MCP SERVER | NO CODE | CLOUD HOSTED

# Interval Calculator MCP

## Know the exact math and theory behind every note combination.

Interval Calculator MCP helps musicians and theorists determine the exact relationship between musical notes. You can calculate raw semitone distances, figure out if an interval is consonant or dissonant, and find the precise frequency (in Hz) of any note using standard tuning systems.

A+ Quality Score 100/100

music | intervals | frequency | theory | tuning

MAINTAINED BY VINKIUS CLOUD FOR AI AGENTS · JUNE 2026
vinkius.com/mcp/interval-calculator

SCAN TO VIEW

# The infrastructure that powers AI agents in the real world.

Vinkius connects AI to the world's software through secure, enterprise-grade infrastructure — enabling real-world execution at scale, built on the Model Context Protocol (MCP).

# Your AI Connections Run Through Vinkius Cloud

The world's largest managed MCP catalog

Vinkius is the cloud infrastructure where AI agents connect to the software your business already runs. We handle the hosting, the security, the credentials, the uptime — you get agents that actually do things.

We operate the world's largest managed MCP catalog. Major SaaS platforms, CRMs, databases, and cloud providers — running, monitored, production-ready. This MCP server is hosted and maintained by the Vinkius Cloud for AI Agents.

> *The agent doesn't manage credentials, doesn't manage uptime, doesn't manage security. Vinkius does.*
>
> — Architecture principle

## Four Pillars of the Vinkius Runtime

**01 — Security by design**

Credentials stay encrypted at rest via AES-256. The AI agent never touches raw keys — they're injected into a sandboxed V8 isolate at runtime. Actions are logged, and connections have an emergency kill switch.

**02 — Built on MCP Fusion**

This MCP server was built with **MCP Fusion**, the open-source framework (Apache 2.0) that powers the entire Vinkius catalog. Schema-as-firewall strips undeclared fields, compiled PII redaction runs at zero overhead, and cryptographic lockfiles produce git-diffable audit trails.

**03 — Deterministic observability**

Eight immutable metrics per endpoint: request volume, p95 latency, error rate, active connections, cost attribution. A live payload feed logs every tool call with mutation detection.

**04 — Autonomous operations**

Servers are deployed, monitored, and patched autonomously. New capabilities and security patches ship weekly. Zero-downtime deployments ensure continuous availability across all managed MCP servers.

| AES-256 | Ed25519 | 24h TTL | V8 Isolate |
|---|---|---|---|
| Encryption at rest | PKI vault signatures | Ephemeral session keys | Sandboxed execution |

## One Token. Instant Access.

Every MCP server on Vinkius is accessed through a **Connection Token**. Tokens are generated in the cloud dashboard and produce a unique MCP endpoint URL. Paste this URL into any MCP-compatible client — no SDK required.

A single token can serve **multiple AI clients simultaneously**, or you can issue separate tokens per client for granular access control. Each token tracks its own request count, last activity timestamp, and can be individually enabled or revoked.

MCP ENDPOINT

```
https://edge.vinkius.com/{token}/mcp
```

Claude · Cursor · VS Code · Windsurf · Grok · Gemini

## Security Is the Architecture

Security in Vinkius is not a feature — it's the foundation of the runtime. The gateway enforces multiple independent protection layers between AI agents and third-party APIs.

**01 — Ed25519 PKI Vault**

Every workspace has an Ed25519 Master Key. Session keys are generated ephemerally (24h TTL) and signed by the Master Key. Credentials never leave the vault boundary.

**02 — V8 Isolate Sandboxing**

Tool code runs inside isolated-vm V8 isolates with 64 MB memory caps and per-request timeouts. No filesystem access, no network access except through the SSRF-guarded fetch bridge.

**03 — SSRF Guard**

All outbound HTTP requests are DNS-resolved and validated before execution. Private IP ranges (10.x, 172.16-31.x, 192.168.x, AWS metadata 169.254.x) are blocked at the network layer.

**04 — DLP & PII Redaction**

A ResponseGuard pipeline intercepts every tool response. Configurable redaction patterns strip sensitive fields (emails, SSNs, card numbers) before data reaches the AI agent.

**05 — Cryptographic Audit Trail**

Every request is signed into a SHA-256 hash chain with Ed25519 signatures. Events form a tamper-proof, SIEM-exportable forensic record.

