

RENCONTRES #EDMUS  
TOULOUSE 14/03/2015

# CRÉER UNE APPLICATION MUSICALE AVEC MAX



@vieroux44  


# PRÉAMBULE

Bienvenue dans le monde merveilleux des programmeurs d'applications musicales ! D'ici quelques minutes, tu devrais être capable de concevoir de petites apps' très efficaces, et ceci sans avoir à toucher une seule ligne de code !

## MAX OU PURE DATA ?

Ces deux logiciels sont issus d'un seul et même logiciel : *Patcher*. Développé par Miller Puckette à l'Ircam dans les années 80, ce logiciel révolutionnaire permettait d'assembler des objets préprogrammée pour répondre à des besoins musicaux très spécifiques... et souvent uniques.

Que choisir ? Voici un petit tableau bien subjectif...

	MAX	PURE DATA
+	Interface agréable Tutoriels bien conçus Documentation très fournie et aide omniprésente Mises à jour régulières Partie vidéo très développée Création d'interfaces simplifiée Export d'applications autonomes Passerelle possible avec <i>Ableton Live</i> .	Libre MacOS, Windows et Linux Version portable (exécutable sans installation préalable) Très léger
-	Payant (250\$ = prix enseignant) ou 59\$ / an. MacOS et Windows uniquement	Interface peu engageante Quelques bugs Moins de fonctionnalités

Aujourd'hui, nous aborderons principalement Max, la version de démonstration étant pleinement fonctionnelle durant 30 jours.

Par la suite, la logique des deux logiciels étant la même, il n'est pas très difficile de passer d'une plateforme à l'autre.

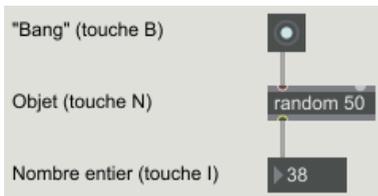
# LA LOGIQUE

Max est un peu comme un jeu de LEGO numérique. Chaque « brique » possède des caractéristiques qui lui sont propres, mais différents agencements permettent de concevoir aussi bien un château fort qu'un bateau, qu'une voiture de course ou qu'une réplique grandeur nature de l'Arc de Triomphe.

Nos briques seront appelées « Objets ». Il en existe des centaines, mais nul besoin de tous les connaître pour commencer à programmer, bien au contraire !

Les objets peuvent être reliés entre eux, en partant du haut vers le bas.

Pour passer du mode édition au mode lecture et inversement, clique sur le verrou en bas à gauche !

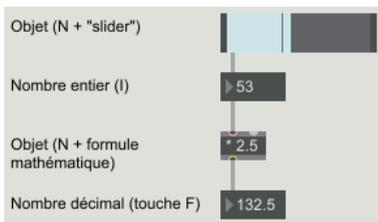
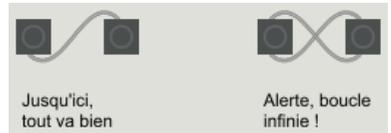


Nous voyons dans cet exemple l'objet **Bang** déclencher l'objet **random**, lequel détermine un nombre aléatoire de 0 à 50 et envoie le résultat vers un nombre.

Nous retrouvons bien la lecture de haut en bas !

Autre exemple de cette logique, avec un **bang** qui déclenche un second **bang**.

Nous voyons dans le deuxième exemple la création de ce qu'un musicien appelle le larsen !

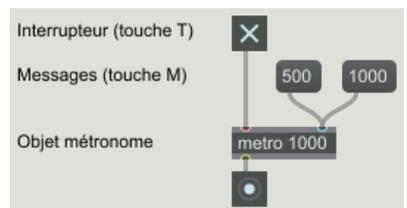


Ici, nous découvrons l'objet **slider**. Pour le créer, place un objet vide sur le plan de travail, et tape « **slider** » à l'intérieur. L'objet se transforme en une barre que l'on peut faire glisser à la souris.

Plus bas, nous voyons l'utilisation d'une fonction mathématique (multiplication par 2.5), dont le résultat s'affiche sous forme décimale.

