



## Partiels, une suite logicielle dédiée à l'analyse des fichiers audionumériques

*Partiels, a software suite dedicated for the analysis of digital audio files*

**Pierre Guillot**

*Chercheur et développeur, Institut de Recherche et Coordination Acoustique/Musique Centre  
National de la Recherche Scientifique - CNRS, UMR 9912 STMS  
pierre.guillot@ircam.fr*

### Résumé

Cet article présente Partiels, une application open-source développée à l'Ircam pour l'analyse de fichiers audionumériques et l'exploration des caractéristiques sonores. Cette application utilise des plug-ins Vamp pour extraire des informations variées sur différents aspects du son, telles que le spectre, les partiels, la hauteur, le tempo, le texte et les accords. Partiels succède à AudioSculpt en offrant une interface moderne et flexible pour visualiser, éditer et exporter les résultats d'analyse en répondant à des enjeux variés allant de la pratique musicologique à la création sonore en passant par la recherche en traitement du signal. L'article décrit les fonctionnalités clés de Partiels, incluant l'organisation des analyses, la gestion des fichiers audio, la visualisation et l'édition des résultats, ainsi que les options d'exportation et de partage de données et son interopérabilité avec d'autres logiciels comme Max et Pure Data. De plus, il met en lumière les nombreux plug-ins d'analyse développés par l'Ircam, reposant notamment sur des modèles d'apprentissage machine, ainsi que l'extension Vamp de l'Ircam qui surmonte certaines limitations du format Vamp original.

**Mots-clés :** Analyse, visualisation, apprentissage machine, plug-in, réseaux de neurones, open-source.

### Abstract

This article presents Partiels, an open-source application developed at Ircam for analyzing digital audio files and exploring sound characteristics. This application uses Vamp plug-ins to extract various information about different aspects of sound, such as spectrum, partials, pitch, tempo, text and chords. Partiels inherits from AudioSculpt by offering a modern and flexible interface for viewing, editing, and exporting analysis results, addressing a variety of issues ranging from musicological practice to sound creation and signal processing research. The article describes the key features of Partiels, including the organization of analyses, audio file management, visualization and editing of results, as well as options for exporting and sharing data and its interoperability with other software such as Max and Pure Data. In addition, it highlights the numerous analysis plug-ins developed at Ircam, based in particular on machine



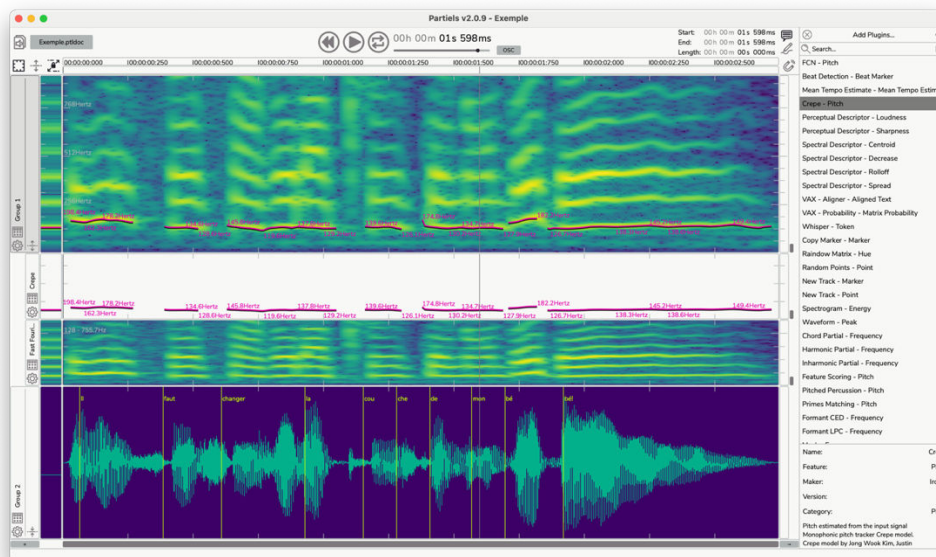
learning models, as well as the Ircam Vamp extension, which overcomes certain limitations of the original Vamp format.

**Keyword:** Analysis, visualization, machine learning, plug-in, neural networks, open-source.

## 1. Introduction

Partiels<sup>1</sup> est une application destinée à l'analyse de fichiers audio numériques conçue pour offrir une interface dynamique et ergonomique permettant d'explorer le contenu et les caractéristiques du son (spectre, fréquence fondamentale, tempo, accords, barycentre spectral, analyse de la parole, etc.). L'application Partiels est disponible pour Windows, macOS et Linux.

Cette application repose sur le format de plug-in d'analyse Vamp, développé au C4DM (Centre for Digital Music) de l'université Queen Mary à Londres (Cannam, 2006 ; Cannam 2010). En parallèle du développement de l'application principale, des plug-ins Vamp sont développés à partir de moteurs audio de l'Ircam – notamment SuperVP (Depalle 1991), PM2 (Depalle, 1993 ; Röbel, 2006), IrcamBeat (Peeters, 2007) et IrcamDescriptors (Peeters, 2004) – et de technologies extérieures – notamment Whisper (Radford, 2022) et Crepe (Kim, 2018) –. De plus, Partiels permet d'utiliser l'ensemble des plug-ins d'analyse Vamp préexistant<sup>2</sup> (tels que ceux de la BBC, du C4DM, de l'Université Pompeu Fabra, etc.).



**Figure 1 :** Aperçu de l'application Partiels contenant deux groupes d'analyses. Un premier groupe avec une analyse de type sonagramme et une estimation de la fréquence fondamentale, un deuxième groupe avec la forme d'onde et la transcription du texte.

1 <https://github.com/Ircam-Partiels/Partiels>

2 <https://www.vamp-plugins.org>



L'application Partiels et ses plug-ins sont développés à l'Ircam au sein de département Innovation et Moyens de la Recherche depuis octobre 2020. La première version du logiciel Partiels a été publiée sur le Forum de l'Ircam en septembre 2021. En décembre 2023, l'application et les plug-ins sont devenus gratuits et depuis septembre 2024, l'application Partiels est disponible en open-source sur GitHub sous licence GPL v3.

