MCP SERVER | NO CODE | CLOUD HOSTED

# FRED GeoFRED Regional Data MCP

## Map Economic Trends Across US Geography.

FRED GeoFRED — Regional Economic Data connects your AI client to comprehensive U.S. economic metrics. It provides unemployment, income levels, and GDP data broken down by state, county, MSA, or Federal Reserve District. Get cross-sectional regional comparisons and necessary geographic boundaries for detailed analysis.

A+ Quality Score 100/100

regional-economics | geographic-data | demographics | economic-indicators | mapping | spatial-analysis

MAINTAINED BY VINKIUS CLOUD FOR AI AGENTS · JUNE 2026
vinkius.com/mcp/fred-geofred-regional-economic-data

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.

# FRED GeoFRED — Regional Economic Data MCP

3 tools available
Cloud-hosted on Vinkius

If you're building a dashboard that needs to show how economics play out across the country, this MCP is what you need. It takes standard national economic time series data—like unemployment rates or median income—and breaks it down into specific geographical regions. You can pull regional snapshots for any U.S. area type, including states, counties, and metro areas. Plus, it gets you the GeoJSON-compatible shape files required to actually map that data visually. When your agent needs to compare performance across different geographic boundaries or build a spatial analysis piece, this MCP makes that possible. You'll connect through Vinkius, giving your AI client access to thousands of other specialized tools alongside these economic datasets.

## Core Capabilities

**01 — Map Geographic Boundaries**

Retrieve the GeoJSON shape files needed for mapping data across specific U.S. regions.

**02 — Discover Data Availability**

Check which types of geographic breakdowns (like state or county) exist for a given economic metric.

**03 — Get Cross-Sectional Regional Metrics**

Fetch specific regional economic data points, such as unemployment rates or income levels, broken down by geography.

## One Click on Vinkius — From Prompt to Execution

Available at **vinkius.com/mcp/fred-geofred-regional-economic-data** — connect your AI agent in three steps.

01 First, use the get_series_group tool with a FRED series ID (like UNRATE) to figure out what geographic breakdowns and data units are available.

02 Next, call get_regional_data. This sends the specific metric, the desired region type (e.g., county), and any necessary filters for your agent to pull the actual cross-sectional numbers.

03 Finally, if you need to plot this data visually, use get_geo_shapes to retrieve the corresponding boundary files needed for mapping.

The bottom line is that you guide your AI client through three specific calls: discovery of available regions, retrieval of regional metrics, and fetching the map boundaries.

## Built For

This MCP serves real estate analysts and economic policy makers who deal with location intelligence daily. If you're tired of manually pulling data from different spreadsheets or dashboards to compare an MSA's income against a county's unemployment rate, this is for you.

**Regional Economist**

Compares historical economic metrics across state lines, using the MCP to pull cross-sectional data sets that would otherwise require multiple manual database queries.

**Location Intelligence Analyst**

Builds geographic dashboards by pairing regional economic indicators with shape files to visualize trends like median income variation across counties.

**Policy Advisor**

Gathers diverse data points—like unemployment and poverty rates—for specific Federal Reserve Districts (FRB) to inform policy recommendations.

## What Changes When You Connect

01 See regional snapshots of key metrics. You can pull specific data like unemployment, income levels, or poverty rates for any state, county, or MSA using get_regional_data.

02 Validate your data scope instantly. Before pulling figures, use get_series_group to confirm exactly what geographic breakdowns exist for a given economic indicator and its units.

03 Build professional maps quickly. The MCP provides necessary GeoJSON-compatible shape files via get_geo_shapes, allowing you to visualize regional boundaries alongside the pulled metrics.

04 Compare performance across diverse areas. This tool lets you compare regions from various groupings (like BEA or FRB) without switching data sources or spreadsheets.

05 Avoid data gaps in your reports. By checking available region types through get_series_group, you ensure your agent doesn't miss necessary geographic scope when running analyses.

## Real-World Applications

**A Housing Market Analysis**

A real estate consultant needs to compare median income across three specific metro areas (MSAs) and see which one has the lowest housing price index. The agent first uses get_series_group to validate the data, then calls get_regional_data for the metrics, and finally asks for get_geo_shapes to create a visual comparison map.

**Policy Briefing on Unemployment**

A policy advisor needs to show the difference in unemployment rates between states versus counties. The agent uses get_series_group to confirm both region types are available, then runs get_regional_data twice—once for the state level and once for the county level—to create a comprehensive report.

**Academic Research on Poverty**

A researcher needs standardized boundaries for mapping poverty rates across all 50 states. The agent first uses get_series_group to validate the 'poverty' series, then calls get_geo_shapes to get the state outlines, and finally runs get_regional_data to populate the metrics.

**Inter-State Business Comparison**

A corporate strategy team wants to compare GDP growth across different Federal Reserve Districts (FRB). The agent uses get_series_group to confirm FRB is a valid region type, then calls get_regional_data for the specific time period and district grouping.

## Patterns to Avoid

### Assuming data exists by default

✗ AVOID

Trying to run regional analysis directly without knowing if the metric supports county-level breakdowns. This often results in partial or incorrect national averages being returned.

✓ INSTEAD

Always start with get_series_group. It confirms the series ID and tells you exactly which region types (county, msa, state) are supported before attempting to use get_regional_data.

