Combobulator User Guide

Version 1.1

by DataMind Audio

DATAMIND
AUDIO

System Requirements

The Combobulator requires Windows 10 or Mac OS 12 and up and is compatible with Intel and Apple Silicon.

The Combobulator can be installed as a VST or Audio Unit. Every major DAW is supported (except for Avid Pro Tools).

Introduction

Welcome to Combobulator by DataMind Audio!

The Combobulator resynthesises an audio input signal by performing a timbral “style-transfer” on it. It uses a revolutionary new synthesis method, neural synthesis, to reinterpret real-time audio through a selected “Artist Brain” (AKA Model, AKA Neural Network). Neural synthesis does not use samples. The neural synthesis engine synthesises the output from scratch, based on what each model has “learned” about sound during model training.

Each model is trained by our in-house specialists, working directly with each artist to curate and refine the training data, to bring you the highest-quality collection of ethically-sourced models available anywhere in the world. With your purchase of each model, artists make 50% of the gross profit from each sale.

NOTICE: Combobulator will not make music for you. The AI models are not designed to faithfully replicate the artists they were trained on or to make you magically sound like the professional music producer your model is named after. Instead, think of it as a synthesis device that has essentially learned how to hallucinate complex timbres based on patterns acquired in the training material.

While the Combobulator functions like an audio effect, it is designed to be played as a neural synthesis instrument. Each model is like a “black box” that contains a universe of timbres that can only be explored by altering the input signal and modulating the latents, a neural synthesis concept we will clarify in section 2. Each model can be used to create drums, bass, textures, melodies, and harmonies, all together or individually.

Installation

After purchasing the Combobulator you will need to download the installer package from the DataMind Audio website. Once downloaded, run the installer.

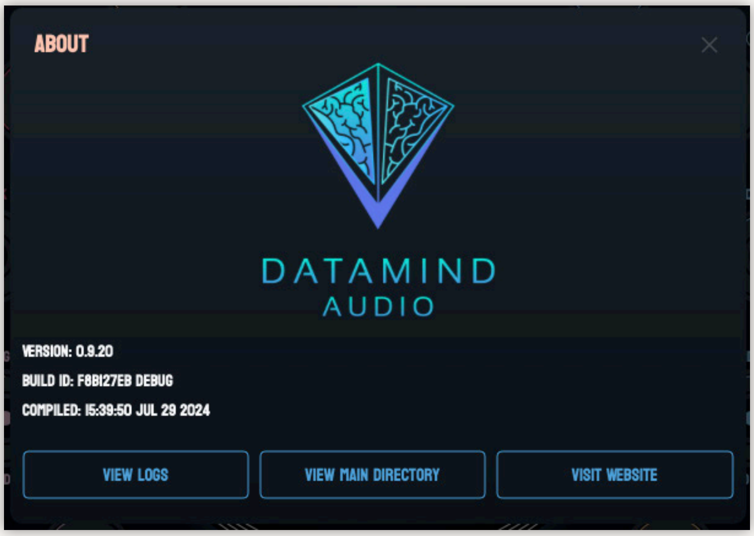
Open your DAW of choice and wait for it to scan the new plug-in. This may take a few seconds. If you don't see it show up, make sure both the VST3 and/or AU paths are enabled in your DAW's settings, then rescan and restart your DAW.

Once the plug-in has finally loaded, enter your username and password, which you will obtain after creating an account on the DataMind Audio website. Ensure you have a working internet connection while doing this.

That's all!

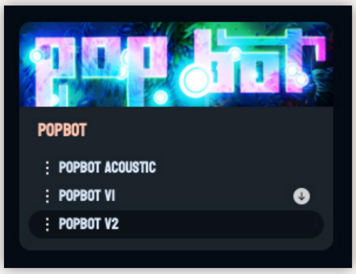
Enjoy!

P.S. Information about the plug-in build, directories used by the app and log files can be found by clicking on the datamindaudio.ai *button in the top-left. If you encounter any issues and would like to file a bug report, please e-mail* support@datamindaudio.com *or post a message on our discord channel* https://discord.gg/KaDdaKbg *and attach the Log.txt file that can be found in the Logs folder. Thank you!*



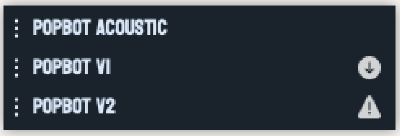
Section 1: Model Browser

In the Model Browser you will find a list of all available models in our store. Various model groups are arranged into cards, with each group containing at least one model.



