

Apache Doris x AI 应用的全场景解析

张彬华@SelectDB资深解决方案架构师



打造全球最知名开源软件基金会 Apache 的顶级项目



Open-Source Real-Time Data Warehouse

2013 Project Creation

13k+ GitHub Stars

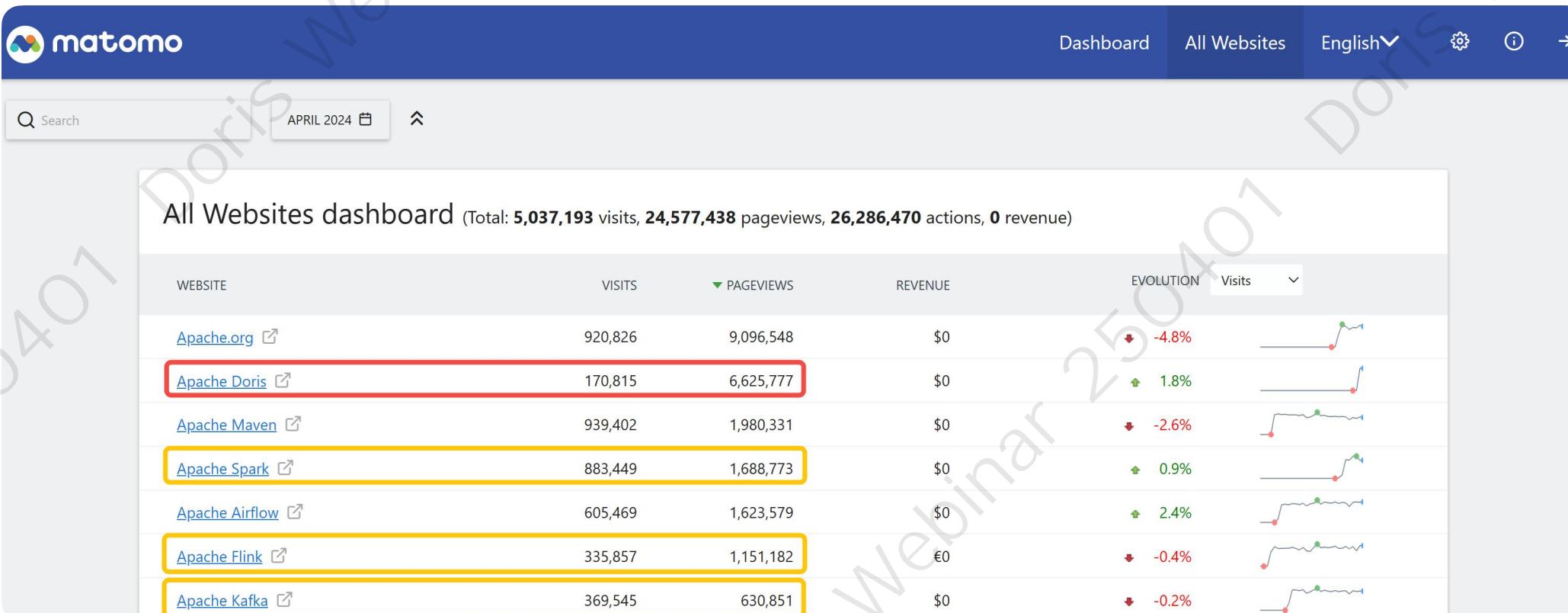
2017 Open Source

650+ Contributors

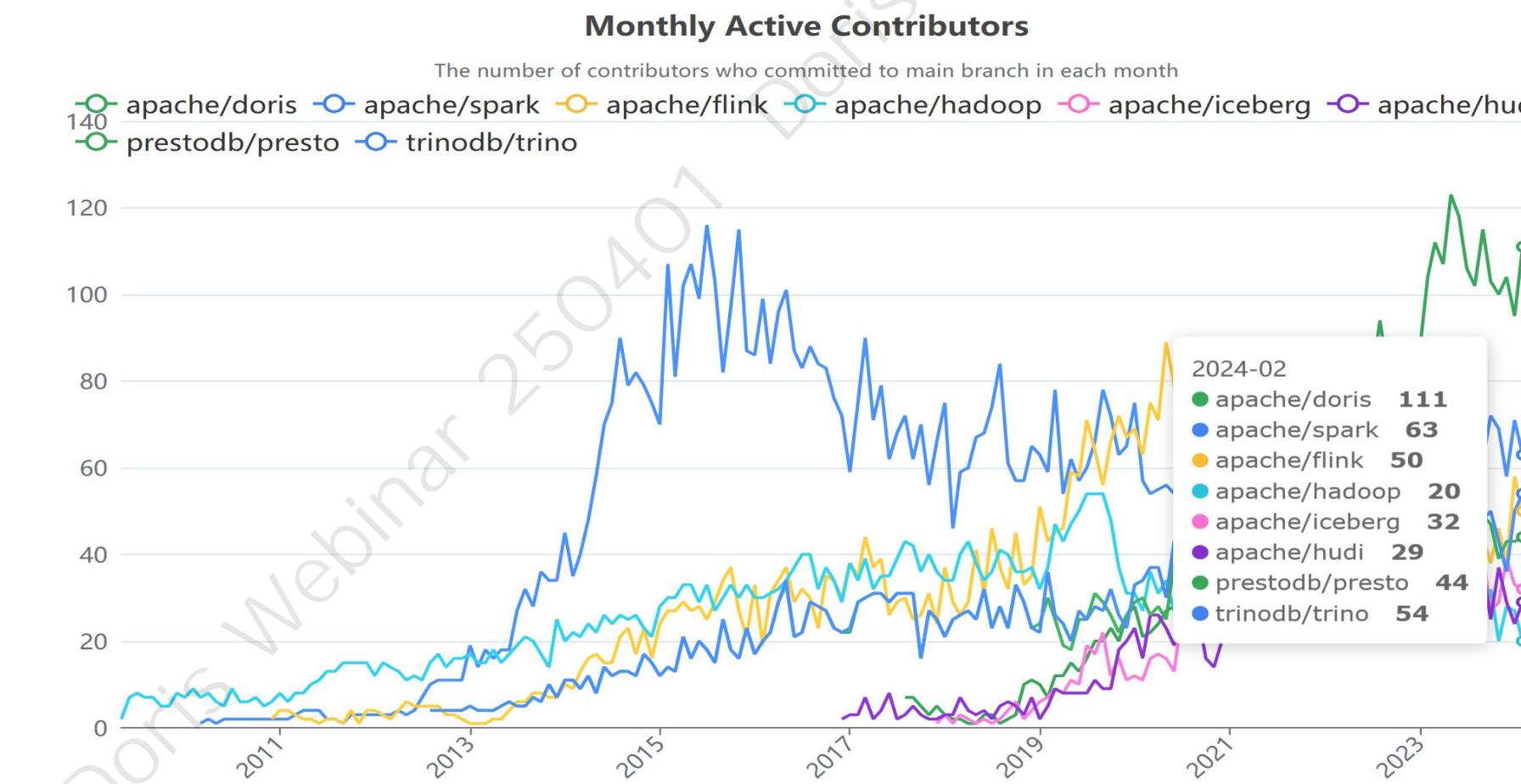
2022 ASF Top Project

5000+ Enterprises

Apache 所有项目中 官网每日浏览量 No.1



开源大数据、数据库项目中 月活开发者 No.1



目录

Apache Doris x AI 关键特性

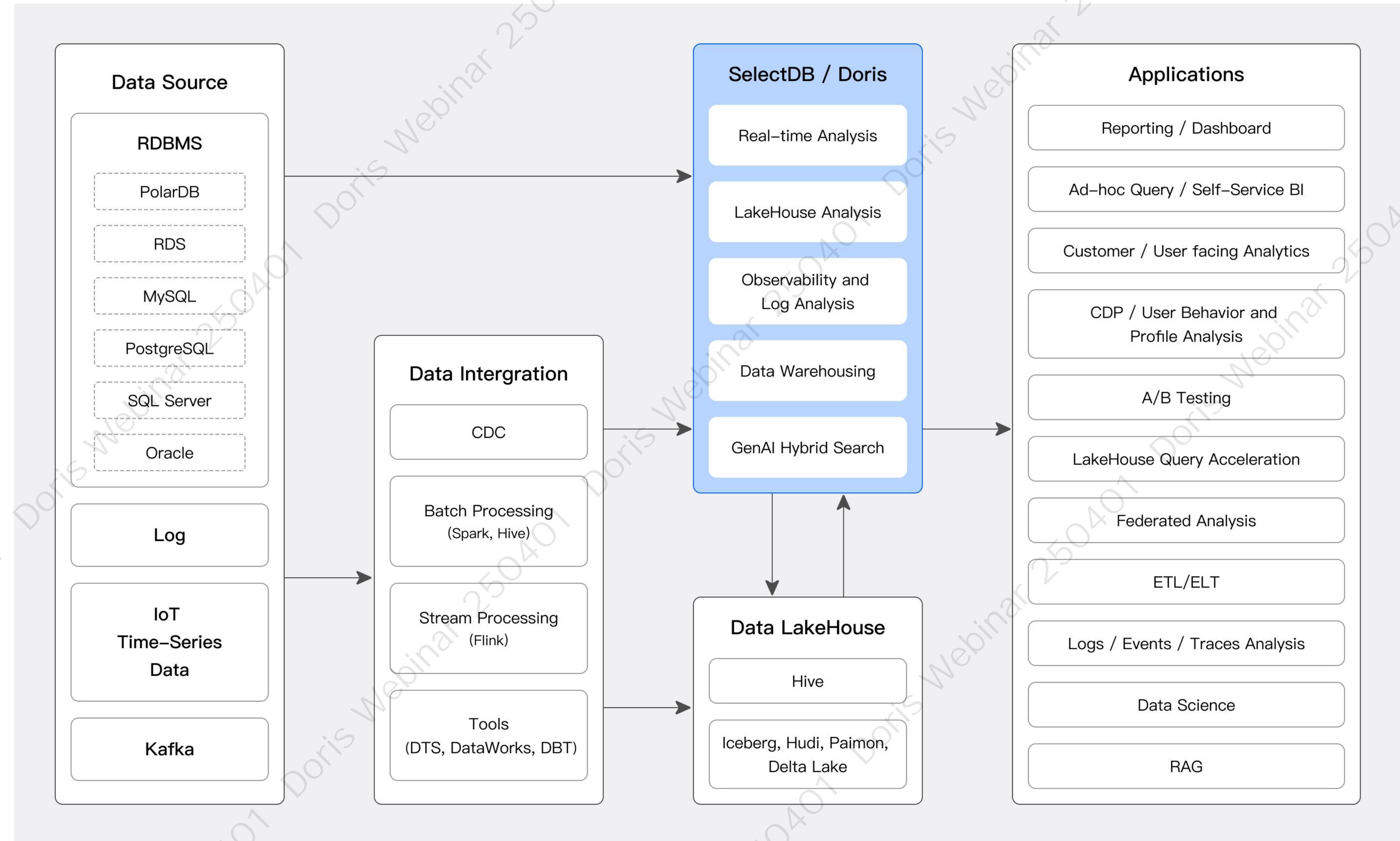
Apache Doris x AI 应用场景

Apache Doris x AI 案例演示

Apache Doris x AI 未来规划



数据源经过各种数据集成和加工处理后，通常会进入实时数据仓库 Doris 和离线湖仓（如 Hive、Iceberg、Hudi、Paimon），广泛应用于 OLAP 分析场景，同时也可以作为 LLM 上下游的数据底座。



兼容 MySQL 的语法和协议

兼容 MySQL 协议，学习成本低，和各类 LLM 项目及 BI 工具兼容性好

使用 MySQL Client 连接

```
> mysql -h 59.110.10.92 -P 11827 -u admin

Welcome to the MySQL monitor. Commands end with ; or \g.
Your MySQL connection id is 119952
Server version: 5.7.37 SelectDB Core version: 2.0.0

Copyright (c) 2000, 2022, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql>
```

