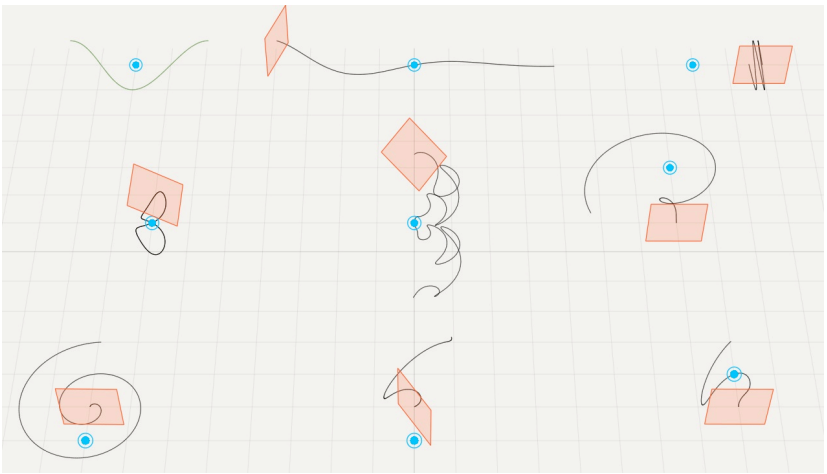
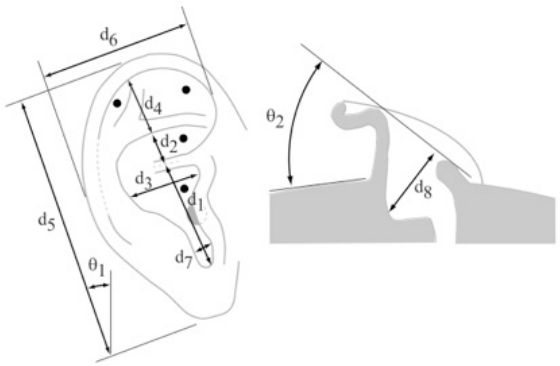
ENCELADE

www.encelade.info

Iannix et Binaural en MIDI

Tutoriel v1.0 (06/02/16)





1. Préliminaires 2
1.1 Présentation 2
1.2 Prérequis 3
2. Mise en place 5
3. Iannix 7
3.1 Aide contextuelle 7
3.2 Fichiers d'exemple 8
3.3 Les objets de Iannix 8
3.4 Générer des ordres MIDI 9
3.5 Calibration des messages MIDI 12
4. Iannix + Binaural 14
4.1 le plugin binaural cSPAThrtvf 14
4.2 Piloter le plugin par Iannix 15
5. Perspectives 17

1. Préliminaires

1.1 Présentation

IanniX est un séquenceur graphique open-source, multi-plateforme Linux, Mac et Windows, inspiré des travaux de Iannis Xenakis et destiné à la création numérique.

Iannis Xenakis est un compositeur contemporain qui a, notamment, créé la musique stochastique - une musique basée sur des évènements probabilistes, c'est-à-dire définis par les mathématiques des probabilités. Les travaux de Xenakis ont largement porté sur une approche mathématique, formelle, des sons et de leur nature, dans le but de transposer les outils de composition classique, et la composition elle-même, dans le monde informatique. Il a introduit un changement de paradigme dans la manière de considérer des évènements sonores, et d'écrire la musique.

Ce tutoriel a pour but d'illustrer un des usages possibles de Iannix : contrôler la position dans l'espace d'un son pour une écoute binaurale, en se basant sur des objets suivant des trajectoires mathématiques définies dans Iannix.

L'écoute binaurale reproduit un espace 3D sonore plus complet que la seule panoramique gauche-droite ; en utilisant des algorithmes basés notamment sur certains paramètres physiologiques de la tête (modélisation), il est possible de reconstruire, lors d'une écoute au casque, une image sonore semblant placée à gauche, à droite, en avant, en arrière, plus haut ou plus bas que la tête de l'auditeur ; et créer aussi l'illusion d'une source plus ou moins lointaine sur ces trois axes.

Malgré une certaine complexité visuelle, et un workflow de création parfois un peu déroutant, Iannix est un logiciel qui s'avère relativement simple une fois qu'on a compris ses concepts. Après la lecture de ce document, il sera possible d'utiliser Iannix et Ableton Live (ou toute autre DAW supportant le MIDI In) en synchronisation, et d'utiliser Iannix pour piloter les paramètres de la DAW et de ses plug-ins. En ajoutant quelques sessions de travail personnelles sous Iannix pour creuser les points particuliers à chacun, cet environnement devrait pouvoir être maitrisé sans trop de difficultés. Iannix fait bien plus que parler le MIDI : il est compatible avec les protocoles OSC, série (Arduino), http... ce qui le rend très versatile dans ses usages, et utile à de nombreux contextes. C'est en l'étudiant et en le pratiquant qu'il vous révèlera son intérêt, et que vous comprendrez sa puissance, et la manière dont il peut vous servir.

Ce tutoriel, réalisé par Encelade[1], a pour objectif de permettre une première approche de Iannix à toute personne désirant explorer les vastes possibilités de ce logiciel. Si vous y trouvez des erreurs, ou des améliorations possibles, n'hésitez pas à me corriger ou à participer à l'élaboration du document si vous le désirez (contact@encelade.info).

[1] www.encelade.info

1.2 Prérequis

- **Nous supposons que le lecteur possède une connaissance de base du protocole MIDI.**

  Un message MIDI est composé d'un numéro de canal, d'un ordre de type NoteOn/Off, Continuous Control (CC), Program Change (PC)... et en fonction de l'ordre, de une ou plusieurs valeurs associées : hauteur de note et vélocité pour NoteOn, numéro de contrôleur et valeur pour CC, numéro de PC...

  Internet regorge de sources de connaissances sur ce protocole très largement répandu dans le domaine de la Musique Assistée par Ordinateur, notamment le site officiel du MIDI[2].