Each model card contains a list of models, and we are regularly adding more and more of these as time passes. When a new model becomes available, it will appear automatically in the model browser.

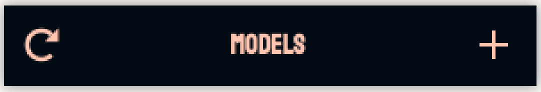
To start using a model, make sure that it has been downloaded to your system and is ready to go. If you see a download icon, click on the item and it will begin downloading.



If you see a warning icon, this means the model file is corrupt. In this case, click on the model name and then the "re-download" button.

If you see a lock icon, this means the model has either not been purchased, or you will need to update your license. To do so, click on your username in the top right, insert your credentials, and then submit. You should now see the lock disappear.

To gain more insight about each model, click on the three dots icon to the left of the model name, and a window will pop up with more information.

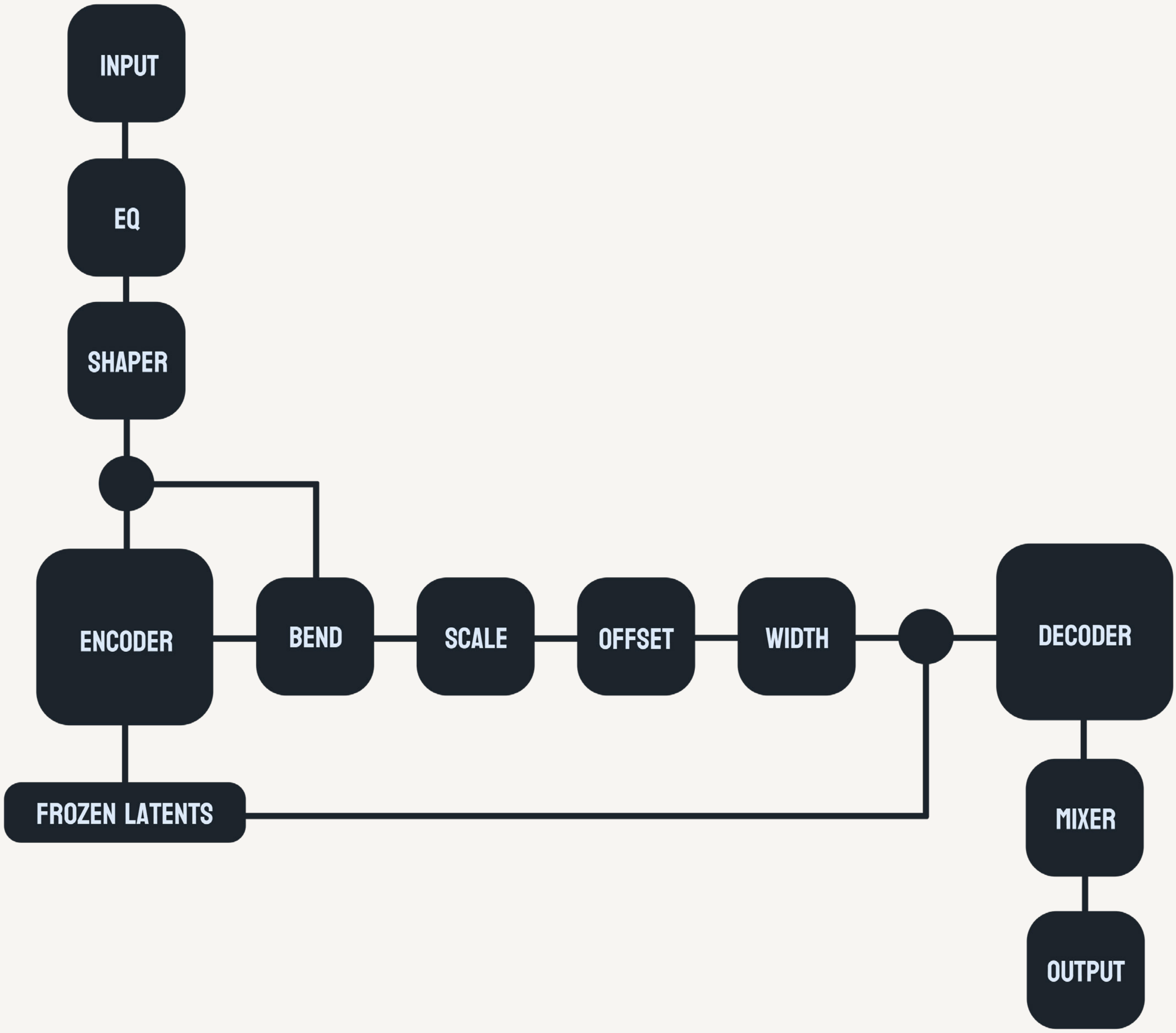


In the case you are manually installing models into the DataMind Audio models folder, you can refresh the browser by clicking the circular arrow icon in the top-left.

To access our store and purchase a model, click on the plus icon. Each new model you purchase pays a generous split to the Artist. Thank you for supporting the Arts with your use of our AI technology!

Section 2: Audio Processing Chain

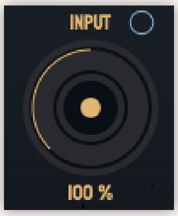
The Combobulator has a series of fourteen parameters that process incoming audio in three stages: the input stage, the model inference stage, and finally the output stage.



2.1: The Input Stage

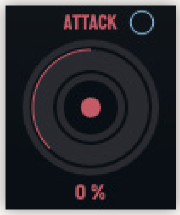
There are many ways of affecting the way the neural network synthesises the audio from an input signal. To accommodate some basic use cases we have provided a transient shaper and filter in the input stage.

Input



This is the gain knob (or "pre-gain") for the input signal. In the same way a distortion plug-in will react differently to a higher volume input, so does the Combobulator. Most importantly, this dial can be used to make some models generate chaotic sounds when the dial is brought down to 0%.

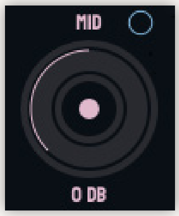