**06 — Honeytoken Trap System**

Phantom credentials are injected into isolated environments. If a honeytoken is used outside Vinkius infrastructure, the server is quarantined instantly.

# Emergency Kill Switch

EU AI Act Art. 14(1) Compliant

The kill switch is an **emergency halt** mechanism — not a simple toggle. When triggered, it executes three actions atomically:

**01 — Server deactivated**

The MCP server is immediately taken offline across the entire cluster.

**02 — All tokens revoked**

Every connection token is invalidated. Total lockout — reconnection blocked until new tokens are issued.

**03 — WebSocket connections killed**

Active connections terminated via Redis pubsub broadcast. Propagates to every runtime node in the cluster.

## Full Visibility. Zero Guesswork.

The Vinkius cloud dashboard includes a full MCP Governance suite — real-time analytics and security controls for production AI operations.

**Control Plane**

KPI dashboard with request volume, latency, success rate, token consumption, and AI-generated operational briefings.

**Agent Activity**

Which AI clients are connecting, how often, and what they're doing — real-time session tracking.

**FinOps**

Cost tracking per tool, payload compression savings, budget optimization signals, and consumption trends.

**Tool Health**

Slowest and most error-prone tools, with actionable root-cause insights and performance baselines.

**Firewall & DLP**

PII redaction activity, sensitive data protection counters, and security event timeline.

**Incident Log**

Error trends, failure rates, status-code breakdowns, and forensic audit trail access.

> Get started at **cloud.vinkius.com** — connect your AI agent in under 60 seconds.

# Interval Calculator MCP

3 tools available
Cloud-hosted on Vinkius

Need to know exactly how far apart two notes are? This MCP handles all the math for music theory. You don't just get a number; you figure out the musical character—is it a perfect fifth, or something more unstable? It determines interval qualities and stability like consonant versus dissonant. Plus, if you need the actual physical vibration frequency of a note in Hz, this MCP calculates that too, supporting both A4=440Hz and A4=432Hz tuning standards. Instead of flipping between separate calculation sheets or searching dense theory texts, your agent handles all these calculations instantly. Vinkius hosts this MCP so you connect once through any compatible client and get access to powerful tools like this one. It's pure, reliable musical data at your fingertips.

## Core Capabilities

**01 — Determine interval character**

Identifies the specific name and stability (consonant or dissonant) of a given musical interval.

**02 — Calculate raw semitone distance**

Finds the precise numerical difference in half steps between any two specified notes.

**03 — Determine physical note frequency**

Computes the actual vibration frequency, measured in Hertz (Hz), for a single given pitch.

## One Click on Vinkius — From Prompt to Execution

Available at **vinkius.com/mcp/interval-calculator** — connect your AI agent in three steps.

01 You input two specific notes and their octaves (e.g., C4 and G4) into your agent.

02 The MCP runs the calculation, determining both the raw semitone distance and the formal interval properties of those pitches.

03 It returns a structured report detailing the interval's name, stability classification, and the total number of half steps moved.

The bottom line is you get precise, multi-layered musical data without doing any arithmetic yourself.

## Built For

Anyone dealing with music theory or audio composition needs this. It’s for the composer stuck on a chord progression that won't resolve, the student needing to verify interval definitions quickly, or the sound designer who must match exact frequencies. This MCP cuts through ambiguity and gives you hard data.

**Music Theorist**

Needs to verify complex chord voicings or analyze historical musical pieces by determining precise interval qualities and their mathematical relationships.

**Audio Engineer**

Must match specific frequencies (Hz) for synthesis, tuning, or restoration work while confirming the semitone distance between notes in a patch.

**Composer/Songwriter**

Is exploring new chord structures and needs quick feedback on whether an interval they've paired up sounds consonant or dissonant to guide their creative choices.

## What Changes When You Connect

01 Determine musical character instantly. Instead of flipping through a textbook to classify an interval, use `identify_interval_properties` to know if D4 to F4 is Minor 3rd or something else. It tells you how stable that sound will be.

02 Get raw, verifiable distances between notes. Need to calculate the exact half-step jump from C4 to G4? The `calculate_semitone_distance` tool gives you the pure number—7 semitones—so you can use it in any calculation.

