Imagisonne — premier atelier de lutherie

Cyclonium · Prototype 1.0 · 17 octobre 2026

Dessiner l’évolution d’un son, l’entendre, puis retrouver ce son comme un instrument : ce premier prototype matérialise la facette lutherie de Phonogramme. Il ne contient pas de document de composition. L’instrument exporté constitue un point de passage vers les futurs ateliers ; aucun consommateur externe n’est encore raccordé automatiquement.

Prise en main

L’exemple initial est audible immédiatement avec Écouter ou les touches du clavier. Dessiner ajoute un trait : l’axe horizontal représente le temps, l’axe vertical la fréquence. Le pinceau choisit les rapports de partiels. La force règle leur amplitude ; un stylet ajoute sa pression. Les traits sont persistants jusqu’à une modification ou un rechargement, mais il n’y a pas de sauvegarde automatique : exporter le JSON avant de fermer.

Choisir une tranche permet de délimiter une durée à écouter seule. Tout voir rétablit l’ensemble du document et enlève la tranche. Déplacer la vue fait défiler une vue agrandie. Deux doigts déplacent et agrandissent le temps ; leur séparation horizontale commande le zoom. Repérer indique la fréquence, la note approximative et le temps sans tracer.

Réglages & fichiers ouvre un panneau qui réduit la place disponible pour la toile. Les commandes ne sont jamais superposées au dessin. Sur téléphone, le panneau occupe au plus 40 % de la hauteur de la fenêtre. La toile demeure visible ; les détails se font défiler à l’intérieur du panneau.

Livraison

Le dossier contient les sources HTML, CSS et JavaScript séparées, une page autonome imagisonne.1.html, les tests, ce rapport MD/PDF et un ZIP. Le site HTTPS est le chemin conseillé pour AudioWorklet et Web MIDI. Un fichier local peut fonctionner selon le navigateur, mais son contexte de sécurité ne garantit pas ces fonctions. Aucun service externe, bibliothèque JavaScript distante ou serveur applicatif ne participe au calcul sonore.

Le document sonore et ses lois

Un trait contient une suite de points (temps, fréquence, amplitude) et un nom de timbre. Les grandeurs sont stockées en secondes et hertz avec des nombres JavaScript, sans quantification à la grille MIDI. Le nom, la durée, l'attaque, le relâchement, l'image de référence et des métadonnées accompagnent les traits.

Interpolation et phase

Entre deux points successifs triés dans le temps, la fréquence et l'amplitude sont interpolées linéairement. Un geste revenant en arrière n'est pas un parcours sonore : le tri en fait une courbe temporelle. À temps pratiquement identique, le dernier point est conservé. Une future version pourra proposer des gestes véritablement multivalués.

$$f(t) = (1-u)f_i + uf_{i+1}, \quad u = \frac{t - t_i}{t_{i+1} - t_i}$$

$$\varphi_{n+1} = \varphi_n + 2\pi\frac{f_n}{F_s} \pmod{2\pi}$$

$$x_n = C\,E_n \sum_j a_j(n)\,b_j(n)\,q_j(n)\sin\varphi_j(n)$$

Fs est la cadence d'échantillonnage ; E est l'enveloppe générale. b adoucit les extrémités d'un trait sur 12 ms à gauche et 25 ms à droite. q atténue les partiels entre 45 % et 49 % de Fs, puis les supprime. C vaut 0,65 divisé par la racine du nombre de pistes partielles. Ce choix réduit les écarts de niveau, sans constituer une normalisation perceptive.

Le calcul utilise des nombres flottants double précision avant l'écriture dans la sortie float32 du navigateur. Math.sin reste une fonction numérique ordinaire : le prototype ne prétend pas produire un étalon métrologique indépendant du CPU. La suppression des porteuses trop hautes ne garantit pas un signal parfaitement sans repliement : les modulations et la saturation douce peuvent encore créer des composantes supplémentaires.

Quatre pinceaux

Sinus pur : rapport 1. Harmoniques doux : 1, 2, 3, 4 avec poids 1, 0,35, 0,16, 0,07. Harmoniques impairs : 1, 3, 5 avec poids 1, 0,33, 0,15. Cloche inharmonique : 1, √2, 2,71, 4,07 avec poids 1, 0,5, 0,25, 0,12. Le dessin affiche le geste principal, pas toutes les trajectoires des partiels produits.

Échelles : voir et transformer

La hauteur normalisée h vaut zéro en bas et un en haut. Trois lois réversibles couvrent de 80 à 4 000 Hz. Ce sont des lois d’affichage ; le document garde ses fréquences physiques.

$$f_{\text{lin}}(h) = 80 + 3920h$$

$$f_{\text{log}}(h) = 80 \times 50^{h}$$

$$f_{\text{quad}}(h) = 80 + 3920h^{2}$$

Changer l’échelle déplace donc les traits sans changer le son. Le bouton Réinterpréter avec la nouvelle échelle applique volontairement la loi nouvelle aux hauteurs qu’occupaient les traits dans l’échelle précédente. Il s’agit alors d’une modification du document, inscrite dans l’historique.

$$f_{\text{nouveau}} = L_{\text{nouvelle}}\left(L^{-1}_{\text{ancienne}}(f)\right)$$

Un second outil rapproche ou éloigne les fréquences autour de Do central. Il agit sur la fréquence principale de chaque trait ; le pinceau conserve ensuite ses rapports de partiels. Ce n’est donc pas encore une compression individuelle de tous les partiels internes du timbre.

$$f' = f_0\left(\frac{f}{f_0}\right)^{\alpha}, \qquad f_0 = 261.625565 \text{ Hz}$$

Pour $0 < \alpha < 1$, les fréquences se rapprochent de f0 ; pour $\alpha > 1$, elles s’en éloignent. Ces transformations sont bornées à 20–20 000 Hz. Les fréquences hors de la fenêtre visible restent dans le document. La loi quadratique inverse rabat les valeurs inférieures à 80 Hz sur sa borne : sa réinterprétation hors domaine n’est pas bijective.