Dans un premier temps, cet article présente les tenants du projet et les objectifs qui en découlent pour, par la suite, décrire les principales fonctionnalités offertes par l'application<sup>3</sup>. Enfin il expose les développements actuels et les perspectives de ce projet.

## 2. Origines du projet

Partiels et les premiers plug-ins d'analyses qui lui sont associés découlent de l'application AudioSculpt (Bogaards, 2004a ; Bogaards, 2004b), un logiciel construit autour de bibliothèques et d'outils en ligne de commande, SuperVP, PM2, Ircam Descriptors et Ircam Beat, développés à l'Ircam au sein de l'équipe Analyse et Synthèse des sons. AudioSculpt offrait une interface graphique permettant de représenter et interagir avec les résultats des analyses et de transformer les sons. Le logiciel est aujourd'hui déprécié car l'architecture monolithique du logiciel et ses dépendances ont empêché sa mise à jour ainsi que son adaptation aux matériels et systèmes d'exploitation actuels<sup>4</sup>. La disparition de cet outil, populaire dans la communauté de l'Ircam, a créé une lacune importante chez les chercheur·euse·s en traitement du signal, musicologues, compositeur·rice·s, réalisateur·rice·s en informatique musicale, enseignant·e·s, designer·eus·e du son, etc. Il était devenu urgent d'offrir une alternative répondant à cette demande en évitant les limitations qui ont amené à l'obsolescence du logiciel.

L'enjeu était donc de proposer une solution à court terme à ce problème. Cependant, bien que possédant des pratiques et des usages qui se recoupent, les publics concernés par cette attente ont des besoins et des usages sensiblement différents. En traitement du signal, par exemple, il est important de pouvoir comparer les résultats de différents moteurs d'analyse. Les musicologues ont quant à eux besoin de générer des images et diagrammes afin d'analyser les sons. En composition, une des demandes fréquentes est de pouvoir exporter des résultats d'analyse pour les utiliser dans d'autres logiciels tels que Max<sup>5</sup>, Pure Data (Puckette, 1997), OpenMusic (Bresson, 2011) afin de produire des sons ou des partitions. De plus, depuis la dernière version d'AudioSculpt, de nouvelles problématiques et de nouvelles pratiques sont

---

3 L'application comprend un nombre important de fonctionnalités qu'il n'est pas possible de décrire en détail dans le présent article qui vise à offrir un aperçu général de l'outil. Le lecteur souhaitant des informations complémentaires concernant une fonctionnalité spécifique peut se référer au manuel d'utilisation de Partiels.

4 AudioSculpt est donc une application principalement graphique ayant une approche monolithique (tous les modules de traitement et d'analyse sont embarqués dans l'application). Le code de l'application repose entièrement sur l'API Carbon d'Apple, non multiplateforme, déprécié depuis 2012 et totalement obsolète sur les ordinateurs 64 bits ce qui fait que ce logiciel ne fonctionne plus sur les ordinateurs Apple depuis quelques années. La demande d'une mise à jour de ce logiciel de la part des utilisateurs et utilisatrices se faisait donc de plus en plus présente. Or de par cette approche monolithique et l'obsolescence de la quasi-totalité du code, le développement d'une nouvelle version, même élémentaire, nécessiterait de moyens qui ne sont pas à disposition de l'Ircam.

5 <https://cycling74.com>



apparues, suivant l'évolution des usages en audionumérique et plus largement de l'informatique. Ainsi, les principales spécifications fonctionnelles étaient :

- de lancer et de modifier rapidement et facilement les analyses et leurs paramètres,
- de corriger textuellement et graphiquement les résultats des analyses,
- d'améliorer la visualisation et la compréhension des analyses,
- de faciliter la communication et la perméabilité avec d'autres environnements logiciels,
- d'appliquer facilement un ensemble d'analyses à une série de fichiers audio,
- de comparer plusieurs fichiers audio et/ou les différents canaux des fichiers,
- de partager les analyses et les résultats avec d'autres utilisateurs.

Il était nécessaire de développer un outil suffisamment puissant pour répondre à ces pratiques complexes et hétérogènes. L'objectif était d'offrir une interface relativement simple et ergonomique afin d'être accessible à un public varié et potentiellement non initié, tout en utilisant une architecture flexible et dynamique permettant d'évoluer et d'intégrer de nouvelles technologies et de nouvelles analyses.

D'autres outils tels que Praat (Boersma, 2006), iAnalyse (Couprie, 2008), Spek<sup>6</sup> et Sonic Visualizer (Cannam, 2010 ; Cannam, 2015), destinés à l'analyse du son existent déjà et offrent chacun des solutions adaptées à des besoins spécifiques et variés. Mais aucun de ces logiciels ne répondait à l'ensemble de nos spécifications. L'un des obstacles importants réside dans le fait que bien que Partiels soit open-source, les algorithmes développés par l'équipe Analyse et Synthèse des sons sont quant à eux privés, empêchant leur intégration directe dans un autre logiciel. Sonic Visualizer, qui se rapproche le plus de nos besoins, possède quant à lui une architecture qui repose sur des plug-ins permettant donc l'intégration de technologies extérieures. Mais les analyses développées à l'Ircam possèdent des modèles de données complexes, multidimensionnelles avec des types différents, et nécessitent des entrées dynamiques, qui ne permettaient pas leur intégration dans ce logiciel. La nouvelle solution logicielle proposée repose sur le format de plug-in Vamp (Cannam, 2006 ; Cannam, 2010) en intégrant une extension, Ircam Vamp Extension (IVE)<sup>7</sup>, qui permet de répondre aux besoins spécifiques.

### 3. Fonctionnalités

Partiels permet avant tout de contrôler et appliquer des analyses sur des fichiers audionumériques, d'organiser, de visualiser, d'éditer les résultats et de naviguer dans leurs représentations. De plus, l'export des données peut se faire sous forme d'images et dans différents formats textuels. Le logiciel offre par ailleurs des fonctionnalités complémentaires comme la possibilité de faire du traitement par lot, d'interagir via l'interface en ligne de commande ou encore d'envoyer les résultats via Open Sound Control (OSC) (Wright, 1997).

