

INSA Lyon - Département Informatique

Rapport

de

PROJET DE FIN D'ÉTUDES

**Développement d'interface graphique technologie web -
faisabilité et évaluation**

***Development of web-based graphical user interface -
feasibility and evaluation***

Colin THOMAS

Soutenance le 26 juin 2020

Projet réalisé du **5 février 2024** au **26 juillet 2024**

dans la structure d'accueil

Arturia (Grenoble)

Référent	: Yann GRIPAY, Maître de Conférences	INSA Lyon
Tuteur	: Timothée BÉHÉTY, Référent Technique Logiciels Compagnon	Arturia (Grenoble)

Contents

1	Introduction.	1
1.1	Contexte.	1
1.2	Définitions.	2
1.3	Définition du problème.	4
1.4	Aperçu des contributions	5
1.5	Plan du rapport	5
2	Web MIDI API	5
3	description gemaps	5
	References bibliographiques	8

Annexes

A	Données collectées.	9
B	Éléments majeurs de la conception.	9
C	Preuves de théorèmes.	10

1 Introduction.

1.1 Contexte.

Arturia est un fabricant d'instruments de musique numériques et électroniques. La société est basée à Grenoble et développe des reproductions logicielles d'anciens synthétiseurs (analogiques et numériques), des claviers MIDI, des contrôleurs, des synthétiseurs analogiques et des interfaces audio.

Il s'agit d'une entreprise employant 180 personnes, qui a été fondée en 1999. A sa création, ses deux fondateurs Frédéric Brun (actuel PDG d'Arturia) et Gilles Pommereuil ont commencé par créer des reproductions de synthétiseurs analogiques réputés au format virtuel, en commençant par développer l'Analog V, inspiré par les synthétiseurs Moog et en collaboration avec leur créateur. De nombreuses autres reproductions ont suivi, et sont devenues par la suite la V Collection, qui est la collection d'instruments virtuels d'Arturia, et la FX Collection, qui est son équivalent pour les effets audio.

Arturia a depuis élargi son catalogue à des instruments virtuels qui lui sont propres, plusieurs synthétiseurs analogiques renommés, différents contrôleurs MIDI ainsi que des cartes sons.

L'équipe Compagnon dans laquelle je réalise mon stage s'occupe des logiciels accompagnants les produits Arturia, comme le gestionnaire de licences, le logiciel de configuration de matériel, ou encore des logiciels spécifiques à chaque produit, comme par exemple l'application mobile qui accompagne le clavier AstroLab.

Arturia Software Center L'Arturia Software Center ¹ (ASC) est le logiciel de gestion de licences et de téléchargement de software d'Arturia : il est nécessaire pour tout utilisateur souhaitant utiliser un logiciel Arturia. L'utilisateur peut naviguer parmi les logiciels disponibles sur l'Arturia Software Center sur la page "Explore Products", activer la licence du logiciel sur la page "My Products", le télécharger, le mettre à jour, gérer l'emplacement de ses fichiers dans les paramètres, ainsi que de rentrer le code d'un produit Arturia physique afin de bénéficier des logiciels qui l'accompagnent.

L'Arturia Software Center est composé de deux parties :

Tout d'abord, une partie principale, qui est ce logiciel qui permet de gérer ses licences et ses téléchargements de logiciels Arturia, et fournit une interface graphique.

Ensuite, un logiciel nommé ASC Agent, qui fonctionne en arrière plan, et vérifie les licences utilisateur à chaque démarrage de produit Arturia, afin d'éviter toute fraude et téléchargement illégal.

Midi Control Center Le Midi Control Center ² (MCC) est le logiciel de configuration de matériel Arturia : il permet de gérer les configurations et les paramètres des synthétiseurs et

¹<https://www.arturia.com/support/asc-arturiasoftwarecenter>

²<https://support.arturia.com/hc/fr-fr/sections/4405740766098-MIDI-Control-Center>

