

# Comprendre la compression de données EXR

Informations compilées par Nicolas Dufresne pour Rainbow Productions et Rainbow Laboratory, Contactez-nous si vous trouvez une erreur ou si vous avez des suggestions sur ce document. <https://rainboxlab.org>  
Mille mercis à Dino Muhić pour son expertise à propos du DWA.

Lorsque vous utilisez le format Open EXR, il peut être difficile de déterminer quelle est la meilleure compression de données à utiliser, et cette question revient régulièrement. Ce document tente d'y répondre, selon vos besoins et le type d'image que vous souhaitez stocker.

Une compression est dite bonne pour un besoin spécifique quand sa vitesse de lecture ou d'écriture est la meilleure selon l'usage, et lorsque la taille du fichier est la plus petite possible.

## Type d'images

Dans ce document, nous distinguerons les types d'images suivants :

### Vidéo ou animation avec du grain

Images photographiques (incluant images de synthèses réalistes) ou film d'animation avec grain

### Vidéo

Images photographiques (incluant images de synthèses réalistes) sans grain

### Animation, illustrations

Images très stylisées comme l'animation 2d, le motion design, l'animation stylisée 3D, ou les passes 3D comme la Z-Depth (couche de profondeur), la passe de Normal...

### Couleurs unies, grands aplats de couleur

Images majoritairement constituées de couleurs unies, comme une couche alpha ou une passe d'ID

### Textures

Fichiers multi-résolution

## Résumé

### Exports (finaux) avec perte

Particulièrement pour les exports finaux, vous n'avez vraisemblablement pas besoin d'une compression sans perte, et avec les bons réglages, la taille du fichier exporté peut être relativement petite avec une perte de qualité minimale, et le Master ou le dossier de sauvegarde de votre film pourra être stocké de cette manière.

Dans ce cas **DWAA** (avec un petit niveau de compression) est très efficace dans tous les cas. Attention, **le DWA n'est pas (encore) supporté par ffmpeg**. Dans ce cas le **PXR24** est une bonne alternative (sans perte).

Les aplats de couleurs (couches alpha) peuvent être compressés très efficacement en utilisant le RLE sans perte de qualité

Si vous n'exportez pas pour du compositing complexe (ex. : fond vert) et surtout depuis une source vidéo, ou si l'export est destiné à être utilisé plus tard comme un Master exporté en YUV 422 ou 421 (comme du h264), l'option **Luminance/Chrominance** divisera la taille du fichier par 2, avec une très petite perte de qualité, imperceptible. Attention, **ffmpeg ne gère pas (encore) l'option Luminance/Chrominance**.

### Exports (intermédiaires) sans perte

Si le fichier est utilisé dans un logiciel de compositing par exemple, vous aurez à exporter sans perdre de qualité

Dans tous les cas vous rendez une image avec AOV (généralement depuis un logiciel 3D) et que vous pouvez accepter une (très légère) perte de qualité, **DWA** est la meilleure option, puisqu'il ne compressera que les couches RVB (ou Y,R,Y,B,Y en cas de Luminance/Chroma) en utilisant le RLE pour la couche alpha et le ZIP pour les autres couches.

Dans ce cas, faites attention car DWA utilise des noms de couches (sensible à la casse)

• couches avec perte: R, G, B, Y, RY, BY

• RLE: A

• ZIP: tous les autres noms (Red, red, r, Green, green, g, Blue, blue, b, x, y, z, U, u, V, v, etc.)

Attention avec les noms des couches. Par exemple si vous utilisez XYZ, la couche Y perdra en qualité. Vous pouvez plutôt utiliser « xyz »

Quand **les images n'ont pas de grain** :

Si vous n'avez pas besoin d'une précision maximum 32-bits flottant, **PXR24** est la meilleure compression que vous pouvez utiliser.

Si vous avez besoin d'une précision 32-bits flottant, **ZIP** est la meilleure option.

Pour les **images avec du grain**, **PIZ** est toujours la meilleure option.

Pour les images **stéréoscopiques**, **ZIP** est la meilleure option.

Pour les aplats de couleur (images unies) comme des couches alpha, le mieux est d'utiliser **RLE**.

## Cas spéciaux

Il y a d'autres utilisations plus spécifiques pour l'Open EXR :

Sur un système de **lecture en temps réel**, **B44** est le mieux (ou B44A pour les couches alpha et couleurs unies).

Si vous avez besoin de rendre **un proxy de petite taille** vous pouvez utiliser **DWAA** avec un gros niveau de compression.

Les images en **niveau de gris** bénéficieront beaucoup de l'option **Luminance/Chroma**, sans perdre de qualité.

## Images Luminance/Chroma

Encoder des images plates avec une couche de luminance et deux de chrominance, au lieu des données RGB, permet une forme de compression avec perte de données simple mais efficace qui est indépendante des méthodes de compression listées ici. Les couches Chroma peuvent être stockées à une résolution plus basse que la couche de luminance. Ce qui réduit significativement la taille des fichiers, avec seulement une perte de qualité d'image minime et généralement imperceptible.

C'est une méthode similaire à celle appelée YUV 422 pour la vidéo.