<sup>6</sup> <https://www.spek.cc>

<sup>7</sup> <https://github.com/Ircam-Partiels/ircam-vamp-extension>



### 3.1. Organisation et gestion des analyses

Un document Partiels est associé à un ou plusieurs fichiers audio. Ce document possède un ou plusieurs groupes qui comprennent eux-mêmes des pistes d'analyse. Ces dernières se superposent les unes par-dessus les autres dans un groupe. Les groupes se répartissent les uns au-dessus des autres dans un document (Figure 1).

Partiels est un logiciel hôte de plug-ins. Sans ces derniers, il ne peut réaliser que des analyses de la forme d'onde et du sonagramme. L'ensemble des analyses complexes sont réalisées par les plug-ins Vamp chargés dynamiquement. Au lancement, l'application recherche les plug-ins installés sur la machine<sup>8</sup> et détermine les différentes analyses disponibles qui sont alors offertes à l'utilisateur. Ces analyses peuvent générer trois types de résultats tels qu'établi par le format Vamp et qui définissent leur mode de représentation graphique : des marqueurs (temps, durée, texte) affichés sous la forme de traits verticaux auxquels sont potentiellement associés du texte, des points (temps, durée, valeur numérique) affichés sous la forme de segments, des vecteurs (temps, durée, liste de valeurs numériques) affichés sous la forme d'une image, le plus souvent pour un sonagramme.

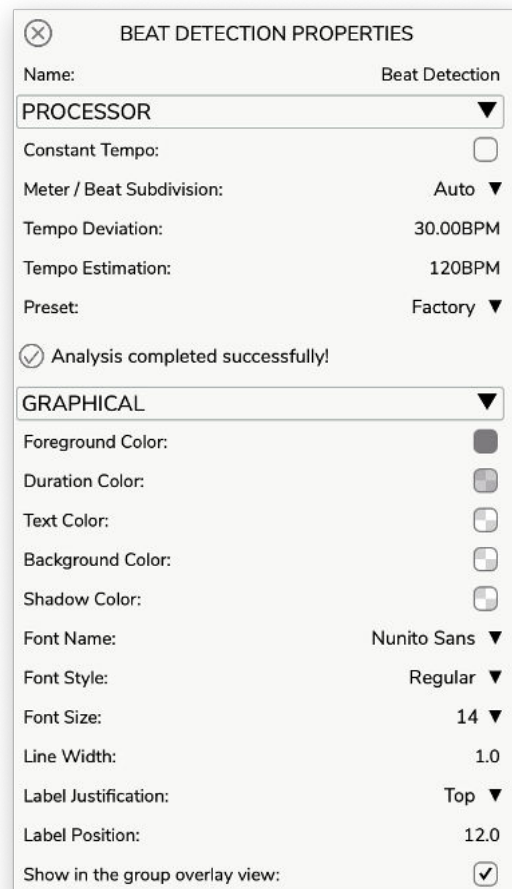
Lorsqu'une analyse est créée, celle-ci est nécessairement ajoutée à un groupe. Il est possible par la suite de modifier l'organisation à l'intérieur d'un groupe et de déplacer ou dupliquer une analyse d'un groupe à l'autre. Il est aussi possible de modifier les paramètres de la piste d'analyse via la fenêtre des propriétés. Ces paramètres sont répartis en deux catégories (Figure 2) :

- Les paramètres du moteur d'analyse (tels que la taille de fenêtrage d'une transformée de Fourier, le seuil de détection de transitoire, le modèle d'un réseau de neurones, etc.) qui génère une nouvelle analyse à la volée lors de changement.
- Les paramètres de rendu graphique (tels que les couleurs, la police de caractère, le positionnement du texte, l'échelle et la plage de valeurs, la grille, etc.) qui génère une mise à jour à la volée des représentations.

Cette approche permet d'accorder les représentations graphiques aux types d'analyse et à leur contexte, afin notamment de mettre en avant certaines données par rapport aux autres. La mise à jour à la volée des analyses et de leurs représentations est très utile notamment pour expérimenter, comprendre les paramètres et les adapter selon les besoins.

---

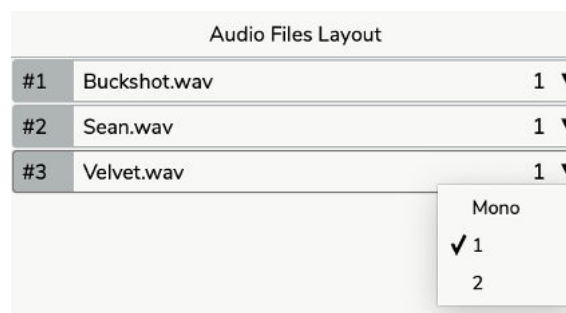
<sup>8</sup> Il est possible de configurer les paramètres de recherche des plug-ins et les chemins d'accès dans les propriétés de l'application.



**Figure 2** : Fenêtre de propriété d'une piste d'analyse de détection de tempo avec une section correspondant aux paramètres du moteur d'analyse et une section correspondant aux paramètres de représentation graphique.

### 3.2. Lecture et gestion des fichiers audio

Partiels offre un moteur avec un système d'agencement des fichiers audio (Figure 3) et de gestion du multicanal lié à l'analyse. Chaque canal de lecture est associé à un canal (ou à la somme des canaux) d'un fichier.



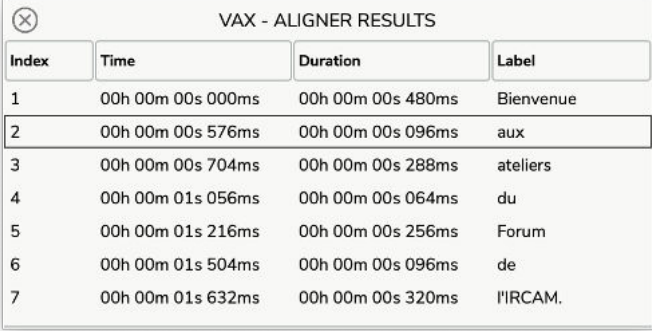
**Figure 3** : Fenêtre de d'agencement des fichiers audio pour l'analyse et la lecture où trois fichiers différents sont utilisés respectivement sur trois canaux.