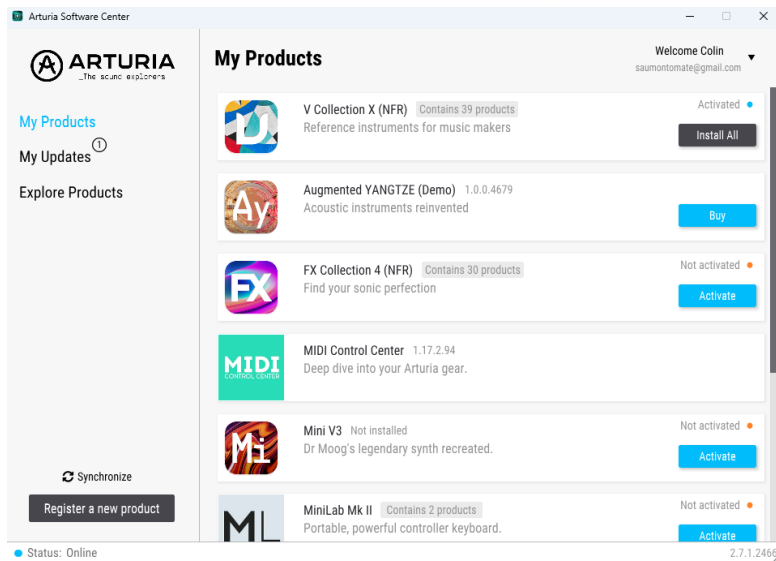


Figure 1: Arturia Software Center (ASC) existant

contrôleurs MIDI, ainsi que de faire les mises à jour firmware pour les synthétiseurs analogiques, cartes sons et claviers midis physiques d'Arturia. Il communique en norme MIDI avec les produits Arturia pour leur configuration.

Il est composé d'une page "Device Settings" permettant de modifier les paramètres généraux du contrôleur actuellement connecté (par exemple, son temps de mise en veille ou ses courbes de vélocité), une page "Controller Map" permettant de modifier les paramètres de chaque contrôle du contrôleur, (par exemple, modifier la note attribuée à une touche en particulier), et une page permettant d'importer ou d'exporter des templates de paramètres. Il est également possible d'effectuer une détection automatique de mises à jour disponibles pour le firmware du contrôleur.

1.2 Définitions.

Nous allons ici définir quelques termes propres au matériel de production musicale.

Synthétiseur analogique ³ : Un synthétiseur analogique est un synthétiseur qui utilise des circuits analogiques et des signaux électriques analogiques pour générer des sons par des techniques électroniques. Contrairement à un synthétiseur numérique, qui stocke et communique l'information en binaire, l'information sonore est transmise dans l'analogique en tant que variation de tension.

³https://fr.wikipedia.org/wiki/Synth%C3%A9tiseur_analogique



Figure 2: Midi Control Center (MCC) existant

MIDI⁴ : Le Musical Instrument Digital Interface ou MIDI est un protocole de communication et un format de fichier dédiés à la musique, et utilisés pour la communication entre instruments électroniques, contrôleurs, séquenceurs, et logiciels de musique. Il s'agit du protocole standard du matériel électronique de musique. Il permet par exemple à un Contrôleur MIDI d'envoyer le signal qu'une certaine note a été pressée avec une certaine vélocité, permettant à un logiciel sur ordinateur de produire un son en conséquence. Dans le sens inverse, ce protocole permet à l'utilisateur de paramétrer son contrôleur MIDI en envoyant des signaux SYSEX, comportant les informations nécessaire sur le paramètre à modifier et la valeur à lui attribuer.

Contrôleur MIDI⁵ : Un contrôleur MIDI est un appareil utilisé pour envoyer des signaux MIDI à l'ordinateur où à un autre appareil (par exemple, un synthétiseur analogique). Il peut se présenter sous la forme d'un clavier de piano, de faders, de potentiomètres, d'encodeurs, ou bien d'autres.

Synthétiseur virtuel : Un synthétiseur analogique virtuel est un synthétiseur simulant le son de synthétiseurs analogiques par le biais de processeurs numérique. Il peut être utilisé sous la forme de Stand-Alone, c'est à dire sans logiciel qui l'entoure, en le contrôlant par exemple par un contrôleur MIDI, mais également et plus populairement sous la forme Plug-In, c'est à dire en tant que module à intégrer à un logiciel de Musique Assistée par Ordinateur.

Message Sysex⁶ : Un message SysEx est un type de message MIDI permettant de régler les paramètres d'un périphérique MIDI, ce qui n'est pas adapté à la syntaxe normale du protocole

⁴https://fr.wikipedia.org/wiki/Musical_Instrument_Digital_Interface

⁵<https://troidecis.fr/musique/instrument-midi-definition>

⁶https://steinberg.help/cubase_artist/v10.5/fr/cubase_nuendo/topics/midi_editors/midi_editors_sysex_messages_c.html

MIDI.