- **Ableton Live + Max4Live, ou toute autre DAW MIDI**

  Nous allons lier Iannix à un logiciel capable d'interpréter (comprendre) des messages MIDI, pour piloter certains de ses paramètres. Nous utilisons pour ce tutoriel la Digital Audio Workstation (DAW) Ableton Live, et son module Max4Live pour le plugin binaural ci-dessous. Toute autre DAW compatible MIDI, permettant l'assignation MIDI sur mesure, pourra être utilisée pour suivre ce tutoriel. Il faudra simplement transposer les concepts de configuration MIDI (localisation des ports MIDI de Iannix dans votre DAW) et d'assignation des messages MIDI (MIDI Learn ou équivalent) dans votre DAW, et utiliser un autre plugin de modélisation binaurale

- **Plug-in de modélisation binaurale**

  C'est cet élément qui sera piloté par Iannix. Ce tutoriel a été construit sur le plugin csPAThrtf, téléchargeable gratuitement ici[3]. Une fois l'archive zip téléchargée et décompressée, suivre les instructions du fichier ReaMeFirst pour l'installation concernant Max4Live. Sur la page du proje, site de Cyclin74, nous lisons que la restitution binaurale se base sur des modèles physiologiques d'une tête humaine et que par défaut, un seul modèle est distribué avec cSPAThrtf. Il est possible d'en essayer d'autres[4]. Placer les fichiers téléchargés dans le répertoire « lib » de cSRPAth[5].

  Pour celles et ceux qui n'utilisent pas Live, Binauralizer[6] et 3-D Binaural Spatializer[7] par exemple, pourront être utilisés.

---

[2] https://www.midi.org/articles/categories/getting-started-with-midi
[3] https://cycling74.com/toolbox/cspathrtfv1/, bouton "tool site".
[4] http://4-to-one.dk/fftBinPan/All%20HRTF%20matrix%20files.rar
[5] Pour retrouver ce répertoire sous Ableton Live : clic droit sur le périphérique chargé dans une piste, « Afficher périphérique dans la fenêtre de navigateur », clic droit sur le périphérique dans le navigateur et «Afficher dans le Finder ». Le répertoire racine, contenant le périphérique, est alors localisé ; le répertoire « lib » se trouve dans cette racine.
[6] http://www.noisemakers.fr/binauralizer/
[7] http://www.dklingler.com/projects/spatializer

- **Iannix**[8]

  Iannix définit un nouveau mode de partition : les propriétés d'un son vont être liées à des déplacement d'objets (appelés « curseurs ») dans l'espace 3D selon des trajectoires mathématiques. On va pouvoir ainsi faire varier la hauteur d'une note, la quantité de réverbération, etc... en se basant sur le comportements de curseurs suivant des trajectoires définies mathématiquement. Nous reviendrons plus loin sur ces concepts ; de manière synthétique, Iannix est à la fois un mode d'écriture, et une vaste télécommande, elle-même télécommandable. En soi, Iannix ne produit ou n'interprète aucun son ; il s'agit uniquement de message de commande, envoyé à, ou reçu par, d'autres processeurs numériques (DAW) ou physiques (synthétiseur).

- **Un sniffer MIDI**

  Le sniffer est un outil logiciel qui surveille des ports et écoute les informations qui y passent. Cela nous sera utile pour comprendre le flux d'information, et pour débugger. Sur Mac OS, MidiMonitor[9] est un très bon outil, pratique, léger et simple d'utilisation. Des sniffers équivalents sont disponibles pour Windows[10] et Linux[11].

Ce tutoriel a été réalisé avec les versions suivantes des logiciels :

- Mac OS 10.8.5
- Ableton Live 9.6
- Max 7.1.0 (pour le plugin binaural Max4Live)
- cSPAThrtf V1
- Iannix 0.9.16 beta

[8] http://www.iannix.org/
[9] http://www.snoize.com/MIDIMonitor/
[10] http://obds.free.fr/midimon/
[11] http://kmidimon.sourceforge.net

2. Mise en place

Nous allons dans un premier temps apprendre à connecter Live et Iannix :

- synchronisation temporelle de Live vers Iannix ;
- réception des messages MIDI de Iannix vers Live.

Nous discuterons ensuite plus en détails le fonctionnement de Iannix.

1. Après vous être assuré de posséder les pré-requis discutés plus haut, lancer Ableton Live, Iannix, et le sniffer MIDI. L'ordre de lancement n'importe pas entre Live et Iannix, par contre préférez lancer le sniffer en dernier (pour être sûr qu'il active bien tous les ports IN/OUT disponibles à son lancement, et éviter une opération de configuration dans le sniffer).

2. La fenêtre principale de Iannix (figure 1) est composée de 4 zones :

   - une barre d'outils, en haut ;
   - un inspecteur de navigation/configuration, à droite ;
   - une barre de transport ; time line, sniffer et afficheur de performances, en bas ;
   - la partition (un espace 4D : 1 temps/ 3 espace), au centre.

   La barre de menu, comme pour toute application chargée dans votre système d'exploitation, complète l'interface utilisateur graphique.

3. Dans l'inspecteur de Iannix (figure 1), cliquer sur l'onglet « Files » puis double-cliquer sur le fichier « Curve 3D equation ». Ce fichier d'exemple donne un bon aperçu des différentes possibilités de trajectoires et comportements des objets de Iannix.

Figure 1 : la fenêtre principale de Iannix

3. Synchronisation MIDI de Live et Iannix.

3.1. Aller dans le Menu « Live → Préférences → MIDI Sync » (figure 2)

3.2 Output : chercher « To Iannix ».

3.3 Mettre sur « On » le bouton « Sync ».

3.4 Vérifier que le port « Input : From Iannix » est bien sur « On » pour Piste et Telec, et sur « Off » pour Sync. Fermer la fenêtre de préférences.

3.5 Dans l'inspecteur de Iannix, vérifier à l'onglet « Config », sous-onglet MIDI, que le port de MIDI Out par défaut est bien « From Iannix ».