Cette approche permet de comparer différents fichiers audio et/ou différents canaux d'un même fichier afin, par exemple, de mettre en avant des différences d'interprétations d'une même pièce, les artefacts de différents encodages ou encore permettre des analyses des effets de spatialisation audio (comme l'ouverture d'un *panning* stéréophonique). Lors de la lecture audio à proprement parlé, chaque piste du système d'agencement peut être envoyée à un canal spécifique de la carte son via une matrice de routage afin de s'adapter au contexte d'analyse et au matériel de restitution. Enfin, le transport audio offre un système de boucle qui peut s'aimer sur les pistes des marqueurs afin de se focaliser sur les sections temporelles spécifiques.

### 3.3. Visualisation des résultats d'analyse

La représentation graphique principale représente l'évolution des résultats dans le temps sur laquelle la tête de lecture se déplace automatiquement lors de la lecture audio (ou manuellement à la souris).

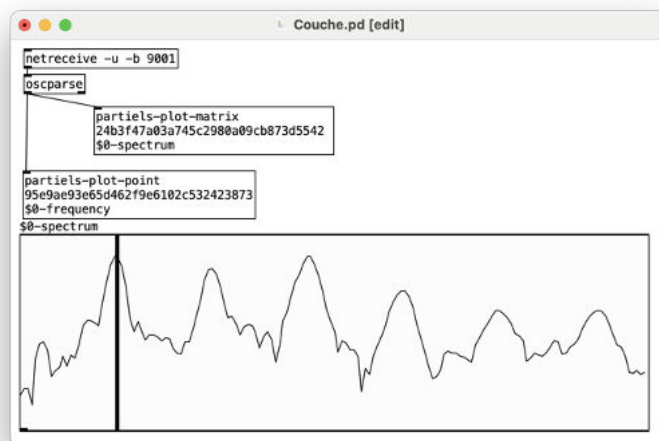


Index	Time	Duration	Label
1	00h 00m 00s 000ms	00h 00m 00s 480ms	Bienvenue
2	00h 00m 00s 576ms	00h 00m 00s 096ms	aux
3	00h 00m 00s 704ms	00h 00m 00s 288ms	ateliers
4	00h 00m 01s 056ms	00h 00m 00s 064ms	du
5	00h 00m 01s 216ms	00h 00m 00s 256ms	Forum
6	00h 00m 01s 504ms	00h 00m 00s 096ms	de
7	00h 00m 01s 632ms	00h 00m 00s 320ms	I'RCAM.

**Figure 4** : Tableau de valeurs contenant les résultats de type marqueur d'une analyse avec le plug-in VAX.

L'application offre aussi une représentation instantanée des résultats sur la partie gauche de l'interface globale ainsi qu'une représentation sous la forme d'un tableau de valeurs numériques et textuelles dans une fenêtre flottante (Figure 4). Ces représentations complémentaires sont synchronisées sur la tête de lecture et permettent de se référer conjointement aux différentes interfaces pour tirer parti simultanément des avantages de chaque type de visualisation.

Par ailleurs, les résultats peuvent être envoyés à des applications tierces telles que Pure Data ou Max via OSC (Figure 5) afin de générer d'autres représentations (et/ou de sonifier les résultats). La distribution de Partiels fournit des patches d'exemple à cet effet.



**Figure 5 :** Patch Pure Data affichant les valeurs d'un spectrogramme et d'une estimation de la hauteur d'un son à partir des données envoyées par Partiels en OSC.

### 3.4. Navigation et édition des résultats d'analyse

Partiels propose deux modes d'interaction inspirés des logiciels Max et Pure Data : la navigation et l'édition. Le mode de navigation est dédié à l'exploration des résultats, il offre une série d'actions et de raccourcis permettant de contrôler les zooms verticaux et horizontaux afin de se déplacer aisément dans les résultats des analyses. La fenêtre d'infobulle permet d'afficher les résultats correspondant à un temps donné (et à la position sur l'axe vertical lorsque c'est pertinent). Le mode d'édition permet de modifier les résultats d'analyse directement sur les représentations graphiques, en dessinant des courbes, créant des marqueurs, copiant, collant des résultats, etc. Ces opérations peuvent aussi être réalisées textuellement via le tableau des résultats. Lorsque les résultats des analyses sont modifiés par l'utilisateur, les données deviennent indépendantes de l'analyse des plug-ins et sont enregistrées dans des fichiers binaires associés au document. Cela permet non seulement de partager le document avec les modifications mais aussi de défaire et refaire les actions.

### 3.5. Export et partage des analyses

Un des enjeux importants du projet est de proposer un outil qui s'intègre dans les usages et les pratiques existantes des utilisateurs. Pour cela, Partiels offre des interopérabilités avec les logiciels pour la création musicale tels que OpenMusic, Max et Pure Data (ou d'autres stations de travail audio numérique) mais aussi des outils de bureautique de type tableur (ou plus simplement des éditeurs de texte).

Partiels permet d'exporter les résultats des analyses dans différents formats selon les besoins et les contextes d'utilisations. Il supporte les images aux formats PNG ou JPEG et en plus des caractéristiques usuelles relatives à la taille, Partiels offre la possibilité d'exporter les pistes indépendamment ou de préserver les représentations groupées. Il supporte aussi différents formats textuels dont le format JSON qui offre notamment la possibilité d'embarquer les informations relatives à l'analyse (pour recréer l'analyse a posteriori si besoin). Il offre aussi



le support du format CSV et de ses dérivés qui peuvent être facilement chargés dans différents environnements logiciels comme Pure Data, Max, Reaper<sup>9</sup>, etc. Cela permet par exemple d'utiliser ces données pour contrôler des synthétiseurs ou plus simplement pour segmenter des fichiers audio numériques. D'autres formats d'export plus spécifiques tels que SDIF ou CUE sont détaillés dans le manuel d'utilisation. Il est, par ailleurs, possible d'exporter seulement une analyse, un groupe d'analyses ou l'ensemble d'un document.

