Calamus Input Engine: Specification v1.2

This document was created at the very beginning of the process. Most concepts now have their own document specification and may have evolved from the description given here.

Vision

The Input Engine is Calamus's notation and composition interface — a system for writing music that captures gesture, displays parameters visually, and communicates with the sound engine. It breaks from traditional staff notation while preserving the familiar: pitch vertical, time horizontal, notes on lines.

Core philosophy: The scale is the infrastructure. The shape is the compositional atom.

The Paradigm Shift

Traditional Staff Limitations

The conventional staff takes the chromatic gamut as fixed infrastructure:

- Lines and spaces represent a 12-note chromatic assumption
- Scales are mentally filtered from this grid
- Spacing is uniform regardless of actual interval size
- Expression markings are afterthoughts (dynamics, articulation symbols)
- The score captures almost nothing of how music actually sounds

Calamus Approach

The scale itself becomes the infrastructure:

- Lines *are* the scale degrees
- Spacing reflects actual pitch intervals

- Color encodes harmonic function
- Note shapes show parameter evolution
- Gesture is captured, not approximated

A traditionally-trained musician sees familiar grammar. Then discovers capabilities that didn't exist before.

Organic Without Random: Input Engine Implications

The foundational principle "Organic Without Random" shapes how the Input Engine handles musical elements. Non-linearity emerges from physics and easing functions, never from noise.

Rhythmic Physics

Musical time elements can have mass/inertia:

Tempo: Heavy mass = resists tempo changes, has momentum, overshoots slightly when accelerating or decelerating. Light mass = responsive, tight.

Timing offset: Individual notes or beats can pull early or late based on physics, creating rubato that emerges from rules rather than being programmed note-by-note.

Duration: Notes can stretch or compress based on mass — heavy notes sustain longer, light notes are crisp.

Each element can have its own easing function and mass setting. The composer chooses what to apply where.

Harmonic Physics

Harmonic movement can also have physical properties:

Tension as mass: Heavy = harmony accumulates complexity (diminished 9ths, augmented 13ths appear as "overshoot"). Light = harmony stays pure, gravitates toward consonance.

Harmonic function weighting: Different mass for tonic vs. fifth vs. third — each scale degree responds differently to gestural force.

Attraction basins: Harmonic targets act as attractors. The path to resolution has freedom; the destination has tendency.

This opens a realm of harmonic exploration: you apply force through gesture, and complexity emerges from physics rather than being chosen chord by chord. Edit afterward if the overshoot lands somewhere unwanted.

Easing Functions for Musical Movement

A library of trajectory shapes governs how parameters move from state A to state B:

Elastic: Overshoots target, bounces back, settles **Bounce:** Multiple rebounds before settling **Spring:** Oscillates around target with decay **Wobble:** Irregular oscillation **Smooth step:** S-curve, gentle acceleration/deceleration

Mathematical: Linear, quadratic, cubic, sinusoidal, exponential

Variations: Each function has ease-in, ease-out, ease-in-out variants

Custom: User-defined parameters, chained segments

These apply to:

- Pitch glide between scale degrees (discrete mode snaps, but *how* does it snap?)
- Dynamics curves (crescendo/decrescendo shapes)
- Tempo changes within phrases
- Formant trajectories (consonant character)
- Any parameter transition

The Prescribed-Emergent Spectrum

PRESCRIBED <----------> EMERGENT

At prescribed: The easing function defines exactly what happens. Repeatable, sculptable, predictable.

At emergent: Physics simulation only — mass, springs, damping. The system finds its own path within constraints.

In between: Easing shapes the tendency, physics adds life on top.

This can be set globally or per-parameter. A piece might have elastic rhythm, prescribed dynamics, and emergent harmonic movement — composer's choice.

The Visual Notation System

Scale-Centric Staff

Aspect	**Traditional**	**Calamus**
Lines represent	Fixed chromatic positions	Scale degrees
Spacing	Uniform	Proportional to interval
Tuning	12-TET assumed	Any system native
Scale visibility	Mental filtering required	Direct — lines ARE the scale

The staff encompasses all registers of all active instruments. Lowest line = lowest frequency. The vertical dimension maps directly to pitch space.