Finalement, ce patch permet de déclencher un métronome à l'aide d'un interrupteur. Par défaut, celui-ci envoie une impulsion toutes les 1000 millisecondes.

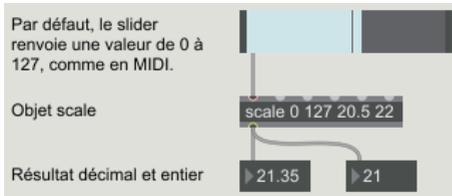
Deux **messages** sont connectés à la seconde entrée du métronome, et permettent de changer le paramètre par défaut.



## DES MATHS...

Max est une formidable machine à calculer. Nous avons déjà vu un opérateur mathématique (la multiplication) ainsi qu'un objet générant un nombre aléatoire (*random*).

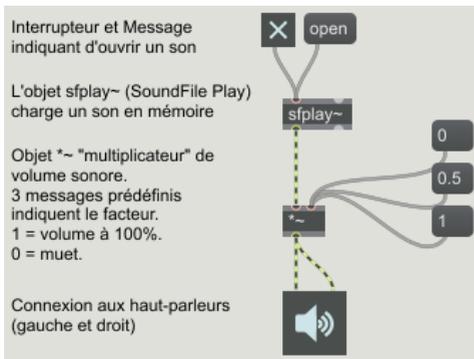
Maintenant, à toi d'expérimenter un nouvel objet, nommé *scale*. Celui-ci permet de convertir une plage de données (par exemple de 0 à 127) vers une autre échelle (par exemple de 20,5 à 22).



L'objet *scale* va convertir la valeur par défaut du slider vers une autre plage de valeurs, ici beaucoup plus restreinte.

*Scale* renvoie le résultat vers deux objets : un nombre décimal (valeur exacte) et un nombre entier (valeur arrondie).

## DU SON...



Les objets sonores se repèrent dans Max grâce au tilde (~) à la fin de leur nom. Visuellement, les connexions entre objets n'ont pas non plus le même aspect.

Ici, nous voyons un patch de gestion du volume.

La partie du haut sert à charger un son. La partie centrale change le volume. La dernière partie envoie le son aux enceintes.

## DES MATHS ET DU SON...

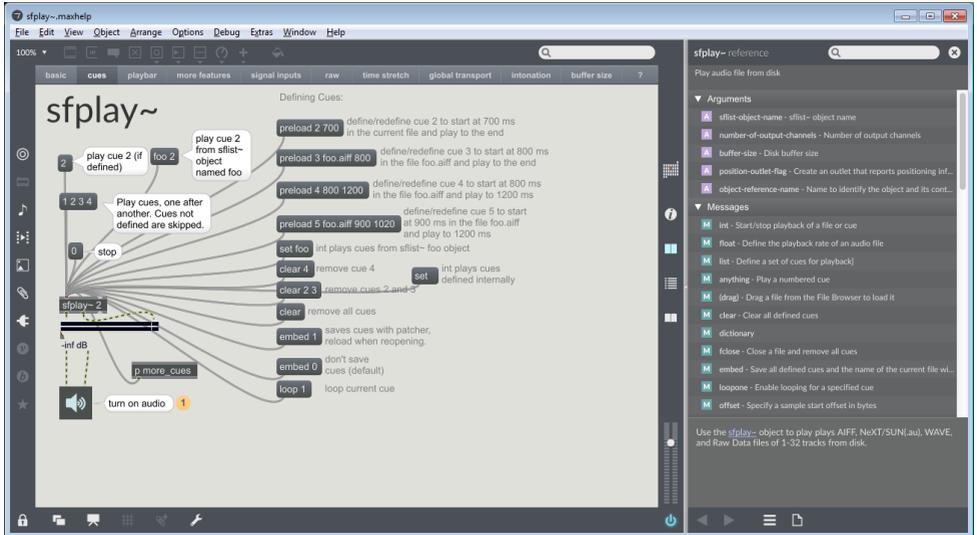
À toi d'imaginer un patch servant à régler la balance stéréo d'un enregistrement donne. Il te faudra :

- Un *slider* horizontal pour l'utilisateur.
- Deux *sliders* verticaux, « témoins » du volume des deux canaux.
- Les objets musicaux *sfplay~*, *ezdac~* et *\*~*.
- L'objet mathématique *scale*.

