image avec BrawlBox. Depuis le menu *File* : *Open*. Puis comme pour TPL Editor exporter/importer les images à modifier.



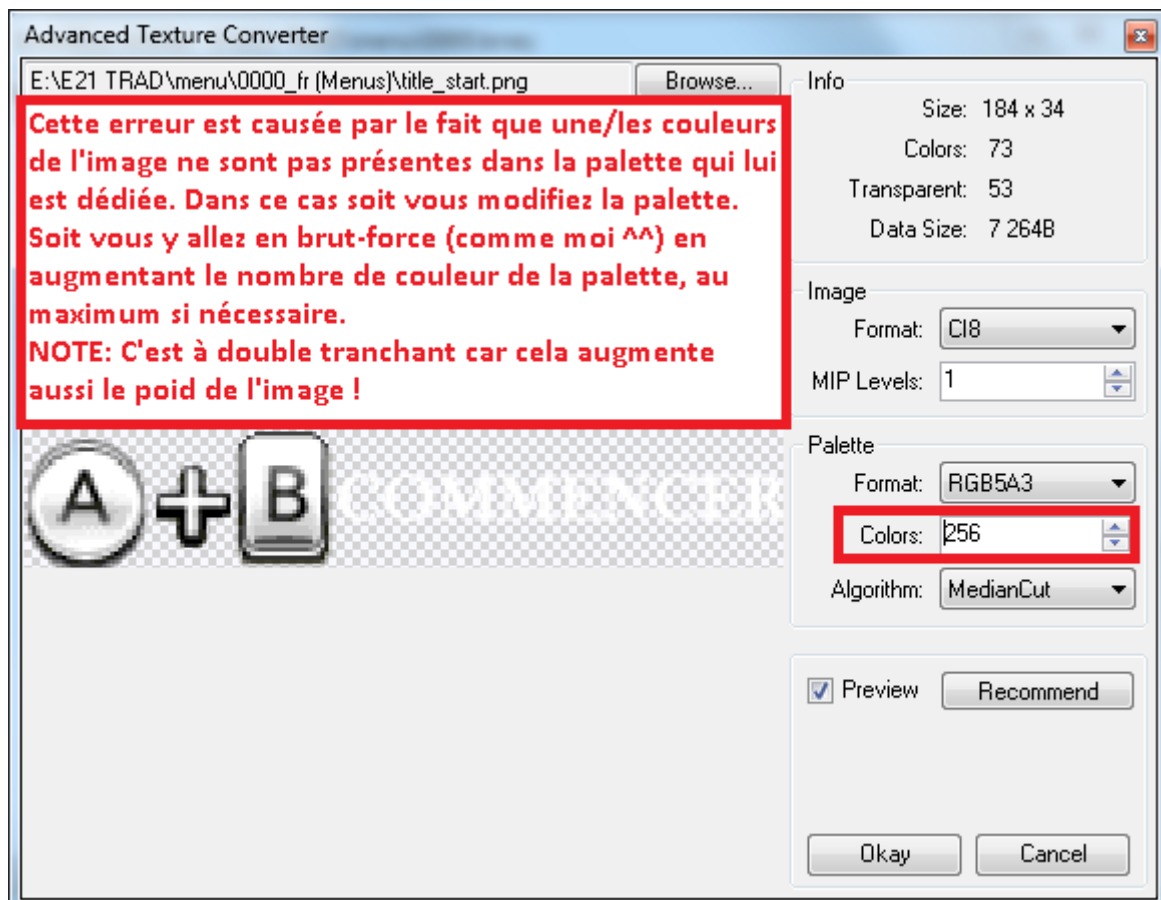
Voici l'interface qui vous permet d'importer une image. Cliquer sur « Okay », si tout s'est bien passé, tant mieux.



Sinon si vous aurez ce message, il vous faudra faire quelques modifications à l'image avec les options proposées lors de l'importation.

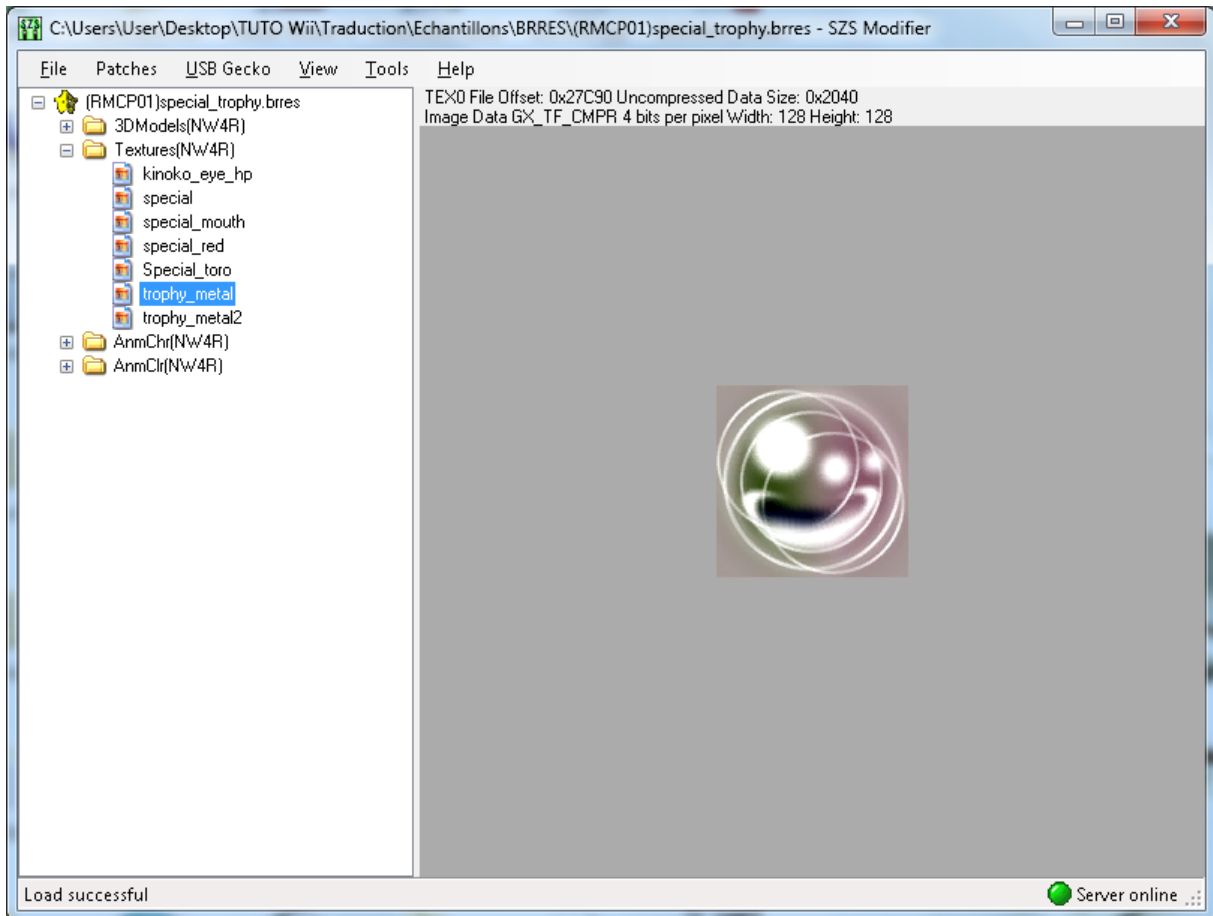


En règle générale, il suffira de modifier le nombre de couleur dans la palette pour la résoudre. Si vous avez des bugs d'affichage de textures (superposition, cf. [Harry Potter Effect](#)) lorsque vous jouez, il faudra suivre la marche à suivre indiquée sur le site.

