MCP SERVER | NO CODE | CLOUD HOSTED

watsonx Discovery MCP

Search deep documents with plain chat commands.

watsonx Discovery connects your AI agent directly to massive, unstructured data collections. This MCP gives you a cognitive search engine that doesn't just keyword match; it understands natural language and surfaces hidden patterns from documents across your enterprise. Stop wading through complex cloud consoles—just ask questions about your knowledge base.

A+ Quality Score 100/100

cognitive-search | nlp | unstructured-data | semantic-search | text-analytics | enterprise-search

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

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.

watsonx Discovery MCP

6 tools available
Cloud-hosted on Vinkius

You can connect your AI agent to IBM watsonx Discovery and treat your entire document repository like a single, searchable conversation. Instead of manually running queries or digging through technical console dashboards, you simply chat with the system using natural language. The MCP uses advanced text analytics to read everything—from legal contracts to internal reports—and surface only what you need. You can ask complex questions about multiple documents at once and get actionable answers immediately. This capability is hosted on Vinkius, making it easy for your AI client to access deep enterprise knowledge without needing specialized coding skills. It's like having a data scientist who lives inside your chat window.

Core Capabilities

01 — Search across collections

You perform natural language or DQL queries against multiple data sources to find relevant information.

02 — Map project contents

The MCP lists all available data collections and the specific documents within them, helping you understand your scope.

03 — Analyze document structure

You retrieve technical metadata for single indexed files, checking ingestion status or identifying key details.

04 — Check data quality and health

The system verifies project component configurations and monitors the overall health of your discovery environment.

05 — Review applied intelligence models

You list all NLP enrichments, like Sentiment or Entity extraction, to see what type of analysis is running on your data.

One Click on Vinkius — From Prompt to Execution

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

01 Subscribe to this MCP and provide your watsonx URL, API Key, and Project ID.

02 Your AI agent uses the credentials to connect and verify access to your cognitive data collections.

03 You ask a complex question in plain language; the system runs the query and returns targeted answers drawn from your documents.

The bottom line is that you never have to leave your chat window to analyze deep, enterprise-level document data.

Built For

Anyone who spends time sifting through massive document repositories or running repetitive database checks needs this. Specifically, Knowledge Analysts and Data Scientists who are tired of manual console navigation.

Knowledge Analyst

You use the MCP to quickly search across thousands of documents and audit which NLP enrichments (like Sentiment) were applied during ingestion.

Data Scientist

You test and refine complex DQL queries or monitor data ingestion status directly via chat, rather than building dedicated dashboard widgets.

Enterprise Developer

You build grounded AI applications by using the semantic search tools to pull highly relevant context from your company's documents.

What Changes When You Connect

01 Find answers instantly. Instead of manually building complex queries, you simply ask your agent a question like, 'What are the termination requirements for contract X?' and get the answer directly from the data using `query_discovery_content`.

02 Audit your data source easily. Use `list_available_enrichments` to see exactly what kind of analysis (like keyword extraction) has been run on your documents—no more guessing if the data is clean.

03 Track project health in real time. The MCP runs checks using `get_component_settings`, letting you know instantly if an ingestion pipeline failed or needs attention, saving hours of dashboard clicking.

04 Understand your entire scope. Start with `list_discovery_collections` to map out every data set available. This gives you a clear view of everything the system can search before you write a single query.

05 Verify document integrity. If you need to know the status or metadata for one file, use `get_document_details`. It's a quick way to check if a specific record is ready and indexed correctly.

Real-World Applications

Finding obscure contract details

A legal analyst needs to know every document mentioning 'indemnification clause' across three different collections. They use their agent with `query_discovery_content` and the MCP aggregates results from all relevant data sets, providing a consolidated summary they can read immediately.

Onboarding new team members

A new developer needs to know what data sources are available for their project. They simply call `list_discovery_collections`, instantly receiving an inventory of all possible knowledge bases and where to start querying.

Troubleshooting failed pipelines

The data science team notices some documents aren't indexing correctly. They use `get_component_settings` to check the system health, immediately pinpointing which component is failing before having to manually investigate logs.

Checking document readiness

A product manager needs to confirm if a specific policy document was successfully indexed. They use `get_document_details` and instantly get the full metadata and ingestion status, confirming it's ready for search.

Patterns to Avoid

Manual collection mapping

✗ AVOID

The analyst has to navigate the IBM Cloud console, clicking between 'Projects,' then selecting a collection name, and finally opening individual document folders just to list what exists.

✓ INSTEAD

Instead of manual clicks, use `list_discovery_collections` first. This gives you an immediate, clean inventory of all available collections so your agent knows exactly where to search.