### Using the wrong geographic scope

✗ AVOID

Pulling data for a metro area (MSA) when the analysis actually requires only state-level data. This mixes metrics and makes comparisons inaccurate.

✓ INSTEAD

Use get_series_group first to see if your target region type is 'msa' or 'state'. Then, specify that exact scope in your call to get_regional_data.

### Forgetting the map boundaries

✗ AVOID

Getting a list of numbers for various regions but having no way to visualize where those regions actually are on a map.

✓ INSTEAD

After getting the regional data, run get_geo_shapes. This provides the necessary GeoJSON files so your agent can build accurate choropleth maps.

## The Right Fit

Use this MCP if your core need is to compare economic performance (like income or unemployment) across different defined geographical boundaries—be it a county, state, or metro area. You must be working with data that requires spatial context for analysis.

Don't use it if you just need a single national time series value (e.g.,

the national average GDP). For simple point-in-time metrics across the whole country, other general economic tools might suffice. Also, if your goal is purely structural database modeling without needing geographic boundaries, this MCP adds unnecessary complexity.

If you need to pull data *and* map it, use get_regional_data combined with get_geo_shapes. If you're just figuring out what metrics are even available for a region type, start with get_series_group.

## Comparing US regions used to be a spreadsheet nightmare.

If you work in location intelligence or policy, you know the drill: You pull a national metric for unemployment. Then, you have to open 20 different tabs and download regional breakdowns —one sheet per state, another for counties, another for MSAs. Copy-pasting these numbers into a master Excel sheet is tedious, error-prone work that takes hours.

With this MCP, your agent handles the complexity. You tell it you need unemployment data broken down by MSA; the agent finds the right series group, pulls all the regional metrics, and provides clean, structured data ready for immediate comparison.

## get_geo_shapes gives you map boundaries instantly.

The hardest part of mapping is getting the correct shape files. Manually sourcing GeoJSON for 50 states, 3000 counties, and various MSAs and ensuring they all align is a massive data engineering lift that usually stalls projects.

Now, your agent uses get_geo_shapes to pull standardized boundary data instantly. The result isn't just clean coordinates; it’s the ready-to-use foundation for any complex visualization you build.

# FRED GeoFRED Regional Economic Data - 3 Tools

Use these tools to discover available regions for any metric, pull cross-sectional data across US geography, and retrieve the necessary map boundaries.

| # | TOOL | DESCRIPTION |
|---|---|---|
| 01 | `get_regional_data` | Retrieves cross-sectional regional economic data for specified U.S. areas like states or counties. |
| 02 | `get_series_group` | Determines the available region types and units for any given FRED series ID to guide your analysis. |
| 03 | `get_geo_shapes` | Downloads standardized geographic shape files, including boundaries for counties and MSAs, ready for mapping. |

# 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 unemployment rate by state?

**Unemployment Rate by State (Latest)**

| STATE | RATE |
|---|---|
| Nevada | 5.4% |
| DC | 5.2% |
| California | 5.1% |
| ...national avg... | 3.7% |
| South Dakota | 2.0% |
| Vermont | 1.9% |
| North Dakota | 1.8% |

51 regions returned from GeoFRED.

**U** Compare median household income across metro areas

**Median Household Income — Top 5 MSAs**

1. San Jose-Sunnyvale: $145K
2. Washington-Arlington: $118K
3. San Francisco-Oakland: $115K
4. Seattle-Tacoma: $105K
5. Boston-Cambridge: $102K

National median: $75K

U Get the geographic boundaries for U.S. states

**🗺 State Boundaries — GeoJSON**

Returned shape data for all 50 states + DC.
Includes: FIPS codes, state names, boundary coordinates.

Use with regional data to build choropleth maps.

# Frequently Asked Questions

01 **How do I know if a metric is available by county using get_series_group?**

You run get_series_group with the desired FRED series ID. The resulting metadata will list all supported region types, confirming whether 'county' or other local breakdowns are valid for that specific metric.

02 **What if I need to compare data across different regions like BEA and FRB?**

You can use get_regional_data multiple times. You just need to confirm the region type (bea or frb) is valid for your metric first, using get_series_group.

03 **Does FRED GeoFRED support international data?**

No, this MCP is specialized for U.S. internal economics. It supports US region types like state, county, and MSA, but not global country comparisons.

04 **Can I get the boundaries for multiple regions at once with get_geo_shapes?**

Yes, you specify the desired shape type (e.g., 'state' and 'county') in a single call to get_geo_shapes, ensuring consistency across your mapping project.

05 **What kind of data can I pull using get_regional_data?**

You can retrieve various cross-sectional economic metrics including unemployment, median income, poverty rates, and GDP breakdowns for specific US regions.

# 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 `"fred-geofred-regional-economic-data": { "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

# FRED GeoFRED — Regional Economic Data 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



## DOCUMENT INFORMATION

| | |
|---|---|
| Generated | June 2026 |
| MCP Server | FRED GeoFRED — Regional Economic Data MCP |
| Server ID | `019d759f-1813-70da-b02a-9fdb79346965` |
| Platform | Vinkius Cloud for AI Agents |
| Endpoint | `https://edge.vinkius.com/{token}/mcp` |

## LICENSE & USAGE