使用 DBeaver 连接



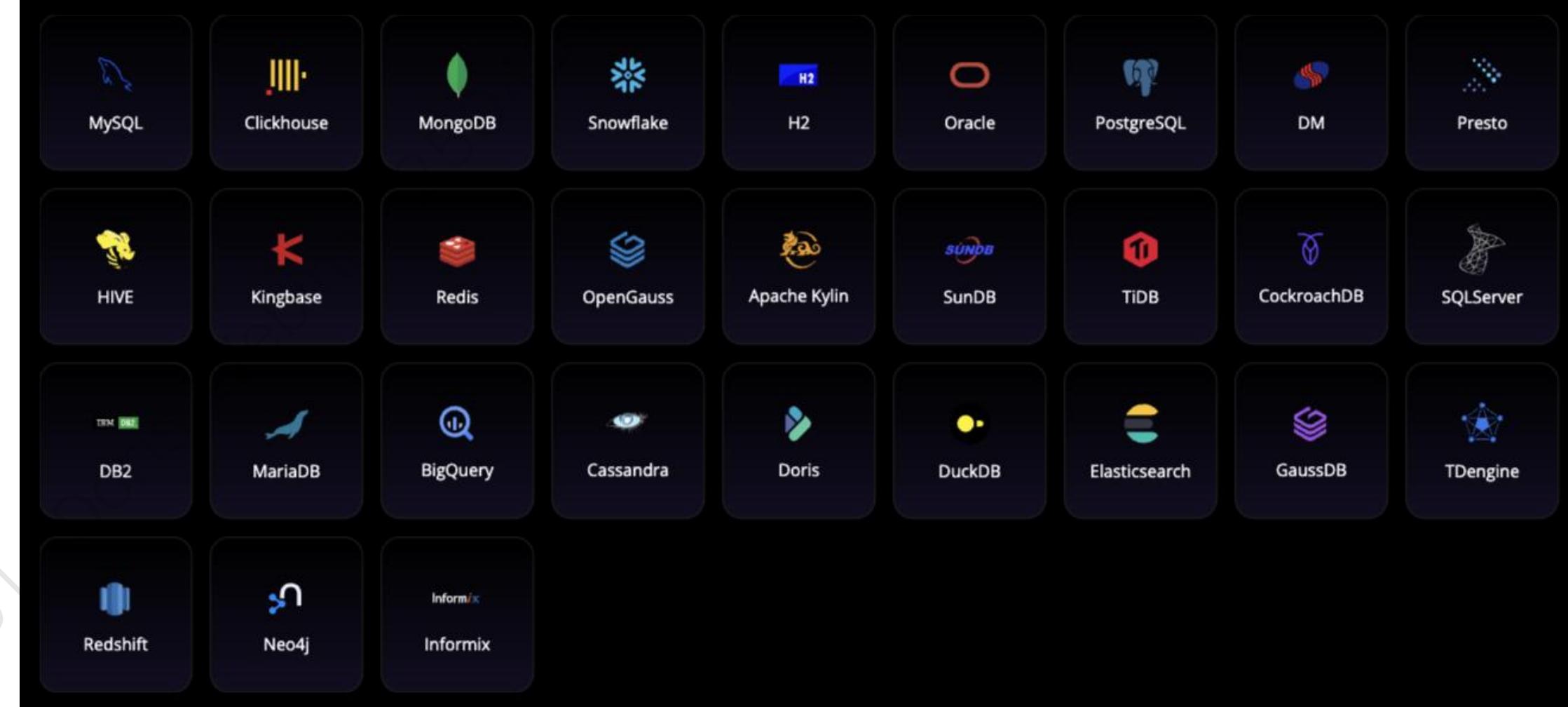
Text2
SQL
项目

SQL Chat is built by [Next.js](#), it supports the following databases and will add more over time:

- MySQL
- PostgreSQL
- MSSQL
- TiDB Cloud

支持多种数据库

告别频繁切换工具的烦恼！现在你可以通过Chat2DB轻松连接和管理任意数据库，无论什么样的数据库类型，我们都提供了无缝集成，让你能够专注于深入分析数据并挖掘有价值的数据。