## BESOIN D'AIDE ?

À ce stade, certains objets peuvent nécessiter quelques explications plus approfondies. L'un des atouts de Max est l'omniprésence de l'aide, car chaque objet est très documenté. Mieux encore, il est particulièrement simple de naviguer dans l'aide, à la recherche d'une fonction dont on ne connaîtrait pas le nom.

Pour accéder à la rubrique d'aide d'un objet, clique-droit sur celui-ci et sélectionne *Help* (nombreux exemples d'usages) ou *Reference* (documentation exhaustive).



Le menu d'aide permet de tester de très nombreux paramétrages de l'objet, grâce à la navigation par onglets.

Le dernier onglet (« ? ») liste d'autres objets connexes, dont la fonction est proche ou complémentaire de l'objet initial. On peut aussi y retrouver des liens vers les tutoriels inclus dans le logiciel.

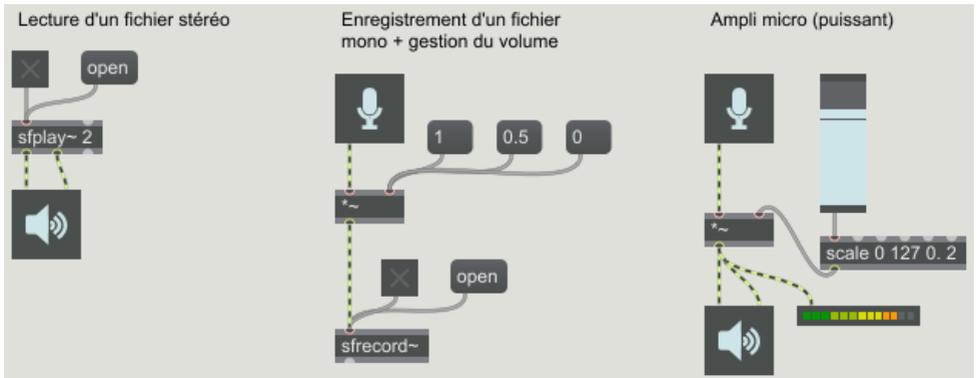
## PIMP MY PATCH

Les objets peuvent être personnalisés, tant dans leur apparence que dans leur fonctionnement. Pour cela, ouvre l'inspecteur (clic-droit sur un objet ou bouton « i » dans la barre de droite).

# OBJETS MUSICAUX

Les objets musicaux incontournables pour débiter sont les suivants :

sfplay~	suivi éventuellement du nombre de canaux (1 = mono, 2 = stéréo) Lit un son, chargé au préalable avec le <b>message</b> « open ». Nécessite un interrupteur.
sfrecord~	suivi éventuellement du nombre de canaux (1 = mono, 2 = stéréo) Enregistre un son, chargé au préalable avec le <b>message</b> « open ». Nécessite un interrupteur.
*~	Module le volume du son. Attention dès que l'on dépasse le 1 !
ezadc~	« Easy Audio-Digital Converter » = entrée son.
ezdac~	« Easy Digital-Audio Converter » = sortie son.
meter~	VU-mètre indiquant visuellement le volume sonore d'un objet musical.

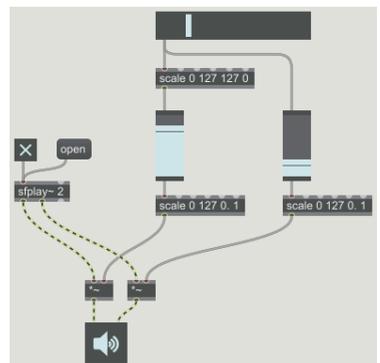


## CORRECTION DE LA PAGE 4

Une valeur de 0 à 127 est envoyée par l'utilisateur via le **slider** en haut du patch.

Les deux **sliders** suivants servent d'indicateurs visuels ; celui de gauche est à l'opposé de son voisin grâce à l'objet **scale**.

Les valeurs de 0 à 127 sont finalement converties en valeurs de 0 à 1, pour moduler le volume des canaux gauche et droit indépendamment.

