

25-Analyse MIDI

introduction

Ce didacticiel décrit quelques objets basés sur le MIDI offrant des fonctionnalités étendues pour la programmation MIDI. L'objet *midiparse* peut être utilisé pour examiner les composants d'un flux MIDI brut, tandis que *midiformat* est utilisé comme un guichet unique pour la création des messages MIDI les plus couramment pris en charge. Nous utilisons également l'objet *xbendin* pour voir les valeurs étendues fournies par de nombreux contrôleurs MIDI pour la valeur du pitch bend.

En raison de la nature de ces objets, il s'agira d'un tutoriel «de démonstration» plutôt que d'un tutoriel de «construction». Cependant, prenez le temps de comprendre le flux de données à travers ces patches, car ils constituent la base de la plupart des manipulations d'une configuration MIDI complexe.

Lors de la création d'un patch contrôlé par MIDI, il peut être nécessaire d'avoir accès à toutes les notes, aux modifications de contrôleur et autres messages générés par un dispositif MIDI. Plutôt que d'utiliser de nombreux objets individuels (tels que *notein*, *ctlin*, *pgmin* et autres), nous pouvons utiliser un seul objet *midiparse* pour décoder le flux MIDI entrant. Nous pouvons également utiliser l'objet *midiformat* pour générer un flux MIDI pouvant contenir une grande variété de messages MIDI. Ces deux objets sont très utiles pour les patches de contrôle MIDI complexes.

Nous examinons également les valeurs étendues fournies pour certains types de messages, en particulier les valeurs étendues (14 bits) fournies par la plupart des claviers MIDI pour la molette de pitch bend. Comme ces messages peuvent être produits très rapidement, nous utiliserons également l'objet *speedlim* pour ralentir l'affichage des valeurs, évitant ainsi la surcharge de notre patch lors de la génération de messages à grande vitesse.

Pour ouvrir le patch du didacticiel, cliquez sur le bouton vert **Ouvrir didacticiel** dans le coin supérieur droit de la fenêtre de documentation.

Voisualisation de *midiformat* en action

Le *midiformat* est un objet complémentaire à *midiparse*, ce qui nous permet de créer de nombreux types de messages différents et de les transformer en un flux MIDI acceptable pour l'objet *midiout*. La moitié inférieure de notre patch de test montre tous les messages pouvant être créés et fournit des objets *sliders* pouvant être utilisés pour générer ces messages. Pour tester le résultat, double-cliquez sur l'objet *midiout* et sélectionnez le périphérique de sortie MIDI approprié. Ensuite, déplacez un *slider* connecté à l'objet *midiformat* - vous devriez entendre les résultats (lorsqu'ils sont audibles) à partir de votre périphérique MIDI.

Il convient de prendre certaines précautions lors de la création de messages pour *midiformat*. Comme avec tout autre périphérique basé sur des notes, vous devez associer des messages de note-on MIDI avec des messages de note-off MIDI. Vous devez également limiter le nombre de messages de contrôleur envoyés à n'importe quel périphérique, car une trop grande densité de messages de contrôleur peut potentiellement perturber un synthétiseur MIDI. Enfin, vous devez vous assurer que le canal MIDI que vous utilisez est approprié pour le périphérique que vous tentez de contrôler.

Suivi des contrôleurs MIDI étendus à l'aide de *xbendin*

Dans la plupart des cas, les valeurs de 7 bits (0-127) envoyées et reçues des contrôleurs MIDI sont suffisantes pour traiter le contrôle des synthétiseurs et des échantillonneurs. Cependant, dans quelques cas, les valeurs de 7 bits sont insuffisantes pour des informations nuancées. L'exemple le plus courant est la commande de pitchbend. Lorsqu'un synthétiseur est réglé sur un bend de +/- une octave, les pas individuels de la commande de pitchbend peuvent correspondre à des pas de hauteurs grossiers et peu musicaux. La spécification MIDI définit pitchbend pour utiliser 14 bits de données, fournissant des valeurs comprises entre 0 (bend maximal vers le bas) et 16383 (bend maximal vers le haut - l'état d'absence de bending est 8192). Cela permet un contrôle beaucoup plus fin de la hauteur et offre de meilleures chances d'obtenir des résultats musicaux. Peu de contrôleurs fournissent tous les 14 bits de précision, mais 8 à 10 bits sont très courants.

Midiparse ne fournit que les 7 bits les plus significatifs des données de bending, mais nous pouvons suivre les informations de pitchbend étendues en utilisant l'objet *xbendin*. *Xbendin* prend une entrée MIDI brute et émet des messages de pitchbend qui incluent cette plage étendue. Même si votre contrôleur MIDI n'envoie pas de véritables messages 14 bits, ils seront analysés comme si la totalité des 14 bits complets était disponible. Malheureusement, même le contrôleur MIDI le plus stable peut générer un nombre excessif de messages de pitchbend MIDI lorsqu'il suit la valeur étendue - le hardware ne peut pas se stabiliser sur une valeur unique et peut produire un nombre supérieur à l'acceptable de très petits changements. Bien que cela n'affecte pas le matériel connecté, cela peut entraîner un comportement lent d'un patch Max ou, du moins, de rendre difficile le débogage d'un patch complexe.

La solution pour cela est l'objet *speedlim*, qui prendra la sortie de n'importe quel objet (dans ce cas, *xbendin*) et limitera la sortie à un seul message dans une période de temps donnée. Dans l'exemple de patch, l'argument 250 indique à *speedlim* de générer un message au maximum toutes les 250 millisecondes (un quart de seconde). L'objet rejettera tous les messages sauf le dernier dans cette période, ce qui nous donne un ensemble moins dense de messages à gérer dans notre patch. Lorsque vous envoyez des informations de pitchbend à notre patch d'exemple, vous constaterez que le *button* et la boîte de *nombres* connectés à *xbendin* se mettent à jour beaucoup plus souvent que ceux de *speedlim*; la boîte de *nombres* connectée à la sortie de *speedlim* sera toujours mise à jour dans un délai d'un quart de seconde pour afficher la dernière valeur transmise.

Résumé

Lors de la création d'un patch complexe contrôlé par MIDI, on souhaite souvent utiliser tous les messages MIDI disponibles. Plutôt que d'avoir un grand nombre d'objets spécifiques aux messages, nous pouvons utiliser les messages *midiparse* et *midiformat* pour transmettre des données MIDI depuis et vers les périphériques tout en ayant accès à tous les types de message. L'utilisation de *midiparse* et de *midiformat* est au cœur de tout système de patching basé sur le MIDI, vaste et complexe.

Avoir accès à des informations de contrôle étendues peut nous permettre de créer des patches plus sensibles sur le plan musical. Utiliser *xbendin* pour recevoir les informations de pitch bend étendues de pitch bend, peut nous aider à créer des synthétiseurs au son plus stable dans Max. Cependant, comme certaines de ces informations peuvent être reçues trop rapidement, utiliser *speedlim* pour ralentir la réception de ces messages peut nous aider à garder nos patches réactifs.