Attack & Sustain



These knobs control an internal transient shaper. They can be used to summon sharper, more percussive sounds from the neural network or softer more sustained timbres.

High / Mid / Low

These knobs control an internal three-band equaliser. The Combobulator tends to “latch on” to the dominant frequency of the input signal, so if the output sounds too muddy or too bright, you can use these knobs to modify the timbre of the incoming signal to alter the model’s interpretation.



2.2: The Model Inference Stage

The model inference stage is where the real magic happens, and we have provided many controls to give users the power to push the models beyond their apparent capabilities and into unexplored territory.

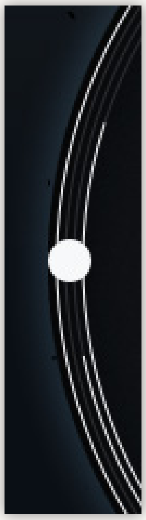
To understand how this next group of parameters works and affects how the model processes the input signal, we must understand a few basics regarding this style of neural synthesis.

The neural network works by taking an input signal and passing it through what is known as the *encoder*, we call the encoded signal the *latents*. Finally, the latents are passed to another neural network called the *decoder*, which converts our latents back into a regular audio signal. The encoding and decoding stages of this process break down the audio and then reconstruct it, swapping details in the input signal for imagined details in the output. This is what is referred to as a style transfer.

A nice analogy for this process is that of an artist drawing a highly detailed picture of a tree, then passing it off to a friend who erases all the details from it (the encoding phase) and subsequently passing it on to yet another friend, who will to the best of their abilities attempt to reconstruct the lost details in their own style (the decoding phase).

It is also possible to take greater control of the latents in between the encoder and decoder, to go beyond simple style transfer and venture into a more abstract and unpredictable realm of sound. These hacks bring out the true character of the models and we have provided many useful tools for this style of exploration. We will go into more detail regarding this in section 2.4.

