Building Intelligent Applications with ClickHouse and LangDB

https://langdb.ai

Founders

Matteo Pelati

Co-founder

Veteran software architect and data engineering leader with 20+ years of experience. Expert in building large-scale data analytics and ML platforms.

Former Head of APAC Data Engineering @ Goldman Sachs
Former Head of Data Platform @ DBS Bank
Former Chief Architect @ DataRobot

Vivek Gudapuri

Co-founder

Technology leader with proven track record in scaling startups. Expert in building innovative platforms in data, logistics, and fintech.

Chief Technology Officer @ OpenFabric
Chief Technology Officer @ Yojee
Member of Founding Team @ PaySense

Backed by

SEQUOIA

Gradient Ventures

JANUARY CAPITAL

Agenda

01 Introduction to agentic applications and real-time data challenges

02 What is an AI gateway and why it's important ?

03 Introducing LangDB as an AI gateway

04 Why ClickHouse?

05 Key LangDB features

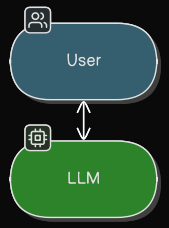
06 Usage patterns and a real-world example

07 Demo

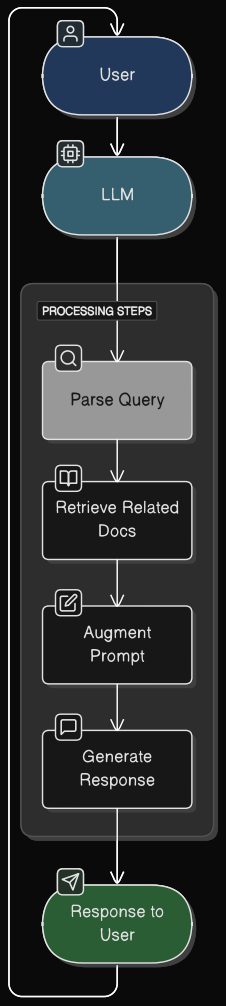
08 Q&A

LLMs and Tools

Large Language Model Interaction



LLMs with Hardcoded Tools



Large Language Models (LLM)

- A basic LLM system where users interact directly with the model.
- Responses are generated only from the model's training data, without external sources.

Example:

User: "Who discovered gravity?"

LLM: "Isaac Newton discovered gravity."

LLMs + Hardcoded Tools

- An enhanced LLM system with a fixed pipeline for retrieving and processing external data.
- The system parses the query, fetches relevant info, augments the prompt, and generates a response.

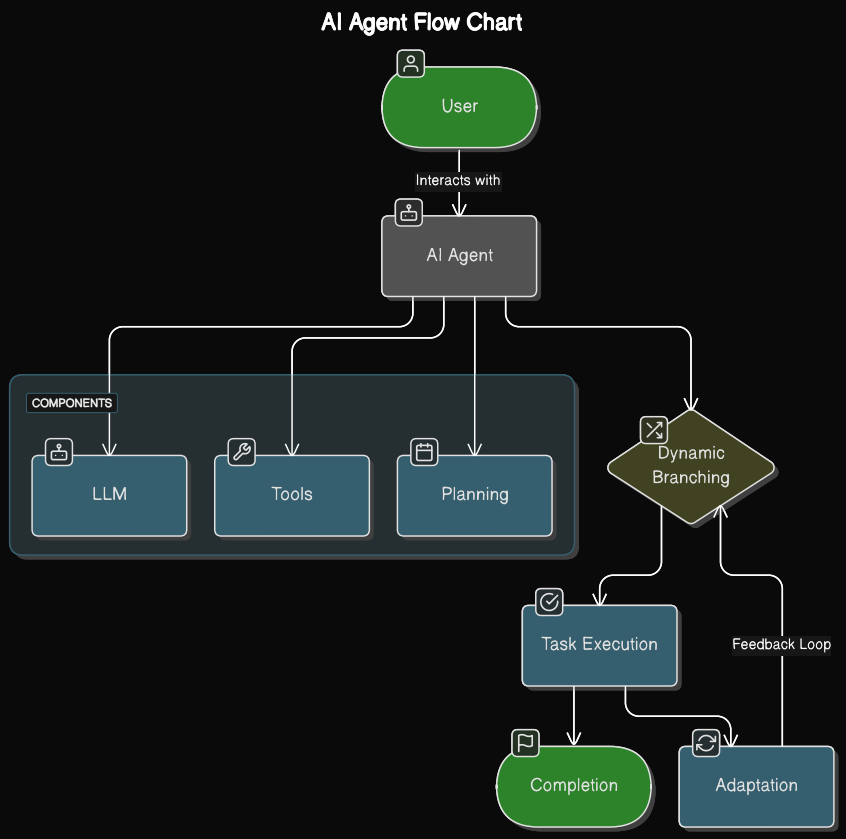
Example:

User: "Where is my order?"

System: Retrieves tracking info → Augments query → LLM responds:

"Your order #12345 was shipped on Feb 20 and will arrive by Feb 25. Track it [here]."

AI Agents



Description

- A fully autonomous AI system that integrates an LLM, tools, and planning to execute complex, multi-step workflows.
- Unlike Diagram 2, this model is not rigid—it dynamically decides which tools to use and what actions to take.
- The workflow branches out, allowing adaptive decision-making rather than following a fixed sequence.

Example: AI-Powered Research Assistant

User: "Summarize the latest AI trends and suggest experts to follow."

AI Agent Workflow:

1. Retrieve recent AI papers and articles.
2. Extract key trends using NLP tools.
3. Identify top AI experts based on citations and social influence.
4. Generate a structured report with sources.

Final Response: "Recent AI trends include multimodal models and efficient fine-tuning. Key experts: Yann LeCun, Andrej Karpathy, Fei-Fei Li. Full summary [here]."

This system mimics human reasoning, making it ideal for automated decision-making, research, and task automation.

Challenges of building AI Agents

Multi-Agent Complexity

Understanding and managing interactions between multiple specialized agents in the workflow

Model Selection & Testing

Rapid experimentation with different models to find the optimal balance of performance and efficiency

Integration Challenges

Coordinating multiple tools: social connectors, web scrapers, search engines, and content summarizers

Quality vs Cost

Maintaining high-quality responses while managing costs across numerous interactions

