Annexe 1-QuickTime Confidentiel

> *Note* : Certaines des informations contenues dans cet article sont obsolètes. Les fichiers Quicktime .mov sont l'un des formats de conteneur pris en charge par Jitter. Les autres formats sont .mp4, .m4v et .avi. La prise en charge de chaque format de conteneur et codec de piste dépend du moteur vidéo et de la plate-forme activés.
>
> Les recommandations en matière de codecs et de dimensions sont également obsolètes. Au moment où nous écrivons ces lignes, il est possible d'obtenir une sortie vidéo de 4K et même de 8K, en fonction du processeur, du processeur graphique, de la vitesse du disque, du codec et du moteur vidéo. Avant tout, la manière la plus efficace de traiter la vidéo avec Jitter est de s'assurer que ***output_texture*** est activé pour les objets *jit.movie*, *jit.playlist*, et *jit.grab*.

Le temps dans QuickTime

Contrairement à la plupart des médias basés sur le hardware, QuickTime n'utilise pas un système de temps basé sur les images. Au lieu de cela, QuickTime utilise un système basé sur l'échelle de temps et les valeurs de temps. L'échelle de temps est une valeur entière qui indique combien de valeurs temporelles composent une seconde de film. Par défaut, une nouvelle vidéo QuickTime a une échelle de temps de 600, ce qui signifie qu'il y a 600 valeurs temporelles par seconde.

La fréquence d'images est déterminée par le concept de temps intéressant - les moments où le film est modifié. Si un film change toutes les 40 valeurs temporelles, sa fréquence d'images effective est de 15 images par seconde (puisque 600 divisé par 40 est égal à 15). Lorsque Jitter détermine le nombre d'images d'un film à l'aide du message *getframecount*, il recherche dans le film les moments intéressants et en indique le nombre. Les messages *frame* et *jump* déplacent la tête de lecture entre différents points de temps intéressants.

Pour plus d'informations sur la relation entre l'échelle de temps et la fréquence d'images, reportez-vous au *Tutoriel 4* : *Controlling Movie Playback*.

Dans Jitter, les opérations d'enregistrement vous permettent de spécifier une fréquence d'images, et Jitter effectue des calculs pour vous lorsqu'il conçoit un nouveau film. Les opérations d'édition, en revanche, utilisent des valeurs temporelles pour déterminer leur gamme d'effets.

Optimisation des films pour la lecture dans Jitter

Bien que Jitter puisse lire n'importe quel film que vous lui proposez, il y a quelques directives que vous pouvez suivre pour améliorer la qualité. Malheureusement, il n'existe pas de recette précise pour atteindre la perfection - les performances dans une application en temps réel comme Jitter sont le résultat de l'interaction entre les codecs de piste d'un film (qui affectent la bande passante et la charge du processeur), les formats du film, la fréquence d'images et, dans une certaine mesure, la complexité du média traité.

Codecs

Les médias visuels, en particulier, contiennent de grandes quantités de données qui doivent être lues sur le disque et traitées par votre ordinateur avant de pouvoir être visualisées. Les codecs, ou compresseurs/décompresseurs, sont utilisés pour coder et décoder les données. Lors de l'encodage, l'objectif est généralement d'alléger les données de sorte que moins d'informations soient lues sur le disque lors de la lecture du film. La lecture des données sur le disque est une source majeure de surcharge lors de la lecture des films.

Si vous avez suffisamment de RAM, vous pouvez envoyer le message *loadram* à *jit.movie* pour copier le média (compressé) d'un film en RAM. L'accès à la RAM étant nettement plus rapide que l'accès à un disque dur, la lecture des films s'en trouvera généralement améliorée, bien que l'objet *jit.movie* doive toujours décompresser chaque image pendant la lecture du film. Pour mettre en mémoire tampon les données de la matrice décompressée en RAM, utilisez l'objet *jit.matrixset*.