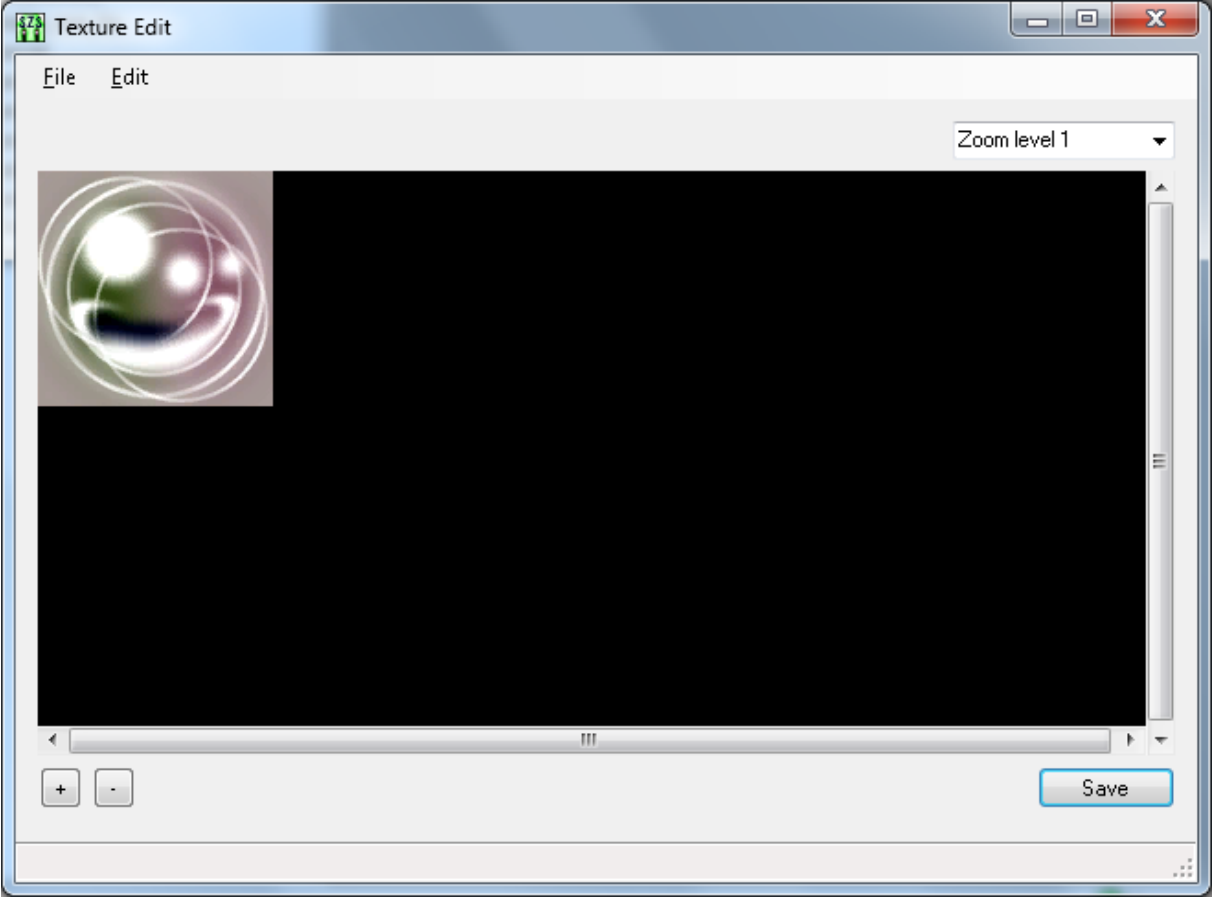


3 - SZS Modifier / (CTool) SZS Tool

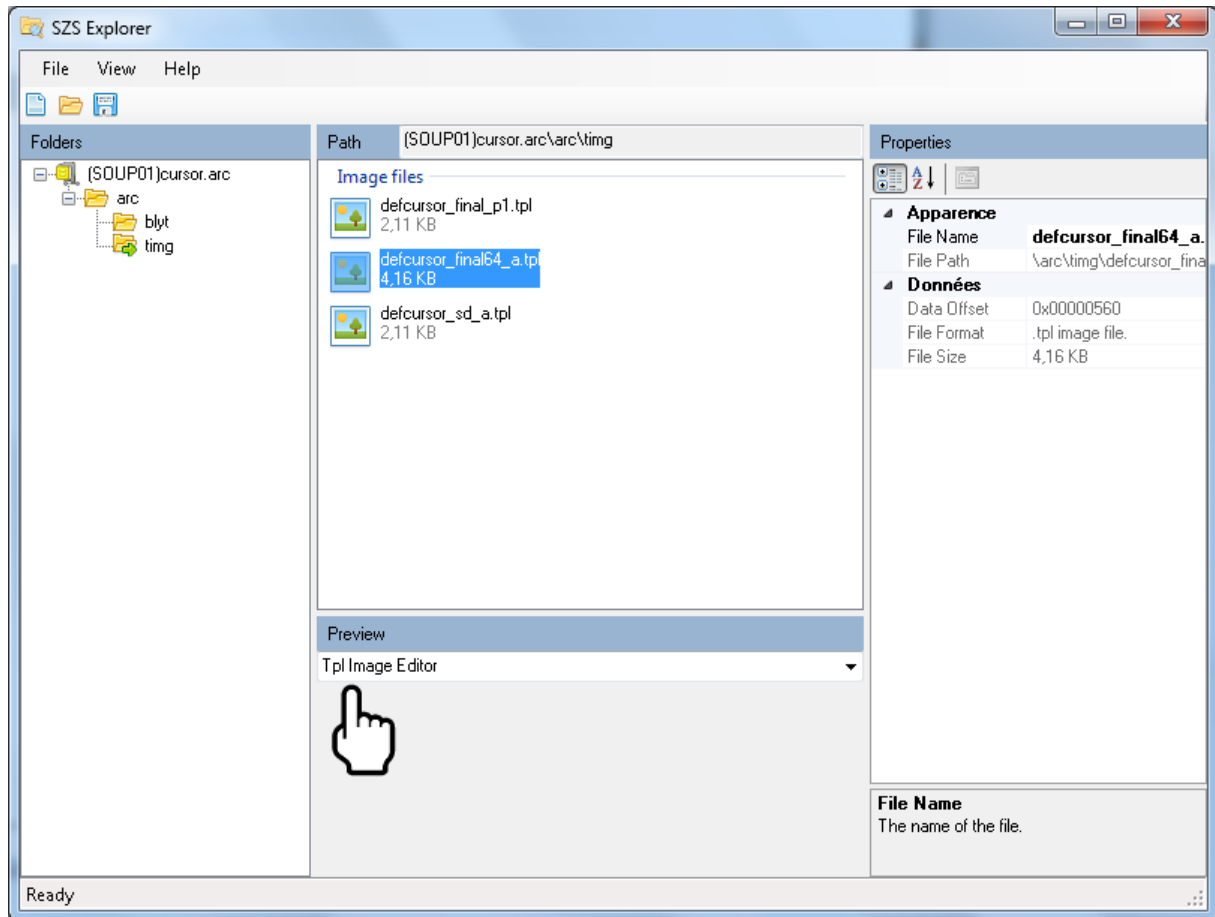
Ici on se servira de ces outils pour ouvrir des archives comme des fichiers ARC ou autres et remplacer des fichiers. Même s'il on peut utiliser l'éditeur intégré dans ces logiciels pour modifier les textures/images. (Personnellement je préfère utiliser BrawlBox.)



