MCP SERVER | NO CODE | CLOUD HOSTED

SurrealDB MCP

Control your data layer from natural conversation.

SurrealDB connects your AI agent directly to a multi-model database instance, letting you manage complex data operations using natural language conversation. Execute raw SurrealQL queries, create new records, update existing tables, and move data in or out of the system without ever leaving your chat interface.

A+ Quality Score 100/100

multi-model | surrealql | graph-database | relational-database | document-store | query-execution

MAINTAINED BY VINKIUS CLOUD FOR AI AGENTS · JUNE 2026
vinkius.com/mcp/surrealdb

SCAN TO VIEW

The connectivity layer between AI and the world's software.

Vinkius sits between AI and every application. All communication passes through Vinkius Cloud via the Model Context Protocol (MCP) — with governance, observability, and security at every layer.

Your AI Connections Run Through Vinkius Cloud

The world's largest managed MCP catalog

Vinkius is the connectivity layer where AI connects 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.

SurrealDB MCP

16 tools available
Cloud-hosted on Vinkius

You can connect your private SurrealDB instance right into your AI client. Instead of writing boilerplate code or opening a separate database console, you talk to your agent about what you need done with your data—and it does it. Need to check if a record exists? Ask the agent. Want to pull every user ID from the 'users' table? Just prompt it. The MCP allows your AI client to run raw SurrealQL statements and handle all the tedious mechanics of database interaction, like creating records or exporting full datasets. If you access this via Vinkius, you get immediate control over a multi-model data layer right where you work. You manage everything from schema inspection using `get_health` to moving entire databases with tools like `import_sql` and `export_data`. It's about treating your database not as an isolated system, but as another service available in conversation.

Core Capabilities

01 — Execute advanced queries

Run complex, raw SurrealQL statements to pull specific data or run custom logic against the database.

02 — Manage record lifecycles

Create new records with a random ID, update specific entries using `update_record`, or delete individual records when they are retired.

03 — Perform bulk data operations

Modify all records in a table at once, or clear out an entire table entirely by selecting or deleting all data.

04 — Inspect database status and health

Check the current operational status of your instance using `get_status` and verify its overall stability with `get_health`.

05 — Migrate and back up data

Move information in or out of your database by importing raw SQL queries or exporting structured datasets.

One Click on Vinkius — From Prompt to Execution

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

01 First, subscribe to this MCP on Vinkius and provide the necessary SurrealDB URL along with your credentials (namespace, database name, and authentication key).

02 Next, connect your preferred AI client—like Cursor or Claude—to the configured endpoint. Your agent now recognizes the full set of available data tools.

03 Finally, simply ask your agent to perform a task, such as 'Select all users who haven't logged in this month.' The MCP executes the necessary database calls and returns the structured results.

The bottom line is that you talk to your AI client like it's an extension of your terminal, giving it direct data access.

Built For

This MCP is for developers and engineers who hate switching context. If you find yourself jumping between a coding environment, a database GUI, and a query console just to verify data, this tool saves you time. It lets you keep your entire workflow inside one chat window.

Backend Developer

You use it to rapidly test complex queries or inspect the current database state without ever leaving your IDE's terminal.

Data Engineer

You manage schema changes and perform quick data migrations, like importing raw SQL into a testing environment.

Product Manager

You query live business metrics directly from the database using natural language prompts instead of waiting for an analyst to run a report.

What Changes When You Connect

01 You stop switching between tabs. Instead of opening a separate database console to test logic, you can execute raw SurrealQL statements using `execute_sql` directly within the chat interface.

02 Data maintenance is faster than manual clicks. Need to clean up old entries? Use `delete_all` or `modify_record` to manage records without writing complex scripts outside your environment.

03 Migration and auditing are simplified. If you need to move data, use `import_sql` to load queries or grab a full dataset using `export_data`, all prompted naturally.

04 You get immediate visibility into the system. Use `get_status` and `get_health` to confirm your instance is up and running before starting any critical tasks.

05 The process of getting data out is straightforward. You can select everything in a table with `select_all`, or just pull one specific piece using `select_record`, then pass that result on.

Real-World Applications

Verifying database schema changes

A data engineer needs to confirm if a new table structure is compatible with old scripts. Instead of running multiple manual checks, they prompt their agent to run `execute_sql` for various queries and check the system's response.

Cleaning up test environments

A backend developer finishes a feature branch and needs to wipe all generated user data. They simply ask the agent to use `delete_all` on the 'test_users' table, confirming cleanup without running SQL in a separate terminal.

Ad-hoc reporting for PMs

A product manager needs to see all posts from last quarter. They ask their agent to `select_all` records from the 'posts' table, getting an immediate list of IDs and metadata without involving the BI team.

Onboarding new data sources

A data scientist has a CSV dump that needs to populate a live database. They use `import_sql` through their agent, feeding the raw queries directly into the MCP for immediate ingestion.

Patterns to Avoid

Treating the AI like a search engine

✗ AVOID

Asking 'How do I modify user records?' and getting only documentation links. The agent won't guess what you want.