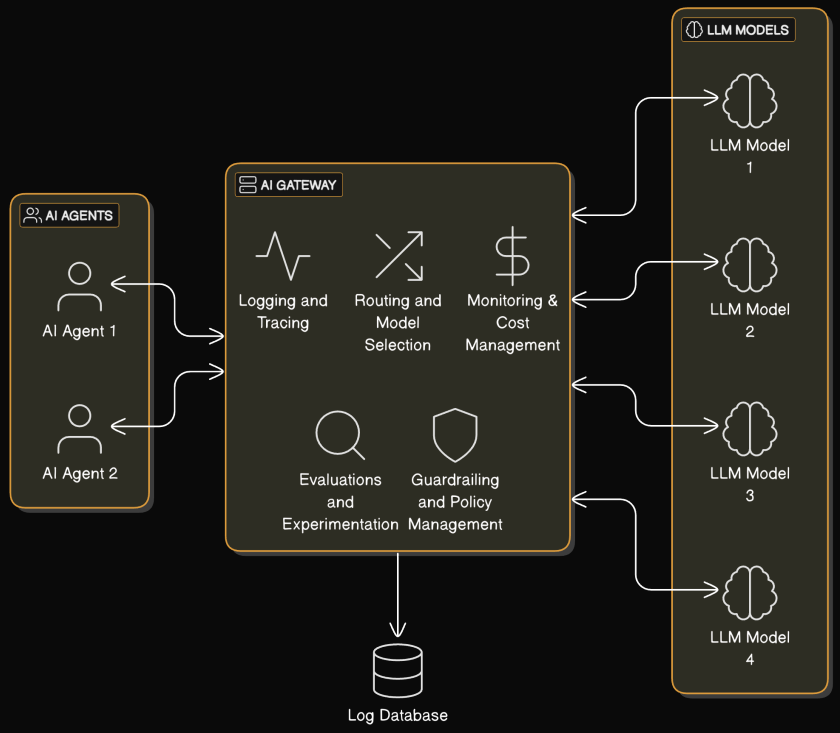
Dynamic Routing

Implementing real-time traffic routing between different models while the system is running

Prompt Engineering

Managing complex, extensive prompts while maintaining consistency and effectiveness

AI Gateway Architecture



An AI Gateway is a service that simplifies, secures, and governs access to LLMs.

Acts as a centralized platform for managing AI workflows, giving developers a consistent interface (or endpoint) to interact with models from different providers

Key Features

1. **Logging and Tracing**
   Centralized request tracking and audit trails
2. **Routing and Model Selection**
   Intelligent request routing across providers
3. **Monitoring & Cost Management**
   Real-time spend tracking and budget controls
4. **Evaluations & Experimentation**
   A/B testing across different model versions
5. **Guardrailing & Policy Management**
   Content filtering and compliance enforcement

LANGDB

Why LangDB?

Fastest Enterprise AI Gateway

Built in Rust for unmatched performance and scalability

Cost Reduction up to 70%

Smart routing automatically selects the most cost-effective models

Access to 250+ Models

Unified API for seamless integration with hundreds of models

Flexible Deployment

Available as hosted service or open source solution

Multi-Agent Tracing

Advanced debugging for complex agent interactions

Advanced Analytics

Deep insights into your AI operations with Clickhouse

Why ClickHouse

1 Observability Powerhouse

ClickHouse is widely used as a storage for observability tools. An AI gateway needs to capture and store a lot of analytical and conversational data, making ClickHouse ideal.

2 Customer Control

Data remains fully accessible for database customers. They can even choose to host their own ClickHouse instance and run LangDB on top of it.

AI Functions

- `ai_completions` Generate AI completions from various models
- `ai_embed` Create embeddings from text

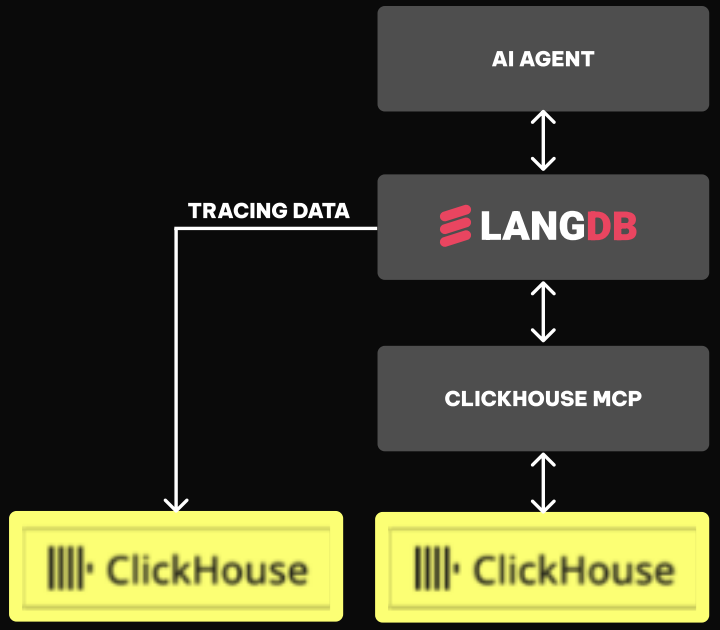
The Unstructured Challenge

While ClickHouse excels at structured data, most LLM data is textual and unstructured. How do we bridge this gap?