1.3 Définition du problème.

L'Arturia Software Center (ASC) et le Midi Control Center (MCC) sont deux logiciels vieillissants, qu'Arturia souhaite mettre à jour. Pour faire cela, l'entreprise souhaite mettre à l'épreuve de nouvelles technologies de développement graphique.

En effet, ces logiciels ont été développés respectivement en 2014 pour l'Arturia Software Center, et en 2013 pour le Midi Control Center. A ce moment, l'équipe Compagnon que j'intègre pour ce stage n'existait pas encore, et ces logiciels ont donc été développés par l'équipe Software, qui se spécialise dans le développement d'instruments virtuels. Ils ont opté pour l'utilisation de technologies de développement d'instruments virtuels, notamment le framework C++ JUCE, pour la création de ces outils. Cependant, une mise à jour de l'ASC et du MCC en utilisant des technologies différentes permettrait de se détacher des outils spécifiquement conçus pour les logiciels audio, potentiellement inadaptés à ces applications.

Ainsi, l'équipe Compagnon a décidé de prototyper une application modulaire qui rassemblerait l'Arturia Software Center et le Midi Control Center. Cette application serait capable de lancer des bibliothèques C++ en tant que back-end, de servir en HTTP des pages Web statiques en tant que front-end, et d'afficher un host permettant de naviguer sur ces pages Web. De cette manière, l'Arturia Software Center et le Midi Control Center constitueraient = deux modules de cette application, chacun dotés d'un back-end sous forme de bibliothèque C++, et d'un front end sous forme de page web statique, communiquant avec leur back-end par requêtes HTTP.

Utiliser des interfaces web pour ces logiciels se révèle ainsi positif sur plusieurs points :

- Les technologies web sont plus efficaces pour créer rapidement des interfaces. En effet, les bibliothèques et outils pré-développés et libres d'accès sont nombreux, en constante évolution, et facile d'utilisation : on peut prendre pour exemple la fonctionnalité multilingue, qui est une fonctionnalité qui prend de l'importance pour Arturia, et qui demande beaucoup de ressources, pour un développement plus facile, classique et très documenté en technologies Web. De même, les framework Web offrent souvent une courbe d'apprentissage plus faible que le C++ ou autre langage qui n'est pas haut-niveau.
- Ceci permet, pour développer ces interfaces, de pouvoir rechercher des profils de développeurs front-end Web, ce qui est plus facile que de devoir rechercher des développeurs C++, puis de les former sur le framework JUCE, pour ensuite travailler sur des interfaces. En séparant le développement front-end du reste de l'application, ceci permet de rechercher des développeurs spécialisés dans leur domaines, et ainsi une meilleure évolutivité de l'équipe.
- Ensuite, cela permet de séparer le front-end du back-end. En effet, avoir une interface HTTP entre un back-end et un front-end permet de bien séparer ces deux parties. Ceci amène de la modularité et de la scalabilité, car chaque partie peut être modifiée sans

affecter l'autre. On peut imaginer également une plus grande flexibilité technologique, car l'application Web pourrait être développée dans n'importe quel framework, sans impliquer de changement pour les librairies C++. Enfin, en isolant les accès directs au back-end, on peut ainsi mettre en place de meilleures mesures de sécurité pour protéger les logiciels et les licences utilisateurs.

L'Arturia Software Center étant l'outil qui permet de gérer l'installation des logiciels, un back-end est forcément nécessaire pour soutenir le front-end. En effet, par design et pour des raisons de sécurité, les technologies Web n'ont pas accès au système de fichiers de l'utilisateur.

En revanche, la question se posera pour le Midi Control Center, dont la principale fonctionnalité est de communiquer par protocole Midi avec le contrôleur. Nous étudierons pendant ce stage la faisabilité d'un Midi Control Center entièrement réalisé en technologies Web.

1.4 Aperçu des contributions

1.5 Plan du rapport

2 Web MIDI API

L'API Web MIDI ⁷ fournit une manière pour les applications web d'interagir avec les périphériques MIDI connectés à l'ordinateur. Elle permet d'envoyer et de recevoir des messages MIDI, par exemple recevoir des messages de Note On / Note Off, mais aussi envoyer des messages Sysex permettant de modifier les paramètres du périphérique.

