# Pd-Multitouch : Interagir avec des patches pure data en mode multitouch

Jean-Yves GRATIUS, jyg@gumo.fr

https://github.com/jyg/multitouch

https://github.com/jyg/mob

### Résumé

Cet article présente les projets **pd-multitouch** et **pd-mob**, ensemble d'abstractions permettant, grâce à des bibliothèques externes courantes, d'utiliser Pure Data en mode multitouch, sur plateforme fixe ou mobile.

### Mots clés

Pure Data. Multitouch. Hid. Linux. MobMuPlat.

### Introduction

Pure Data est un logiciel de programmation graphique pour la création musicale et multimédia en temps réel de type patcher<sup>1</sup>

Sa communication avec l'environnement extérieur peut s'effectuer au travers de différentes interfaces, audionumériques, midi, réseau, vidéo, clavier / souris, tablettes graphiques, manettes de jeu, ou tout autre périphérique d'entrées / sorties². Sur les appareils mobiles, il est également possible d'utiliser les capteurs physiques intégrés (accéléromètre, gyroscope, GPS, boussole...).

Ce sont d'ailleurs ces derniers dispositifs qui ont popularisé depuis 2010 l'utilisation des écrans tactiles multitouch, jusqu'alors limités aux installations muséographiques.

L'interêt et la pertinence des interfaces multitouch pour les applications d'informatique musicale est illustré notamment par le nombre d'applications musicales mobiles qui ont depuis vu le jour, tirant pleinement parti des possibilités qu'offrent les écrans tactiles des téléphones, tablettes et autres périphériques mobiles.

#### Qu'en est-il avec Pure Data, en 2020 ?

Dans un premier temps, nous passerons en revue les solutions existantes qui permettent d'interagir plus ou moins directement avec Pure Data au moyen d'une interface multitouch, sur plateforme fixe ou mobile. Nous évoquerons également les solutions intégrés qu'utilisent des logiciels musicaux tiers. Ensuite, nous présenterons le projet pd-multitouch en expliquant le principe de son fonctionnement, ainsi que les prérequis matériels et logiciels permettant de l'utiliser. Complémentaire, le projet propose des modules d'interface pd-mob, qui élaborés utilisables dans les patches et compatibles multitouch, sera également mentionné, notamment pour sa fonction d'export simplifié des patches vers la plateforme mobile MobMuPlat. Enfin, nous discuterons des perspectives futures pour ces projets. comme l'élargissement à d'autres O.S. que celui actuellement utilisé (Linux).

# 1.1 Multitouch & PureData via interfaces externes

## a) Contrôleur multitouch dédié

Historiquement, avant l'apparition de solutions hardware homogènes, l'interaction avec Pure Data en mode multitouch passait par des interfaces externes spécifiques, communiquant avec le patch via un protocole réseau, par exemple OSC.



*Fig.* 1 - multitouch via OSC

C'est par exemple le cas du Lémur, interface matérielle spécifiquement dédiée aux applications musicales et commercialisée en 2004 par la société JazzMutant, sous forme d'un contrôleur modulaire à écran tactile multitouch, qui ensuite s'est décliné sous forme logicielle (application pour iOS ou Androïd).

Un autre protocole, Tuio, a été introduit lors de la mise au point du synthétiseur RéacTable<sup>3</sup> et du module RéacTiVision<sup>4</sup>. C'est une couche rajoutée au dessus du protocole OSC.

Puckette M."The Patcher", Proceedings of the International Computer Music Conference1988, p. 420-429, Cologne, Allemagne, 1988.

<sup>2</sup> Par exemple, voir la <u>malinette.info</u>

<sup>3 &</sup>lt;a href="http://reactable.com/">http://reactable.com/</a>

<sup>4 &</sup>lt;a href="http://reactivision.sourceforge.net/">http://reactivision.sourceforge.net/</a>

Ce protocole a permis la mise au point de solutions multitouch utilisant des capteurs grand public (Wiimote, webcams infrarouge, Kinect, etc.) associés à Pure Data au moyen d'un programme serveur générant des événements Tuio<sup>5</sup>.

## b) Interface sur appareil mobile

Avec l'essor des smartphones et tablettes à partir des années 2010, un certain nombre d'applications mobiles sont apparues, exploitant les fonctionnalités multitouch de ces appareils pour proposer des interfaces tactiles permettant de communiquer avec un programme cible sur un autre ordinateur, que ce soit Pure Data ou tout autre logiciel de production musicale / de régie spectacle. Citons TouchOSC<sup>6</sup>.