On peut se servir de SZS Modifier pour connaître le Zoom Level (i.e. MIP Level dans BrawlBox) d'une image en double cliquant dessus.



Ou se servir de SZS Tool/Modifier pour ouvrir des archives compressées.



c) Limites

Ici la limite, n'est pas à la quantité de texte. Ce qui est un plus ^^ . Mais aux propriétés de l'image. Si vous conserver les dimensions, le même format vous n'aurez aucun soucis. En revanche cela peut rendre le texte plus dur à traduire. Cependant garder un œil sur la taille de l'image modifiée pour ne pas trop alourdir l'ISO.

La traduction et ses outils

Ici nous allons aborder les différents points que je n'ai pas trop abordés en détail. Notamment les outils de traductions que j'utilise. Ainsi que les différents fichiers compressés.

a) Les fichiers compressés

Je n'en ai pas trop parlé car cela peut différer d'un fichier à l'autre donc je n'ai pas voulu m'étendre sur toutes les possibilités. Normalement les fichiers portant la même extension sont censés être conçu de la même manière. Mais des fois il y a quelques variantes. Par exemple dans l'échantillon que j'ai d'une quinzaine de fichier ARC je me suis rendu compte que certains étaient compressés et donc ne pouvant être ouvert avec mes logiciels. Pour plus d'infos sur les logiciels dont nous parlerons plus bas c'est là : [Wii File Type Database](#).

Bref, vous y trouverez la plupart des outils que j'utilise pour traduire un jeu Wii. Maintenant on va apprendre à décompresser/recompresser un fichier. Si vous connaissez d'autres types de compressions utilisés non-cités ni ici, ni dans [Wii File Type Database](#), n'hésitez pas ;v).

Compression LZ11 : Très facile à voir car soit le fichier porte « LZ » dans son extension, soit le premier octet du fichier est \$11 (visible avec MadEdit). Dans ce cas il faudra utiliser LZ11 pour décompresser et recompresser l'archive. Pour ce faire copier le fichier en question dans le même dossier que LZ11 puis le glisser sur « LZ11.bat ». **NOTE** : Je vous conseille de faire un doublon du fichier à (dé)compresser car il me semble qu'il écrase l'original (à vérifier).

Compression U8 : Si les premiers octets correspondent à ça : « U.8- » alors c'est un fichier compressé U8. Là plusieurs choix s'offrent à vous : vous pouvez utiliser SZS Modifier, (CTools) SZS Tool ou Wiimms SZS Tool. En général, vous n'aurez rien à faire car c'est le logiciel qui gère tout pour vous. Par contre [Wiimms SZS Tool](#) c'est le mieux mais il ne faut pas avoir peur de se frotter à des lignes de commande en mode console.

Compression Yaz0/Yaz1 : Les premiers octets correspondent soit à « Yaz0 » pour le Yaz0 et « Yaz1 » pour le Yaz1. Là encore les trois logiciels cités pour la compression U8 fonctionnent. Cependant il me semble que seul Wiimms SZS Tool prend en charge correctement la compression Yaz1.

b) La traduction avec des images

C'est une chose très délicate de devoir traduire du texte avec le texte sous forme d'image. Là vous avez deux options soit vous maîtrisez JWPce et vous utilisez la Table SHIFT-JIS pour taper les caractères, soit vous utiliser un OCR. Qu'est-ce donc ? *Optical Character Recognition* permet de reconnaître les caractères textuels présents dans une image. Pour ce faire il suffit de se rendre [ici](#), il existe d'autres sites et logiciels le faisant. Attention ce site concerne uniquement les caractères japonais. Divers options s'offre à vous, soit vous tentez de faire reconnaître une ligne de texte, soit caractère par caractère. Par expérience je dirais que le mieux est la deuxième option, elle est bien plus fiable car il vous sera proposé plusieurs caractères. Cependant il faudra isoler le caractère à la main avec votre logiciel de retouche photo. Et faire en sorte d'éviter les dégradés donc le mieux est qu'il n'y ai que deux couleurs le blanc et le noir. Ensuite vous retouchez un peu votre caractère en refaisant les bords, les formes mal faites, etc. En gros optimiser le caractère pour que la détection se fasse le mieux possible. Puis go NHocr.

c) Comment bien traduire un texte ?

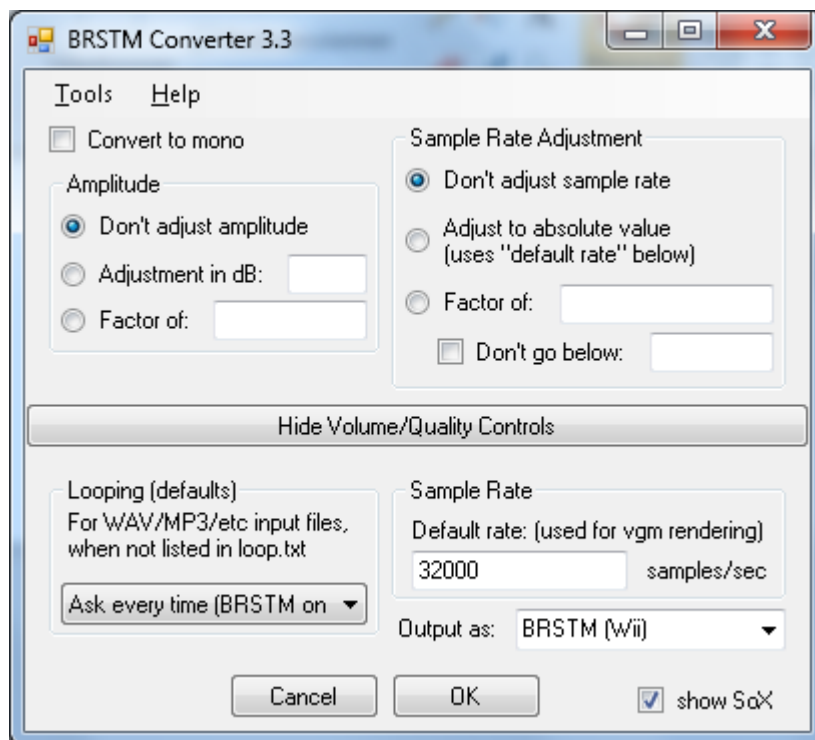
Si vous ne connaissez pas la langue source, ne serait-ce qu'un peu c'est embêtant. Le mieux je dirais est d'utiliser différents traducteurs pour voir ce que ça donne et un dictionnaire. En ce qui me concerne, traduisant du Japonais, j'utilise un [dictionnaire-japonais](#) et Google Traduction. Il va de soi que s'il on ne maîtrise pas la langue, évitez de traduire les dialogues, surtout ceux d'un RPG ><. Je tiens aussi à rappeler que dans certaines langues il y a des mots « anglicisé », « francisés », etc. Par exemple en japonais il m'est arrivé d'être face à un texte que Google ne voulait pas traduire, c'était en fait des mots anglais "japonisé" s'il on peut dire. Regarder les mots transparents s'il y en a et faire attention aux faux-amis bien sûr ;v).