Partiels offre aussi un système de traitement par lot avec une interface graphique utilisateur qui permet d'appliquer une série d'analyses à un ensemble de fichiers audio et d'exporter automatiquement les résultats dans le format souhaité. Dans un contexte de développement, l'interface en ligne de commande permet, de même, d'appliquer un ensemble d'analyses à un fichier audio et d'exporter les résultats. Afin de faciliter la reproduction d'analyses, il est possible d'utiliser un document existant comme modèle *template* afin d'appliquer ses analyses à d'autres fichiers audio numériques. Cette fonctionnalité est disponible via l'interface graphique ou via l'interface en ligne de commande.

Enfin, Partiels offre la possibilité de consolider les documents pour faciliter le partage des analyses et de leurs résultats d'un utilisateur à un autre ou d'un ordinateur à un autre. Cette action crée un dossier associé au fichier Partiels qui contient les fichiers audio numériques et des versions binaires des résultats d'analyse. Ainsi, même si la machine qui reçoit le document Partiels ne possède pas les plug-ins nécessaires aux analyses, les résultats restent néanmoins disponibles.

## 4. Les plug-ins de l'Ircam

L'application Partiels repose sur les plug-ins Vamp. Il est compatible avec l'ensemble des plug-ins d'analyse d'ores et déjà disponibles<sup>10</sup> et qui sont distribués par de nombreuses institutions telles que l'Université Queen Mary, la BBC, l'Université Pompeu Fabra, etc. Mais le projet s'accompagne aussi de nombreux plug-ins d'analyse développés à l'Ircam. Un travail important a été réalisé afin d'assurer la compatibilité des analyses avec les principaux systèmes d'exploitation (macOS, Windows et Linux) afin de rendre accessibles au plus grand nombre ces outils. Ce travail s'est révélé particulièrement important pour les plug-ins reposant sur des technologies d'apprentissage machine et notamment les bibliothèques TensorFlow (Abadi, 2016) et PyTorch (Paszke, 2019).

### 4.1. Les technologies de l'Ircam

Une partie de ces plug-ins repose sur des technologies développées à l'Ircam par l'équipe Analyse et Synthèse des sons du laboratoire STMS :

- SuperVP : Ce plug-in offre de nombreuses analyses qui reposent sur le vocodeur de phase développé à l'Ircam par l'équipe Analyse et Synthèse des sons (Depalle, 1991). Il offre notamment des estimateurs de hauteur – pour les instruments mélodiques, les instruments percussifs et un basé sur le réseau de neurones FCN (Ardaillon, 2019), des analyses de type

---

<sup>9</sup> <https://www.reaper.fm>

<sup>10</sup> <https://www.vamp-plugins.org>



sonagramme résultant de FFT, LPC, Cepstrum, *True Envelop*, une détection des transitoires, la génération de marqueurs par détection de différence spectrale, une analyse des formants, ou encore une analyse de la fréquence de coupure de la voix.

- IrcamBeat : Ce plug-in permet une estimation du tempo d'un fichier audio ainsi que la génération de marqueurs à chaque temps qui s'adapte aux fluctuations de tempo (Peeters, 2007).
- IrcamDescriptors : Ce plug-in propose quatre descripteurs spectraux – *centroid*, *decrease*, *roll-off* et *spread* – et deux descripteurs perceptifs – *loudness* et *sharpness* – (Peeters, 2004).
- PM2 : Ce plug-in offre des estimations des partiels harmoniques et inharmoniques avec les informations de fréquence, de phase et d'amplitude et un modèle reposant sur une segmentation par marqueurs pour la détection d'accords (Depalle, 1993 ; Röbel, 2006).
- VAX : Ce plug-in repose sur des modèles de réseau de neurones pour l'alignement du texte sur l'audio. Il permet d'afficher une matrice de probabilité des caractères latins dans le texte et de segmenter et réaligner un texte donné en entrée (Doras, 2023).

## 4.2. Les technologies externes

Une autre partie de ces plug-ins repose sur des technologies externes à l'Ircam :

- Crepe : Ce plug-in offre une estimation de la hauteur à partir d'un ensemble de modèles de réseau de neurones développés par Jong Wook Kim, Justin Salamon, Peter Li and Juan Pablo Bello (Kim, 2018) et qui reposent sur TensorFlow<sup>11</sup> (Abadi, 2016), un framework d'apprentissage profond optimisé pour l'entraînement distribué.
- Whisper : Ce plug-in permet de transcrire de l'audio à partir des modèles de réseaux de neurones développés par OpenAI (Radford, 2022). L'intégration s'appuie sur la bibliothèque C/C++ Whisper.cpp développé par Georgi Gerganov<sup>12</sup> pour une inférence optimisée en temps-réel.
- Basic Pitch : Ce plug-in permet une estimation des hauteurs de sons multiphoniques à partir des modèles développés par Spotify (Bittner, 2022) et repose sur TensorFlow (Abadi, 2016).