Color Coding

Line colors indicate harmonic function:

- Tonic (to be defined — likely most prominent)
- Fifth
- Third
- Other scale degrees

Background color indicates key/tonal center:

- Changes visually at modulations
- Enables unprecedented capability: tuning system modulation mid-piece
- Moving from just intonation to maqam as smoothly as a key change

Note Shapes

Notes are not dots or ovals. They are **visualizations of parameter evolution**:

- Shape shows how a chosen parameter changes over the note's duration
- Length shows duration
- Selectable parameter: amplitude (waveform-like), spectrum, brightness, others

The note's visual form *is* its expressive content — what traditional scores leave to performer interpretation.

Input Methods

The Wacom Art Pen

Primary input device with six continuous dimensions:

Dimension	**Range**	**Primary Mapping**
X position	Continuous	Time (in drawing mode)
Y position	Continuous	Pitch
Pressure	2048+ levels	Assignable
Tilt X	~60°	Assignable
Tilt Y	~60°	Assignable
Rotation	360°	Assignable

Six simultaneous control streams from a single hand gesture — more than a violin bow.

Two Capture Modes

Drawing Mode:

- X = time, Y = pitch
- You draw a curve directly in pitch-time space
- Fast horizontal stroke = long note
- Vertical movement = melodic motion
- The curve IS the melody

Recording Mode:

- Time scrolls automatically (tempo-determined)
- You control Y (pitch) as time advances
- Curve trails behind like a seismograph
- You see it emerge as you play

Pitch Interpretation

After capture, the curve's relationship to the scale-lines can be:

Discrete: Curve snaps to lines. Hand draws gesture, system provides scale pitches. Muscle memory of shape, precision of tuning. The *easing function* determines how the snap behaves — instant, elastic, bouncy.

Continuous: Curve is literal. Glissando, portamento, microtonal inflection. Lines become reference, not constraint.

This choice can be:

- Made during capture (e.g., pressure determining mode)
- Applied after capture (same gesture, different interpretation)
- Varied within a single line

Visual Channels

A curve can carry multiple simultaneous visual properties:

Channel	**Potential Mapping**
Position (X, Y)	Time, Pitch
Thickness	Pressure/Dynamics
Color hue	Voice identifier
Color saturation	Assignable parameter
Color brightness	Assignable parameter
Opacity	Assignable (with care — see multi-voice)
Edge quality	Sharp vs. feathered — assignable

Each pen input dimension can map to a visual property, and each visual property to a sound parameter. **The drawing shows what it will sound like.**

Time Handling

Horizontal axis = time. Three modes:

Regular division: Traditional meters (4/4, 3/4, 6/8, etc.)

Irregular patterns: Repeating cycles of different divisions. Example: 5/4 + 6/7 + 7/3 repeating. The system accommodates complex metric structures.

Free flow: No grid. Millisecond precision. Rubato and natural timing captured exactly as performed. No quantization unless requested.

With physics: Any mode can have mass/inertia applied. Regular division with heavy mass creates organic stretch/compression without losing the underlying pulse.

The Curve System

Curves as Compositional Atoms

The shape has always been the compositional atom. Calamus gives it explicit existence.

Curves can be:

- Captured from pen gesture
- Generated from mathematical functions (sine, Bezier, Lissajous, fractals)
- Generated from easing functions (elastic, bounce, spring, wobble)
- Saved as reusable motifs
- Rebound to different parameters (same shape, different meaning)
- Transformed (invert, mirror, stretch, compress)

Hierarchical Curves

Curves operate at every temporal scale:

Level	**Scope**	**Example**
Macro	Entire piece	20-minute arc of tension/release

Level	Scope	Example
Section	Movement/part	Each section's shape within the macro
Phrase	Breathing unit	Natural musical sentences
Note	Individual gesture	Single expressive mark

Fractal potential: The same curve shape at every level. The piece breathes as a note breathes. Self-similarity as compositional principle.