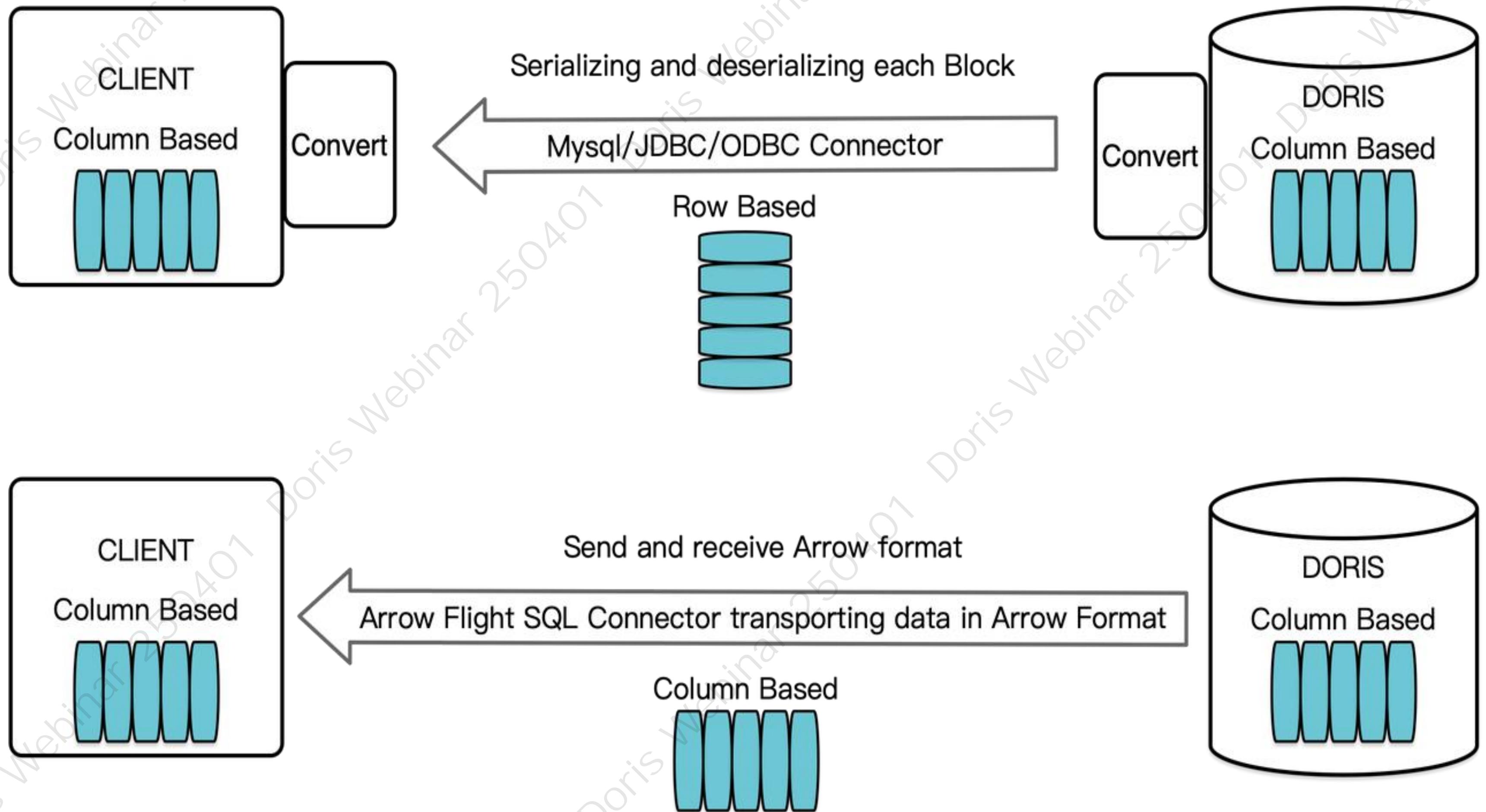


支持 Arrow Flight 高速读取传输服务数据科学、机器学习等AI 场景

MySQL 具有良好兼容性和广泛的工具支持，但在数据科学、大规模数据导出场景，FE 容易成为瓶颈、文本协议效率差。

引入 Arrow Flight 实现高速数据读取、数据直接通过 BE 传递到 Pandas 客户端，列式数据传输。

Pandas (NLP预处理/模型训练)
测试数据吞吐提升 100 倍！



极致的分析性能

执行引擎

- MPP 执行框架
- Pipeline 数据处理模型
- 向量化执行算子