### c) Interface depuis navigateur web

Les écrans multitouch sont ensuite apparus sur certains ordinateurs portables, et des O.S. comme Windows, Linux ou Chrome ont commencé à prendre en charge des applications tirant parti de ces fonctions (notamment certains navigateurs web).

Depuis 2014, le projet Open-Stage-Control, développé par Jean-Emmanuel Doucet, permet de charger une interface GUI depuis un navigateur web connecté à un serveur qui, à son tour, communique avec le programme client via OSC. Le multitouch est pris en charge sur les plateformes mobiles, ainsi que sur les OS Windows et Linux, pour peu que le navigateur utilisé prenne en charge le multitouch.

Cette solution, comme les précédentes, est basée sur une architecture client-serveur. Dans le cas d'une utilisation avec Pure Data, c'est encore à partir d'une interface spécifique, mais cette fois-ci depuis le navigateur web, que l'on communiquera avec le patch. L'interface est modulaire et éditable, la communication OSC est bidirectionnelle, et il est possible de synchroniser plusieurs instances sur le même réseau, ce qui est intéressant dans le cadre d'une configuration avec plusieurs interfaces / utilisateurs. Le projet est très riche et le développeur très (ré)actif.

# 1.2 Multitouch & PureData mobile

Des déclinaisons de Pure Data pour appareils mobiles existent depuis plusieurs années, permettant naturellement d'accéder aux fonctionnalités multitouch de ces appareils, mais les patches qui leurs sont destinés doivent être préalablement écrits et mis au point sur une plateforme fixe, avec la version standard de Pure Data, plus éventuellement des outils / externals pour adapter les patches en vue de leur utilisation sur la plateforme cible. Parmi ces déclinaisons, citons notamment MobMuPlat, PdParty, PdDroidParty, Pof, Ofelia, ainsi que les diverses applications mobiles dérivées de la bibliothèque libpd<sup>7</sup>.

Certaines de ces applications sont capables de restituer directement l'interface telle qu'elle apparaît dans le patch (avec les objets *slider*, sélecteurs, *number*, *toggle*, etc.. disponibles dans Pure Data).

D'autres nécessitent de redéfinir entièrement l'interface, soit avec un éditeur externe (ex: MobMuPlatEditor), soit à l'intérieur d'une fenêtre de rendu graphique et d'une bibliothèque spécifique (Pof, Gem, Ofelia). Dans ce cas, même si la bibliothèque prend en charge la gestion d'événements multitouch, même si l'ordinateur utilisé est doté un écran multitouch et même si l'OS gère les événements multitouch (cas de certaines distributions Linux ou Windows), ce d'interaction n'est pas forcément fonctionnel<sup>8</sup>.

# 1.3 Multitouch & autres logiciels musicaux

Il existe des logiciels orientés création multitmédia qui gèrent le multitouch en natif. Citons notamment, sans exhaustivité, Usine Hollyhock<sup>9</sup>, Bitwig Studio<sup>10</sup>, Sonar<sup>11</sup>, Kivy<sup>12</sup> et Processing<sup>13</sup> (ces deux derniers étant davantage à considérer comme des environnements de programmation multimédia).

Certains plugins audio sont également multitouch, mais nous ne sommes pas en mesure de le vérifier<sup>14</sup>.

Voir le site <u>tuio.org</u> qui présente différents ponts logiciels permettant d'interconnecter hardware et logiciels clients.

<sup>6 &</sup>lt;a href="https://hexler.net/products/touchosc">https://hexler.net/products/touchosc</a>

<sup>7</sup> Ces dernières sont souvent constituées d'une interface graphique et d'un patch pure data non-modifiables.

<sup>8</sup> En tout cas à la date d'aujourd'hui (mars 2020)

<sup>9</sup> www.brainmodular.com

<sup>10</sup> www.bitwig.com

<sup>11</sup> www.cakewalk.com

<sup>12</sup> kivy.org Kivy est une bibliothèque libre et open source pour Python, utile pour créer des applications tactiles multitouch pourvues d'une interface utilisateur naturelle (wikipedia).

<sup>13 &</sup>lt;a href="https://processing.org/">https://processing.org/</a>

<sup>14</sup> Nous nous référons à certains fils de discussion sur le net, notamment celui-ci : <a href="https://www.kvraudio.com/forum/viewtopic.php?te-454288">https://www.kvraudio.com/forum/viewtopic.php?te-454288</a>

# 2.1 Pd-multitouch : gestion directe du multitouch dans Pd

La solution proposée ici consiste à rendre *multitouch - compatible* un patch Pure Data existant, simplement en y insérant l'objet [multitouch] du projet pd-multitouch. Cette abstraction écoute les événements touch générés par l'interface de l'écran tactile et les convertit en messages qui, à l'intérieur du patch Pure-Data, interagissent avec les éléments d'interface graphique (GUI) que ce dernier contient.