La modification de fichiers audio

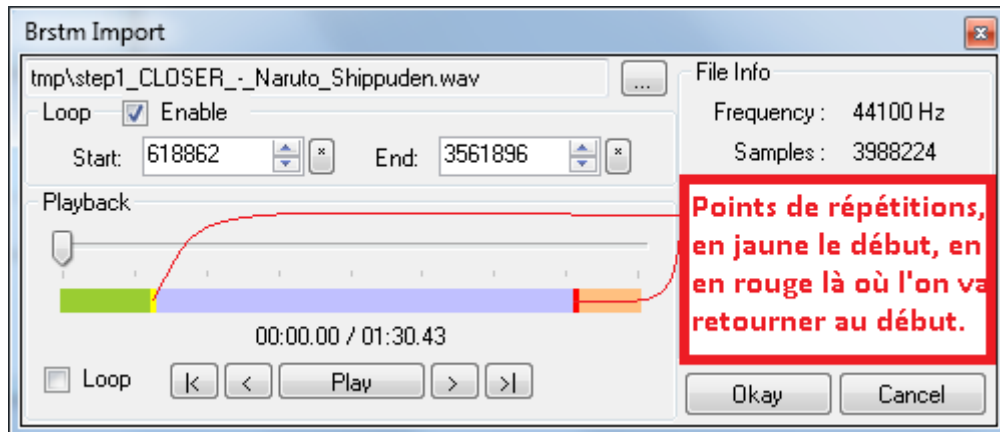
Alors ici, il est fortement conseillé d'avoir les connaissances pour modifier les fichiers audio avec des logiciels de retouche car on ne les abordera pas ! Sachant qu'on parlera des modifications à apporter aux fichiers BRSTM et BRASAR qui sont parfois liées.

a) Créer un fichier BRSTM Mono/Stéréo

Pour commencer il faudra lancer BRSTM Converter. Il utilise la BrawlLib de BrawlBox en plus d'autres bibliothèques pour convertir quasiment n'importe quel flux audio. Ainsi vous aurez quasiment aucune manipulation de conversion à faire.



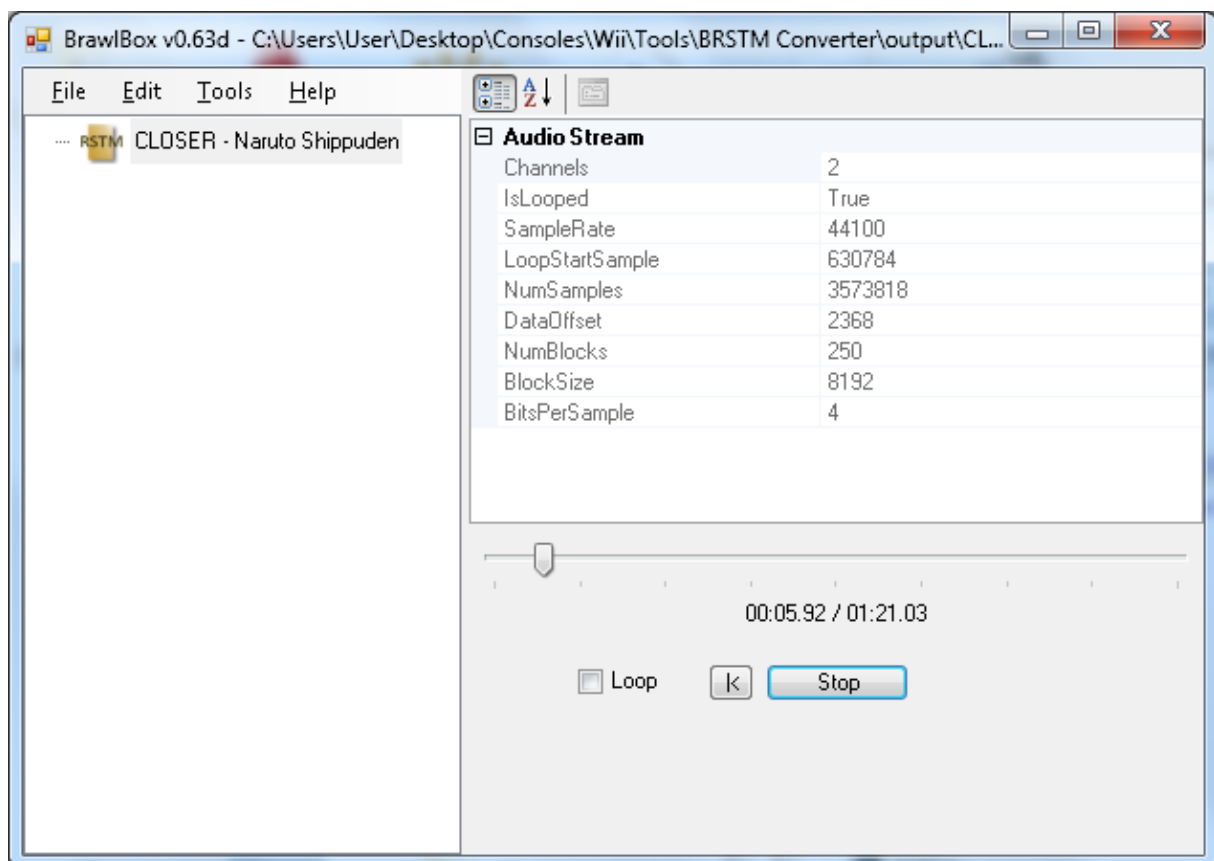
La seule chose à faire est de régler les paramètres souhaités c'est à dire le nombre de canaux audio stéréo/mono. Il est important de conserver le même nombre de canaux sinon il se peut qu'il y ai des silences ou que la musique ne se charge pas. Il est conseillé de mettre le "Looping (Defaults)" sur « Ask Every Time » ainsi il vous sera toujours demandé si et où vous voulez répéter la musique. Essayer également de conserver le même nombre de SampleRate que le fichier original (visible à l'aide de BrawlBox). Puis faites OK. Il ne vous reste plus qu'à choisir la musique à convertir.



Ensuite vous serez face à cet écran et vous pourrez choisir si vous répétez la musique ou non et d'où à où. **NOTE** : Si vous avez des problèmes avec des musiques non-répétées, il faudra choisir comme point de début de répétition un multiple de 14336.

Magique votre fichier BRSTM est lu par BrawlBox, il ne reste plus qu'à remplacer le fichier original avec WiiScrubber et modifier le fichier BRSAR qui contient toutes les informations relatives aux fichiers audio.

NOTE : J'ai vu que BrawlBox v0.67 n'arrivait plus à lire les fichiers BRSTM correctement chez moi. Du coup j'utilise la version 0.63. Mais ce n'est pas important car on ne se servira de BrawlBox que pour collecter des informations et non pour remplacer des fichiers dans une archive.



b) Créer un fichier BRSTM (avec 1 à 8 canaux)

Alors peu de jeux utilisent des fichiers audio à plus de deux canaux (stéréo) mais certains font office d'exception comme Mario Kart Wii. J'ai à force de persévérance pu trouver une solution ne faisant pas recourt au fichier du SDK Nintendo ^^ :

[Télécharger BRSTM Maker \(SP\)](#) :

Cela n'aurait jamais été possible sans wdrev que j'ai pu avoir grâce à un des membres de [NSMBW: Retro Remix danch744](#) que je remercie au passage. Pour créer un BRSTM Multi-Channel, wdrev.exe (qui est basé sur revb.exe) a besoin de fichiers DSP qui sont obtenable avec le programme du SDK Nintendo. Sachant que c'est illégal de cette façon-là et que je ne possède pas le dit programme, j'ai du procédé autrement.