3.6 Appuyer sur Play dans Live : vous devriez voir bouger les curseurs (plans orange) dans Iannix, ainsi que défiler la tête de lecture dans sa barre de transport. Déplacez-vous dans la time line de live (mode de vue Arrangement) : la tête de lecture de Iannix suit celle de Live.

3.7 Observez également le témoin MIDI In de votre DAW (figure 3) : vous devez observer un transfert d'information, un clignotement par exemple dans Live.

3.8 Enfin, regardez votre sniffer MIDI (figure 4). Vous devriez voir des informations défiler dans le sniffer. Si vous ne réceptionnez pas d'ordre MIDI, vérifier que le port "from Iannix" est bien écouté dans le sniffer. Dans MIDI Monitor, déplier les Sources, déplier "MIDI sources", et cocher "From Iannix". Les ordres MIDI devraient maintenant défiler. Il est également possible de voir le signal de synchronisation temporelle (clock) envoyé par Live à Iannix : Sources/Spy on output to destinations, cocher « To Iannix ». Un message « Clock » apparaît dans le MIDI Monitor.

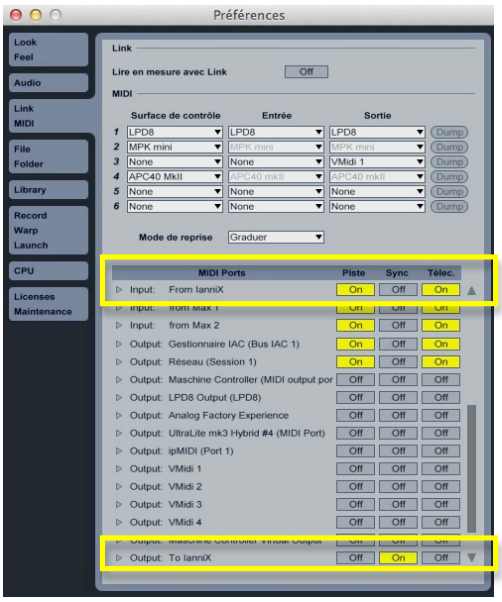


Figure 2 : Préférences MIDI de Live

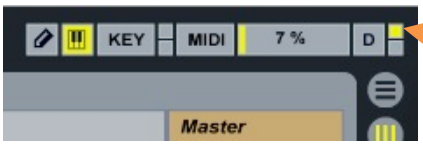


Figure 3 : indicateur MIDI In DAW OK

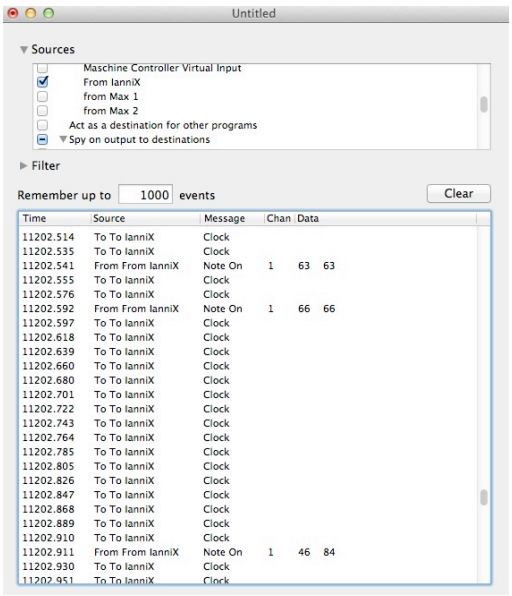


Figure 4 : sniffer MIDI Monitor OK

3. Iannix

Il n'y a pas de manuel pour Iannix. Des ressources diverses sont disponibles sur Internet, mais des outils internes à Iannix, suffisants pour commencer, sont à disposition pour son apprentissage : **l'aide contextuelle**, et **les fichiers d'exemple**.

Note : Iannix peut se manipuler à la souris via la fenêtre principale (figure 1), ou directement en ligne de script (code informatique). Une fenêtre donnant accès au code (« Script Editor » figure 5) s'ouvre en même temps que la partition. Ce mode est réservé aux usagers plus avancés. Nous ne nous intéresserons pas à cette méthode ; vous pouvez fermer cette fenêtre, et ne garder que la fenêtre principale, et celle d'aide, ouvertes.

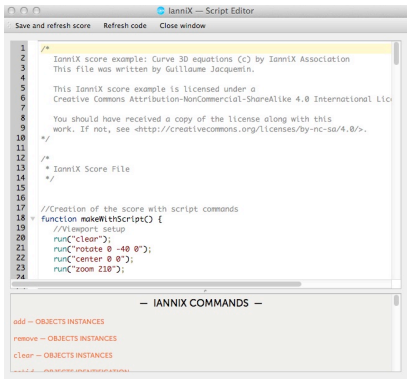


Figure 5 : fenêtre "Script Editor"

3.1 Aide contextuelle

- Appuyer sur le bouton « Help » de la barre d'outils, qui ouvre une fenêtre d'aide contextuelle (figure 6).
- Survoler les divers éléments avec la souris : outils de la barre d'outils, vue générale, inspector, onglets de l'inspecteur…). Le contenu de la fenêtre d'aide se met à jour automatiquement.
- Pour la suite des opérations, je vous conseille de laisser la fenêtre Help ouverte, et de la regarder régulièrement pour apprendre plus en détails le logiciel au fur et à mesure de votre avancée dans ce tutoriel.

