

Plate-forme : Mac OS/PC Marque : Steinberg

Sujet : VST/32 et 32 bit float

Cubase VST/32 a vu l'introduction de la possibilité d'enregistrer avec une résolution de 32 bit float. Ce format a déjà été implémenté il y a quelque temps dans WaveLab et plus récemment dans Nuendo, mais quelques rappels peuvent sans doute être utiles. Commençons tout d'abord par dissiper ce qui est un malentendu malheureusement assez répandu. Le "bus" interne audio de Cubase est au format natif appelé virgule flottante (floating point). Ceci signifie que Steinberg utilise le type de nombre en virgule flottante supporté à la fois par leur système de développement et votre système d'exploitation.

Que sont les nombres à virgule flottante (floating-point numbers) ? C'est simplement l'une des nombreuses façons de représenter une valeur dans un petit "coin" de la mémoire de votre ordinateur, ou dans une section de votre disque dur. Il existe en fait deux types de nombre : les nombres entiers (integers) et les nombres à virgule flottante (floating-point numbers).

Les nombres à virgule flottante diffèrent des entiers par le fait qu'ils peuvent "changer d'échelle" de façon interne afin de pouvoir représenter aussi bien de très grands nombres que de très petits sans perdre aucun détail important. L'autre type, les entiers, ont une plage fixe, finie : un entier 16-bit peut représenter 65535 valeurs discrètes différentes. Un nombre 24-bit peut représenter 16777216 valeurs discrètes différentes. Et un nombre ENTIER 32-bit peut représenter 4294967296 valeurs discrètes différentes. Tout ceci paraît déjà énorme, mais tous ces nombres entiers ont le même désavantage : à mesure que le nombre qu'ils doivent représenter diminue, le nombre de bits utilisé pour décrire le nombre lui-même décroît lui aussi. L'ERREUR SIGNIFIANTE dans la représentation d'un petit échantillon analogique augmente à mesure que le nombre devient de plus en plus petit, car ce nombre peut seulement être 1, 2, 3, n. le nombre 1 et 1 bit n'existe pas par exemple.

C'est exactement ce que permettent les nombres à virgule flottante : un nombre avec un point décimal mobile. Un nombre peut alors être représenté en parties fractionnaires (fractional parts). Le nombre peut en effet être 1.00001, ou 2.55555, ou 95522041.1456, peu importe. Dans la représentation interne d'un nombre à virgule flottante, certains bits (12 je crois) sont simplement utilisés pour représenter le nombre entre 0.000000 et 0.999999, et les 20 bits restant sont utilisés pour ajuster le nombre à la plage voulue.

Pour les matheux : $2.3 * 10^3$ $4.4589 * 10^{-4}$... vous verrez ici l'analogie.

Donc un nombre à virgule flottante peut à la fois représenter des nombres très importants ET maintenir une grande précision pour les très petits nombres. Cubase VST utilise un niveau d'opération nominal dans cette plage de nombres flottants, de telle façon qu'il y ait une précision plus que suffisante pour pouvoir représenter les détails les plus fins MAIS disposer cependant d'une réserve énorme. Quelqu'un a calculé une fois que cela représentait une réserve de 1500 dB, même si j'ai personnellement tendance à penser que ce chiffre donne une fausse impression). La réserve impressionnante est ce qui vous permet d'empiler des pistes, et de baisser le master fader lorsque la somme des signaux ne peut plus être représentée par les nombres entiers de votre carte audio. Baisser le master fader revient à changer la valeur d'échelle comme les valeurs à virgule flottante sont converties en entier dans votre carte audio. L'avantage est le suivant : vous pouvez empiler les pistes, et si la sortie finale sature, vous pouvez tout simplement baisser le master fader sans être obligé de baisser le niveau de chacune des pistes jusqu'à faire disparaître la saturation et sans dénaturer le signal.

La précision à bas niveaux est la raison pour laquelle vous n'avez pas à vous préoccuper de l'optimisation des niveaux des départs auxiliaires, etc. Avec du matériel analogique, l'utilisation de faibles niveaux pourrait potentiellement augmenter de bruit de fond, alors que dans VST, ce n'est pas réellement un problème critique. Pour résumer, la représentation de l'audio par des nombres à virgule flottante sert à améliorer la définition des signaux faibles et très faibles, et permet d'être pratiquement "à l'abri" de toute saturation.

Mais la première chose que je tenais à éclaircir est que le **"bus" interne de Cubase est en mots de 80 ou 96 bit**, parce que c'est la taille en bit des nombres à virgule flottante dans la mémoire de la plupart des machines (selon le système de développement et la plateforme). Le "bus" interne de Cubase utilise des nombres bien plus grands que 32 bit virgule flottante (80 ou 96, voir ci-dessus).