## a) Principe de fonctionnement

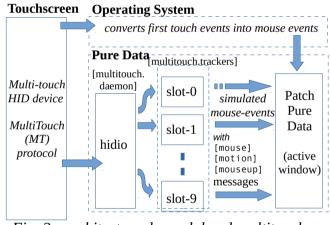


Fig. 2 - architecture du module pd-multitouch

Il n'était pas envisageable de modifier le code de l'interface graphique tcl/tk pour ajouter la prise en charge des événements multitouch. Nous avons donc opté pour l'interception directe des événements multitouch générés par le périphérique tactile, depuis Pure Data, au moyen de l'external hidio<sup>15</sup>. Ces événements sont ensuite convertis par l'abstraction [multitouch] en séries d'événements « souris » virtuels. Pour chaque contact tactile établi à l'écran, – affecté à un « slot » spécifique au moment de son apparition –, une salve de trois messages est envoyée à la fenêtre du patch actif : mouse (click de souris), motion (message de déplacement de souris avec un

offset correspondant au déplacement du doigt), puis mouseup (relâché de souris).

Ce processus est répété pour chaque itération, c'est à dire à chaque fois que le périphérique renvoie une nouvelle trame d'événements multitouch pour l'ensemble des slots<sup>16</sup>.

Le traitement des événements liés au slot-0 est un peu différent, car comme l'O.S. est susceptible de générer automatiquement des événements souris pour le premier contact avec l'écran tactile, il faut veiller à ne pas produire de tels événements en double.

# b) Configuration utilisée

• O.S.: Linux

• PureData > 0.50

 externals : hidio, iemguts, hcs/screensize (ce dernier étant facultatif)

Le système sur lequel nous avons développé la bibliothèque pd-multitouch était un ordinateur portable Lenovo Thinkpad T440s avec écran multitouch et système d'exploitation Linux (18.04.4 LTS).

L'adaptation vers un système Windows nécessiterait une version pour Windows de l'external hidio. Jusqu'à présent, nos tentatives d'interception des événements générés par le périphérique multitouch avant qu'ils soient capturés par la fenêtre active ont échoué.

Il est important de désactiver la prise en charge par l'O.S. des gestes multitouch (*pinch*, *3-finger tap*, etc..) de manière à ce qu'il n'envoie pas des messages inadéquats (*scroll*, *zoom*...) au patch Pure Data. En revanche, l'émulation de la souris en mode monotouch est nécessaire pour la calibration et le bon fonctionnement de [multitouch]<sup>17</sup>.

# c) Installation

- Vérifier la présence des externals requis : iemguts, hidio (obtenu ici : <a href="https://github.com/jyg/hidio">https://github.com/jyg/hidio</a>).
- Vérifier le bon fonctionnement de hidio en ouvrant le patch hidio-help<sup>18</sup>.
- Ouvrir le patch *multitouch-help.pd*.
- 16 Selon le protocole MT (<a href="https://www.kernel.org/doc/Documentation/input/multi-touch-protocol.txt">https://www.kernel.org/doc/Documentation/input/multi-touch-protocol.txt</a>), chaque numéro de slot est affecté à un seul et unique contact tactile, de l'apparition jusqu'à la disparition de celui-ci.
- 17 [multitouch] compare les messages de hidio et les événements souris reçus par le patch, pour calculer les offsets en x et y, qui dépendent de la position de la fenêtre du patch sur l'écran.

<sup>15</sup> Il s'agit d'une version corrigée de hidio disponible à l'adresse suivante <a href="https://github.com/jyg/hidio">https://github.com/jyg/hidio</a>.

- Dans la fenêtre *multitouch.settings.pd* qui apparaît (fig. 3), suivre les instructions de la configuration en trois étapes. Il s'agit principalement d'identifier le périphérique touchscreen dans la liste des périphérques hid du système, puis de vérifier les résolutions d'écran, et enfin d'ajuster, le cas échéant, les réglages d'orientation de l'affichage (normal / inversé).