03 Match precise frequencies for synthesis work. If your project requires A4 at 432Hz, the `calculate_note_frequency` function handles that tuning standard and returns the exact Hz value you need.

04 Avoid guesswork in composition. By running an interval through this MCP, you move past vague feelings about sound and get concrete data on consonance or dissonance.

05 Compare multiple tuning systems easily. This MCP supports both A4=440Hz and A4=432Hz standards for frequency calculations, letting you compare the same note across different historical tunings.

## Real-World Applications

**Analyzing an antique piece of music**

A musicologist is analyzing a 17th-century manuscript and needs to know if a specific chord voicing was intended to be dissonant or consonant. They ask their agent about the interval between D4 and F4, getting immediate confirmation that it's classified as 'Dissonant.' This lets them accurately describe the composer's intent.

**Setting up a synthesizer patch**

An audio engineer is designing a complex arpeggiator patch and needs to ensure every note hits an exact frequency. They use `calculate_note_frequency` for their key notes, guaranteeing that the physical vibration matches the required 432Hz standard before committing the sound design.

**Writing a song with specific intervallic jumps**

A songwriter wants her melody to jump exactly seven semitones. She uses `calculate_semitone_distance` between C4 and G4, confirming the exact numerical distance. This allows her to build harmonic movement based on hard mathematical facts.

**Comparing modern vs. historical tuning**

A composer wants to hear how a standard key sounds when tuned using 432Hz versus the common 440Hz pitch. They run A4 through `calculate_note_frequency` for both standards, getting two distinct Hz values and deciding which tone best fits their piece.

# Patterns to Avoid

### Guessing interval qualities

✗ AVOID

Thinking that because a note sounds 'nice' together, it must be consonant. This can lead to incorrect theory and composition.

✓ INSTEAD

Don’t trust your ears alone. Use `identify_interval_properties` to get the official classification of the interval—it tells you if it is mathematically classified as Consonant or Dissonant.

### Using only one frequency standard

✗ AVOID

Assuming that every musical instrument and piece was tuned to 440Hz, when in fact many historical pieces used different tunings.

✓ INSTEAD

Check both standards. Use `calculate_note_frequency` with A4=440Hz and then run it again using A4=432Hz to get the full picture for your composition.

### Calculating only pitch names

✗ AVOID

Knowing that the distance is between C and G, but not knowing if they are exactly seven semitones apart or if octaves change the calculation.

✓ INSTEAD

Always provide both notes and their octave numbers. `calculate_semitone_distance` gives you the precise numerical difference in half steps every time.

# The Right Fit

Use this MCP if your problem requires concrete, quantifiable data about pitches—specifically semitones, frequencies, or formal interval names. You need to know *why* two notes sound a certain way, not just that they do. If you're studying theory, checking for consonance/dissonance, or engineering synthesis patches based on Hz values, this is your tool. Don’t use it if you simply want

suggestions for 'pretty-sounding' chords; those require creative input, not math. Also, don't use it just to name notes—use the `calculate_semitone_distance` tool instead when you need a raw number.

## The Manual Pain of Music Theory

Right now, if you need to know the relationship between two pitches, you're either cross-referencing dense textbooks or running through multiple online calculators. You calculate the distance in semitones, then you check a second chart for the formal name (Major 3rd?), and finally, you have to manually look up the precise frequency for your synthesizer patch, constantly copy-pasting values between tabs.

With this MCP, all that manual cross-referencing vanishes. You ask your agent for the interval data, and it returns a single source of truth: semitones, formal name, stability classification, *and* the correct Hz frequency—all in one step.

## Interval Calculator MCP Provides Full Pitch Analysis

You stop having to juggle three different kinds of data points: raw number distances from `calculate_semitone_distance`, musical labels from `identify_interval_properties`, and physical Hz values from `calculate_note_frequency`. The system gathers it all for you.

The result is immediate, comprehensive, and always accurate. It’s the single source of data every composer or theorist needs.