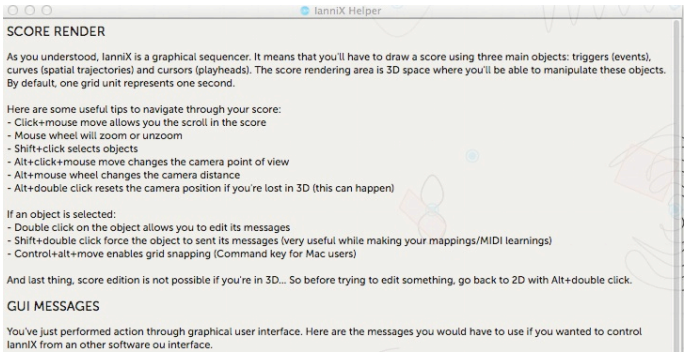


Figure 6 : outil "Aide" (en haut) et fenêtre d'aide contextuelle (à droite)

3.2 Fichiers d'exemple

- Dans la fenêtre Inspector/Files (figure 7), différents partitions pré-établies sont à disposition.
- Après avoir lu ce tutoriel, chargez quelques partitions, et lancez Iannix. Observez les comportements, fouillez dans les propriétés des objets, avec la fenêtre d'aide contextuelle à l'écran. Cela permettra de se faire une idée sur les possibles.

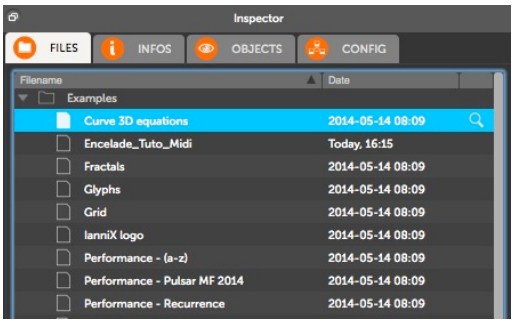


Figure 7 : les fichiers exemples

3.3 Les objets de Iannix

Une partition de Iannix n'est pas composée d'une clé, d'une armure, de barres de mesure, ni de notes de hauteur et durée différentes ; mais de **curseurs**, associés à des **courbes**, et à des **triggers** éventuels (Figure 8). Le temps défile aussi dans la partition, et les curseurs et courbes bougent selon leurs lois de comportement temporel : vitesse du curseur sur une courbe, contact curseur/trigger, évolutions de la trajectoire dans le temps ou par message externe ou interne à Iannix...

Figure 8 : les trois objets de base de Iannix

- **Curseur** : un plan dans l'espace, qui va suivre une courbe au fur et à mesure que le temps avance. Le plan suit la plupart du temps une courbe, qu'il intercepte en son centre. Le curseur peut, par exemple, envoyer des informations liées au positionnement du plan : coordonnées dans l'espace et orientation (angles).
- **Courbe** : la trajectoire qu'un curseur va suivre. Il est possible de définir une trajectoire par une relation mathématique, incluant plusieurs paramètres que l'on peut faire varier, notamment le temps. Ex : allez dans l'Inspecteur/Infos/3D Space (figure 9). Survolez ensuite une des courbes de la partition (elle devient rose), et regardez le champ « Curve equation ». On obtient par exemple « sin(2*t*PI*param2) », soit l'équation d'un sinus, dépendante du temps t, et d'un paramètre "param2" que l'on peut modifier via la barre de paramètres sous l'équation.
- **Triggers** : un point particulier de l'espace, qui va déclencher un ordre une fois qu'il est atteint par un curseur.

Figure 9 : équation d'une courbe

Nous allons d'abord explorer les objets existants dans le fichier d'exemple pour comprendre la manière de générer des ordres MIDI dans Live, puis nous créerons nos propres objets.

3.4 Générer des ordres MIDI

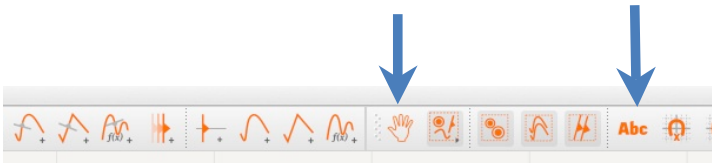
1. Partir du fichier d'exemple "Curve 3d equations". Alt + double clic gauche pour centrer la vue et la mettre à plat.

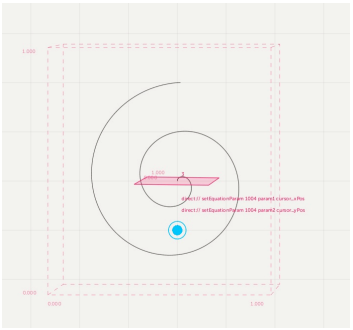
2. Effacer tous les objets, sauf ceux de la spirale située en bas, à gauche (voir figure 1). Clic gauche sur chaque objet, qui devient bleu - pour le curseur, une boite en pointillés apparaît ; puis touche du clavier "efface". OU : dans la barre de menu "Edit/Select All", puis Shift + clic gauche sur les trois objets de la spirale à conserver, puis touche du clavier "efface".

3. Se rapprocher de la spirale : cliquer sur une zone vierge de la partition, et observer la fenêtre d'aide (la rubrique « Score render » doit y apparaître). On y apprend les commandes de base pour se déplacer dans l'espace : "Alt + Double click" remet la vue au centre, zoomée, et à plat ; « Clic + mouse move » pour translater l'image selon les axes verticaux et horizontaux, « Mouse wheel » pour zoomer / dézoomer, etc. Placer la spirale au centre de la vue, en vision proche.

4. Survoler le curseur : une boite en pointillés apparaît, ainsi que les informations envoyées (deux messages "direct://...") , le tout en rose (figure 10). Si rien n'apparaît, vérifier que l'outil "Locks Object Position" n'est pas activé (figure 10).

Figure 10 : curseur verrouillé et messages associés (à droite), et outils "Locks Objet Position" et "Objects label" (en bas)





5. L'inspecteur se met à jour dynamiquement lorsque vous survolez ainsi différents objets (par exemple and Infos/General). Promenez-vous dans la partition pour observer deux ou trois objets.

6. Renommons les objets de la spirale de manière explicite, pour pouvoir les retrouver facilement. Activer la présentation des labels dans la partition avec l'outil "Object Label" (figure 10). Rien n'apparaît, car par défaut les propriétés de label ne sont pas renseignées dans ce fichier d'exemple.