查询优化器

- 结合RBO和CBO，在处理多表 Join 等复杂查询时，能够游刃有余，轻松应对

缓存加速

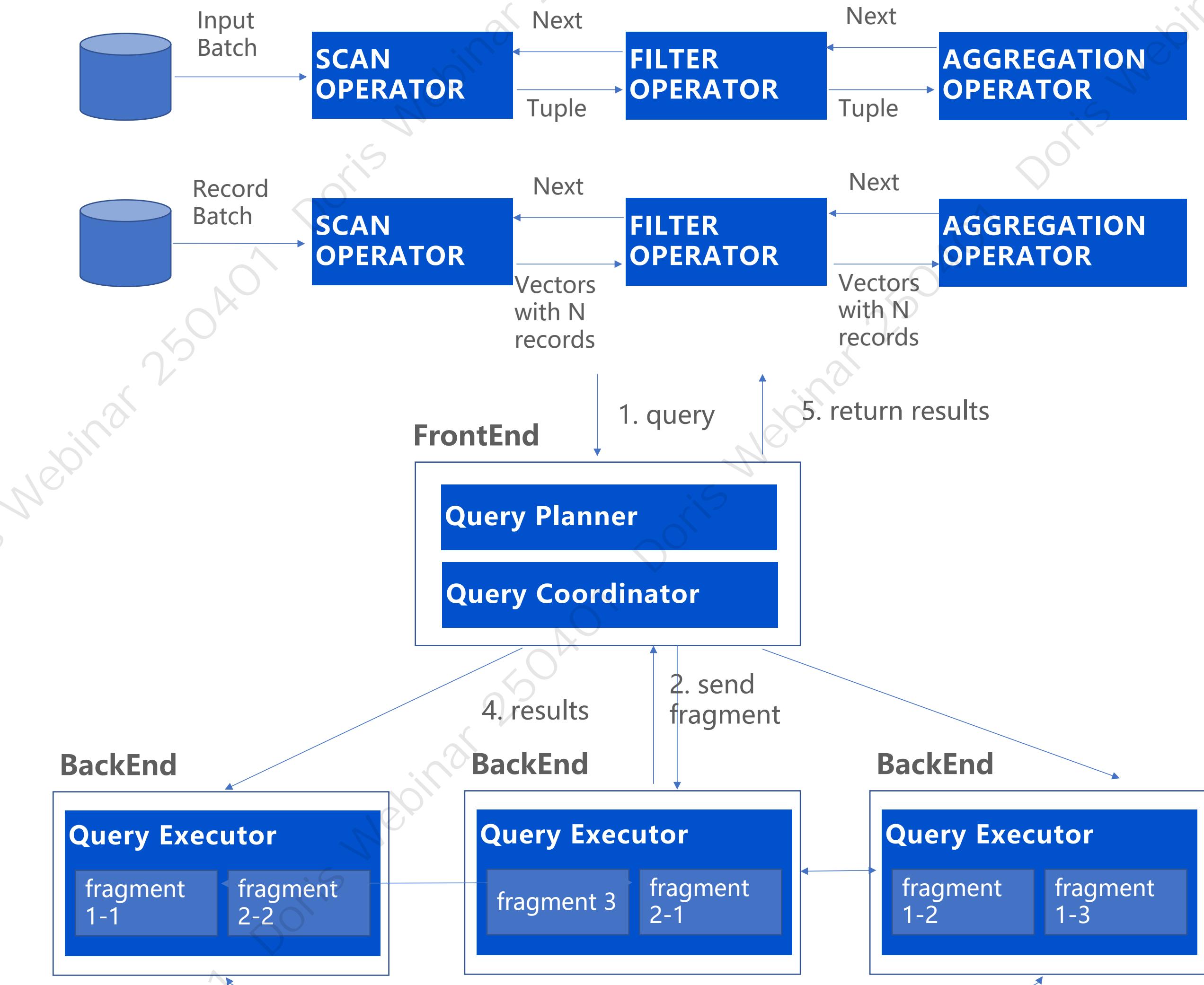
- Data Cache
- SQL Cache

物化视图 透明加速

- 同步物化视图
- 多表异步物化视图

智能索引

- 倒排索引，全文检索（LLM日志分析）
- 自动创建前缀索引 和 ZoneMap 索引



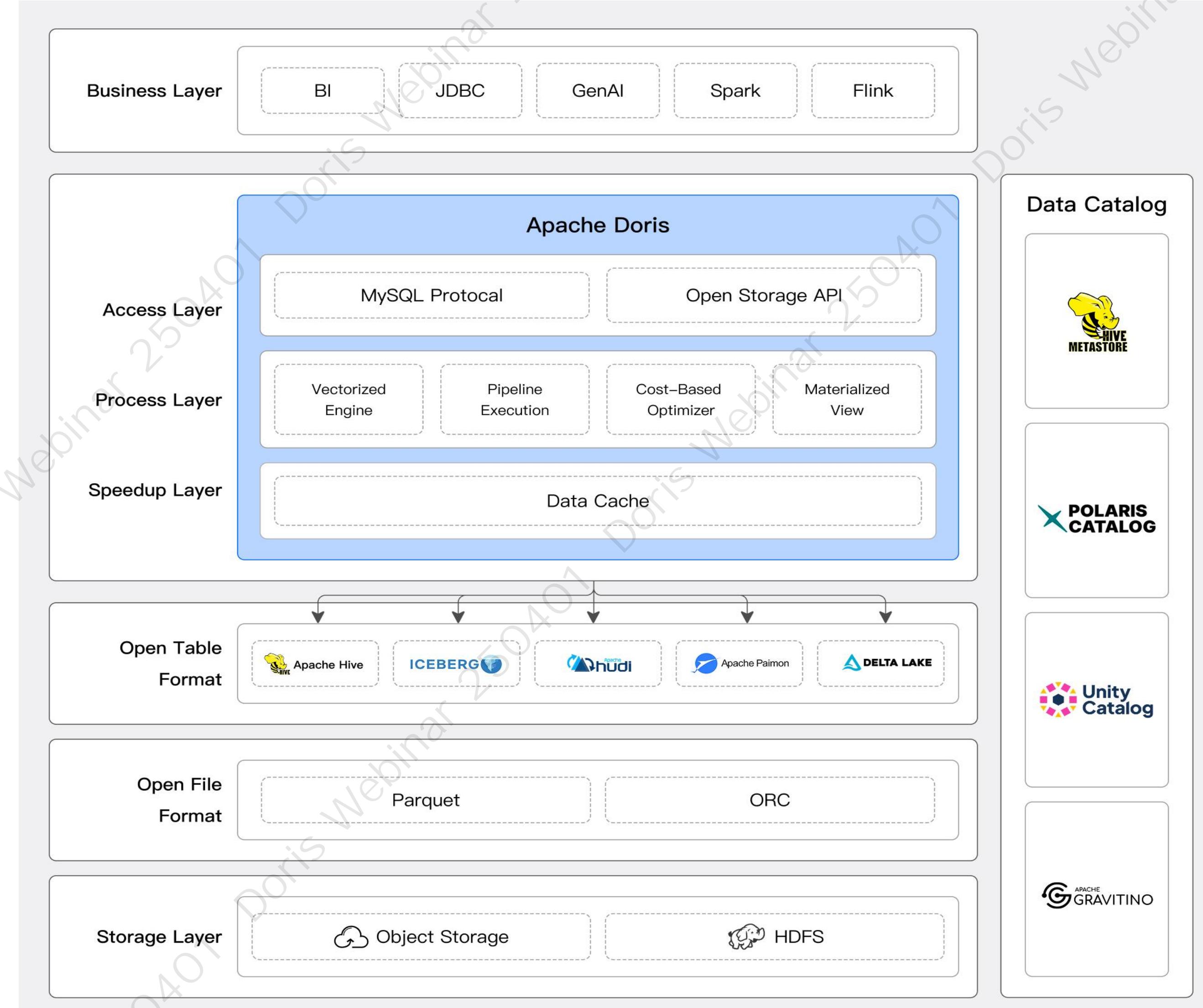
开放的湖仓一体

Doris 通过可扩展的连接器框架、存算分离架构、数据生态开放性和极致的分析性能，为用户提供了优秀的湖仓一体解决方案。

1) 可扩展的连接器框架：Doris 定义了标准的数据目录(Catalog)、库(Database)、表(Table)三个层级，帮助开发人员快速对接企业内部特有的数据源(S3/HDFS：溯源分析/跨域模型训练)，实现数据快速互通。

2) 存算分离架构：不同时间点使用不同规模的计算资源服务业务请求，按需使用计算资源，节约成本。

3) 数据生态开放性：2.1 版本起，Doris 支持多种 SQL 方言，如 Presto、Trino、Hive、PostgreSQL、Spark、Clickhouse 等等。



