

30-capture de geste

introduction

Ce tutoriel se concentre sur la capture et la lecture de données. En gros, nous allons effectuer un séquençage. Cependant, contrairement au séquençage présenté dans les didacticiels MIDI, il s'agit du séquençage de données arbitraires (mouvement de la souris, dans ce cas). Nous allons développer notre utilisation de *coll*, cette fois en collectant et en reproduisant les données pour un patch de dessin intéressant.

Le cœur de nombreux outils de performance est la capacité de capturer et de reproduire un mouvement, qu'il s'agisse de données de capture de mouvement, de données de capteurs ou de mouvements de souris. Ce didacticiel présentera les bases des systèmes de capture et de lecture et introduit également quelques modèles de conception intéressants.

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

Exécuter le programme, visualiser les données

Jetez un coup d'œil au patcheur du tutoriel. C'est un patch assez complexe, et l'ensemble sert un seul but: capturer le mouvement de la souris et le transformer en dessin intéressant. Ce qui est intéressant, c'est que, tout en recevant les mouvements de la souris et en dessinant l'image, le patch enregistre également les mouvements. Comme ce patch dépend d'un timing régulier, indépendamment des autres activités de l'ordinateur, vous pouvez activer **Overdrive** dans le menu **Options**.

Conseil important: vous ne voulez pas interagir avec ce patch en utilisant la souris, car cela aurait tendance à perturber votre dessin. Remarquez plutôt la section basée sur les objets *key* située juste au-dessus de l'objet *lcd*: elle associe trois touches (espace, «r» et «p») à trois fonctions du patch («clear», «record» et «playback»). Pour commencer, appuyez sur la touche “r” (les minuscules sont importantes!) pour lancer le processus d'enregistrement. Maintenant, déplacez votre souris sur l'écran - les boucles et les grands mouvements sont particulièrement efficaces. Lorsque vous avez terminé, appuyez à nouveau sur la touche “r” pour désactiver l'enregistrement.

Vous avez maintenant un dessin intéressant, ressemblant à un script, sur le *lcd*. Appuyez sur la barre d'espace pour effacer l'écran, puis appuyez sur la touche “p” pour commencer la lecture. Vous verrez le dessin se produire: non seulement le dessin a été capturé, mais le mouvement réel de la souris (et le timing) a été capturé et reproduit.

Ce patch contient trois objets *coll*, mais tous font référence aux mêmes données: les données d'un *coll* nommé **thegest**. (L'argument **-1** empêche Max de rechercher un fichier de ce nom lorsque le patch est chargé.) L'accès à la copie de droite est le plus pratique. Cliquez deux fois dessus pour révéler les données capturées. Le contenu de *coll* montre que notre mouvement est suivi et capturé à intervalles de 20 millisecondes. Voyons comment cette capture s'est produite.

La partie enregistrement

- Double-cliquez sur l'objet *patcher d'enregistrement* (**p**) pour l'ouvrir.

La section RECORD du patch est lancée avec le gros objet *toggle* rouge. Lorsqu'il est activé, il réinitialise le patch (via l'objet *sel 1* connecté à la sortie droite du *trigger*), puis lance un *metro* qui se déclenche toutes les **20** millisecondes. Cela détermine le *taux d'échantillonnage* du suivi de la souris et sert également de déclencheur pour les autres objets. Le *metro* envoie un message **bang** à deux objets: un subpatch (la version réduite de **WTHITM** que nous utilisons) et un objet *int* (abrégé "i").

L'abstraction **WTHITM** devrait déjà être familière maintenant - elle suit la position actuelle de la souris et produit un ensemble de coordonnées X et Y mises à l'échelle dans la plage donnée en argument. Dans ce cas, le subpatch émettra des valeurs comprises entre **0-320** et **0-240** chaque fois qu'il recevra un message **bang**. Afin de fournir un retour immédiat, la sortie du subpatch est envoyée à un subpatch de dessin qui génère des commandes de dessin pour l'objet *lcd*.

L'objet *int* attaché au *metro* et l'objet **+** qui lui est connecté réalisent une fonction de comptage. Les messages **bang** entrants produiront la valeur actuelle (comme indiqué dans la boîte de *nombres*). Cependant, la valeur de sortie ira également à l'objet **+**, qui ajoutera un à la valeur et la stockera dans l'entrée droite de l'objet *int* (**i**) pour attendre le prochain message **bang**. Le résultat est que chaque message **bang** génère une valeur, mais l'incrémente également de **1** - un peu comme un objet *counter*, mais sans limite supérieure. C'est un moyen utile d'obtenir un nombre toujours croissant dans des situations où vous ne savez pas quelles sont les limites.