✓ INSTEAD

Be specific: 'Use `update_record` to change the email of user:123 to new@email.com.' Use the action name, not a question.

Running complex logic manually

✗ AVOID

Having to write and execute five different raw SQL queries in separate tabs just to compare data across tables.

✓ INSTEAD

Group them into one prompt: 'Run an `execute_sql` query that joins users and orders where the status is pending.' The MCP handles the full logic flow.

Forgetting data scope

✗ AVOID

Trying to delete all records but accidentally deleting a production table because you didn't specify the target table.

✓ INSTEAD

Always confirm the action. For mass deletion, use `delete_all` and explicitly name the table *before* prompting.

The Right Fit

Use this MCP if your core pain point is context switching. If you are a developer or data professional who needs to execute database commands (queries, CRUD operations, schema checks) but hates

opening separate terminals or GUIs, this is for you. The ability to talk to the database via natural language and get immediate action is the value here. Don't use this if all you need to do is *read* public data—you might be better off with a simple read-only API connector. Also, don't use it if your primary goal is running complex machine learning pipelines; for those, a dedicated vector store or specialized agent framework will work better than pure SQL execution.

The Friction of Data Context Switching

Right now, doing anything non-trivial with your database is a multi-step ritual. You open your IDE, you switch tabs to the DB GUI, you write raw SQL into a separate query window, and then copy/paste results back into your main code file just to verify something. It's slow, it's tedious, and half the time you get distracted before you finish.

With this MCP connected via Vinkius, that workflow vanishes. You tell your agent what needs changing—whether it’s running `select_all` or using `create_record`. The entire cycle happens within one conversation window. It's immediate action, not staged process.

SurrealDB MCP: Data Management in Conversation

The manual steps of verifying table names, remembering the correct SQL syntax for updates, or running separate scripts just to check health are gone. Your agent handles the mechanics of `get_health`, `update_record`, and complex queries through `execute_sql` automatically.

You stop thinking about database commands and start focusing on outcomes. The data is simply available when you need it.

SurrealDB: 15 Database Operation Tools

These tools let you perform every common database action—from reading single records to importing entire SQL dumps—all through your AI agent.

#	TOOL	DESCRIPTION
01	`create_specific_record`	Builds and inserts one new, defined record into a target table.
02	`delete_all`	Wipes every single record out of an entire specified table.
03	`delete_record`	Removes one specific, targeted record from a table.
04	`export_data`	Export data
05	`get_health`	Checks the current health and operational status of your SurrealDB instance.
06	`import_sql`	Ingests external data by running raw SurrealQL statements into the database.
07	`modify_all`	Overwrites or updates every single record in a table using merge operations.
08	`modify_record`	Updates one specific, targeted record with new data via merge operations.
09	`select_all`	Retrieves every single record that exists in an entire specified table.
10	`select_record`	Retrieves the details of a single, specified record by its unique ID.
11	`create_record`	Inserts a new record into a table and assigns it a randomly generated ID.
12	`execute_sql`	Runs arbitrary SurrealQL statements, allowing variables to be passed in via URL parameters.
13	`get_status`	Checks the web server status and operational uptime of your SurrealDB instance.
14	`update_all`	Applies updates to every single record in a table, overwriting existing data.
15	`update_record`	Updates one specific, targeted record with new information.

#	TOOL	DESCRIPTION
16	get_version	Returns the exact version number of the SurrealDB software currently running on the instance.

See It in Action

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

U Check the health and version of my SurrealDB server.

I've checked your instance. The server is healthy (200 OK) and currently running SurrealDB version 2.1.0.

U Select all records from the 'users' table.

I've retrieved all records from the 'users' table. There are 3 records: 'user:admin', 'user:john_doe', and 'user:jane_smith'. Would you like to see the details of a specific one?

U Execute a SurrealQL query to find all posts where status is 'published'.

Running the SurrealQL query... I found 12 posts with the 'published' status. The most recent one is 'post:mcp-guide' created on 2023-10-27.

Frequently Asked Questions

01 **How do I run a query using SurrealDB MCP?**

You use the `execute_sql` tool by prompting your agent with the specific SurrealQL statement. The system executes it against your live database instance and returns structured results.

02 **Can I delete all data from a table using SurrealDB MCP?**

Yes, you can use the `delete_all` tool to wipe every record in an entire table. Be careful with this action; it's irreversible.

03 **What is the difference between create_record and execute_sql?**

`create_record` generates a new record with a random ID for simple inserts. `execute_sql` lets you run any complex, raw SurrealQL statement, giving you full control over the query logic.

04 **Does this MCP handle data migration?**

Yes, it handles migration using tools like `import_sql`, which allows you to feed external SQL queries into your instance for ingestion.

05 **What if I only need one specific record? Should I use select_record or execute_sql?**

If you know the exact ID, use `select_record`. If you need to filter records using complex criteria (like 'all active users'), it's better to wrap that logic in an `execute_sql` query.

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 `"surrealdb": { "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

SurrealDB 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 SurrealDB. 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	SurrealDB MCP
Server ID	`019e38f6-6c6b-712f-9baf-7ee14f53045c`
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/surrealdb.