7. Cliquer gauche sur la courbe pour verrouiller la vue. La courbe devient bleue, vous pouvez déplacer la souris sans perdre les informations à l'écran ; l'inspecteur ne se met maintenant plus à jour automatiquement, par survol

d'objet. Pour déverrouiller la vue, cliquer gauche une nouvelle fois sur la partition, dans une zone vierge. Verrouiller à nouveau la courbe.

8. Dans Inspecteur/Infos/General : entrer "Courbe" dans le champ "Label" et "Spirale" dans le champ "Group ID". Le label de la courbe s'affiche dans la partition. Faire de même pour le curseur (Label Curseur, Group Spirale) et le trigger (Label Trigger, Groupe Spirale). Déverrouiller ensuite la vue (figure 11).

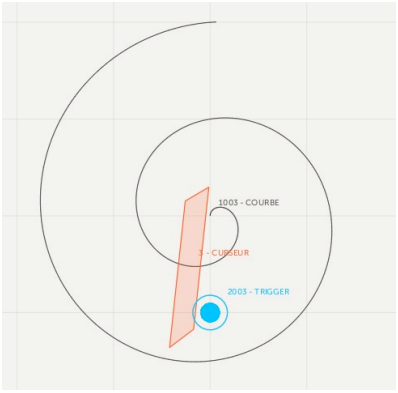


Figure 11 : les 3 objets Spirale avec leur label

9. Appuyer sur Play dans Live. Les messages MIDI apparaissent dans le sniffer et dans Live, en fonction des interactions dans Iannix. N

10. Nous avons vu en figure 10 que d'autres messages que le MIDI peuvent être envoyés. Pour avoir la liste détaillée des ordres envoyés et reçus par Innaix, aller dans Inspector/Config/Message Log. Un ensemble d'ordre défilent : direct, MIDI... Si ça n'est pas le cas, assurez-vous d'avoir bien déverrouiller la vue en cliquant une zone vierge de la partition. Nous voyons ici que le mode de verrouillage d'un objet est aussi un moyen simple de filtrer le flux d'information qui arrive dans le Message Log : en verrouillant le trigger, le flux du curseur n'apparaît plus.

11. Pour voir les messages envoyés par un objet particulier, le survoler à la souris (mode déverrouillé) . Le trigger de notre spirale envoie trois ordres : un OSC, un MIDI et un TCP.

12. Seuls les messages du curseur nous intéressent pour le moment, nous n'allons observer qu'eux. Inspector/Objects pour localiser les objets du groupe "Spirale". Appuyer sur Solo pour le curseur et le trigger. En retournant dans l'onglet Config (ou via votre sniffer MIDI), seuls les messages du trigger et curseur spirale apparaissent, à savoir, *à chaque fois que le plan curseur touche le trigger* : canal 1, un NoteOn F#1 de vélocité 70 et durée 1 seconde, puis un NoteOn F#1 de vélocité 0 (équivalent à un NoteOff). En désactivant le solo du curseur par exemple, les ordres ne sont plus envoyés car l'interaction Curseur/Trigger n'existe plus (seul le Trigger est en Solo)[12].

13. Pour modifier le comportement temporel de Iannix, il est possible de jouer sur le facteur temps dans la barre de transport : en augmentant ou en diminuant ce facteur multiplicateur, les curseurs bougeront plus ou moins vite.

14. Double clic gauche sur le Trigger affiche la fenêtre d'édition des messages (figure 12). Il est aussi possible de l'ouvrir en verrouillant la vue sur le trigger (un seul clic gauche), puis dans Inspector/Infos/Messages, cliquer sur « Edit Message ».

[12] Il est possible que cette étape de Mute/Solo ne marche pas tout à fait correctement ; il faut parfois rendre muet ou en solo plusieurs fois pour obtenir un filtre des messages corrects

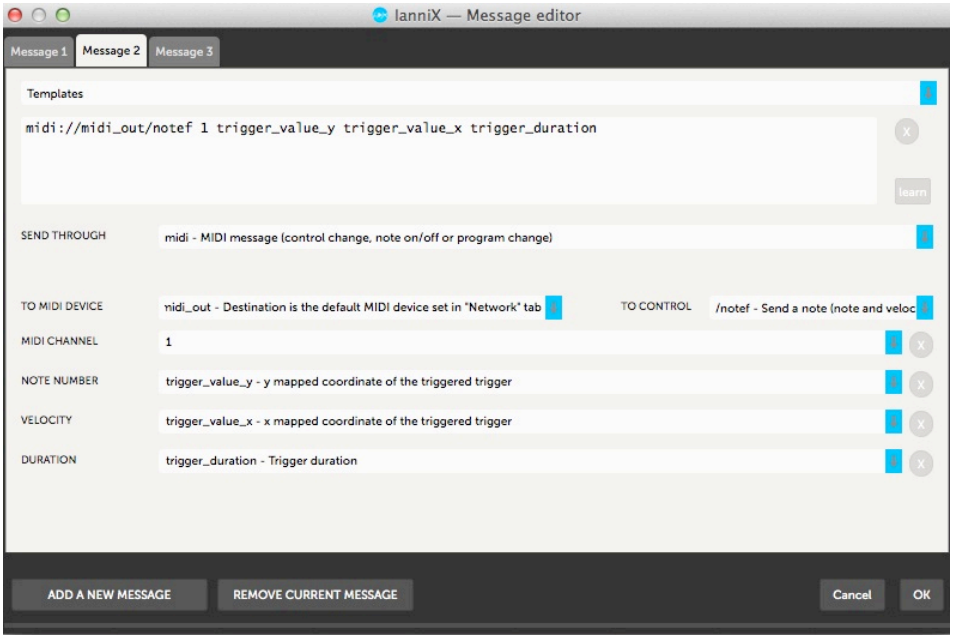


Figure 12 : fenêtre d'édition des messages

15. Trois messages sont associés au trigger de la spirale : onglets Message 1, 2 et 3. Chaque onglet est composé de la même manière : le champ Template, un résumé du message qui sera envoyé, et les paramètres constitutifs du message ("Send Through", "To host...", "value..."). Les paramètres affichés sont directement reliés à la nature du message (MIDI, OSC, TCP...).