Ce pack contient deux fichiers batch le premier du nom de BRSTM_Maker.bat qui est basé sur le batch de wdrev qui m'a généreusement été donné :vP. Apparemment il se sert du programme du SDK Nintendo pour faire une conversion wav (PCM - Mono) -> DSP -> BRSTM (multi). Comme je n'ai pas ce fichier je n'ai pas pu tester ce batch, j'y ai juste fait 2, 3 retouches. Histoire que ce ne soit pas comme ces fichiers batch volatiles où tu ne sais pas ce qu'ils font. J'y ai aussi mis un peu de français pour ceux qui ne comprendrait pas.

Ensuite vient un second fichier batch qui cette fois est fait de ma main, donc si vous y trouvez des erreurs, bugs faites les moi savoir ;v). **NOTE** : Il est déconseillé d'y mettre des fichiers avec des chemins/noms avec caractères spéciaux comme &,|, etc. Lui se base sur wdrev et revb. Mais contrairement à l'autre, il effectue une conversion BRSTM (stéréo) -> DSP -> BRSTM (multi).

En somme, c'est un pack contenant les outils pour pouvoir créer des fichiers BRSTM Multi-Channel le tout légalement !

Pour commencer il y a deux choix possibles soit vous possédez le programme du SDK Nintendo et dans ce cas vous n'avez qu'à lancer le batch et suivre les indications en mode console. Vous devrez alors convertir votre fichier audio en wav (PCM Mono) via BRSTM Converter en mettant comme Output : WAV et en cochant convert Mono.

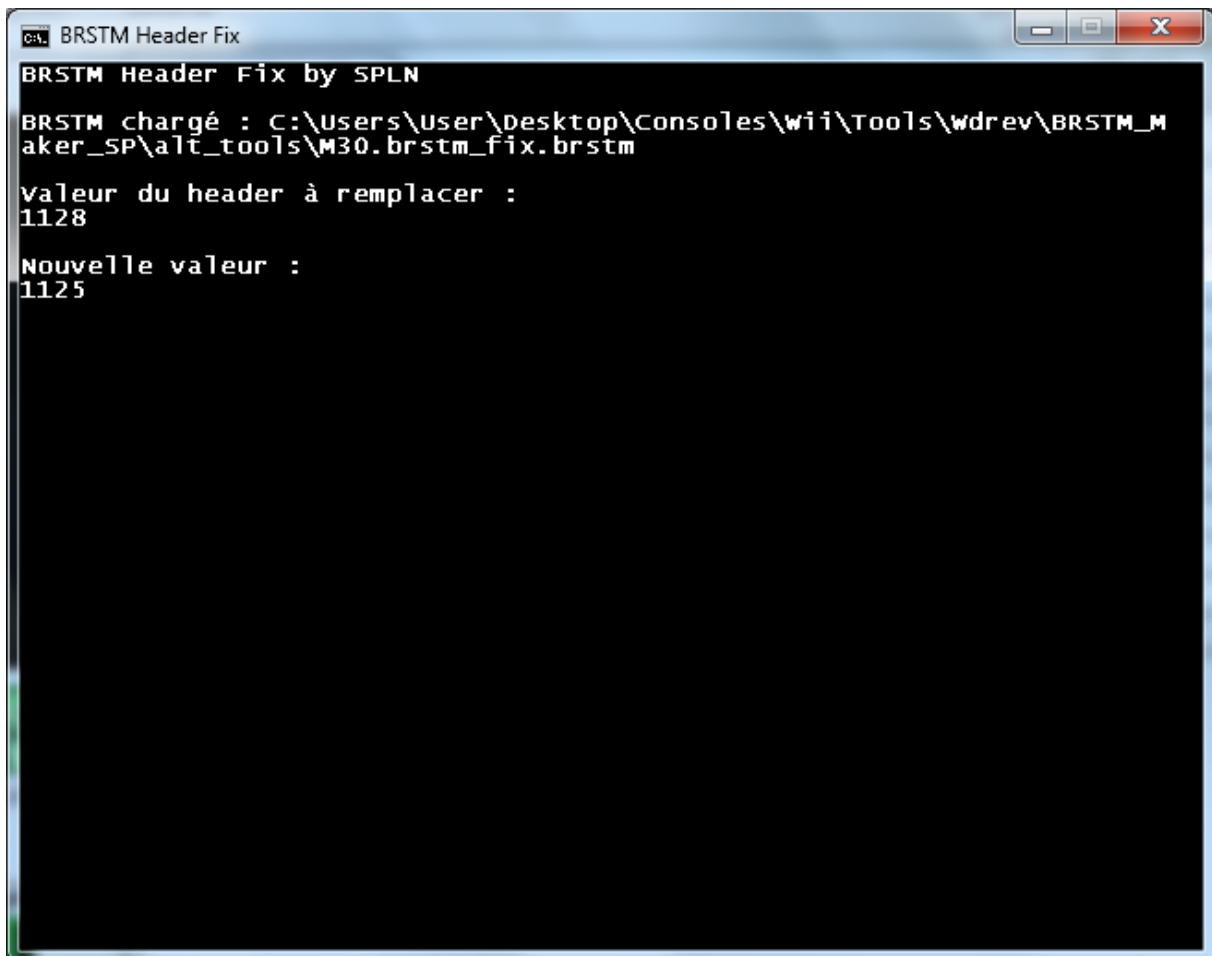
Soit vous êtes dans le même cas de figure que moi. Dans ce cas vous devrez créer un premier fichier BRSTM (stéréo) puis lancer BRSTM_Maker_SP.bat puis le glisser sur la fenêtre en mode console. Cette méthode a cependant l'avantage de conserver les propriétés du BRSTM original ;v). Une fois la console lancée, on vous demandera le nombre de canaux audio (= channel) puis faites entrer. Si vous êtes chanceux tant mieux sinon vous obtiendrez ceci :

```
C:\Windows\system32\cmd.exe
channel 0: 1.dsp
channel 1: 2.dsp

RSTM size = 6e9840
RSTM chunks:
 chunk 0 'HEAD' offset: 0x00000040 size: 0x00000100
 chunk 1 'ADPC' offset: 0x00000140 size: 0x00000de0
 chunk 2 'DATA' offset: 0x00000f20 size: 0x006e8920
Stream info:
 Codec:                2 <4-bit DSP ADPCM>
 Loop flag:            1 <looped>
 Channels:             2
 Sample rate:         44100 Hz
 Loop start:          114688 samples
 Total samples:       6338480
 Data start:          0xf40
 Block count:         443
 Block size:          0x2000
 Samples per block:   14336
 Last block used bytes: 1128
 Last block samples:  1968
 Last block size:     1152
 Samples per ADPC entry? 14336
 Bytes per ADPC entry? 4
Computed last block used bytes 1125 != value in header
```

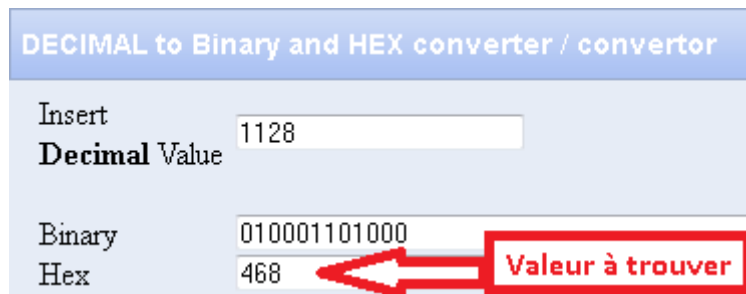
Si tel est le cas le batch devrais lancer un programme que j'ai fait à l'arrache sous GM8 >< (Si j'ai le temps, je le recoderais en C++). Il ne vous restera plus qu'à rentrer les valeurs indiquées par des rectangles rouges dans le screenshot plus haut, comme le montre le screenshot ci-dessous.