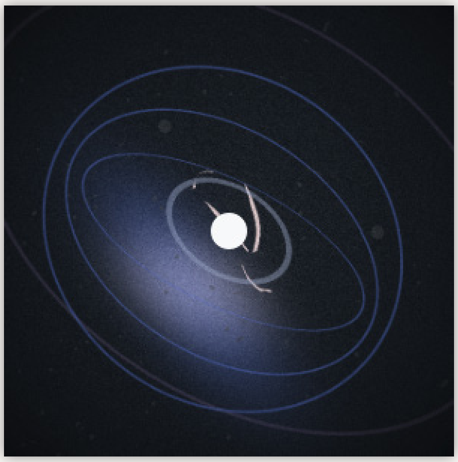
2.3: The Visualiser



At the heart of Combobulator’s neural synthesis engine is the concept of manipulating latents. Each latent represents a learned “feature” contained in a model, collapsed into a single variable. These features control the dynamics, timbre, and every other aspect of the sound.

Each model has anywhere between sixteen and sixty-four latents that come together to synthesise the audio. The first four latents represent the most dominant features in a model. You can scale and offset the latents with the *scale/offset* faders, and you can see the modulation for the first four latents.

We allow you to modulate the relative positions of the individual latents using the



stagger option inside of the modulators. These can also be controlled directly using the MIDI Input modulators. Generally speaking, the further apart the latents are, the further the model is pushed to sound different from the input material.

Note: Only the first four latents are visualised, but you are modulating all of them behind the scenes.

In the centre of this circular plane is the latent visualiser, which is a vectorscope for displaying latent phase offsets. The Combobulator visualises the relative distances between latent variables currently active in the neural network, almost akin to seeing synapses fire inside of an invisible brain.

2.4: Hacking The Brain

Beyond scaling and offsetting the latents, we have come up with a series of modules that allow for a much more explorative sound design process with the Combobulator. It is important to understand that the latents are simply a short list of numbers, and by altering them drastically, we obtain drastic results. Some experimentation will be necessary to become familiar with these tools, but they will often yield the most interesting results. There is a lot of sound to be discovered in each model and by combining this next set of parameters with the modulators, wild results will surely emerge!

Scale

Scale affects how broadly the neural network's timbres are selected. Turning *scale* all the way down will "freeze" the timbre, as the input signal would no longer have an impact on the latents in the neural network. Turning *scale* all the way up will summon timbres from all across the neural network, maximising dynamic and timbral variety in the model's interpretation of the input signal. *Scale* is compatible with the *stagger* parameter found in the LFO and Envelope Follower modulators.

Offset

Offset shifts where inside the neural network the sound is being synthesised from. Moving this fader will radically alter the resulting timbre, with the outer edges pushing the model into gradually more extreme, experimental, and alien territories. Each model's *offset* fader does something a little bit different because the features represented by the latents are decided by the machine learning algorithm during the training process. One direction is likely to make the sound brighter and louder, the other direction darker and quieter, as is the natural duality of dynamics and timbre. *Offset* is compatible with the *stagger* parameter found in the LFO and Envelope Follower modulators.

Encode

Encode mixes the latents generated from the encoder with a baseline block of latents which initially are all set to zero. When loading the plug-in from scratch, bringing the encode dial down to zero gives the same result as does bringing the scale fader down to its lowest position. We can modify the baseline block of latents by triggering the *freeze* and *reset* buttons. More on these last two below.

Bend

Bend is inspired by classic circuit bending. It mixes audio samples from the input signal directly into the latents, bypassing the encoder. To hear the effect of this, bring down the *encode* dial all the way down to 0%, and bring the bend parameter all the way up to 100%, then send a simple sine tone into the Combobulator.

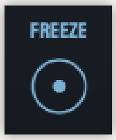
Dilation

Dilation determines the spacing between each sample to be used in bending. Used in conjunction with *bend*, this will change the patterns that emerge while using the *bend* parameter.

Width

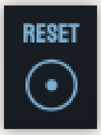
Width applies a gentle offset to the latents used to synthesise the left and right channels. A greater difference in between the each channel creates variance, re-sulting in a wider stereo image.

Freeze



Freeze captures the latents just before being decoded and saves them as a baseline to be mixed with the *encode* dial. The frozen latents are by default all set to zero. To trigger this parameter from your DAW, send a C-2 MIDI message to channel 16.

