

4: Metro et Toggle

Actions automatisées

Ce tutoriel se concentre sur les actions automatisées et la possibilité de les contrôler. Presque toutes les actions automatisées sont construites autour d'un objet *Metro*, ce qui produit des messages bang à un rythme constant. Nous allons d'abord présenter l'objet d'interface utilisateur *toggle* comme un contrôle simple pour l'objet *metro*. Nous examinerons également plus en détail les arguments de l'objet et sur le remplacement de ces arguments.

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

Jouer avec l'objet toggle

Jetez un coup d'œil au patcheur du tutoriel. Le patch le plus à gauche comprend une série de boîtes de messages et un bouton connecté à une boîte carrée. Commencez par cliquer dans cette case - nous voyons le faible X de l'intérieur s'éclaircir. Un message **1** a également été envoyé de l'objet à la boîte de nombres connectée en dessous. En cliquant à nouveau dans l'objet, le X s'assombrit et envoie un **0**. C'est tout ce que fait cet objet. Lorsqu'il est modifié, il génère une valeur nulle ou différente de zéro, selon le nouvel état (désactivé ou activé). Le nom officiel de cet objet est *toggle* et c'est l'un des objets les plus utilisés dans la programmation Max.

La boîte à cocher *toggle* peut être activée ou désactivée en lui envoyant des messages. Cliquez sur les boîtes de *messages* qui sont connectées à *toggle* pour voir leur effet sur l'état de l'objet. En regardant la boîte de *nombres* connectée, nous voyons que tout nombre autre que 0 l'active (la marque X est lumineuse); le *toggle* renvoie également la valeur du nombre qui l'a activée. Un clic sur le message 0 désactive le *toggle* et génère un 0.

L'action de l'objet *bouton* est encore plus intéressante. Un clic sur bouton permet d'activer ou de désactiver l'objet *toggle*, et la sortie (visible dans la console Max et dans la boîte de *nombres*) alterne entre 0 et 1. Il s'agit de la racine du nom de l'objet *toggle* - sa capacité à "basculer" sur la base des messages **bang** entrants.

Introduction à metro: l'objet métronome

Le patch au centre introduit un nouvel objet: *metro*. Le *metro* produit des messages **bang**, un peu comme l'objet *bouton*. Cependant, contrairement à bouton, les objets *metro* envoient ces messages de manière répétée selon l'intervalle spécifié dans son argument. Une fois lancé, le *metro* continuera à envoyer des messages jusqu'à ce qu'il soit arrêté. Cliquez sur le *toggle* qui est connecté au *metro* et observez le résultat - il envoie un message bang (dans ce cas, à un objet bouton) une fois par seconde. Les *metros* sont activés lorsqu'ils reçoivent une valeur non nulle. Nous pouvons le désactiver en décochant le *toggle*, qui envoie au *metro* un 0. L'objet *metro* est l'un des objets Max pouvant être utilisés pour créer des actions automatisées ou auto-exécutées dans nos programmes.

Si nous cliquons sur le *bouton* qui est connecté au *metro*, nous voyons que cela fait également fonctionner le *metro*. Si nous cliquons à nouveau sur le bouton, nous voyons que cela déclenche un bang immédiat du *metro*. Cliquez rapidement sur le *bouton*, et il y a un bang pour chaque clic. Lorsque nous nous arrêtons, il y aura une seconde d'attente pour le prochain bang. Quand un *metro* reçoit un bang, le *metro* se remet en marche et programme le prochain bang à partir du moment où nous l'avons déclenché. Lorsque nous voulons l'arrêter, nous pouvons cliquer sur le message **stop**.

Comment détermine-t-on la vitesse à laquelle le *metro* produit des messages bang ? Nous déterminons la vitesse d'un *metro* en définissant sa période. L'objet *metro* prend un **argument** (le nombre 1000, dans ce cas) qui définit l'intervalle de temps (en **millisecondes**) entre les messages. 1000 millisecondes est égal à 1 seconde, c'est pourquoi nous voyons les messages bang sortir à ce rythme. Comment pouvons-nous changer ce taux ? C'est la fonction de l'**entrée droite** de l'objet *metro* - elle vous permet d'envoyer un nouveau paramètre d'intervalle de temps sans avoir à saisir une nouvelle valeur d'argument. La nouvelle valeur prendra normalement effet une fois l'intervalle actuel écoulé. (L'exception à cette règle est un changement d'intervalle qui est le résultat direct du bang du *metro* - qui sera effectif immédiatement.)