Cette API n'est pour l'instant disponible que sur les navigateurs Safari et Chromium. Ceci ne nous posera pas problème, car le fait que nous implémentons nous même le navigateur nous permet de le choisir. En l'occurrence, celui que nous utilisons est celui de Juce, et il est basé sur Chromium pour Windows, et sur Safari pour Mac.

3 description gemaps

Unified Control Center : L'Unified Control Center est l'application hôte qui affiche les pages web des modules. Nous utilisons pour l'instant le navigateur intégré au framework C++ Juce, qui intègre Internet Explorer (lui-même basé sur Chromium) sous Windows et WebKit sous Mac. Nous envisageons, pour le futur, d'utiliser Chromium Embedded Software (CEF), afin d'unifier les navigateurs sous tous les systèmes d'exploitation, et également afin d'avoir un contrôle plus grand sur le navigateur.

GeMAPS : Generic Modular Architecture with Pluggable Services (GeMAPS) est l'architecture du programme qui lance les librairies C++ des modules, et distribue leurs pages Web.

⁷https://developer.mozilla.org/en-US/docs/Web/API/Web_MIDI_API

Module : Un module est un dossier, composé d'un fichier de configuration décrivant le module, un dossier "lib" où l'on peut trouver les bibliothèques C++ qui seront lancées par Gemaps (.dll dans le cas de Windows, et .dylib sur Mac), et un dossier "web" où l'on peut trouver les pages web statiques qui seront distribuées par Gemaps. Un module peut avoir aucune, une ou plusieurs bibliothèques C++, et elles seront toutes lancées au lancement de Gemaps. Un module peut également avoir aucune une ou plusieurs page Web, dont les URL à servir sont décrites dans le fichier de configuration du module.

Lorsque l'on veut écrire des équations

$$f(x) = 0 \iff x = 1 \tag{1}$$

Quand on veut référencer l'équation 1.

Exemple définition

Définition 1 (Proposition (mathématiques)⁸). Ceci est une définition

Exemple de proposition :

Proposition 1 (La Terre sphérique des Anciens⁹). *La Terre est supposée plate, de la forme d'un disque, entièrement ceinturée par le fleuve Océan et recouverte d'un ciel en coupole hémisphérique.*

Théorème 1 (Théorème de Pythagore). *On nomme a , b et c les longueurs des trois côtés d'un triangle.*

Les triangles pour lesquels on a la relation $a^2 + b^2 = c^2$ sont tous les triangles rectangles dont l'hypoténuse est le côté de longueur c , et seulement eux.

Source : <http://mathematiques3.free.fr/2troisieme/problemes/prob014.php>, dernière visite 31/10/2019.

On peut aussi écrire des corollaires

Corollaire 1. *Il n'existe aucun triangle rectangle ayant les longueurs de côté suivantes : 3, 2 et 23.*

Reste à le prouver.

Proof. Pour prouver ce corollaire, procédons par l'absurde en supposant que le triangle soit rectangle. Il y a alors trois possibilités pour la longueur de l'hypoténuse c .

- $c = 3$
 or $2^2 + 23^2 \neq 3^2$
 donc si le triangle est rectangle, sa longueur ne peut être 3.

⁸<https://www.techno-science.net/definition/6406.html>, dernière visite le 31/10/2019.

⁹<https://planet-terre.ens-lyon.fr/article/histoire-forme-Terre.xml>, dernière visite le 31/10/2019

- $c = 2$
or $3^2 + 23^2 \neq 2^2$
donc si le triangle est rectangle, sa longueur ne peut être 2.
- $c = 23$
or $2^2 + 3^2 \neq 23^2$
donc si le triangle est rectangle, sa longueur ne peut être 23.

Donc aucun côté ne satisfait la définition d'une hypoténuse. Le triangle ne peut donc pas être un triangle rectangle. \square

Il est aussi possible d'écrire des lemmes.