Vous pouvez donc utiliser cette option si vous exportez depuis, ou vers, un standard YUV 422 ou 421 sans perte de qualité. Si vous exportez depuis, ou vers, YUV 444 ou RGB, la perte de qualité est toujours très faible.

Comme la luminance est stockée en pleine qualité en utilisant l'option luminance /chroma, cette méthode peut stocker les images en niveau de gris encore mieux que le standard RGB sans aucune perte de données.

Note : cette option n'est pas (encore) supportée par ffmpeg.

## Les formats que vous ne savez pas que vous les connaissez déjà

**ZIP** est la méthode de compression utilisée par les fichiers **Portable Network Graphics (.png)**. Elle **n'est PAS** basée sur le format de fichier **ZIP**, malgré son nom). Ce qui signifie que pour toute mention de la compression ZIP dans ce document, vous pouvez aussi utiliser des fichiers PNG (si l'on a pas besoin d'AOV, en gardant à l'esprit que la plupart des logiciels sont plus lents avec le PNG, en particulier *After Effects*).

**RLE** est la compression utilisée pour les fichiers **TGA**, et elle ressemble à la méthode de compression de données de **Quicktime Animation** parfois utilisées dans les fichiers **.mov**. Ce qui signifie que toute mention du RLE dans ce document peut aussi être remplacée par un fichier TGA (si l'on a pas besoin d'AOV).

**DWAA** est proche de la méthode de compression des fichiers **JPEG**. Une séquence JPEG peut faire office de séquence EXR en DWA à condition de faire attention aux taux de compression (et si l'on a pas besoin d'alpha ou d'AOV)

## Résumé par utilisation et type d'images

### • Rendu final/export

Animation, illustration ou vidéo

**DWA**, (un peu) de perte (utiliser un petit niveau de compression) – non compatible avec ffmpeg (pour l'instant).

**PXR24**, sans perte\*

Si vous exportez en YUV 422 ou 421 vidéo (ex. : h.264) ou en niveaux de gris, utilisez l'option Luminance/Chroma – non compatible avec ffmpeg (pour l'instant).

Animation et vidéo avec grain

**DWA**, légère perte – non compatible avec ffmpeg (pour l'instant).

**PIZ**, sans perte

Si vous exportez en YUV 422 ou 421 vidéo (ex. : h.264) ou niveaux de gris, utilisez l'option Luminance/Chroma – non compatible avec ffmpeg (pour l'instant).

couleurs unies, aplats (ex. : couche alpha ou passe d'ID)

**RLE**

\* Dans le cas spécifique d'un export final, PXR24 est considéré comme sans perte, dans le cas où vous n'aurez jamais besoin des données en 32 bit flottant

### • Rendu intermédiaire 32 bits flottant

Textures, animations, illustration ou vidéo

**ZIP**

**DWA** si vous pouvez vous permettre une petite perte de qualité – non compatible avec ffmpeg (pour l'instant).

Notez que DWA compressera uniquement les couches R, G, B (ou Y, RY, BY avec l'option Luminance/Chroma) et sélectionnera automatiquement RLE pour la couche Alpha ou ZIP pour les AOV (Z, U, V, Normal...).

Vidéo ou animation avec grain

**PIZ**

Couleurs unies, aplats (couche alpha et passe d'ID)

**RLE**

### • Rendu intermédiaire 16/24 bits flottant, 16/32 bits entier

Textures, animation, illustrations ou vidéo

**PXR24**, si non disponible : ZIP

**DWA** si vous pouvez vous permettre une petite perte de qualité – non compatible avec ffmpeg (pour l'instant).

Notez que DWA compressera uniquement les couches R,G,B (ou Y,R,Y,BY avec l'option Luminance/Chroma) et sélectionnera automatiquement RLE pour la couche Alpha ou ZIP pour AOV (Z, U, V, Normal...).

Vidéo ou animation avec grain

**PIZ**

Couleurs unies, aplats (couche alpha et passe d'ID)

**RLE**

- **Images stéréoscopiques**

Textures, animation, illustrations ou vidéo

**ZIP**

Couleurs unies, aplats (couche alpha et passe d'ID)