16. Sélectionner le Message 2. Le message envoyé par le curseur est résumé en haut de la fenêtre d'édition du message :

    midi://midi_out/cc setEquationParam 1004 param1 cursor_xPos

    Il s'agit donc d'un ordre MIDI envoyé par Iannix, avec des paramètres définis selon la logique de Iannix :

    - **Send Through** : le message est actuellement de type MIDI. En cliquant sur la flèche bleue pointant en bas, à droite du champ, il est possible de définir des envois selon d'autres protocoles : interne à Iannix, OSC, serial, http...
    - **To MIDI device** : permet de spécifier le port MIDI de sortie (ici, celui sélectionné par défaut dans Inspector/Config/MIDI/DefaultMIDI_Out).
    - **To control** : spécifie le type d'ordre MIDI, Note On, par exemple.
    - Le canal MIDI, la hauteur de la note, sa vélocité, et sa durée peuvent être ensuite définies, de manière absolue (en entrant un chiffre dans le champ), ou relative (lié à d'autres évènements, externes à Iannix, ou produits par la partition), *et c'est là un des points forts de Iannix*. Par exemple, en Note Number, "trigger_value_y - y mapped coordinate of the triggered trigger" signifie que la hauteur de la note sera égale à la coordonnée Y du trigger. Dans notre exemple, ce trigger étant fixe dans la partition Iannix, la valeur

qu'il envoie est constante. La vélocité est, elle, commandée par la coordonnée X du trigger (constante également). Le trigger possédant aussi une propriété de durée (Inspector/Time), nous envoyons donc à notre DAW un Note On de hauteur fixe (valeur Y du trigger) et vélocité fixe (valeur X du trigger), de durée une seconde (envoi d'un NoteOn de vélocité 0 après une seconde).

17. Le trigger toujours verrouillé, aller dans Inspector/Infos/3D space. Les coordonnées X, Y et Z du trigger sont affichées, respectivement : -8 s, -6 s et 0s.[13]

18. Appuyer sur Play dans Live, et observer le sniffer MIDI. L'ordre MIDI envoyé à chaque passage du curseur sur le trigger est un Note On F#1 de velocité 70, suivi après une seconde d'un NoteOn F#1 de vélocité 0. Changer la coordonnée X du trigger de -8 s à -7 s : la vélocité du premier F#1 est passée à 98. En passant Y à -6.5, la hauteur de la note passe à F#0. Il y a également moins d'ordres MIDI qui sont envoyés, car le trigger est plus éloigné du centre de la spirale : il y a moins de contacts entre le trigger et le curseur durant sa course, donc moins de messages déclenchés.

19. Nous allons créer un messages MIDI CC liés à la courbe : canal 3, numéro de contrôleur CC 2, valeur de CC : la position X du curseur. Arrêter Iannix et double-cliquer gauche sur le curseur pour ouvrir la fenêtre d'édition des messages. Supprimer les messages 1 et 2 (en bas à gauche de la fenêtre), puis ajouter un nouveau message, et entrer :

    Send Through : MIDI
    To MIDI Device : midi_out - destination is the default MIDI device set in "Network" tab
    To Control : CC
    Midi Channel : 3
    Controler ID : 2
    Value : cursor_value_x - x mapped coordinate of the running cursor

    Appuyer sur Play dans Iannix, et observer le sniffer : un message Control est envoyé sur le canal 3, pour le contrôleur 2. Mais sa valeur est constante à 0.

Tout est correct, et cela nous amène à la dernière étape à comprendre : la calibration des messages MIDI.

3.5 Calibration des messages MIDI

Une propriété très importante, associée au principe de partition dans Iannix, est **la calibration des échelles**.

En fonction du protocole utilisé, seules des valeurs comprises sur une certaine échelle sont possibles. En général, les valeurs des ordres MIDI (vélocité, hauteur, valeur CC, valeur PC) sont compriss entre 0 et 127. En modifiant les échelles de sortie X, Y et Z d'un

[13] L'unité est la seconde, car l'espace de la partition est dimensionné en temps : 1 carreau fait 1 s x 1 s x 1 s de côté. L'épaisseur et la larguer du curseur, etc. sont donc elles aussi dimensionnées en temps.

objet dans l'inspecteur/Infos/Messages, il est possible de s'assurer que les données envoyées sont à l'échelle adaptée : complète MIDI de 0 à 127 (hors pitchwheel) ; ou de 60 à 80, de 96 à 96...

Des valeurs en dehors de celles admises par le protocole bloqueront à l'extrema concerné : par exemple de 0 à 200 dans Iannix donnera un signal MIDI de 0 à 127, avec la valeur 127 pour *toutes* les valeurs de Iannix supérieures à 127.

L'échelle de calibration est reliée à un référentiel : les coordonnées des objets de Iannix peuvent être lues dans le repère de base contenant tous les objets, ou dans un repère local à l'objet, par exemple.

1. Laisser Iannix tourner (Play), et le sniffer dans votre champ de vision. Le flux de CC de valeur 0 est continu.

2. Dans Inspector/Info/Messages, changer "X Output To" à 127 : les valeurs de X vont de 0 à 127 maintenant, et le sniffer montre bien la progression des ordres MIDI selon le déplacement du curseur sur la spirale.

3. Différents référentiels sont disponibles dans Iannix. Les plages de valeur sont directement impactée par le référentiel choisi. Verrouiller le curseur, puis aller dans Inspector/Infos/Messages et essayer les différents mappage, en observant le sniffer pour voir l'impact sur les ordres : "Mapped on curves bounding rectangle", "Mapped on curves bouning rectangle including cursor size", etc. Déverrouillez la vue, et bougez la courbe dans la partition tout en continuant d'observer le sniffer. Les coordonnées du curseur ont changé dans le référentiel, les ordres MIDI changent.