Lors du décodage, l'objectif est de ramener les données à leur état pré-encodé aussi rapidement que possible. Les codecs, dans l'ensemble, sont à perte, ce qui signifie que certaines données sont perdues au cours du processus. En tant qu'utilisateurs, notre objectif est de déterminer quel codec offre la meilleure qualité à la meilleure vitesse pour notre application particulière.

Pour déterminer les codecs de piste d'un film QuickTime, utilisez le message *gettrackcodec* de *jit.movie*.

Codecs audio

Des codecs sont disponibles pour les pistes vidéo et audio. Pour la diffusion en ligne, vous pouvez utiliser un codec MPEG 2 niveau 3 (.mp3) ou AAC pour créer des fichiers plus petits. Dans Jitter, cependant, si vous jouez des films avec des pistes vidéo et audio, vous obtiendrez les meilleurs résultats avec de l'audio non compressé (audio PCM) simplement parce qu'il n'y aura pas de surcharge de décompression du codec audio.

Note technique : Les fichiers .mp3 peuvent être lus par *jit.movie* en tant que films audio uniquement (ils sont compressés avec le codec MPEG 2 Layer 3). Bien que vous ne puissiez pas traiter les données audio avec d'autres objets Jitter que l'objet *jit.movie*, vous pouvez utiliser l'attribut ***soc*** de *jit.movie* pour envoyer l'audio à MSP via l'objet *spigot~* (***Note : le spigot~ ne supporte que le moteur QuickTime 32 bits et nécessite l'objet jit.movie pour fonctionner***. Voir le Tutoriel 27 : *Using MSP Audio in a Jitter Matrix* pour plus d'informations).

Codecs vidéo

Les codecs vidéo peuvent être gérés de manière matérielle - vous pouvez avoir une carte vidéo spéciale qui fournit une compression et une décompression matérielle d'un codec particulier, comme MPEG ou Motion-JPEG - ou plus généralement de manière logicielle. Dans Jitter, le support des codecs matériels ne concerne que les composants de sortie vidéo. La lecture de films utilisera toujours un codec logiciel, à l'exception importante de la fonction de composant de sortie vidéo directe de l'objet *jit.movie* (voir le Tutoriel 22 *Video Output Components and the Object Reference entry* pour l'objet *jit.movie* pour plus d'informations).

Note technique : Les cartes vidéo qui offrent un support matériel aux codecs ne les acceptent généralement que pour la décompression des médias à l'écran. Comme Jitter décompresse généralement les médias dans un tampon hors écran (à l'exception de ce qui est indiqué ci-dessus), des codecs logiciels sont utilisés.

Les codecs vidéo utilisent généralement l'un des deux schémas suivants : la compression spatiale et la compression temporelle.

La compression spatiale vous est probablement familière dans le monde des images fixes. Les fichiers JPEG, PNG et PICT utilisent tous des types de compression spatiale. Les schémas de compression spatiale recherchent dans une image unique des motifs et des réplications qui peuvent

être décrits de manière plus simple. La plupart simplifient également les images pour s'assurer qu'elles contiennent ces motifs. Néanmoins, les images plus complexes sont plus difficiles à compresser et donnent généralement lieu à des fichiers plus volumineux. La compression spatiale ne tient pas compte du temps : elle compresse simplement chaque image en fonction de son algorithme de codage.

La compression temporelle est propre au monde des images en mouvement, car elle fonctionne en créant une description du changement entre des images consécutives. En général, la compression temporelle ne décrit pas entièrement chaque image. Au lieu de cela, un film compressé temporellement contient deux types d'images : les images clés, qui sont des images entièrement décrites (généralement compressées dans l'espace également), et les images régulières, qui sont décrites par leur changement par rapport à l'image clé précédente.

Pour les applications où un film sera lu du début à la fin, la compression temporelle est très utile. Des codecs comme *Sorenson* utilisent la compression temporelle pour créer des fichiers extrêmement petits, idéaux pour la lecture sur le Web. Cependant, la compression temporelle n'est pas un très bon outil si vous devez lire votre film à l'envers, car l'ordre des images clés est vital pour décrire correctement la séquence d'images. Si nous lisons un film compressé temporellement à l'envers, les descriptions des changements seront traitées avant l'image clé qui décrit l'état initial de l'image! De plus, la décompression du codec *Sorenson* est assez gourmande en ressources processeur. La lecture d'un film compressé par la méthode *Sorenson* sera plus lente que la lecture d'un film compressé par une méthode plus légère.