Reset



Reset sets the baseline *Latents* back to zero. To trigger this parameter from your DAW, send a C#-2 MIDI message to channel 16.

2.5: The Output Stage

In the output stage, we have a couple more parameters.

Dry/Wet

Dry/Wet mixes the input signal with the affected wet signal.

Output

Output master volume knob.

Section 3: Modulator Browser

The Combobulator has a total of twelve modulators: five LFO's, five Envelope Followers and two MIDI inputs. Each modulator is capable of modulating any parameter that contains a modulation node next to it. This includes the parameters of the modulators themselves.

To enable a modulator, click on the plus icon at the bottom of the modulator browser, and a window will pop up with a list of available modulators. Click on any of the modulators and you'll see them appear in the browser to the right.

To disable a modulator simply click on the exit button on the modulator card and to minimise/un-minimise a modulator, double-click on the modulator card.

To begin modulating a parameter, click on any of the modulator modulation nodes, and drag the patch chord to the modulation node next to any parameter. This will create a connection between the modulator and the parameter.

To control the depth of the modulation, use the dials that appear above the modulation nodes and to disable a connection, double-click on the corresponding dial.

3.1: LFO

The Combobulator's LFO (Low-Frequency Oscillator) has five LFO shapes: sine, triangle, sawtooth, rectangle, and noise.

Beat Sync

Beat Sync switches between a beat synced rate and time synced rate in ms.

Rate

Rate controls the frequency of the LFO oscillation and supports beat synced rate and time synced rate.

Phase

Phase controls the phase offset of the oscillator.

Pulse Width

PW controls the pulse width of each oscillator except for noise.

Scale

Scale controls the depth of the oscillation range.

Stagger

Stagger only works for latent *scale* and *offset*. This feature delays the modulation signal being sent to the latents by multiples of the stagger amount (i.e. creating a rippling effect).

3.2: Envelope Follower

The Envelope Follower converts the amplitude of the input signal into a control signal.

Attack & Release

Attack and *release* control the slew rate of the Envelope Follower.

Sidechain

Sidechain allows an external audio signal to be sent into the Combobulator to control the Envelope Follower.

Scale

Scale controls the depth of the Envelope Follower's range.

Stagger

Stagger only works for latent *scale* and *offset*. This feature delays the modulation signal being sent to the latents by multiples of the stagger amount (i.e. creating a rippling effect).

3.3: MIDI Input

Combobulator has two MIDI Input modules and they automatically operate in two distinct ways. When the modulator is connected to a regular parameter, the highest velocity value of any note determines the modulation value. When the modulator is connected to latent *scale* or *offset*, each midi note in the octave controls a different latent. Note C through B (any octave) corresponds to latents 1–12, and the velocity determines the magnitude of the modulation. This means that a midi sequence can be used to modulate the latents.

Attack & Release

Attack and *release* control the slew rate.

Scale

Scale controls the depth of the MIDI Input's range.

Section 4: Presets

The Combobulator has a useful preset system for saving and loading the state of the plug-in. We have provided a starting bank of presets which can be explored by clicking the left and right arrows on the preset manager button.

To explore all available presets, click on the library icon next to the left arrow icon, and you will see a window will all available presets appear. Click on the preset for it to load.

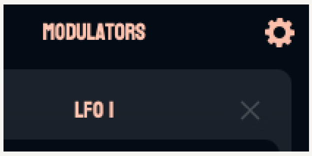
You can easily restore the plug-in to a default state by clicking on the default preset.

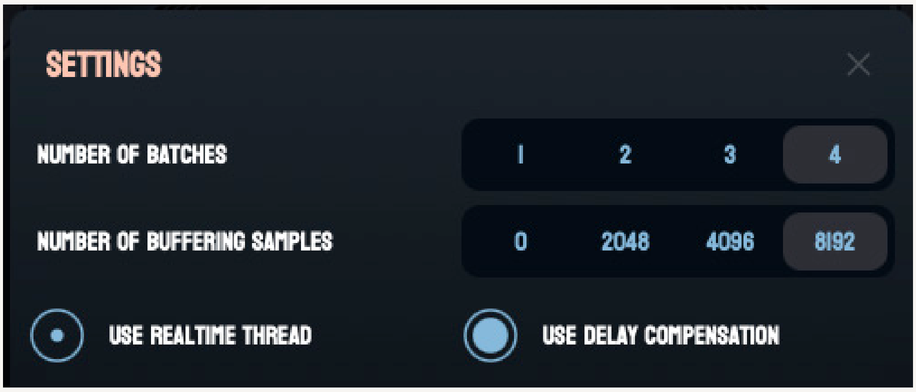
To save a new preset, click on the floppy disk icon to the right of the preset manager, a window will pop up with a prompt to save the current plug-in state. Give the preset a name and then submit.