Dans notre patch, il y a une boîte de *nombres* avec plusieurs boîtes de *messages* qui y sont connectées. Cliquez sur le message 2000 et (après le prochain bang) la sortie ralentira à un bang toutes les deux secondes. Cliquez sur le 500 et le *metro* s'accélère, en envoyant des messages bang deux fois par seconde. Le message 250 causera quatre bangs par seconde. La modification du réglage de l'intervalle ne modifie aucun autre aspect du *metro* (par exemple, qu'il soit "activé" ou "désactivé").

L'intervalle de temps d'un *metro* n'a pas besoin d'être spécifié en millisecondes. D'autres *formats de temps* sont disponibles pour spécifier des intervalles en battements ou en valeurs de rythme. Le tempo indiqué dans les temps ou les valeurs de rythme est contrôlé par un objet de *transport*.

Changer l'intervalle avec un message dans l'entrée droite qui ne change pas non plus l'argument du *metro*. Pourquoi pas ? Ne serait-il pas pratique de voir la nouvelle valeur ? Peut-être, mais cela signifierait que toute modification de l'objet modifierait la valeur sauvegardée avec le patch - et vous souhaitez la plupart du temps conserver l'argument saisi car il s'agit de la valeur de départ par défaut du *metro*. La plupart des objets Max se comportent de la manière suivante: ils ont un ou plusieurs arguments qui peuvent être **remplacés** par des messages envoyés dans leurs entrées.

Exécution en parallèle

Le troisième patcheur de notre didacticiel comprend un seul *toggle* principal associé à trois autres objets *toggle* supplémentaires, chacun d'entre eux étant connecté à un *metro* avec un argument différent. Enclenchez le *toggle* principal et les objets *metro* génèrent un spectacle lumineux au galop. En effet, les trois arguments d'intervalle de temps des objets *metro* sont liés: le premier *metro* se déclenche toutes les secondes (1000 millisecondes), le second se déclenche trois fois par seconde (333,33 ms) et le troisième se déclenche tous les quarts de seconde (250 ms). Il n'y a rien qui gère ce modèle - cela se produit simplement parce que la sortie programmée des objets *metro* est constante et que le modèle continue d'apparaître. Notez que l'intervalle de temps d'un *metro* peut être un nombre **entier** (par exemple **1000**) ou un nombre à **virgule flottante** (par exemple **333,33**).

Développons l'idée de créer des motifs rythmiques en utilisant certaines des autres idées que nous venons d'apprendre. Nous pouvons modifier le modèle en modifiant les valeurs temporelles de chaque *metro*. Commençons par ajouter trois boîtes de *messages*, de valeurs **500**, **200** et **750**, et en connectant une à l'entrée droite de chaque *metro*. Lorsque nous cliquons sur elles pour modifier les intervalles de temps de nos objets *metro*, aucun motif apparent n'existe. En effet, nous devons synchroniser le début des objets *metro* pour voir le modèle. Ceci peut être accompli en activant et désactivant le *toggle* principal. Le résultat est un voyant central clignotant rapidement, avec un motif extérieur syncopé.

Créons un autre motif en utilisant les boîtes de *messages* pour les valeurs **200**, **400** et **600**, respectivement. Cependant, au lieu d'utiliser le bouton *toggle* pour activer / désactiver les objets *metro*, connectez un seul objet *bouton* à l'entrée gauche des trois objets *metro*. Cliquez sur les trois

boîtes de messages, puis sur le bouton pour redémarrer tous les objets metro. Le bouton force les objets metro à redémarrer de manière **synchrone**, nous donnant ainsi le spectacle de lumière continue attendu.

Résumé

Les actions automatisées sont au cœur de nombreux patchs de performance Max, et les objets *metro* sont souvent utilisés comme point de départ pour ces actions. Une fois démarré, le système de programmation de Max fournit un système de timing stable pour la sortie Max et permet d'exécuter les événements sans interaction supplémentaire.

Contrôler les patchs à l'aide de l'objet *toggle*, semblable à une case à cocher, est également un outil de programmation clé, puisqu'un seul *toggle* peut fournir une simple interface marche / arrêt pour des processus beaucoup plus complexes.