目录

Apache Doris x AI 关键特性

Apache Doris x AI 应用场景

Apache Doris x AI 案例演示

Apache Doris x AI 未来规划



Doris x DataAgent

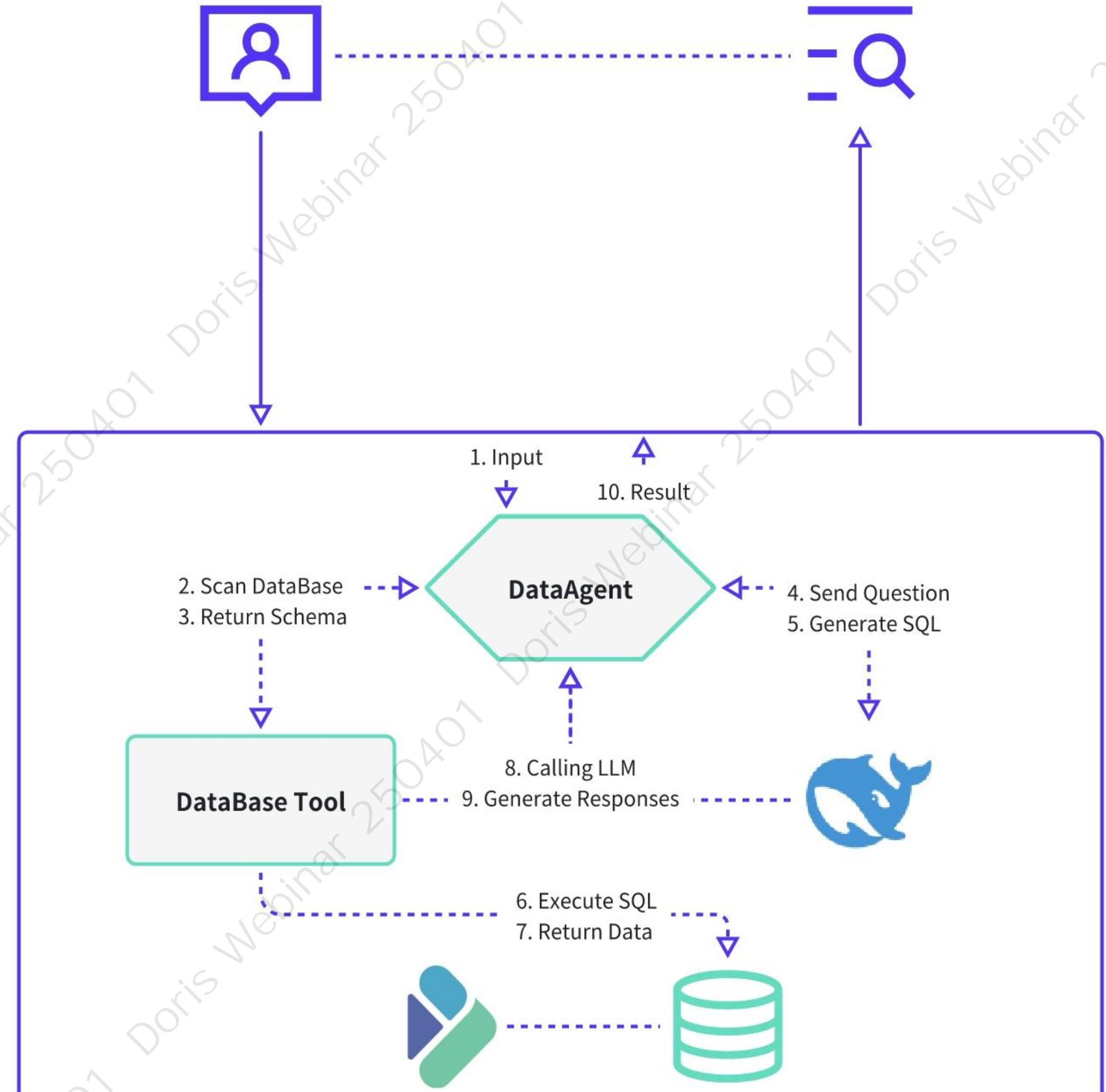
Data Agent 是一种数据智能助手，它能像数据专家一样理解你的问题，自动连接到数据库，结合 LLM 独立完成数据查询、分析和洞察。

应用方案

- 大模型（DeepSeek）根据输入的提示词，以及数据库相关的 Schema 信息，生成查询 SQL
- 基于 Doris，执行上一步生成的 SQL，返回得到的结果数据集
- 大模型（DeepSeek）根据用户的问题，以及第二步得到的结果数据集，生成自然语言回答

应用场景

- **AI 舆情分析：**Doris 存储整合社交、新闻、论坛及客户交互等基础数据，通过 DataAgent 调用大模型进行多维分析，自动进行舆情分类、紧急定义及处理建议输出，并预警潜在风险
- **实时反欺诈决策：**Doris毫秒级处理交易流水（支持每秒百万级写入），结合 DataAgent 通过图神经网络识别异常交易模式，自动拦截高风险交易并生成风险报告