NOTE : Ces valeurs peuvent variées d'un BRSTM à l'autre alors ne vous basez pas sur les valeurs du screenshot mais celles que vous aurez obtenu.



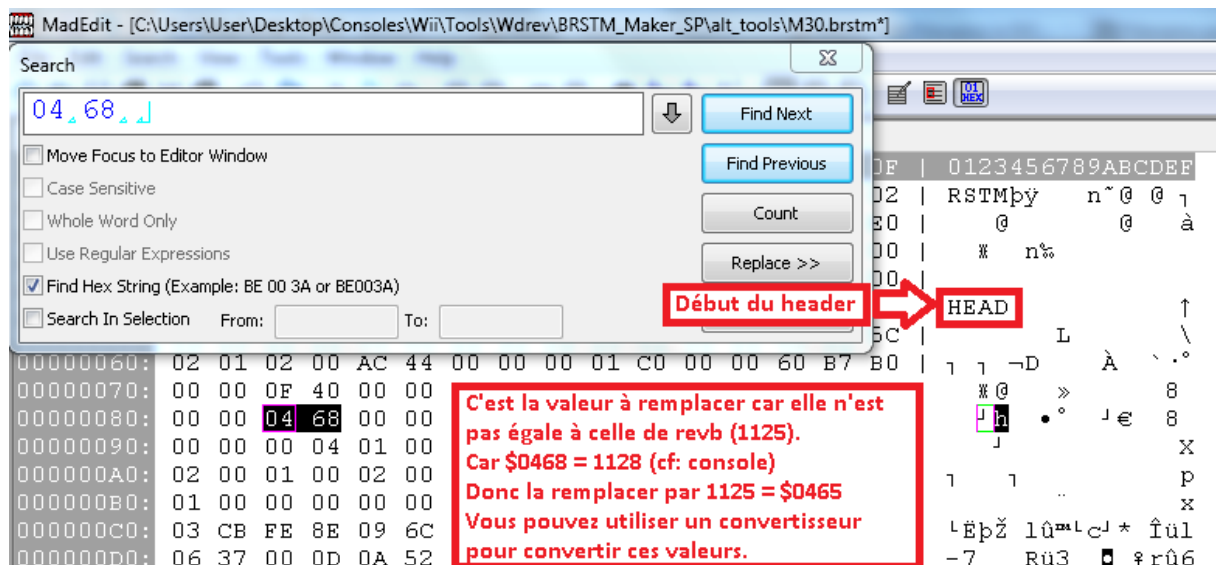
```
BRSTM Header Fix
BRSTM Header Fix by SPLN
BRSTM chargé : C:\Users\User\Desktop\Consoles\wii\Tools\wdrev\BRSTM_Maker_SP\alt_tools\M30.brstm_fix.brstm
valeur du header à remplacer :
1128
Nouvelle valeur :
1125
```


Si tout s'est bien passé, vous devriez avoir votre BRSTM créé dans le dossier output avec comme suffixe « _fix.brstm » ou avec le même nom si vous avez eu de la chance. Cependant si cela s'est mal passé même avec BRSTM Header Fix, voici comment faire la manip' à la main. Dans un premier temps il faut convertir la valeur en hexadécimal (celle obtenue dans le premier rectangle « Last block used bytes: XXXX »). Puis la chercher avec MadEdit.



Une fois la valeur trouvée, convertissez en hexadécimal la seconde valeur (celle obtenue dans le second rectangle « Computed last block used bytes XXXX != value in header »).

NOTE : « != » veut dire « différent de » donc la valeur du header est différente par rapport à la valeur réelle. Donc on va remplacer la valeur du header par celle réelle.



Normalement si vous avez suivi le protocole vous devriez avoir un BRSTM Multi-Channel dans le dossier Output.

c) Patcher un fichier BRSAR

Cette partie est un peu technique, pourquoi ? Car les seuls éditeurs existants n'ont été conçu à la base que pour patcher un jeu, Super Smash Bros. Brawl, Mario Kart Wii, New Super Mario Bros. Wii. Heureusement pour nous le dernier nous permet de modifier n'importe qu'elle BRSAR à condition de créer un fichier musicOffset.xml.

Le fichier XML est structuré de cette façon :

```
<?xml version="1.0" encoding="utf-8" ?>
  <music gameName="New Super Mario Bros. Wii">
    <track>
      <title>5 Star Completion Jingle</title>
      <filename>5stars_cmplt_lr.n.32.brstm</filename>
      <offset>04BA61</offset>
    </track>
  </music>
```

Donc pour chaque musique à modifier rajouter entre les balises music :

```
<track>
  <title>Mon titre pour identifier le fichier</title>
  <filename>nom_du_fichier_original.brstm</filename>
  <offset>Position dans le brsar</offset>
</track>
```

De telle sorte à obtenir quelque chose comme ça :

```
<?xml version="1.0" encoding="utf-8" ?>
<music gameName="Nom de mon jeu">
  <track>
    <title>Mon titre pour identifier le fichier</title>
    <filename>nom_du_fichier_original.brstm</filename>
    <offset>Position dans le brsar</offset>
  </track>
  <track>
    <title>TUTO Trad Wii Info</title>
    <filename>Trolololo.brstm</filename>
    <offset>212342</offset>
  </track>
</music>
```

Bref il va falloir faire quelques recherches de fichiers et de pointeurs mais rien de très fastidieux je vous rassure ^^.

Tout d'abord dans les balises <title> mettre un nom significatif pour reconnaître la musique à remplacer. Dans les balises <filename> le nom du fichier modifié c'est à dire celui du fichier original à remplacer pour mon exemple ce sera « ES21Wii_m30.brstm ». Donc Il faudra chercher avec MadEdit dans le fichier BRSAR le nom du fichier. Puis [...]

Voici la marche à suivre pour trouver la position à rentrer dans le XML.

- 1 - Avec MadEdit chercher le nom du fichier.
- 2 - Une fois trouvé, retourner en arrière comme indiqué par la ligne bleu. Vous trouverez un groupe caractéristique composé de "y" (\$FF).
- 3 - Depuis ce groupe continuer de parcourir le fichier à l'envers jusqu'à être face à autre chose qu'un caractère null/séparation (\$00)
- 4 - Se positionner sur le tout premier octet du groupe (cf: le rectangle vert).

ATTENTION: Vérifier avec la partie hexadécimal de l'editeur car parfois les caractères ne sont pas visibles typologiquement (comme l'espace ci-dessous).

- 5 - Relever la position actuel (cf: CharPos).
- 6 - Puis convertir cette position decimal en hexadécimal. Avec un convertisseur par exemple.
- 7 - Recopier cette position dans le champ prévu à cet effet dans le fichier XML.

Voici la position du pointeur en décimal : 132 353 et en hexadécimal donne : \$020501

Il faudra ensuite rentrer cette valeur dans le fichier xml.