La véritable nouveauté de Cubase VST/32 est que les fichiers audio peuvent être créés au format 32-bit à virgule flottante ! Un fichier 32-bit à virgule flottante possède toutes les caractéristiques d'un nombre 32 bit en mémoire. Il maintient une grande précision même lors de la représentation de très petits chiffres, et permet de disposer d'une réserve très importante au-dessus du niveau d'opération nominal. La question que tout le monde se pose est : en quoi tout cela me concerne étant donné que ma carte audio n'est que 16 ou 24 bit ? Il est sûr que l'enregistrement de données audio 16 ou 24 bit dans un fichier 32 bit n'amène aucun gain particulier, au premier regard, et il ne peut bien sûr CREER plus de précision par exemple.

Mais les avantages sont ailleurs. Considérons l'action d'exporter votre chanson en audio. Dans les versions précédentes de Cubase et la plupart des applications 16/24 bit, dès que vous quittez le monde VST à virgule flottante et prenez la décision de procéder à la conversion en nombres entiers (le master fader) et d'enregistrer le résultat dans un fichier, si vous faites une erreur, le fichier est saturé, ou alors, il n'utilise pas toute la plage de valeurs possible, augmentant alors la distorsion. Avec un fichier d'export à virgule flottante, vous pouvez exporter sans avoir à prendre de décision sur le niveau de conversion en nombres entiers. Ceci veut dire que vous pouvez soit continuer le traitement du fichier dans votre logiciel de mastering préféré, ou continuer à mixer les différentes pistes, sans avoir eu à passer votre musique "à la moulinette" à travers une réduction de qualité (en nombres entiers) non nécessaire.

Considérez l'action du mixage de réduction final, car il est de toute évidence lié à l'export audio. La raison principale de ce mixage final est que l'ordinateur ne peut gérer toutes les pistes et les effets que vous désirez simultanément. Tout en accomplissant ce mixage, vous devez choisir un niveau de conversion adéquat, de telle sorte que le fichier mixé ne "sature" pas. Par contre, avec un niveau trop faible, vous augmenterez le bruit de quantification. Mais si vous mixez dans un fichier à virgule flottante, ces détails n'ont quasiment plus aucune importance. Mixez, tout simplement. Vous perdrez BEAUCOUP moins que lorsque vous mixez dans un fichier en nombres entiers, et vous ne pourrez pas le saturer. A part l'avantage de la suppression de la saturation, le plus agréable est que ces fichiers sonnent vraiment bien ! Superbe dynamique, notamment.

Mais bien sûr, un fichier 32 bit, à virgule flottante ou pas, nécessite 33% d'espace disque supplémentaire qu'un fichier 24 bit. Il est également vrai que cette "largeur de bande" supplémentaire demandera davantage d'effort au disque dur, mais il existe en revanche une récompense : les fichiers 24 bit sont relativement difficiles à lire, puisqu'un ordinateur "aime" gérer les données sur la base de 8, 16, ou 32 bit. Pour jouer des fichiers 24 bit, les données doivent être assemblées à partir d'une partie 16bit et d'une autre 8bit, tandis que le 32 bit float est "naturellement" géré par les unités mémoire de votre ordinateur. Nous n'insisterons pas spécialement sur cet aspect, car le résultat dépend de votre système disque, mais des améliorations de performance sont cependant mesurables dans ce domaine.

Parlons maintenant d'un nouveau système d'enregistrement arrivé avec VST/32 : **TrueTape**. C'est une partie du concept, car il s'insère précisément au point où le signal 24/16 bit est converti en GRAND nombre interne VST à virgule flottante. Il génère alors PLUS de détails que le signal original contenu. Cela semble impossible et cela l'est bien sûr ; TrueTape modélise le processus d'enregistrement d'un magnétophone analogique, et comme ceci est un traitement, il génère des données directement dans le monde à virgule flottante qui est plus détaillé que l'original. Et il sonne en plus ! Si vous êtes donc intéressé par le processus TrueTape, c'est une très bonne raison d'enregistrer des fichiers 32 bit float.

Pour conclure : quand la spécification des cartes audio ASIO 2.0 a été publiée, elle incluait l'option de transfert de données entre l'application et les cartes audio au format 32 bit float. Ceci signifie qu'il existe désormais une application hôte pouvant UTILISER une carte audio à virgule flottante, et donc qu'il pourrait apparaître sur le marché des cartes audio à virgule flottante. Sans vouloir donner de vains espoirs, imaginez les bienfaits qui pourraient en découler : combien d'enregistrements avez-vous été obligés de recommencer à cause d'une crête momentanée ? Et ne vous arrive-t'il pas souvent de "perdre des bits au bout des 24 bits" pour n'en utiliser en fait que 18, juste pour vous assurer de l'impossibilité d'une quelconque saturation ? Vous êtes alors obligé d'accepter l'augmentation du bruit de quantification...

Ce serait bien si on pouvait échapper à ces problèmes, non ?

Dave Nicholson, Cubase Product Manager

Bonne musique, Support Technique Steinberg France

Tech-Info n°26 Steinberg France

p.2 de 2