Note: presets do not hold information about the selected model.

Section 5: Settings

By clicking on the gear icon in the top-right, you will gain access to the Settings panel, which will allow you to tweak options relating to the way in which the processing happens. These can be fine-tuned for resource efficiency or low latency.





Number of Batches

This allows you to specify a multiple of the model input size, drastically improving performance at the cost of extra latency. You can effectively think about it the same way you would the block size in your DAW.

Number of Buffering Samples

This allows you to modify the number of samples used to guard against dips in performance that would otherwise cause clicks. This option only makes sense when Use Realtime Thread is switched off. Bringing this value down reduces latency, but should be kept at a minimum of 2048 when Use Realtime Thread is off.

Use Real-time Thread

This decides if the model processing should happen in parallel. It should be used when attempting to bring down the latency as low as possible, but will cause a very large increase in CPU usage. Practically speaking, you won't gain much advantage by turning this on unless you are specifically using a low-latency model and Number of Batches and Number of Buffering Samples are set to one and zero respectively. In most scenarios, you should keep this off.

Use Delay Compensation

Different models have different amounts of latency. The Combobulator automatically takes care of applying delay compensation to the signal and reports it to the host DAW. However, some models require more than 500ms of delay compensation to be in sync with the incoming audio. This can be annoying while producing music, so to solve this, we have provided a toggle switch, which can be found in the toolbar next to the preset manager. Click the button to toggle delay compensation on or off.

Low Latency Recipe

To be able to play with the models in a real-time setting with the lowest latency possible, you must select one of our low-latency models and then match the settings seen in the image below. It is important to also set your host block size to 512 and sample rate to 44100. If your computer seems to struggle, try increasing the number of batches or your host block size. This is a CPU-heavy process, and not all systems will be able to handle it without clicks.

About DataMind Audio

DataMind Audio was co-founded by Rob Clouth (Lead Programmer), Ben Cantil (Production), Catherine Stewart (Managing Director), Zack Zukowski and CJ Carr (a.k.a. Dadabots) (Lead Researchers). We are a small company created by musicians for musicians, making AI-powered music production and sound design software that empowers artists and inspires new ideas.

On behalf of DataMind Audio, a heartfelt thank you to the generous support and funding from the University of Edinburgh's Creative Informatics Resident Entrepreneur Grant and Creative AI Music & Audio Pilot Project Grant, and from the UK government's Feasibility Studies for Artificial Intelligence Solutions Grant by Innovate UK. We have also received additional support from Edinburgh Innovations, the EPCC, and the University of Edinburgh.

UKRI Innovate UK

CREATIVE INFORMATICS

THE UNIVERSITY of EDINBURGH

EDINBURGH INNOVATIONS

epcc

Based on the RAVE technology developed at IRCAM in the STMS Lab.
Authors : Antoine Caillon, Philippe Esling.

ircam Centre Pompidou

Credits

Product Design: Rob Clouth, Ben Cantil and Luca Ayscough
Lead Programmer: Luca Ayscough
Programmer: Robin Leathart

Lead Data Scientist: Zack Zukowski
Innovation Lead: Dr. Martin Parker
Model Reliability Engineers: Yashique Chalil, Andreas Papaeracleous, and Robin Leathart
Data Science Consulting: CJ Car

UI Design: Jacob Crider
Web Design and Security: Norman Visser, Red Hot Designs
Sustainability Officer: Nick Sumbles

Managing Director: Catherine Stewart
Chief Operating Officer: Ben Cantil

Special Thanks:
Hugues Vinet
Victoria Murphy
Nicola Osborne
Andrew Apanov