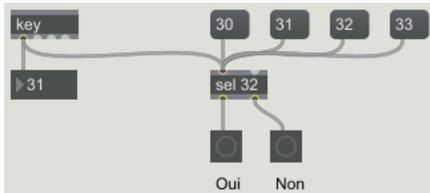


## CRÉATION D'UN SAMPLER

Voyons comment créer un *sampler*, afin de pouvoir déclencher des sons à l'aide d'un clavier d'ordinateur et d'enregistrer sa performance en direct.

Deux nouveaux objets seront utilisés :

- **key** renvoie la valeur de n'importe quelle touche pressée sur un clavier.
- **sel** teste une valeur rentrante afin de la rediriger au bon endroit.



Ce patch teste les valeurs entrantes, soit par le clavier, soit pas l'un des 4 **messages** sur lesquels l'utilisateur peut cliquer. Si la valeur entrante est 32 (sur le clavier, il s'agit de la touche Espace), le premier **bang** s'allume. Dans le cas contraire, il s'agira du second.

Pour notre projet, nous souhaitons pouvoir charger 6 sons différents et les assigner aux touches A, Z, E, R, T et Y.

- Dans un premier temps, note les valeurs de ces 6 touches et insère 6 lecteurs, tous connectés à une (*ezdac~*) et à un enregistreur (*sfrecord~*).
- Ensuite, crée un objet « **sel** » pour la première touche (lettre A).
- Si la valeur renvoyée par « **key** » est la bonne, cela doit déclencher l'interrupteur et la lecture d'un son.

## DIRECTION LA CLASSE !

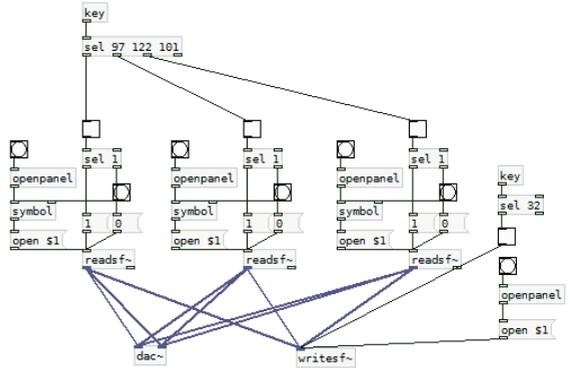
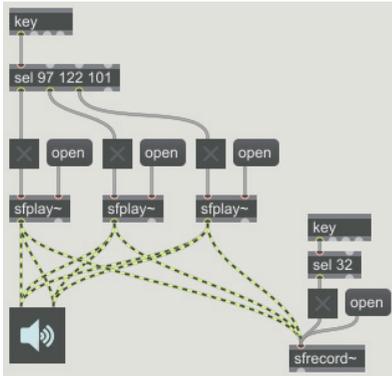
Lorsqu'un patch est terminé, il est utile d'en peaufiner l'apparence afin d'offrir aux élèves un outil dont l'interface est claire et fonctionnelle.

Le bouton « *Presentation mode* » permet de basculer vers une vue nouvelle, où seuls les objets choisis apparaîtront. Pour sélectionner ces objets, un clic-droit sur l'un d'eux puis « *Add to Presentation* » suffisent !

Finalement, le menu « *File / Build collective / Application...* » permet de sauvegarder son programme au format \*.exe ou \*.dmg. Les applications ainsi générées sont librement distribuables et peuvent être utilisées en classe sans restriction aucune.

# COMPARAISON MAX / PURE DATA

Voici un exemple de réalisation sur Max et sur Pure Data. Il s'agit du sampler évoqué à la page précédente... optimisé et sur les seules touches A, Z et E. La touche espace sert à déclencher / arrêter l'enregistrement.



# PILOTAGE DEPUIS UN TERMINAL MOBILE

Max et Pure Data comprennent tous deux le langage OSC, permettant une interaction entre machines.

L'application TouchOSC (iOS + Android) permet de concevoir une interface entre son ordinateur, son téléphone et/ou sa tablette. Les données sont simplement récupérées dans Max à l'aide d'un objet librement téléchargeable sur le site de l'éditeur.