4. "Mapped on curves bounding rectangle" est un bon moyen de s'assurer que les ordres envoyés seront sur toute la plage définie par la courbe : dans ce cas, les valeurs de X seront bien minimales au plus à gauche de la courbe, et bien maximales au plus à droite, par exemple.

Nous avons lié Iannix et Live, et nous avons appris à créer un message MIDI sur mesure et à le calibrer dans Iannix. Il ne nous reste plus qu'à alimenter notre plugin binaural du flux MIDI créé par Iannix.

4. Iannix + Binaural

Nous allons **piloter deux paramètres du plugin binaural par Iannix**:

- **Azimuth** de la source (orientation gauche/droite/devant/derrière), qui suivra une spirale centrée à Azimuth 0
- **Hauteur** de la source (orientation haut/bas), qui suivra une courbe en cloche centrée à Hauteur 0.

4.1 le plugin binaural cSPAThrtvf

1. Après avoir correctement installé ce plugin (voir 1. Prérequis), créer un nouveau projet dans Live, avec une piste audio uniquement. Y charger le plugin cSPAThrtf[14].

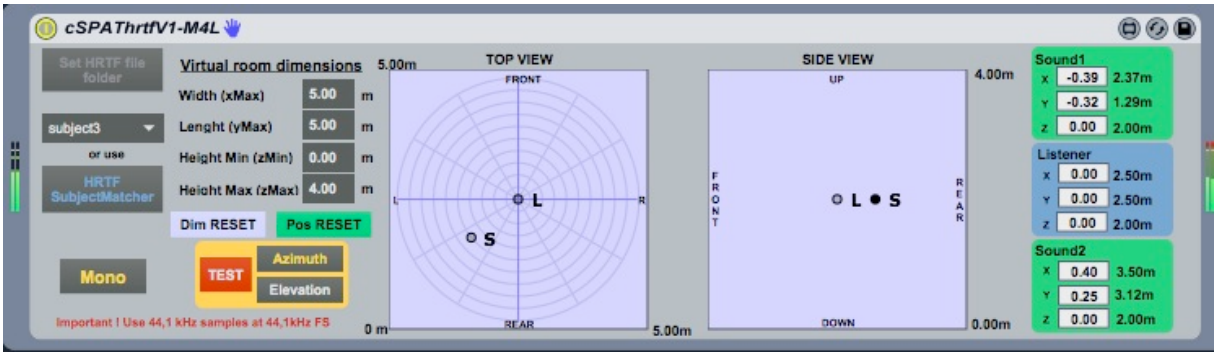


Figure 13 : plugin Max 4 Live cSPAThrtvf

2. Placer un clip de votre choix dans la piste audio. **Muter la piste**, et lancer le clip : vous n'entendez rien bien sûr. Mais en regardant la barre de plugin Live (figure 13), nous voyons que le signal audio entre bien dans le plugin (level meter de gauche), mais que rien n'y sort (level meter de droite). Il faut pour cela choisir le modèle "subject3" dans le plugin. Le level meter de droite montre maintenant du signal, saturé.

3. **Démuter la piste** : le level meter de la piste à la table mixage sature, mais le niveau sonore que vous entendez n'est pas au niveau du level meter... Il semble s'agir d'un "artefact" du plugin : au moment du chargement du modèle, un clip assez sauvage peu apparaître. S**i vous n'aviez pas démuté à l'étape 1**, vous l'auriez entendu, et votre casque et vos oreilles l'auraient potentiellement bien ressenti aussi.

4. Mettre votre casque, appuyer sur le bouton "Mono" pour n'avoir qu'une seule source de modulation du signal audio, puis les boutons "Test" et "Azimuth" : l'audio est affectée, comme si la source audio S tournait à vitesse constante autour de l'auditeur L.

[14] Aller la catégorie "Max for Live" de l'explorateur de fichiers de Live, et taper son nom dans le champ de recherche pour le localiser

4.2 Piloter le plugin par Iannix

Créons dans un premier temps deux objets dans Iannix.

1. Ouvrir le fichier "Curve_3D_Equation" ; sélectionner les objets de la courbe en cloche (en haut à gauche) et ceux de la spirale (à l'aide la touche shift), les copier, puis créer une nouvelle partition. Coller les objets dans la nouvelle partition. Si vous ne voyez rien, dézoomer.

2. Effacer les messages liés aux curseurs, et créer trois nouveaux messages CC :

Curseur	**Cloche**	**Spirale**	
Send Through	MIDI		
To Control	CC		
Midi Channel	1		
Controler ID	1	2	3
Value	cursor_value_x - x mapped coordinate of the running cursor	cursor_value_x - x mapped coordinate of the running cursor	cursor_value_y - y mapped coordinate of the running cursor

3. Pour les deux curseurs, sélectionner dans Inspector/Infos/Message le mappage "Mapped on curves bounding rectangle" et calibrer les échelles X et Y de 0 à 127.

4. Appuyer sur Play dans Iannix et vérifier dans le sniffer que les ordres envoyés sont corrects. Des ordres MIDI venant des triggers vont également apparaître. Supprimer les triggers dans Iannix.

Réalisons maintenant le MIDI Learn dans Live, pour lier les curseurs aux bons paramètres du plugin.

1. Dans Iannix, verrouiller le curseur de la courbe cloche pour n'avoir que son signal. Vérifier dans le sniffer que seul l'ordre CC 1 1 valeur apparaît.

3. Dans Live, sélectionner la piste avec le plugin binaural. Activer le mode MIDI Learn de Live, et appuyer sur le paramètre "Sound1 - z" dans le plugin, à droite : la hauteur est maintenant mappée à la coordonnée X du curseur dans Iannix. Quitter le mode MIDI Learn : vous devriez voir bouger la source de haut en bas dans le plugin.