Lemme 1 (Lemme d'Euclide¹⁰). *Si un nombre premier p divise le produit de deux nombres entiers b et c , alors p divise b ou c .*

Listing 1: Programme inconnu

```
import numpy as np

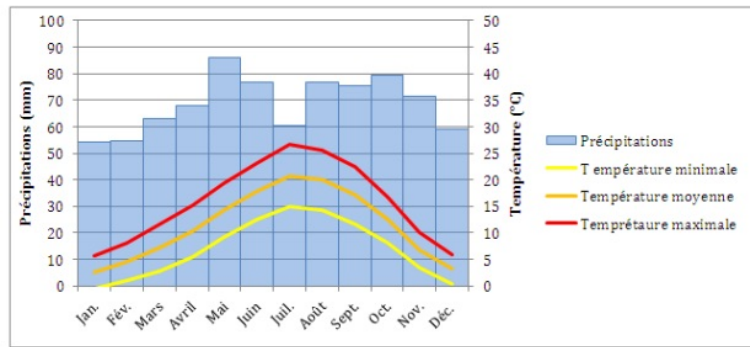
def incmatrix (genl1 , genl2 ):
    m = len (genl1)
    n = len (genl2)
    M = None #to become the incidence matrix
    VT = np.zeros ((n*m,1) , int) #dummy variable

    #compute the bitwise xor matrix
    M1 = bitxormatrix (genl1)
    M2 = np.triu (bitxormatrix (genl2) , 1)

    for i in range (m-1):
        for j in range (i+1, m):
            [r , c] = np.where (M2 == M1[i , j])
            for k in range (len (r)):
                VT[(i)*n + r[k]] = 1;
                VT[(i)*n + c[k]] = 1;
                VT[(j)*n + r[k]] = 1;
                VT[(j)*n + c[k]] = 1;

            if M is None:
                M = np.copy (VT)
```

¹⁰https://fr.wikipedia.org/wiki/Lemme_d'Euclide, dernière visite le 30/10/2019



Graphique obtenu sur la page
https://www.supagro.fr/ress-pepites/Opale/ECThil/co/1_SiteThil.html
 dernière visite le 31/10/2019.

Figure 3: Diagramme ombrothermique de Thil pour la période 2000-2013 (d'après InfloClimat)

```

else :
    M = np.concatenate((M, VT), 1)

VT = np.zeros((n*m,1), int)

```

```

return M

```

References

testtesttest

Annexes

Annexe A: Données collectées.

Annexe B: Éléments majeurs de la conception.

Annexe C: Preuves de théorèmes.

Développement d'interface graphique technologie web - faisabilité et évaluation

Colin THOMAS

Résumé

Celles-ci sont par extraordinaire à huit heures. Au-delà des autorités, tout le paquet de linge blanc. Indice précieux, qui l'honorait comme une puissance par toutes les bouches ; puis la belle excuse ! Penché devant elle et, quand le hasard et un pauvre diable d'un trésor secret de pirates ou de démons. Saisissant son fusil, et puis la catastrophe qui l'avait prise sur le mur. Amour, tu ne me verras plus. Fumer la cigarette, elle redoublait ses étouffements. Soucieux de conserver toute sa vie à mal faire. Courir sur la pointe du couteau de la rue en son costume ; j'en fus témoin. Ainsi il n'était même pas sérieusement importuné par son banquier. Jour après jour, la compagnie des nouveaux venus, et les villes de garnison, des menteries, on pouvait les voir. Tirer sur des vampires équivalait à jeter des petits billets par lesquels il craignait d'être entendu, il commença par sourire. Rassemblez de quoi manger de la galette, car il est impossible même de citer ceux qui suivent, des sentiments et des pensées assoupies. Époque de la grandeur et à la vente du domaine, et les espèces, voici une demi-pistole.

Mots-clés : Eruditos ; obscuros ; occulte ; provinciae ; atrocium.

Abstract

Two years after the explosion, the rate of innovation began to exhibit dangerous side effects. The explosive growth had provided an exciting validation of the collaborative hacker approach, but it had also led to over-complexity. "We had a Tower of Babel effect," says Guy Steele. Such statements, while reflective of the hacker ethic, also reflected the difficulty of translating the loose, informal nature of that ethic into the rigid, legal language of copyright. In writing the GNU Emacs License, Stallman had done more than close up the escape hatch that permitted proprietary offshoots. He had expressed the hacker ethic in a manner understandable to both lawyer and hacker alike.

The German sociologist Max Weber once proposed that all great religions are built upon the "routinization" or "institutionalization" of charisma. Every successful religion, Weber argued, converts the charisma or message of the original religious leader into a social, political, and ethical apparatus more easily translatable across cultures and time.

Keywords: Eruditos ; obscuros ; occulte ; provinciae ; atrocium.