Blind querying

✗ AVOID

The developer runs a broad query without knowing if the data is properly enriched or indexed, leading to vague or incomplete results.

✓ INSTEAD

Before querying, run `list_available_enrichments`. This confirms that Sentiment or Entity analysis is active on your documents, ensuring the agent searches using full cognitive context.

Ignoring project health

✗ AVOID

A team assumes their data pipeline is running fine and runs a query, only to get vague results because the underlying component failed weeks ago.

✓ INSTEAD

Always check `get_component_settings` first. This tool verifies the entire project's configuration and notices, ensuring your search isn't hampered by technical failures.

The Right Fit

Use this MCP if your primary problem is finding specific answers hidden deep inside massive amounts of unstructured data—think legal documents, research reports, or internal memos. Your data exists in collections, but the knowledge is buried in text and needs semantic understanding.

Don't use this if you just need to query a simple database table (SQL lookups are better) or if your search criteria can be reduced to exact keywords across a single, small file. If all you need is metadata from one specific record, the `get_document_details` tool handles that; but for broad, conversational searching and analysis of *collections*, this MCP is necessary.

The pain point: Sifting through enterprise consoles

Right now, finding an answer means navigating a maze of cloud dashboards. You have to manually copy collection IDs, switch between tabs to check metadata, and then write highly technical query language just to get started. It's slow, it’s fragile, and it requires knowing the exact internal architecture.

With this MCP connected through Vinkius, that process vanishes. Your AI agent handles all the setup work behind the scenes. You just ask a question in plain English, and you get immediate, context-aware answers without ever touching a technical ID or a console dashboard.

Get Deep Insights with watsonx Discovery

You no longer need to manually run `list_discovery_collections` just to know what data sources exist, nor do you have to check component status using `get_component_settings`. These tedious setup steps are handled by your agent before it even answers.

The difference is that you move from being a data technician who runs commands, to an informed analyst who simply asks questions. The knowledge retrieval happens automatically.

watsonx Discovery: 6 Tools for Data Retrieval

These tools allow your agent to systematically map, analyze, query, and monitor every aspect of your watsonx Discovery project.

#	TOOL	DESCRIPTION
01	`get_document_details`	Pulls metadata and status for a single indexed document, showing its technical details.
02	`get_component_settings`	Retrieves the configuration settings and health metrics for all project components.
03	`list_discovery_collections`	Lists every data collection available in your current watsonx Discovery project.
04	`list_collection_documents`	Provides a list of all specific documents contained within a selected data collection.
05	`list_available_enrichments`	Lists every NLP model, such as Sentiment or Entity recognition, configured for your project.
06	`query_discovery_content`	Executes a natural language question or DQL query against a specific data collection.

See It in Action

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

U List all my Discovery collections.

I found 3 collections in your project: 'Legal Documents' (ID: col-1), 'Technical Support KB' (ID: col-2), and 'Marketing Research' (ID: col-3). Which one would you like to query?

U Search the 'Legal Documents' collection for 'contract termination clauses'.

I found several matches. The most relevant document mentions that termination requires a 30-day written notice and highlights specific liability limitations. Would you like me to pull the full document text?

U What enrichments are currently active in my project?

Your project has 4 active NLP enrichments: 1. Sentiment Analysis, 2. Entity Extraction (People, Places, Organizations), 3. Category Classification, and 4. Keyword Extraction. These are applied to all documents during ingestion.

Frequently Asked Questions

01 **How do I start searching with the watsonx Discovery MCP?**

You first need to provide your specific watsonx credentials and project ID. Once connected, you can use `query_discovery_content` by simply asking a natural language question.

02 **What if I want to know what data sources are available? Do I need the watsonx Discovery MCP?**

Yes, use the `list_discovery_collections` tool. This function gives you an immediate inventory of every collection in your project so you can plan your query.

03 **Can this MCP help me check if a document is ready to be searched?**

Absolutely. Use `get_document_details` on the specific file ID. This tool retrieves the metadata and ingestion status, confirming it's indexed and available for querying.

04 **Does watsonx Discovery help with NLP analysis? Which tools are involved?**

The MCP lists active enrichments using `list_available_enrichments`. This tells you if the document has Sentiment or Entity tags applied, which enhances your ability to query.

05 **I have multiple documents. Can I search them all at once with watsonx Discovery?**

Yes. By using `query_discovery_content`, you can write a prompt that directs the agent to look across several collections simultaneously, consolidating the findings.

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 `"watsonx-discovery": { "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

watsonx Discovery 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 watsonx Discovery. 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	watsonx Discovery MCP
Server ID	`019d761f-9419-7239-a644-fe12cd123c2b`
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/watsonx-discovery.