4. Répéter l'opération pour le paramètre Azimuth. Cependant ici, le mode MIDI Learn sera perturbé, car deux ordres sont envoyés simultanément par le curseur Spirale. Nous allons dans un premier temps créer dans Iannix un objet uniquement dédié au MIDI Learn :

- dans la barre d'outil, "Adds Trigger", et pointer un endroit de la partition : un trigger apparaît. Double-clic gauche, et effacer tous ses messages. Valider par OK.

- dans la barre d'outil, "Adds circular curves with cursors", et placez le cercle de sorte à ce que son curseur entre en contact avec le trigger durant sa rotation. Double-clic gauche sur le curseur, et vider tous ses messages.

- copier le 1er message du curseur Spirale dans celui du trigger MIDI Learn que nous venons de créer.

- verrouiller le trigger MIDI Learn pour n'envoyer que son message, et appuyer sur Play. Vérifier dans le sniffer que le message envoyé est bien "CC 1 2 une_valeur" à chaque passage du curseur spirale sur le trigger MIDI Learn.

- assigner par MIDI Learn dans Live le paramètre "Sound1 - X" au message envoyé par Iannix. Couper le MIDI Learn. La source doit se déplacer uniquement dans le plan Azimuth, sur un axe horizontal.

- répéter l'opération pour le second message du curseur spirale : copié-collé du message 2 du curseur Spirale dans le message du trigger MIDI Learn (**remplacer ce message, n'en créez pas un second**) ; verrouiller le trigger MIDI Learn ; observer le sniffer, et si l'ordre est bien uniquement CC 1 3 une_valeur, assigner dans Live ce nouveau flux MIDI au paramètre "Sound1 - Y". Couper le MIDI Learn, la source doit se déplacer uniquement dans le plan Azimuth, sur un axe horizontal.

- déverrouiller la vue dans Iannix, et observez le plugin dans Live : la source se déplace sur les trois axes, selon une spirale en azimuth, et une cloche en élévation.

Il ne reste plus qu'à calibrer les ordres : nous voyons en effet dans le plugin que la spirale tourne en sens inverse de Iannix, et que dans le plugin, son centre dans le plan Azimuth n'est pas tout à fait aligné avec le centre réel du ce plan.

- rétablir le sens : dans Iannix, verrouiller le curseur Spirale puis Inspector/ Info/Message : inverser les min et max de l'échelle Y (devient de 127 à 0), Iannix et le plugin tournent dans le même sens.

- recentrer la spirale : dans Iannix, placer son centre à une des intersections du quadrillage de la partition (figure 14). Cliquer sur "Mapped on curves bounding rectangle" dans Inspector/ Infos/Messages. Nous voyons que la distance maximale entre le centre de la spirale et le point le plus haut de la courbe vaut 2 carreaux, soit 2 s ; celle vers le point le plus bas, environ 1.5 s. Idem pour les distances maximales entre le centre, et les points de la courbe les plus à sa gauche et à sa droite (respectivement 1.8 s et 1.3 s environ). Le mouvement du curseur se faisant entre ces extrêmes, le point d'arrêt central du curseur ne produira donc pas les valeurs centrales des plages X et Y. Pour que le centre de la spirale soit au centre des plages de données, il faut activer "Custom Mapping", et modifier les coordonnées "Going From...To" des deux axes X et Y de manière à ce que la spirale soit centrée dans la boite

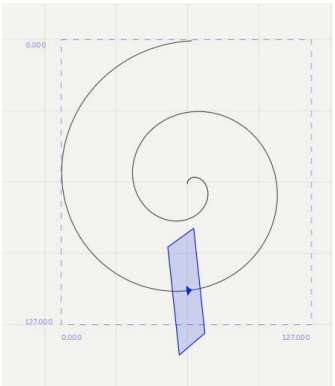


Figure 14 : spirale centrée dans ses plages de sorties

bleue (augmenter par tâtonnement la limite haute de X, et baisser un peu la limite basse de Y, tout en observant la fenêtre du plugin dans Live).

5. Perspectives

Voilà, dans l'essentiel, ce qu'il est nécessaire de savoir pour pouvoir utiliser Iannix comme contrôleur MIDI d'une DAW.

Passez un peu de temps à explorer, en survolant à l'aide de l'outil "aide" les différents champs et icônes de l'interface graphique. Iannix peut parfois être déroutant dans son interface (pas de menu contextuel par exemple, ou des curseurs qui disparaissent...), mais vous apprendrez assez rapidement ses particularités à ce niveau, et développerez un workflow qui vous sera propre et produira des résultats répétables.

Sauvegardez régulièrement vos partitions, sous d'autres noms, pour garder un historique. Enregistrer les automations ainsi produites par Iannix directement dans votre DAW, pour faciliter leurs manipulations et reproductions durant la suite de votre processus de création musicale.

Après avoir fait plusieurs essais de création d'objet, d'ordre, etc. ces opérations deviendront plus naturelles, la logique apparaîtra, et Iannix s'avèrera être un outil tout à fait particulier et performant pour qui s'intéresse à l'interaction en temps réel avec les paramètres d'un processus numérique.

N'hésitez pas à consulter tous les autres fichiers exemples, et à aller sur le site officiel de Iannix, fourni en nombreuses et impressionnantes démonstrations, ainsi qu'à creuser les forums : ce tutoriel n'est qu'un aperçu rapide des possibilités de cet environnement, qui couvrent un spectre bien plus large.

En utilisant les différents modes de déplacement d'un curseur le long d'une courbe (Inspector/Infos/Time, champs "Cursor Loop Pattern" et "Cursor Easing Curve"), les équations 3D, en creusant les possibilités de programmation interne (messages internes à Iannix), en utilisant du MIDI In pour piloter Iannix... vous allez pouvoir bénéficier d'un outil extrêmement puissant pour séquencer des évènements dans le temps, selon des courbes, des curseurs et des triggers dynamiques, et via une programmation graphique créant de véritables partitions d'évènements. Enfin, la versatilité des protocoles avec lesquels Iannix est compatible le rendent à coup sûr performant dans d'autres contextes que la musique.