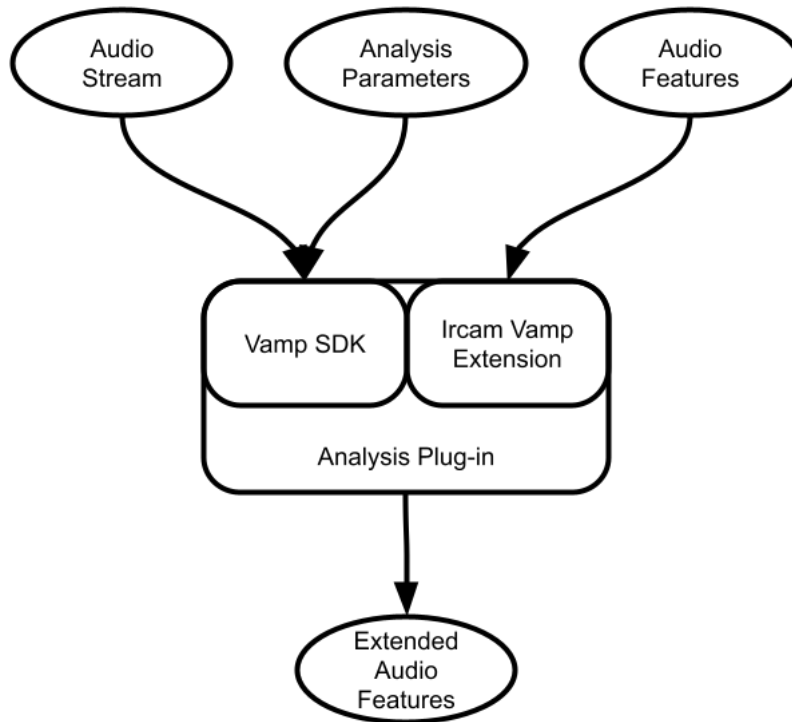
## 5. L'extension VAMP de l'Ircam

Lors de la mise en œuvre des plug-ins, nous nous sommes confrontés à des limitations du format Vamp. La bibliothèque Ircam vamp Extension (IVE), développée dans le cadre de ce projet, permet de palier à un certain nombre de contraintes. Cette bibliothèque, gratuite et libre, ne remplace pas le SDK des plug-ins Vamp mais se greffe à celui-ci afin de rajouter ou compléter certaines fonctionnalités (Figure 6). Cette approche s'inspire de bibliothèques analogues dans le domaine du traitement audio avec les formats de plug-ins d'effets (VST3,

11 <https://www.tensorflow.org>

12 <https://github.com/ggerganov/whisper.cpp>

AudioUnit, AAX, etc.) et notamment les REAPER Plug-in Extensions<sup>13</sup>, les PreSonus Plug-In Extensions<sup>14</sup> ou encore l'extension Audio Random Access (ARA)<sup>15</sup>.



**Figure 6 :** Graphique représentant l'architecture d'un plug-in d'analyse utilisant le SDK Vamp et IVE avec la gestion des données en entrée et en sortie.

Elle permet d'étendre les données partagées entre un plug-in et l'hôte sans modifier ni injecter de code dans le SDK original du format de plug-in afin d'assurer une compatibilité ascendante avec celui-ci et le support des plug-ins ne reposant pas sur l'extension.

### 5.1. Résultats d'analyse en entrée

Le SDK des plug-ins Vamp ne permet pas de chaîner les analyses et n'accepte en entrée que des paramètres numériques de type scalaire et des flux audio. Cette restriction empêche le développement d'approches complexes comme pour l'alignement du texte sur l'audio du moteur VAX qui nécessite de fournir le texte en amont de l'analyse. Elle offre de nouvelles interfaces aux plug-ins, associées à celles proposées par le SDK de Vamp, afin qu'il puisse recevoir les résultats d'autres analyses conjointement aux paramètres d'entrée. Dans Partiels, cela permet de réaliser des chaînes d'analyses. Par exemple, l'analyse Whisper est utilisée pour transcrire le texte d'un fichier audio et les résultats sont envoyés à l'analyse VAX pour aligner ce texte sur l'audio (Figure 7). La mise à jour d'un paramètre d'un plug-in d'entrée ou la

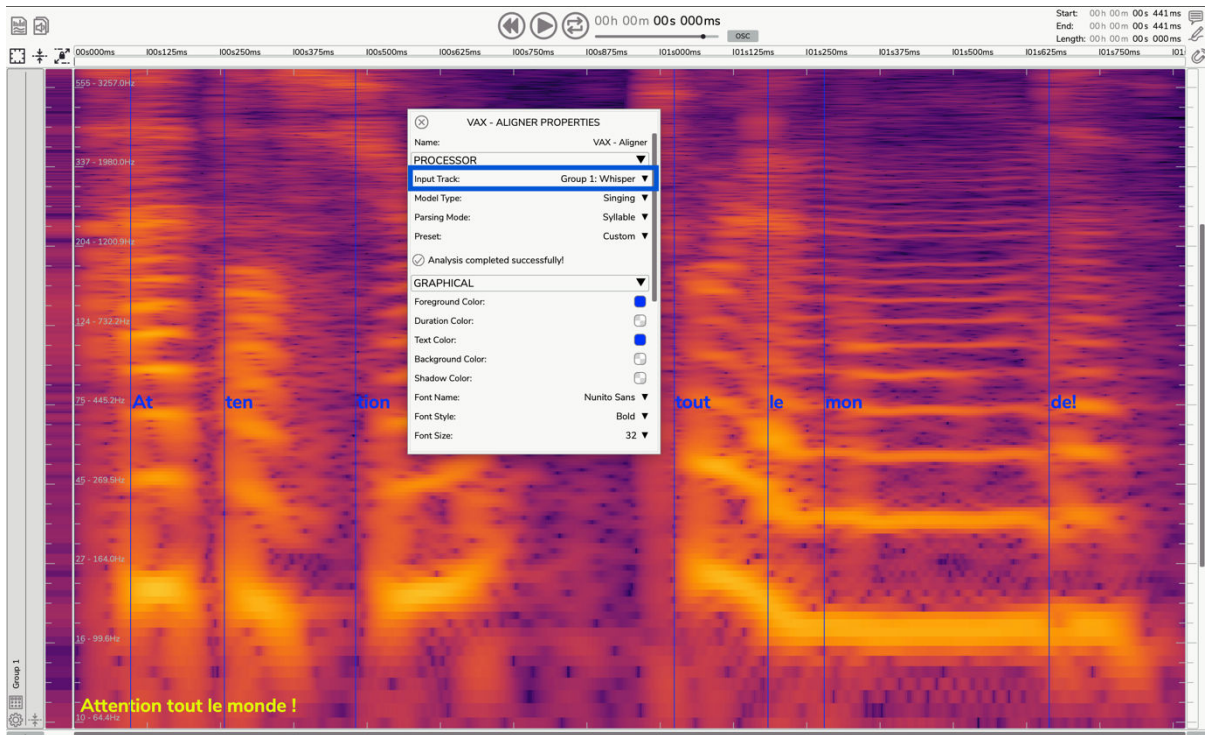
13 <https://www.reaper.fm/sdk/plugin/plugin.php>