Pour la lecture de Jitter, nous recommandons d'utiliser un codec vidéo sans compression temporelle, tel que Photo-JPEG ou Motion-JPEG (les compressions Photo- et Motion-JPEG utilisent la même méthode de compression, mais Motion-JPEG est optimisé pour un support matériel particulier [voir note ci-dessus]). Avec des paramètres de qualité élevés, la compression JPEG offre un bon compromis entre la taille du fichier et la qualité de l'image. Elle est également relativement simple à décoder, de sorte que votre processeur peut être utilisé à meilleur escient que pour décompresser une vidéo.

Si la qualité de l'image est primordiale, le codec Animation est plus performant que Photo-JPEG, mais crée des fichiers beaucoup plus volumineux.

Les différentes versions de QuickTime prennent en charge différents codecs audio et vidéo. Par exemple, QuickTime 5 ne prend pas en charge le codec MPEG-4, alors que QuickTime 6 le fait. (QuickTime 10 a abandonné de nombreux codecs. Pour l'utilisation des anciens codecs, vous pouvez toujours télécharger QuickTime 7 sur Apple.com). Vous devriez expérimenter avec différents paramètres de codecs pour trouver la meilleure solution pour votre utilisation particulière de Jitter.

Formats de la vidéo et fréquence d'images

Par rapport au codec, les formats de la vidéo et la fréquence d'images sont des facteurs plus directs de la performance de Jitter. En termes simples, une image plus grande ou une fréquence d'images plus élevée indique que Jitter a plus de données à traiter par seconde.

Une vidéo en 640x480 produit 1 228 800 valeurs individuelles par image que Jitter doit traiter (640 fois 480 fois 4 (valeurs séparées pour les canaux alpha, rouge, vert et bleu). Une vidéo de 320x240 ne produit que 307 200 valeurs individuelles par image, soit un quart de la taille de la vidéo la plus grande. Sur la plupart des machines, les films de 640X480 donneront de bonnes performances avec un ou deux traitements. Si votre patch est complexe, vous devrez passer à la taille inférieure.

Si vous travaillez avec un support DV, vos films sont enregistrés à 29,97 images par seconde (en NTSC) ou 25 images par seconde (en PAL). Même en utilisant un film de 360x240, Jitter doit traiter 10 357 632 valeurs par seconde en NTSC et 8 640 000 valeurs par seconde en PAL. En réduisant la fréquence des images à 15 ou 20 images par seconde, vous améliorerez considérablement les performances si vous utilisez Jitter pour un traitement lourd.

Nos paramètres préférés

Nous avons constaté que, pour la plupart des films, les paramètres suivants donnent systématiquement de bons résultats :
- Taille d'image 320x240
- 30 images par seconde
- Pistes vidéo : Codec Photo-JPEG, en utilisant un paramètre de qualité spatiale de moyenne à élevée
- Pistes audio : aucune compression

Résumé

Le modèle temporel de QuickTime est quelque peu différent du modèle standard basé sur les images. Il s'appuie sur une valeur d'échelle de temps pour déterminer le nombre de valeurs de temps dans une seconde de média QuickTime. L'objet *jit.movie* permet de naviguer dans la lecture en utilisant à la fois les modèles de trame et d'échelle de temps. Toutes les fonctions de montage de l'objet *jit.movie* utilisent le modèle d'échelle de temps pour déterminer la portée de leur effet.

Déterminer les paramètres idéaux pour un film utilisé dans Jitter est une science inexacte. Les performances réelles dépendent d'un certain nombre de facteurs, notamment le codec, les dimensions, la fréquence d'images et la complexité du média. Pour obtenir les meilleurs résultats sur le hardware actuel, nous recommandons d'utiliser des films en 320x240, 15 fps, compressés en Photo-JPEG.