DECIMAL to Binary and HEX converter / convertor

Insert	132353
Decimal Value	
Binary	00100000010100000001
Hex	20501

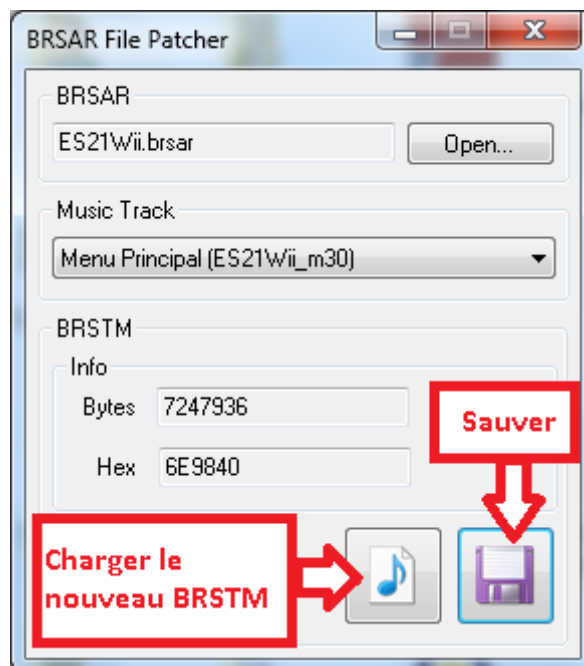
ATTENTION: L'hexadécimal s'écrit par "paire" dans le XML, ce n'est pas le cas ici : "\$20501" => "\$020501"

Car "02 05 01". Même si cette notation est correcte, je ne veux pas courir de risque.

Et voici ce que donne mon XML

```
<?xml version="1.0" encoding="utf-8" ?>
<music gameName="Eyeshield21 Wii">
  <track>
    <title>Menu Principal</title>
    <filename>ES21Wii_m30.brstm</filename>
    <offset>020501</offset>
  </track>
</music>
```

Maintenant, enregistrer le « musicOffsets.xml » modifié et lancer BRSAR Patcher. Puis charger le fichier BRSAR à patcher, le nouveau BRSTM ainsi que la musique à patcher.



Puis remplacer les fichiers BRSAR et BRSTM originaux par ceux que vous avez modifiés (si ce n'est pas déjà fait) via WiiScrubber. Si tout s'est bien passé vous pourrez écouter sans problème la musique que vous avez modifiée.

La modification de fichiers vidéo

Rien de très difficile, il vous faudra installer SUPER (c) qui est un convertisseur de fichier multimédia. Même remarque que pour la partie précédente il faudra savoir modifier des vidéos avec un logiciel de retouche.

Si vous désirez implanter un autre fichier vidéo qu'une modification du fichier original alors vous pouvez sauter l'étape a).

a) Convertir un fichier THP en un autre format vidéo

Pour commencer ouvrir SUPER et choisir comme « Output Container » MP4. Assurez-vous d'avoir décoché « Stream Copy » et « DirectShow Decode » qui font planter la conversion. Puis glisser le fichier THP à convertir sur la fenêtre. Il apparaîtra alors dans une liste dans laquelle vous pourrez le double-cliquer pour en connaître ses propriétés. Ceci permettra de remplir adéquatement les options du convertisseur. Vous aurez de base une qualité de vidéo moyenne mais pas très lourde. Donc si vous voulez avoir la meilleure qualité lorsque vous éditez la vidéo, cocher Hi Quality et Top Quality. Mais le revers de la médaille est que la conversion prendra beaucoup plus de temps et que le fichier obtenu sera très lourd. Cependant ceci n'aura aucun impact sur le fichier THP final. Pour finir, cocher le fichier THP dans la liste et cliquer sur « Encode (Active Files) ».

Si elle échoue dans ce cas tenter différents formats (AVI/WMV/etc.) et différentes options comme les « internal encoders (FFmpeg/MEncoder) ». Une fois la conversion terminée vous pourrez modifier le fichier vidéo obtenu à votre guise.

[1. Select the Output Container] [2. Select the Output Video Codec] [3. Select the Output Audio Codec]

MP4
H.264/AVC
AAC LC

internal encoders are auto-selected to complete this operation

FFmpeg
 MEncoder
 x264
 DirectShow Decode

VIDEO

Disable Video
 Stream Copy
 Video Scale Size More
Aspect
Frame/Sec More
Bitrate kbps
Options

560:304
 720:480
 1:1
 5.994
 15
 Hi Quality

608:336
 720:576
 3:2
 6.25
 23.976
 Top Quality

624:352
 768:576
 4:3
 7.5
 25
 Stretch It

640:368
 960:528
 5:4
 10
 29.97
 44K Audio

640:480
 960:720
 11:9
 11.988
 30
 H264 Profile

704:480
 1280:720
 12:5
 12.5
 50
 Crop / Pad

704:576
 NoChange
 16:9
 14.985
 60
 Other Opts

1248

AUDIO

Disable Audio
 Stream Copy
 Sampling Freq
Channels
Bitrate kbps
Streams Auto Detection

11025
 24000
 1
 64
 Detect All Streams

12000
 32000
 2

16000
 44100
 6

22050
 48000

OUTPUT

Encode to File Format: **MP4**
>>>
 Use Video Codec: **H.264/AVC, 640:480 (4:3), 1248 kbps, 29.97 fps**
 Use Audio Codec: **AAC LC, Stereo, 64 kbps, 32000 Hz, AudioStream [1]**

NOW ENCODING FILE 1 / 1 : Input: 145,956 MB Output: 20,303 MB

E:\E21 TRAD\thp\IC21_m.thp THP [THP , ADPCM_THP]

Active	FileName	Streaming Link	Output Status
<input checked="" type="checkbox"/>	E:\E21 TRAD\thp\IC21_m.thp		

An ERROR has occurred. Click to read more..

b) Convertir un fichier audio/vidéo pour créer un THP

Pour pouvoir créer un fichier THP via les outils dont nous disposons, il nous faut un son au format WAV (PCM 16 Little-Endian) et d'une séquence d'image JPG. Cependant ce procédé possède une limite, tout comme les images il est **TRÈS** fortement conseillé de conserver les mêmes propriétés ! Sachant que les dimensions des images doivent impérativement être multiples de 16 et que la durée de la vidéo NE **DOIT PAS** excéder celle de l'original. En sachant cela on peut se servir de SUPER pour convertir un fichier vidéo en séquence d'images. Pour se faire, choisir comme Output Container : « Sequence of Images ». Puis une fois encore choisir les propriétés adéquates. Il est également préférable de définir comme output folder (dossier de destination), l'endroit où se trouve THP GUI pour ne pas se retrouver avec des tonnes d'images dans un dossier à déplacer à l'aide d'un clic droit puis « Output File Save Management ». Vous pouvez décocher HI Quality car la vidéo de base l'était déjà.

En ce qui concerne le fichier WAV vous pouvez l'obtenir via BRSTM Converter en sélectionnant comme Output WAV.

[1. Select the Output Container] [2. Select the Output Video Codec] [3. Select the Output Audio Codec]

Sequence Of Images | JPG

internal encoders are auto-selected to complete this operation

FFmpeg
 MEncoder

DirectShow Decode

VIDEO

Disable Video
 Stream Copy

Video Scale Size More

- 560:304
- 608:336
- 624:352
- 640:368
- 640:480
- 704:480
- 704:576
- 720:480
- 720:576
- 768:576
- 960:528
- 960:720
- 1280:720
- NoChange

Aspect

- 1:1
- 3:2
- 4:3
- 5:4
- 11:9
- 12:5
- 16:9

Frame/Sec More

- 5.994
- 6.25
- 7.5
- 10
- 10
- 11.988
- 12.5
- 14.985
- 15
- 23.976
- 25
- 29.97
- 30
- 50
- 60

Bitrate kbps

2496

Options

- Hi Quality
- Top Quality
- Stretch It
- 44K Audio
- H264 Profile
- Crop / Pad
- Other Opts

AUDIO

Disable Audio
 Stream Copy

OUTPUT

Encode to File Format: Image Sequence - 1 Image Per Frame >>>

Use Video Codec: JPG, 640:480 (4:3), 2496 kbps, 29.97 fps

Use Audio Codec: NO audio

DROP A VALID MULTIMEDIA FILE HERE - Will try to play or encode

Active	FileName	Streaming Link	Output Status
<input checked="" type="checkbox"/>	E:\E21 TRAD\thp\IC21_m.thp.MP4		02700.jpg

Encode (Active Files) | An ERROR has occurred. Click to read more..