We leveraged ClickHouse UDF functionality to integrate with LangDB, unlocking infinite possibilities of combining LLMs and ClickHouse capabilities.

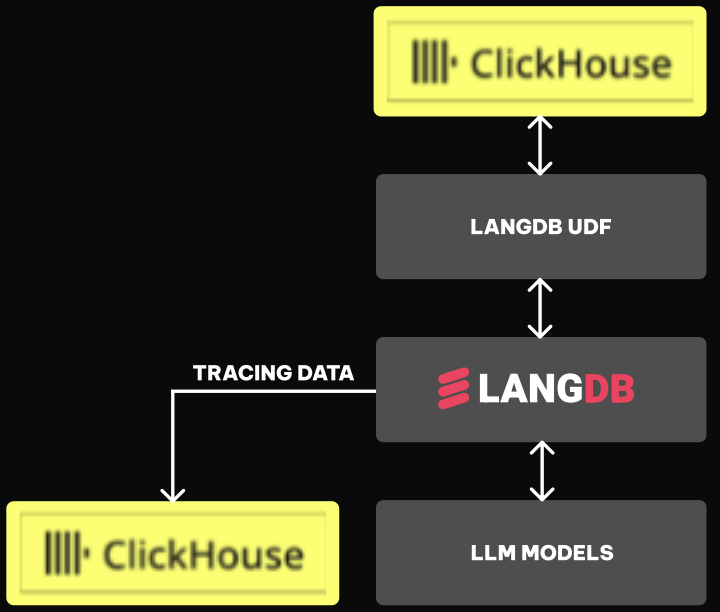
Implementation Patterns

ClickHouse as MCP server



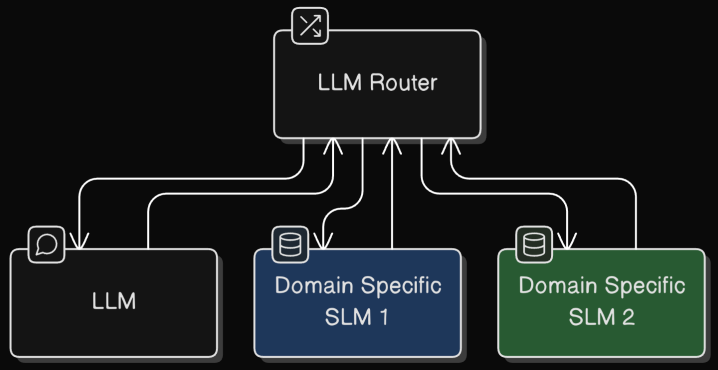
AI agents can leverage LangDB to access multiple models and leverage LangDB's MCP capabilities to access data stored in ClickHouse

AI functions in Clickhouse



ClickHouse can directly leverage LangDB's SQL AI functions to access to multiple LLMs

Least Cost Routing



1 Standard LLM Deployment

Agent is deployed using a normal LLM, providing a foundation for intelligent processing and decision-making capabilities.

2 Data Collection

Usage and preference data is captured in ClickHouse, creating a comprehensive dataset of interaction patterns and performance metrics.

3 SLM Fine Tuning

Data is used to fine tune an SLM based on real usage patterns, improving response quality and task-specific performance.

4 Router Fine Tuning

Router model is trained on usage data to make intelligent decisions about which SLM to use for each request type.

5 Intelligent Routing

Router can now optimize cost by automatically routing requests to the appropriate SLM, ensuring efficient resource utilization and improved performance.

How to use it?