Les valeurs de ces deux objets (le nombre actuel et les positions X et Y mises à l'échelle) sont assemblées avec un objet *pack*, puis envoyées à l'objet *coll*. La première valeur de la liste (le nombre actuel) agit comme un index, indiquant à *coll*. de stocker les autres valeurs à cette position. En conséquence, le *coll* agit comme un enregistreur de données, prenant les valeurs indexées et les stockant pour une utilisation ultérieure. La collecte de données se poursuit jusqu'à ce que la fonction d'enregistrement soit désactivée, soit en cliquant sur la case rouge, soit en appuyant sur la touche «**Γ**».

Prenez un moment pour regarder ce que fait l'objet *select 1* connecté au *trigger* d'enregistrement. Lorsque l'enregistrement commence, un **bang** est envoyé à un deuxième objet *trigger* qui, de droite à gauche, désactive la boîte *toggle* "play" (arrêtant toute lecture), envoie un message **clear** à l'écran *lcd*, efface le contenu du *coll*. (avec un autre message **clear**) et met à zéro l'objet *int* pour réinitialiser le compteur d'index.

La partie lecture

- Double-cliquez sur l'objet *patcher play* (**p**) pour l'ouvrir.

La partie de lecture du patch reprend une grande partie de la section enregistrement, à l'exception du suivi de la souris. Au lieu de cela, elle demande simplement les valeurs indexées à *coll* nommée **thegest**, et utilise le même patch de dessin pour recréer les mouvements de la souris. Elle est capable d'utiliser la même technique d'incrémentation des nombres entiers que celle utilisée dans la section record; cela a l'avantage de ne pas nécessiter la définition d'une limite supérieure, mais d'inconvénient de ne jamais s'arrêter ou de ne jamais boucler sur elle-même.

Comme la fonction d'enregistrement n'a pas *horodaté* les données capturées (elle les a simplement indexées avec la sortie de l'objet *integer*), cela signifie que nous pouvons manipuler la lecture du dessin en modifiant la vitesse à laquelle nous demandons le contenu du *coll*. Le moyen le plus simple consiste à modifier la vitesse du *metro* en lecture.

Déverrouillez le patch et ajoutez une boîte de nombres entiers dans la zone de la section de lecture. Connectez sa sortie à l'entrée droite du *metro* (l'entrée droite contrôle l'intervalle de sortie). Entrez **2** dans la boîte de *nombre*, puis appuyez sur "p" pour la lecture. Le dessin se déroulera comme avant, mais 10 fois plus vite. Arrêtez le dessin (appuyez sur "p") et modifiez la boîte de *nombre* en **40**, appuyez à nouveau sur la touche "p", et le dessin se déroulera à la moitié de sa vitesse. Comme vous pouvez le constater, cette technique de capture de données vous offre une grande flexibilité au moment de la lecture.

Une fois que vous avez obtenu une capture de données intéressante, vous voudrez probablement l'enregistrer pour une utilisation ultérieure. Le *coll* à l'extrême droite (souligné en rouge) montre la méthode simple pour sauvegarder et récupérer le contenu d'un *coll*; un message de **lecture** vous permettra d'importer des données à partir de fichiers de disque, tandis qu'un message **d'écriture** prendra vos données capturées et les stockera sur disque. Vous pouvez ajouter un nom de fichier réel et un chemin d'accès au message si vous avez un fichier spécifique à lire ou à écrire, tandis qu'un message en lecture / écriture sans argument ouvrira une boîte de dialogue de fichier.

Faites quelques dessins et sauvegardez-les en utilisant le message **d'écriture** dans le *coll* de droite. Si vous les relisez en utilisant le message **read**, vous pourrez lire vos dessins sauvegardés. Notez que les fichiers de données créés par les objets *coll* sont simplement des fichiers texte. vous pouvez ensuite les ouvrir dans n'importe quel éditeur de texte ou les importer dans un autre programme pour les visualiser ou les manipuler.

Résumé

Dans ce didacticiel, nous avons vu comment utiliser l'objet *coll* en tant que dispositif de collecte de données et nous avons appris à concevoir un compteur sans limite supérieure. Nous avons également vu comment l'objet *coll* peut être utilisé pour le stockage et la récupération sur disque, ce qui nous permet de sauvegarder et de réutiliser nos données capturées. Enfin, ce patch est un bon exemple d'un ensemble de données utilisé par trois objets *coll*, une méthode de partage des données qui devient très utile avec les patchs volumineux.