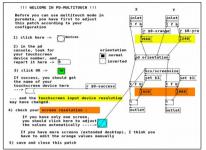


Fig. 3 – fenêtre de configuration multitouch.settings

## d) Limitations du projet

Les objets GUI natifs de Pure Data tels que les sliders ou les sélecteurs fonctionnent correctement en mode multitouch. C'est moins le cas pour les objets bang, number, number2 et surtout pour toggle et les boites de messages. En effet, les salves d'événements [mouse / mouseup] générées à chaque déplacement même infime des points de des conduisent à re-déclenchements intempestifs des boîtes de messages et des toggle. Aux toggle, on préfèrera des sélecteurs à 2 positions. résistent obiets bana mieux déclenchements intempestifs, du fait de l'existence d'un temps d'interruption pendant lequel ils ne sont pas ré-enclenchables.

En mode monotouch ou avec la souris, on peut continuer, après avoir cliqué sur un *slider*, à le contrôler même si le pointeur se déplace en dehors de la zône d'interaction de celui-ci. En mode multitouch, les *sliders* sont actionnés tant que le contact à l'écran reste à l'intérieur de leur zône d'interaction, mais dès que le contact quitte cette zône, l'association avec le *slider* est perdue. Il n'y a pas de capture exclusive d'un contact par un *slider* spécifique. Cela vient du fait que des salves d'événements souris (mouse / mouseup) sont générées pour tous les contacts en cours, et donc que le test d'inclusion est reconduit à chaque itération pour tous les objets GUI natifs de Pure Data<sup>19</sup>.

C'est également pour cette raison que les objets *number* et *number2* ne sont pas utilisables en mode multitouch, car leur surface d'interaction est trop petite. La modification des graphiques (*arrays*) est également possible en mode multitouch, mais nécessite là encore un certain doigté, pour ne pas perdre certains points du tableau en route...

Une autre limitation actuelle du projet est qu'il n'est pas possible d'utiliser plusieurs patchs simultanément en mode multitouch. Seule la fenêtre de patch qui a le focus est active.

# 2.2 Le projet pd-mob

Complémentaire mais indépendante de [multitouch], la collection d'abstractions [mob] a deux finalités :

## a) fournir des objets GUI élaborés

... sous forme d'abstractions, qui permettent de nouveaux types d'interactions dans un patch : boutons rotatifs (*knob*), molettes rotatives (*jog*), menu popup, glissières horizontale et verticale (*hfader / vfader*), pad xy (*xy\_slider*), pad xy multitouch (*xy\_multi*), zone d'affichage graphique (*lcd*). En mode multitouch, ces objets, pour la plupart d'entre eux, sont à même de se « verrouiller » sur un slot pour en capturer tous les messages, ce qui veut dire que l'association avec un contact tactile perdure même si le contact se déplace en dehors de la zone d'interaction de l'objet.



Fig. 4 – quelques objets GUI élaborés de la bibliothèque [mob]

# b) faciliter l'export versl'application mobile MobMuPlat

... des patchs PureData, tout en conservant la disposition des éléments d'interface tels qu'ils y sont définis (fig. 5). Il n'est plus nécessaire d'utiliser le programme tiers MobMuPlatEditor pour redessiner l'interface à partir de zéro.

Cette collection d'abstractions nécessite pour fonctionner quelques externals : mrpeach/binfile, ainsi que les bibliothèques iemguts et iemlib.

<sup>18</sup> Si cela ne fonctionne pas, il est peut-être nécessaire de recompiler l'external. Ouvrez un terminal dans le dossier hidio, puis tapez ./configure et ensuite make.

<sup>19</sup> Les objets GUI du projet pd-mob fonctionnent différemment et permettent la capture exclusive des événements multitouch.

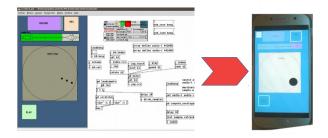


Fig. 5 – Export rapide vers MobMuPlat

# 2.3 Perspectives - reste à faire

Concernant le projet pd-multitouch :

- Dresser la liste des modeles d ecrans multitouch compatibles avec le projet pdmultitouch;
- Envisager une version pour Windows et notamment les tablettes MS-surface pro.

#### Concernant le projet pd-mob :

- continuer le développement d'autres objets GUI et leur export vers MobMuPlat ;
- Étendre la fonction d'export d'interface vers d'autres systèmes que MobMuPlat : Pof, Open Stage Control.

## Liens

#### **Documentation sur le protocol Multitouch**

https://forums.opensuse.org/showthread.php/506695-Multitouch-on-ELAN-touchscreen

https://www.kernel.org/doc/Documentation/input/multi-touch-protocol.txt

#### Démonstration vidéo

https://vimeo.com/292789