1. Get a LangDB API key from the LangDB dashboard
2. Replace the OpenAI URL, the API key, and the model

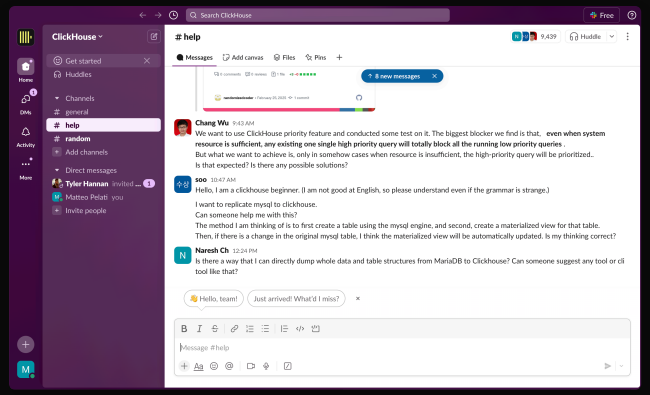
```python
from openai import OpenAI

client = OpenAI(
    base_url="https://api.us-east-1.langdb.ai/9c7ac2c8-b76f-453b-914d-39eaaccec092/v1",  # LangDB API base URL
    api_key=""
)

response = client.chat.completions.create(
    model="openai/gpt-4o-mini",
    messages=[
        {
            "role": "system",
            "content": "You are a helpful assistant."
        },
        {
            "role": "user",
            "content": "What are the earnings of Apple in 2022?"
        }
    ]
)

print("Assistant:", response.choices[0].message)
```

Example: Slack Message Analysis



Multiple LLMs Required

- A Reasoning LLM for metric definitions extraction
- A simple and cost effective LLM for metric values extraction

01 Data Collection

Collect messages from Clickhouse Slack channel and store all raw messages in ClickHouse table

02 Metric Definition

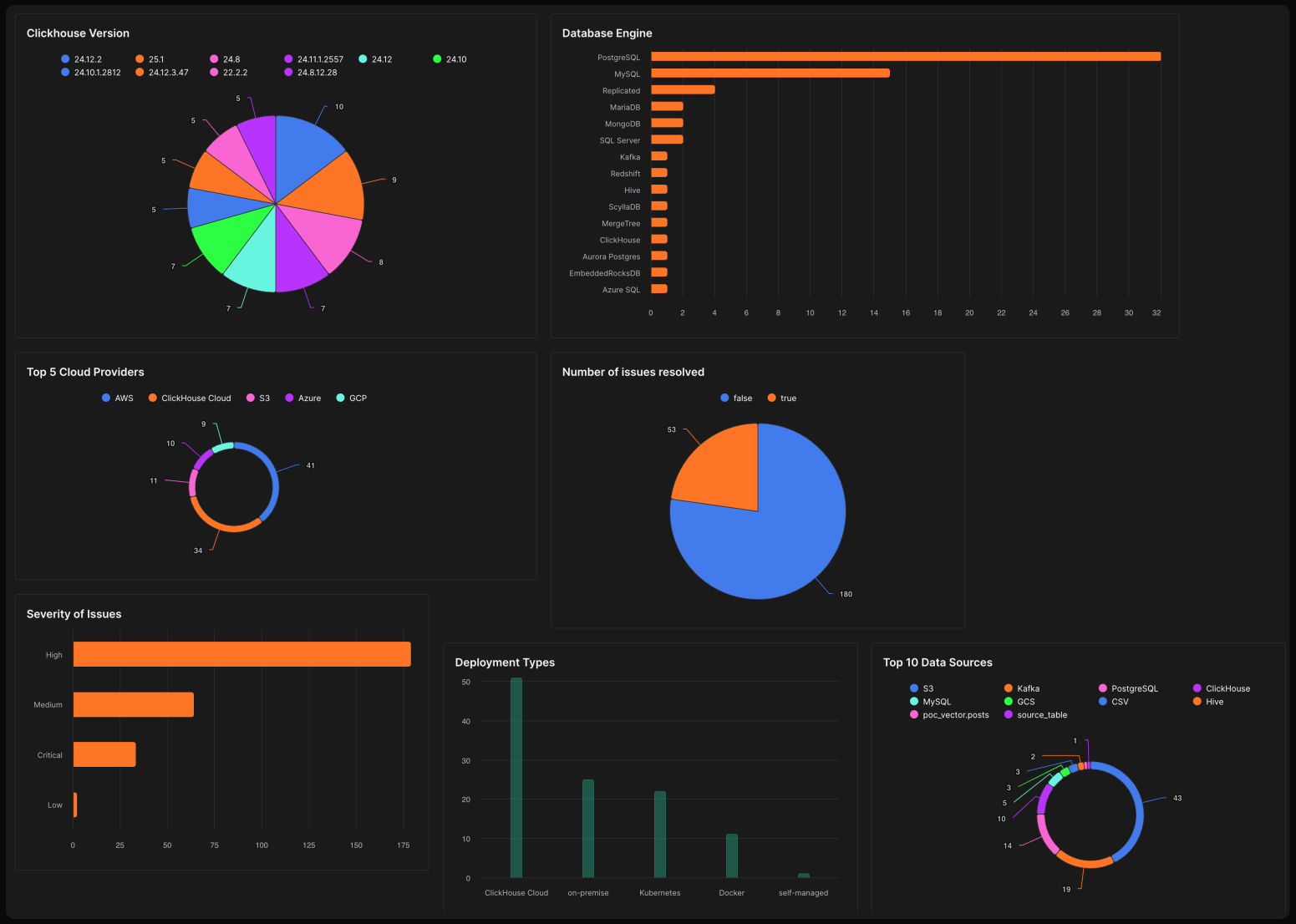
Use LangDB UDF to extract metric definitions from messages using a reasoning

```
{
"category": "Product Issues",
"metric_name": "Issue Type",
"metric_description": "Categorizes the type of issue reported in each co
"sample_values": [
  "Query Failure",
  "Data Type Issue",
  "Metadata Mismatch",
  "High Memory Usage",
  "Connection Issue",
  "Version Upgrade Issue",
  "Performance Issue",
  "Replication Issue",
  "Authentication Issue",
  "Integration Issue"
],
"extraction_prompt": "Categorize the type of issue reported in the conve
"metric_type": "string",
"relevance_count": 37
 }
```