14 <https://github.com/fenderdigital/presonus-plugin-extensions>

15 [https://github.com/Celemony/ARA\\_SDK](https://github.com/Celemony/ARA_SDK)



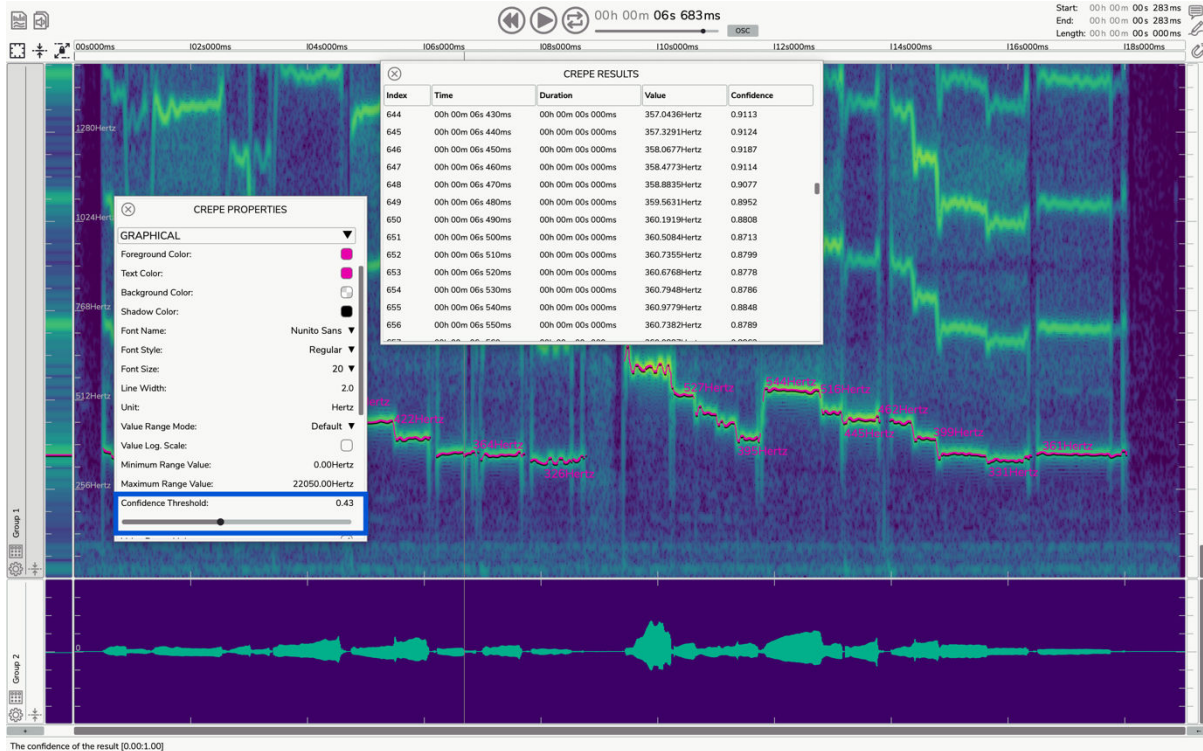
modification de ses résultats déclenchent automatiquement le recalcul de l'analyse qui en découle. Ainsi, le changement du modèle de Whisper ou la correction manuelle du texte relancent l'analyse via le plug-in VAX et réalignent le texte en prenant en compte les modifications.



**Figure 7** : Capture d'écran d'un document comprenant le sonagramme d'un enregistrement d'une voix parlée, le marqueur résultant de la transcription de la voix avec le plug-in Whisper, en jaune, et les marqueurs résultant de l'alignement des syllabes de cette transcription sur la voix avec le plug-in VAX, en bleu. La fenêtre des propriétés de l'analyse VAX montre que la piste Whisper est utilisée en entrée, fonctionnalité rendue possible grâce à la bibliothèque IVE.

## 5.2. Des données complémentaires en sortie

Une autre restriction réside dans les formats de sortie des résultats d'analyse tels que définis par le SDK de Vamp. Le résultat peut être soit un label, soit un scalaire, soit un vecteur de valeurs numériques. Dans de nombreux cas, il est nécessaire d'augmenter ces données de sortie en ajoutant des données numériques supplémentaires, afin d'offrir, par exemple, un score de confiance pour les transitoires ou l'amplitude et la phase associées à la fréquence d'un partiel. La bibliothèque IVE offre la possibilité d'ajouter des données numériques supplémentaires aux données de sortie. Dans Partiels, ces valeurs sont affichées en complément des données de résultats principales. Elles permettent, par ailleurs, de modifier les représentations en filtrant les résultats selon des seuils de valeurs (Figure 8). Ainsi, il est possible de cacher certains partiels dont l'amplitude est inférieure à un seuil minimum.



**Figure 8** : Capture d'écran d'un document comprenant la forme d'onde et le sonagramme d'un enregistrement d'une voix chantée. L'estimation de la fréquence fondamentale de la voix ; représentée par la courbe rose sur le sonagramme, a été réalisée avec le plug-in Crepe. La fenêtre des résultats affiche la valeur en hertz de chaque point temporel ainsi qu'un score de confiance, cette valeur supplémentaire est rendue possible grâce à la bibliothèque IVE. La fenêtre des propriétés de l'analyse Crepe montre que les résultats affichés sont filtrés selon un seuil de confiance.

## 6. Bilan et perspectives

Le projet Partiels est aujourd'hui dans une phase avancée de son développement. Il offre un ensemble d'outils dont les fonctionnalités répondent à un grand nombre de besoins, de contextes et cas d'utilisation ; grâce notamment aux analyses disponibles via ses nombreux plug-ins. Le logiciel est par ailleurs déjà utilisé, à l'Ircam comme à l'extérieur, dans le domaine de la recherche, de la création artistique mais aussi en pédagogie et notamment à Sorbonne Université dans les formations de musicologie et le cursus Sciences et Musicologie.

