

CryoPNG expliqué...

CryoPNG est un outil spécifiquement destiné à réduire la taille des fichiers PNG de type 6 (RVB+Alpha). De nombreux logiciels ou scripts existent déjà pour optimiser les images PNG, toutefois CryoPNG apporte quelque chose de nouveau dans le traitement des pixels totalement transparents.

Suite à l'excellent article de Sergey Chikuyonok publié sur [Smashing Magazine](#) durant l'été 2009 quelques outils tels PunyPNG ou ImageOptim ont intégré le « nettoyage » des pixels totalement transparents, c'est la technique nommée « dirty transparency », CryoPNG pousse cette technique un peu plus loin.



Une image PNG de type 6 (comme celle représentée ci-dessus à gauche) est en pratique composée d'une image 24 bits « Truecolour » (8 bits par composant RVB : Rouge, Vert et Bleu - représentée au centre) ainsi qu'un canal de transparence (8 bits Alpha - représenté à droite).

La transparence peut donc varier de zéro (totalement transparent, représenté en noir) à 255 (totalement opaque, représenté en blanc). Un pixel totalement transparent est par définition invisible, c'est l'élément graphique situé sous l'image qui est visible (ici le damier gris et blanc). Puisqu'il est invisible, ce qui figure dans l'image 24 bits pour représenter ce pixel pourrait en définitive être n'importe quoi... le nettoyage réalisé par le « dirty transparency » consiste à remplacer ce pixel par du noir (le triplet 0,0,0), ce qui est assez judicieux, mais [une image PNG est filtrée...](#)

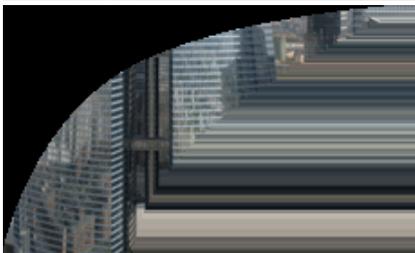


Filtre 0 : aucun (none)

Chaque ligne d'une image PNG est filtrée individuellement avant compression. Un seul filtre est appliqué par ligne parmi les 5 existants.

Le filtre 0 ne réalise aucune transformation, c'est une sorte de non-filtre.

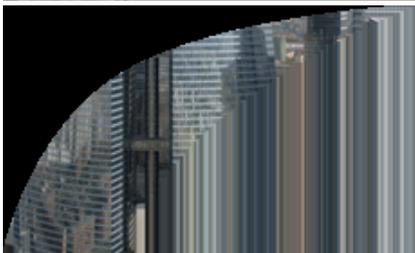
L'image de gauche représente ce qu'il reste de l'image RVB si l'on remplace les pixels totalement transparents par du noir (le triplet 0,0,0), ceci est réalisé par PunyPNG, ImageOptim, PngOptimizer et CryoPNG.



Filtre 1 : soustraction (sub)

Le filtre 1, réalise la soustraction composant par composant de chaque pixel sur la ligne avec celui qui le précède (celui à sa gauche), ceci dans le but de maintenir la majorité des valeurs à compresser proche de zéro.

L'image de gauche représente ce que devient l'image RVB si la valeur en sortie du filtre 1 est zéro pour les pixels totalement transparents.



Filtre 2 : haut (up)

Le filtre 2, réalise la soustraction composant par composant de chaque pixel sur la ligne avec celui de la ligne précédente (celui au-dessus), toujours dans le but de maintenir la majorité des valeurs à compresser proche de zéro.

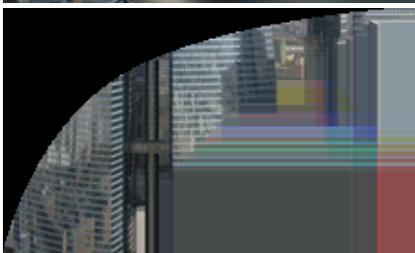
L'image de gauche représente ce que devient l'image RVB si la valeur en sortie du filtre 2 est zéro pour les pixels totalement transparents.



Filtre 3 : moyenne (average)

Le filtre 3, réalise la différence composant par composant de chaque pixel sur la ligne avec la moyenne de celui à sa gauche et de celui au-dessus, toujours dans le but de maintenir la majorité des valeurs à compresser proche de zéro.

L'image de gauche représente ce que devient l'image RVB si la valeur en sortie du filtre 3 est zéro pour les pixels totalement transparents.



Filtre 4 : paeth (paeth)

Le filtre 4, le plus complexe, réalise la différence composant par composant de chaque pixel sur la ligne avec le résultat du prédicteur Paeth (qui retourne la valeur du composant du pixel qui se trouve soit à sa gauche, soit au dessus, soit dans la diagonale au-dessus à gauche).

L'image de gauche représente ce que devient l'image RVB si la valeur en sortie du filtre 4 est zéro pour les pixels totalement transparents.

CryoPNG annule donc les valeurs en sortie de filtre des pixels totalement transparents, mais pour quels bénéfices ?

Voici un rapide comparatif des résultats obtenus avec différents outils d'optimisations.

- Cryogen-d est le script rapide qui accompagne la version Windows de CryoPNG.
- Cryogen-f est une version plus lente mais plus poussée sous Linux.
- PNGSlim peut produire des résultats qui varient un peu (il utilise les optimisations aléatoires de pngout).
- ImageOptim est appliqué 2 fois (sinon ses résultats sont moins bons).

Fichier	Original	Smush.it	PNGSlim	PunyPNG	ImageOptim	Cryogen-d	Cryogen-f
chess	125765	125427	107597	109536	111169	107431	107330
butterfly	105295	104011	61121	62136	61737	59675	59534
dirty-tr-smpl1	75458	74538	28340	28986	28351	27923	27892
automator	23469	23469	22168	22354	22195	22021	22028
coconut	14986	14916	13934	14565	14253	13752	13741
twit	2013	2013	1973	2013	2013	1926	1926

On remarque que Smush.it qui ne réalise aucune forme d'optimisation des pixels totalement transparents a des résultats en retrait, le script Cryogen-d bien plus rapide que Cryogen-f permet déjà de faire mieux que les autres.

Il ne faut aucunement en tirer des conclusions globales, CryoPNG et les scripts Cryogen sont uniquement prévus pour optimiser des images RVB + Alpha, et les fichiers pris en exemple ici ne sont pas nécessairement représentatifs

Peut-on faire mieux ?

Oui, car il est possible d'écrire des scripts plus poussés qui, à l'image de ce que fait PNGSlim, activent le mode aléatoire de pngout ce qui devrait encore permettre de gagner quelques dizaines octets.

Oui, car on peut également traiter les PNG de type 4 (Niveaux de gris + Alpha) de la même manière, ceci devrait être intégré début 2011.

Oui, car pour l'instant CryoPNG ne produit pas de fichier qui exploite plus d'un filtre, il peut le faire (il suffit de demander le filtre 5) mais le résultat m'a toujours semblé moins bon que celui produit par un seul filtre appliqué à toute l'image, c'est un problème d'heuristique de sélection des filtres.

Oui, car pour l'instant CryoPNG repose très étroitement sur OptiPNG (c'est OptiPNG avec des modifications au niveau de la libpng), il doit être possible de s'en affranchir pour le rendre plus rapide.

Peut-on utiliser CryoPNG pour intégrer des PNG optimisés à une application iPhone/iPod/iPad ?

Non, car Apple utilise des fichiers PNG modifiés qui s'éloignent de la norme (les composants RVB sont prémultipliés par le canal Alpha, ce qui en soi annule totalement ce que fait CryoPNG).

Frédéric Kayser
cryopng@free.fr