03 Metric Extraction

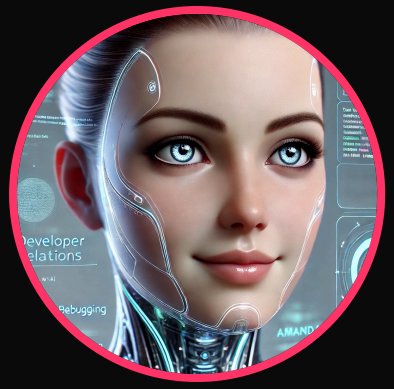
Use LangDB UDF to extract actual metrics from messages using a simpler and cheaper model

Example: Slack Analytics Dashboard



LangDB Demo

Amanda: A Demo PR Agent



The Challenge

Reaching developers where they naturally gather and engage:

- Reddit
- HackerNews
- Slack
- Discord

Amanda's Intelligent Workflow

1 Community Monitoring

Actively identifies discussions about AI gateways and related technologies

2 Context Research

Conducts thorough online research to understand the full context of discussions

3 Relevance Assessment

Intelligently evaluates if and how LangDB can add value to the conversation

4 Personalized Engagement

Crafts thoughtful, context-aware responses that genuinely help the community

LangDB Editions

LangDB Hosted

Available at https://langdb.ai/

1. **Access to 250+ Models**
   Use any model from our extensive collection
2. **Free Signup + Credits**
   Start for free, use credits or bring your own key
3. **Intelligent Routing**
   Smart request distribution and load balancing
4. **Monitoring & Cost Control**
   Track usage and manage expenses in real-time

LangDB Open Source

Available at github.com/langdb/ai-gateway

1. **Built in Rust**
   High performance and reliability
2. **Self-Hosted**
   Deploy in your own environment
3. **Apache-2.0 License**
   Free to use and modify
4. **Active Community**
   Join us on Slack, contributions welcome