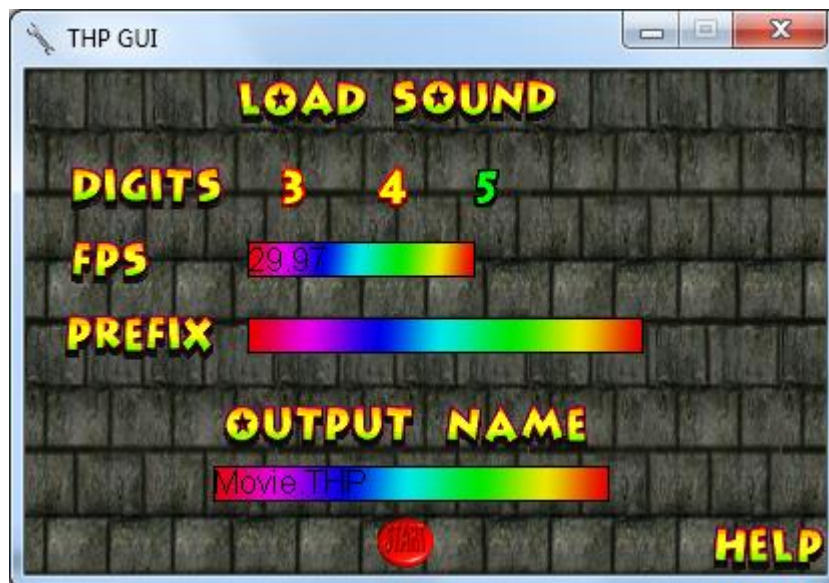
Play (Active Files | Streams) | Preview The Last Rendered File

Player Options | Cancel Current | Cancel All

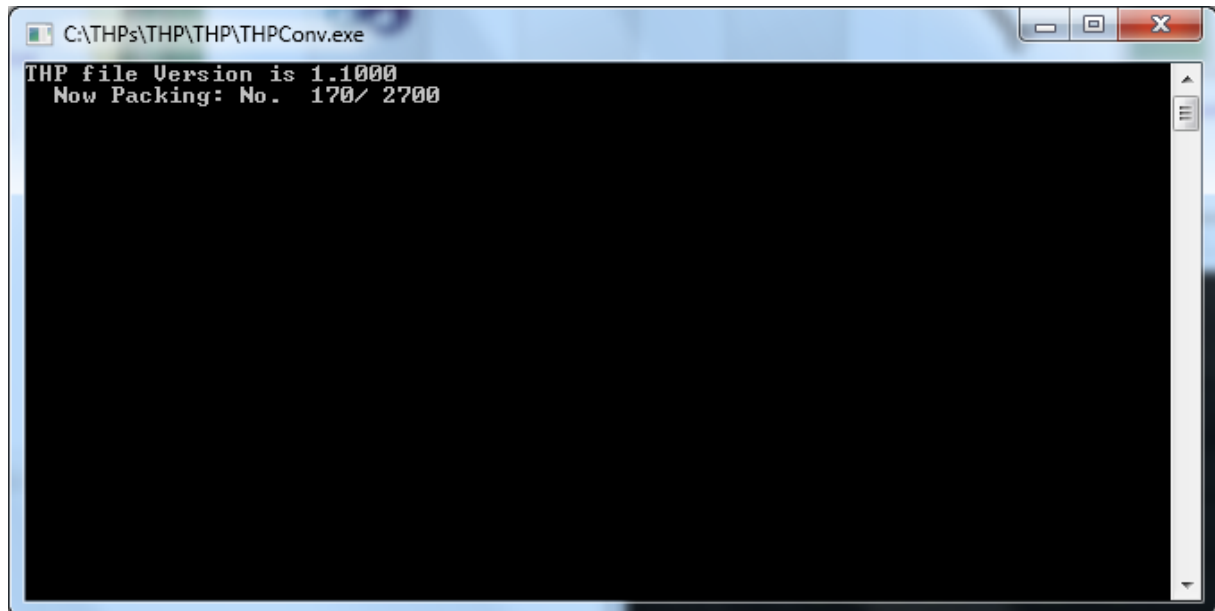
c) Créer un fichier THP

On va se servir de THP GUI pour convertir les séquences de JPG et notre WAV en THP. Tout d'abord assurez-vous qu'il y ai dans le dossier où se trouve THP GUI un dossier "THP" et à l'intérieur duquel doit figurer vos fichiers JPG et votre fichier WAV nommé « sound.wav ». Ensuite l'option « Digits » correspond aux nombres de chiffres présents dans le nom du fichier (i.e. « 00001.jpg » => 5 digits qui est par défaut avec SUPER). Ensuite il faut régler le nombre de FPS, là encore il faut connaître celui de l'original. Même si en général il est de 29.97 FPS pour les fichiers THP. Le préfixe concerne les utilisateurs de Sony Vegas soit le préfixe avant la séquence de nombres dans le nom (i.e. « img_00001.jpg » => préfixe: « img »). Ensuite il ne reste plus qu'à choisir un nom au fichier de sortie.

IMPORTANT : Allez avant de lancer l'opération dans Help puis cliquer sur la clé à molette et mettre **Auto-Delete (OFF)** ><. NOTE: Je pense que je vais faire un GUI moi-même car là il a été bâclé avec Game Maker 8 (que je possède ^^) d'où le « game_error.log » en cas d'erreur. Une fois fait appuyer sur start.



Bref, si tout s'est bien déroulé vous devriez voir l'image ci-dessous. Une fois la conversion terminée vous aurez droit à un aperçu de votre THP. Sauf si vous l'avez désactivé via les options. Ceci étant il ne vous reste plus qu'à remplacer le fichier THP original par celui créé, le tout à l'aide de Wiiscrubber.



Bilan

Voilà c'est ici que s'achève ce tutoriel. Vous aurez appris à manipuler toutes sortes d'outils et de fichiers pour traduire des textes de différents encodages avec MadEdit. Puis manipuler des fichiers via des outils appropriés comme BrawlBox, BRSTM Maker, BRSAR Patcher, THP GUI, etc. Si j'ai un peu de temps je vais faire vite fait un parseur de fichier BRSAR histoire d'automatiser le BRSAR Patcher. Puis voir ce que je peux faire pour les BRSTM Multi-Channel. Ainsi que faire un THP GUI mieux que ça ><. Mais bon, avant toute chose il faut que je trouve le temps de faire cela =vP.

Je tiens également à remercier tous les endroits où j'ai pu glaner des informations, notamment :

- DrThunder et son patch sans quoi je n'aurais jamais appris tout ce que je sais sur les textes
- smashboards
- wiki.tockdom.com
- Tous les sites que j'ai pu oublier =vD

Ainsi que tous les auteurs de ces merveilleux outils que nous pouvons utiliser.