Néanmoins, il reste de nombreux aspects qui peuvent être complétés et améliorés. L'un des enjeux est de continuer à faciliter l'accès de cet outil au plus grand nombre. Nous envisageons notamment d'offrir des tutoriels sous la forme de vidéos mais aussi de proposer des traductions des interfaces (car pour le moment, le logiciel n'est disponible qu'en anglais). D'autre part, nous réfléchissons à l'intégration de certaines fonctionnalités d'IVE directement dans le SDK de Vamp afin que ces améliorations puissent être utilisées dans d'autres hôtes Vamp (Sonic Visualizer, Max, Pure Data, etc.) et que les plug-ins de l'Ircam soient mieux supportés en dehors de Partiels. De façon générale, IVE pourrait servir d'espace d'expérimentation avant l'intégration plus stable au sein du SDK de Vamp. Dans cette optique expérimentale, nous souhaiterions intégrer une API graphique à IVE afin de permettre des



représentations plus spécifiques à certaines analyses, notamment lorsque les résultats sont multidimensionnels. À ce sujet, nous travaillons actuellement sur une adaptation des analyses du Snail (Hélie, 2017) afin de générer un sonagramme à partir de données où la couleur d'un pixel dépend de multiples données (et non pas seulement d'une seule valeur d'amplitude). Nous voulons aussi continuer à développer des patches Max et Pure Data afin d'offrir des représentations alternatives des données et aussi de faciliter la sonification de ces données. Enfin, nous sommes continuellement en train de développer de nouveaux plug-ins d'analyse reposant sur de nouvelles technologies que nous développons à l'Ircam ou sur des modèles extérieurs qui utilisent notamment PyTorch (Paszke, 2019).

## **7. Remerciements**

Nous tenons, avant tout, à remercier l'université Queen Mary et plus spécifiquement Chris Cannam qui a développé Sonic Visualizer et le SDK de plug-in Vamp. Nous souhaitons évidemment remercier l'équipe de recherche Analyse et Synthèse des sons du laboratoire STMS hébergé à l'Ircam et plus spécifiquement Axel Röbel, Frédéric Cornu, Guillaume Doras et Yann Teytaut pour leurs contributions aux moteurs d'analyse. Enfin, nous remercions Matthew Harris qui a contribué au développement d'une partie des plug-ins Vamp SuperVP, IrcamBeat, IrcamDescriptors et PM2.

## **Bibliographie**

Abadi Martin, et al. (2016), « TensorFlow: A system for large-scale machine learning », 12th USENIX Symposium on Operating Systems Design and Implementation (OSDI '16), Savannah, USA.

Ardailon Luc et Röbel Axel (2019), « Fully-Convolutional Network for Pitch Estimation of Speech Signals », Interspeech 2019, Graz, Austria.

Bittner Rachel M., et al. (2022), « A lightweight instrument-agnostic model for polyphonic note transcription and multipitch estimation », Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP), Singapore.

Boersma Paul et van Heuven Vincent (2001), « Praat, a system for doing phonetics by computer », Glot International, vol. 5, no 9/10, p. 341–345.

Bogaards Niels et Röbel Axel (2004), « An interface for analysis-driven sound processing », in AES 119th Convention, New York, USA.

Bogaards Niels, et al. (2004), « Sound analysis and processing with AudioSculpt 2 », Proceedings of the International Computer Music Conference, Miami, USA.



---

Bresson Jean, et al. (2011), « OpenMusic – Visual programming environment for music composition, analysis and research », Proceedings of the 19th ACM International Conference on Multimedia, Scottsdale, USA.

Cannam Chris, et al. (2006), « The Sonic Visualiser: A visualisation platform for semantic descriptors from musical signals », Proceedings of the 7th International Conference on Music Information Retrieval (ISMIR), Victoria, Canada.

Cannam Chris, et al. (2010), « Sonic visualiser: An open source application for viewing, analysing, and annotating music audio files », Proceedings of the ACM Multimedia International Conference, Florence, Italy.

Cannam Chris, et al. (2015), « MIREX 2015 entry: Vamp plugins from the Centre for Digital Music », Proceedings of the International Symposium on Music Information Retrieval (ISMIR), Urbana-Champaign, USA.

Couprie Pierre, « iAnalyse : un logiciel d'aide à l'analyse musicale », Actes des Journées d'Informatique Musicale, Albi, France.

Depalle Philippe et Poirrot Gilles (1991), « SVP: A modular system for analysis, processing and synthesis of sound signals », Proceedings of the International Computer Music Conference, Montréal, Canada.

Depalle Philippe, et al. (1993), « Tracking of partials for additive sound synthesis using hidden Markov models », Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP), Minneapolis, USA.

Doras Guillaume, et al. (2023), « A linear memory CTC-based algorithm for text-to-voice alignment of very long audio recordings », Applied Sciences, vol. 13, no 3, art. 1854.

Hélie Thomas et Picasso Charles (2017), « The Snail: A real-time software application to visualize sounds », Proceedings of the 20th International Conference on Digital Audio Effects (DAFx-17), Edinburgh, UK.

Kim Jong Wook, et al. (2018), « CREPE: A convolutional representation for pitch estimation », Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP), Alberta, Canada.

Paszke Adam, et al. (2019), « PyTorch: An imperative style, high-performance deep learning library », Advances in Neural Information Processing Systems (NeurIPS 2019), Vancouver, Canada.



---

Peeters Geoffroy (2004), « A large set of audio features for sound description (similarity and classification) in the CUIDADO project », IRCAM Publication, IRCAM, Paris.

Peeters Geoffroy (2007), « Template-Based Estimation of Time-Varying Tempo », EURASIP Journal on Advances in Signal Processing, vol. 2007, issue 1.

Puckette Miller (1997), « Pure Data: another integrated computer music environment », Proceedings of the International Computer Music Conference, Thessaloniki, Greece.

Radford Alec, et al. (2022), « Robust Speech Recognition via Large-Scale Weak Supervision » [en ligne], arXiv, URL : <https://arxiv.org/abs/2212.04356> [consulté le 13/10/2025].

Röbel Axel (2006), « Estimation of partial parameters for non stationary sinusoids », Proceedings of the International Computer Music Conference, New Orleans, USA.

Wright Matthew et Freed Adrian (1997), « Open SoundControl: A new protocol for communicating with sound synthesizers », Proceedings of the International Computer Music Conference, Thessaloniki, Greece.