# Interval Calculator MCP with 3 Tools

These tools let you perform complex calculations—from finding raw semitones between pitches to determining the exact frequency in Hertz.

| # | TOOL | DESCRIPTION |
|---|---|---|
| 01 | `identify_interval_properties` | Provides a description of the musical identity and character—like whether it's Major 3rd—of any given interval. |
| 02 | `calculate_note_frequency` | Determines the specific physical vibration frequency (in Hz) for a note, supporting multiple tuning standards. |
| 03 | `calculate_semitone_distance` | Finds the total raw numerical distance in semitones between two specified pitches and octaves. |

# See It in Action

Real prompts you can use once this MCP is connected to your AI agent through Vinkius Cloud.

**U** What is the distance in semitones between C4 and G4?

The distance between C4 and G4 is 7 semitones.

**U** Tell me about the interval between D4 and F4.

The interval from D4 to F4 is a Minor 3rd, which is classified as Dissonant. Its inversion is an Augmented 6th.

**U** What is the frequency of A4 using 432Hz tuning?

The frequency of A4 with a 432Hz tuning standard is 432.0 Hz.

# Frequently Asked Questions

01 **How do I find the semitone distance using Interval Calculator MCP?**

Use the `calculate_semitone_distance` tool and provide two notes along with their octaves (e.g., C4, G4). It returns the raw numerical difference in half steps.

02 **Does Interval Calculator MCP handle different tuning standards?**

Yes. The `calculate_note_frequency` tool supports determining note frequencies using both A4=440Hz and A4=432Hz tuning systems, giving you versatile data.

03 **What is the difference between calculating distance and identifying properties?**

Calculating distance (`calculate_semitone_distance`) gives a raw number (e.g., 7). Identifying properties (`identify_interval_properties`) tells you what that interval means musically, like 'Perfect 5th,' and whether it's consonant.

04 **Can Interval Calculator MCP give me the frequency for two notes?**

You must calculate them separately. Use `calculate_note_frequency` once for each note you want to analyze, ensuring you specify the correct tuning standard (432Hz or 440Hz).

05 **Does this MCP work with any AI client?**

Since it's an MCP hosted on Vinkius, your agent can connect to it using any compatible client application that supports the Model Context Protocol.

# Go Live in 60 Seconds

Get your connection token from **cloud.vinkius.com**, then paste the endpoint URL into any MCP-compatible client.

YOUR MCP ENDPOINT

```
https://edge.vinkius.com/[TOKEN]/mcp
```

| CLIENT | WHERE TO CONFIGURE |
|---|---|
| Claude AI | Profile → Customize → Connectors → "+" → Add custom connector → Paste endpoint |
| Cursor | Settings → Features → MCP Servers → "+ Add New MCP Server" → Type: SSE → Paste endpoint |
| VS Code | Ctrl/Cmd+Shift+P → "MCP: Add Server" → add `"interval-calculator": { "url": "…" }` |
| Windsurf | MCP Settings → `mcp_settings.json` → Add endpoint URL |
| ChatGPT | Settings → Tools & plugins → Add MCP server → Paste endpoint |
| Gemini | Extensions → Add MCP Server → Paste endpoint URL |

## ASK AN AI ABOUT THIS

Let your preferred AI explain this MCP server

- Ask ChatGPT
- Ask Claude
- Ask Perplexity
- Ask Gemini
- Ask Grok

READY TO CONNECT

# Interval Calculator is live on Vinkius Cloud.

Get your connection token, paste it into your AI agent, and start building. No SDK. No deployment. Just results.

**Start at cloud.vinkius.com →**

vinkius.com · support@vinkius.com

## INDEPENDENT PLATFORM DISCLAIMER

Vinkius is an independent platform and is not affiliated with, endorsed by, sponsored by, verified by, or otherwise authorized by Interval Calculator. All third-party trademarks, logos, and brand names are the property of their respective owners. Their use in this document is strictly for informational purposes to identify service compatibility and interoperability.

## DOCUMENT INFORMATION

| | |
|---|---|
| Generated | June 2026 |
| MCP Server | Interval Calculator MCP |
| Server ID | `019f010e-24f3-7394-a873-60cbb224a447` |
| Platform | Vinkius Cloud for AI Agents |
| Endpoint | `https://edge.vinkius.com/{token}/mcp` |

## LICENSE & USAGE

This document is generated automatically by the Vinkius PDF Engine. Content reflects the MCP server configuration at the time of generation and may change as updates are deployed. For the most current information, visit vinkius.com/mcp/interval-calculator.