Physics at every level: Each hierarchical level can have its own mass/inertia settings. Macro-level curves might be heavy (slow to change direction), note-level curves light (responsive).

Curve Modulation

Curves interact across levels:

- A phrase curve operates *within* its section curve
- Multiplication/interaction, not replacement
- Lower-level curves ride on higher-level envelopes

Single Canvas Approach

Unlike traditional scoring with stacked staves:

- All voices share one pitch-frequency canvas
- Total vertical range = all registers of all instruments
- No page turns through stacked staves
- See the counterpoint directly — where lines meet, where they diverge

---

The Left Hand

The computer keyboard serves as command center (unchanged from original spec):

Functions:

- Voice selection (1-4)
- Mode toggles
- Memory/accumulation sculpting
- Navigation (zoom, scroll)
- Shortcuts and modifiers
- Easing function selection (TBD)
- Mass/inertia adjustment (TBD)

Not static — actively sculpts instrument state during composition. Specific mappings defined during implementation.

Editing Paradigm

After capture, gestures are fully editable. The vocabulary includes (at minimum):

Basic operations:

- Select
- Move
- Copy
- Delete
- Split

Transformations:

- Stretch/compress (time or pitch)
- Invert (pitch mirror)
- Mirror (time reverse)
- Nudge (fine adjustment)

Extended (as useful):

- Squiggle, squish — playful deformations
- Mathematical transforms
- Curve substitution (replace gesture with easing function)
- Apply physics retroactively (add mass to captured gesture)

The editing paradigm is open — whatever can be usefully applied to curves, will be.

Navigation

Standard capabilities:

- Zoom in/out (see detail or overview)
- Scroll forward/back through time

Primarily controlled by left hand (keyboard shortcuts, possibly mouse wheel).

New Capabilities Summary

Traditional Limitation	**Calamus Opens**
Chromatic grid assumed	Scale-native, any tuning system
Key changes only	Tuning system modulation mid-piece
Notes clicked/placed	Gesture drawn, shape primary
Grid-quantized time	Millisecond precision, captured rubato
Expression unmarked	Parameter evolution visible in note shape
Form implicit	Hierarchical curves make structure explicit
Discrete pitches	Continuous pitch, discrete optional
Static score	Dynamic visualization of sound parameters
Mechanical timing	Physics-based rhythm with mass/inertia
Random humanization	Organic behavior from easing functions

Bridge to Sound Engine

The Input Engine captures and visualizes. The Sound Engine renders to audio.

Interface point: The curves and shapes carry parameter meaning. The Input Engine outputs gesture data that the Sound Engine interprets through its physics model.

Shared vocabulary: Both engines use the same physics concepts — mass, inertia, easing functions, attractors. A curve captured with "elastic" behavior in the input engine produces sound with elastic character.

Technical Platform

Language: C++

UI Framework: Qt

Audio: RtAudio (cross-platform, low-latency)

Build: CMake

Target platforms: Windows, macOS, Linux

Escape hatch: If specific components need different treatment, JUCE remains available for audio-specific modules.

Design Principles

1. Familiar on-ramp, unprecedented capabilities
2. Scale is infrastructure, not overlay
3. Gesture precedes quantization
4. Visual form reflects sonic content
5. Curves at every scale — from note to piece
6. Organic without random — physics and easing, never noise
7. Open editing vocabulary
8. Documentation before implementation
9. Step by step — build, test, celebrate, repeat

What the Input Engine Is NOT

- Not a real-time performance interface (though sketching is possible)
- Not grid-locked (unless you want it)
- Not chromatic-assumed
- Not click-to-place composition
- Not separated from expression
- Not random — organic behavior emerges from physics, not dice

---

The Name, Revisited

Calamus — the reed and the pen unified.

The Input Engine is the pen: where gesture becomes notation, where shape becomes sound.

Document version: 1.1 Original: December 2025 Updated: December 2025 — Added Organic Without Random section, easing functions, rhythmic/harmonic physics