Image et son

L’image importée est réduite à 1 024 pixels au plus sur son grand côté, puis enregistrée en JPEG. Elle est uniquement un fond visuel de référence, indépendant de l’échelle fréquentielle. Elle n’est ni analysée ni synthétisée. L’ajout ultérieur d’un bitmap sonore exigera sa propre loi de correspondance et une distinction explicite entre changement de vue et réinterprétation.

Exécution, clavier et échange d’instruments

Répartition du travail

Fil principal CPU : interface, gestes, historique, import, dessin Canvas 2D. AudioWorklet : synthèse échantillon par échantillon, phases et enveloppes indépendantes pour chaque note. Worker : rendu hors temps réel et encodage WAV, sans bloquer volontairement le dessin. Le navigateur peut accélérer Canvas, mais aucune synthèse GPU, aucun WebGL et aucun WebGPU ne sont requis.

moteur.js contient les lois, la validation et VoixImagisonne. audio.js crée les contextes audio et Worker à partir des mêmes fonctions de calcul. atelier.js contient les interactions et expose window.imagisonne pour l’inspection dans le débogueur. Les fichiers sont des scripts classiques : cette première livraison ne dépend pas de la configuration MIME .mjs du serveur.

Note tenue et écoute

Au clavier, Do central conserve le dessin original ; les autres notes multiplient les fréquences par 2 puissance (nombre de demi-tons / 12), sans changer la durée. L’attaque est une rampe. À 65 % de la durée, le curseur de synthèse se fige : les phases continuent d’évoluer au spectre atteint. Si aucun trait ne traverse cet instant, le maintien est silencieux. Le relâchement diminue l’amplitude jusqu’à zéro.

Écouter lit le son ou la tranche sélectionnée depuis un état neuf. Aucun contexte musical antérieur n’est reconstruit. Le trait possède aussi ses fenêtres propres, donc une fin de trait peut devenir silencieuse avant la fin de l’enveloppe générale. Les modifications du dessin ne remplacent pas les données des notes déjà démarrées ; une nouvelle note reçoit le document courant.

Web MIDI est demandé seulement après clic. Les notes et les commandes All Notes Off / All Sound Off sont reçues ; vélocité, pitch bend et pédale ne sont pas encore interprétés. Une perte de focus ou une déconnexion arrête les notes. La polyphonie est limitée à huit notes : une nouvelle note est ignorée tant que les huit places sont occupées, relâchements compris.

Fichiers

L’instrument JSON version 1 conserve les points, timbres, paramètres, image et métadonnées. L’import vérifie les valeurs et les tailles. Ce format est documenté par le code et peut servir à un adaptateur futur de composition ; il n’est pas présenté comme un standard Cyclonium définitif. Le WAV mono 48 kHz est en IEEE float32 et possède une trace RIFF/INFO : atelier, version, date, nom encodé et volume. Pour une reprise éditable, conserver aussi le JSON. La sortie applique tanh au mélange : protection douce susceptible de colorer les niveaux élevés.

Vérifications et suite de la refonte

Vérifié automatiquement

Les tests Node valident les inverses des échelles, le rejet de données non finies, un signal à 440 Hz par comptage de passages par zéro, l’absence de NaN lors d’un glissando, le maintien et le relâchement, la suppression d’une porteuse hors bande. Le code AudioWorklet est exécuté dans un contexte simulé ; le Worker réalise un export complet dont l’en-tête est contrôlé.

Un second test simule le DOM : démarrage, treize touches, ajout de trait, annulation/rétablissement, changement d’affichage sans modification du document, réinterprétation annulable, abandon du trait au second doigt et ouverture/fermeture du panneau. Ce test ne mesure pas les dimensions réelles d’un téléphone ni le rendu de son navigateur.

Limites connues

La validation dans un vrai navigateur, sur écran tactile et avec une sortie audio ou un clavier MIDI physique reste à faire. La fluidité avec beaucoup de traits dépend du matériel. Les bornes actuelles sont 64 traits, 1 500 points par trait, 12 secondes et huit notes simultanées. L’historique garde 30 états. Une mauvaise performance au maximum ne doit pas être interprétée comme une garantie temps réel.

Il n’y a pas encore de gomme sélective, déplacement de points, pinceau de bruit/transitoire dédié, conversion sonore bitmap, mode harmonique de composition, enregistrement audio externe, bibliothèque multi-instruments ni connexion automatique avec les anciens ateliers. Les deux doigts ne zooment que le temps. La vue fréquentielle conserve ses bornes fixes.

Prochaine étape utile

Faire jouer ce prototype sur le téléphone cible, puis décider des gestes d’édition et de la sémantique du maintien. Ensuite, extraire un adaptateur d’instrument partagé, puis ajouter la synthèse bitmap avec une loi de fréquence sauvegardée. La composition doit importer une référence/version d’instrument, sans devenir le même document que la lutherie.

Repères techniques

Documentation : MDN, AudioWorkletProcessor.process ; MDN, Web MIDI API ; MDN, Web Workers API. Le principe de la phase intégrée est décrit dans le présent rapport. Aucun accès aux anciennes archives ou sources non présentes n’est supposé pour cette implémentation.

Tests : node tests.cjs puis node tests-interface.cjs. Reconstruction de la page autonome et du ZIP : python construire.py. Reconstruction du PDF : python rapport.py (ReportLab et Matplotlib).