**RLE**

- **Lecture en temps réel**

**B44A ou B44**

Si indisponible: PXR24

- **Proxy**

**DWAA** (utilisez un haut niveau de compression)

Si indisponible: PXR24, ZIP, PIZ

Nom	Perte	Description*	Ratio**	Résumé	Bon pour***
<b>PIZ</b>	Sans perte	A wavelet transform is applied to the pixel data, and the result is Huffman-encoded. This scheme tends to provide the best compression ratio for the types of images that are typically processed at Industrial Light & Magic. Files are compressed and decompressed at roughly the same speed. For photographic images with film grain, the files are reduced to between 35 and 55 percent of their uncompressed size. PIZ compression works well for scan-line based files, and also for tiled files with large tiles, but small tiles do not shrink much. (PIZ-compressed data start with a relatively long header; if the input to the compressor is short, adding the header tends to offset any size reduction of the input.)  PIZ compression is only supported for flat images.	35-55% (photo avec du grain)	- Vitesse d'écriture = vitesse de lecture - Meilleur pour les images avec du grain	- Photo/Video (avec grain)  - Animation 3D (avec grain)
<b>ZIP</b>	Sans perte	Differences between horizontally adjacent pixels are compressed using the open-source zlib library. ZIP decompression is faster than PIZ decompression, but ZIP compression is significantly slower. Photographic images tend to shrink to between 45 and 55 percent of their uncompressed size. Multi-resolution files are often used as texture maps for 3D renderers. For this application, fast read accesses are usually more important than fast writes, or maximum compression. For texture maps, ZIP is probably the best compression method. In scan-line based files, 16 rows of pixels are accumulated and compressed together as a single block.	45-55% (photo sans grain)	- Lecture plus rapide, écriture significativement plus lente - Identique au PNG - Support de la stéréoscopie	<b>Uniquement quand 32bpc flottant est nécessaire (sinon, PXR24 est meilleur):</b>  - Textures  - Photo/Video (sans grain)  - Animation 3D (sans grain)  - Animation 2D, Illustrations
<b>ZIPS</b>	Sans perte	Uses the open-source zlib library for compression. Like ZIP compression, but operates on one scan line at a time.		- Identique au PNG - Support de la stéréoscopie	
<b>RLE</b>	Sans perte	Differences between horizontally adjacent pixels are run-length encoded. This method is fast, and works well for images with large flat areas, but for photographic images, the compressed file size is usually between 60 and 75 percent of the uncompressed size.	60-75% (photo)	- Rapide - Identique au TGA - Meilleur avec de grands aplats (alpha et id) - Support de la stéréoscopie	- Couleurs unies, aplats (couche alpha et passe d'ID)
<b>PXR24</b>	Avec perte (16bit float, 16/32-bit int)  Légère perte $3 \times 10^{-5}$ (32-bit float)	After reducing 32-bit floating-point data to 24 bits by rounding (while leaving 16-bit floating-point data unchanged), differences between horizontally adjacent pixels are compressed with zlib, similar to ZIP. PXR24 compression preserves image channels of type HALF and UINT exactly, but the relative error of FLOAT data increases to about . This compression method works well for depth buffers and similar images, where the possible range of values is very large, but where full 32-bit floating-point accuracy is not necessary. Rounding improves compression significantly by eliminating the pixels' 8 least significant bits, which tend to be very noisy, and therefore difficult to compress. PXR24 compression is only supported for flat images.	Meilleur que ZIP pour 32bpc  Identique au ZIP sinon	- Lecture plus rapide, écriture significativement plus lente - Transforme 32-bit flottant en 24-bit	<b>Uniquement pour 16bpc float ou 16/32bpc int, ou quand 24bpc float est suffisant au lieu de 32bpc</b>  - Photo/Video (sans grain)  - Animation 3D (sans grain)  - Animation 2D, Illustrations  - Textures
<b>B44</b>	Avec perte	Channels of type HALF are split into blocks of four by four pixels or 32 bytes. Each block is then packed into 14 bytes, reducing the data to 44 percent of their uncompressed size. When B44 compression is applied to RGB images in combination with luminance/chroma encoding (see below), the size of the compressed pixels is about 22 percent of the size of the original RGB data. Channels of type UINT or FLOAT are not compressed. Decoding is fast enough to allow real-time playback of B44-compressed OpenEXR image sequences on commodity hardware. The	44%	- Taille de fichier fixe - Lecture très rapide	- Pour systèmes nécessitant une lecture temps réel

		size of a B44-compressed file depends on the number of pixels in the image, but not on the data in the pixels. All images with the same resolution and the same set of channels have the same size. This can be advantageous for systems that support real-time playback of image sequences; the predictable file size makes it easier to allocate space on storage media efficiently. B44 compression is only supported for flat images.			
<b>B44A</b>	Avec perte	Like B44, except for blocks of four by four pixels where all pixels have the same value, which are packed into 3 instead of 14 bytes. For images with large uniform areas, B44A produces smaller files than B44 compression. B44A compression is only supported for flat images.	< 44%	- Taille de fichier fixe - Lecture très rapide - Meilleur avec de grands aplats (alpha et id)	- Pour systèmes nécessitant une lecture temps réel
<b>DWAA</b>	Avec perte	JPEG-like lossy compression format contributed by <a href="#">DreamWorks Animation</a> . Compresses 32 scanlines together.	Varying depending on chosen compression level	- Similaire au JPEG	- Proxies (haute compression)  - Export finaux quand une perte minime est acceptable
<b>DWAB</b>	Lossy	Same as DWAA, but compresses blocks of 256 scanlines.	Varying depending on chosen compression level		- Proxies (haute compression)  - Export finaux quand une perte minime est acceptable

\* La description est celle de la *technical introduction* de <https://www.openexr.com>

\*\* Compressé / non compressé. Un ratio plus faible est meilleur. Par exemple, 45% signifie que le fichier compressé a une taille correspondant à 45% du fichier non compressé.

\*\*\* A file is good for a given use when its read or write speeds best fits how it is going to be used, and has the best compression ratio available.