Doris x RAG

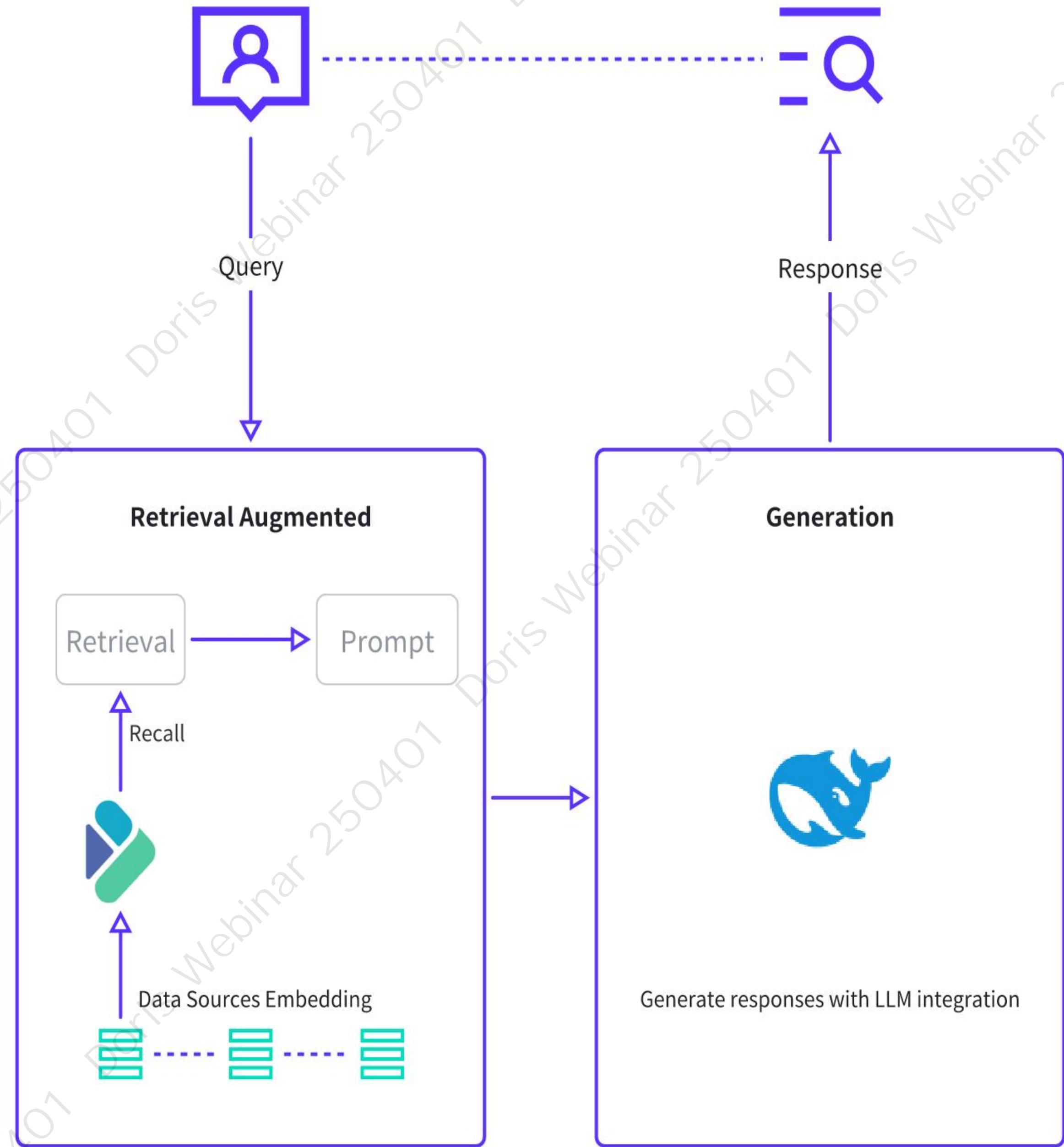
RAG（检索增强生成）是给大模型添加一个知识库，让它遇到问题时先查资料再回答，好比人类解决问题前先查找相关信息一样，也相当于开卷考试。

应用方案

- 本地数据文件进行向量化 Embedding 后存储至 Doris 中
- 结合 Doris（高性能查询）与 **distance** 向量函数（相似检索），实现召回
- 大模型（DeepSeek）选择最合适的数据，生成回答

应用场景

- 问答系统：**RAG 在问答系统中用于处理复杂查询与长尾知识。例如法律咨询场景，系统可检索法律条文、案例库及历史咨询记录，生成带引用来源的答案。
- 企业知识库：**RAG 技术通过结合检索与生成能力，解决传统知识库 FAQ 维护成本高、结构化/非结构化数据难以混合、知识更新滞后等问题。



Doris x ChatBI

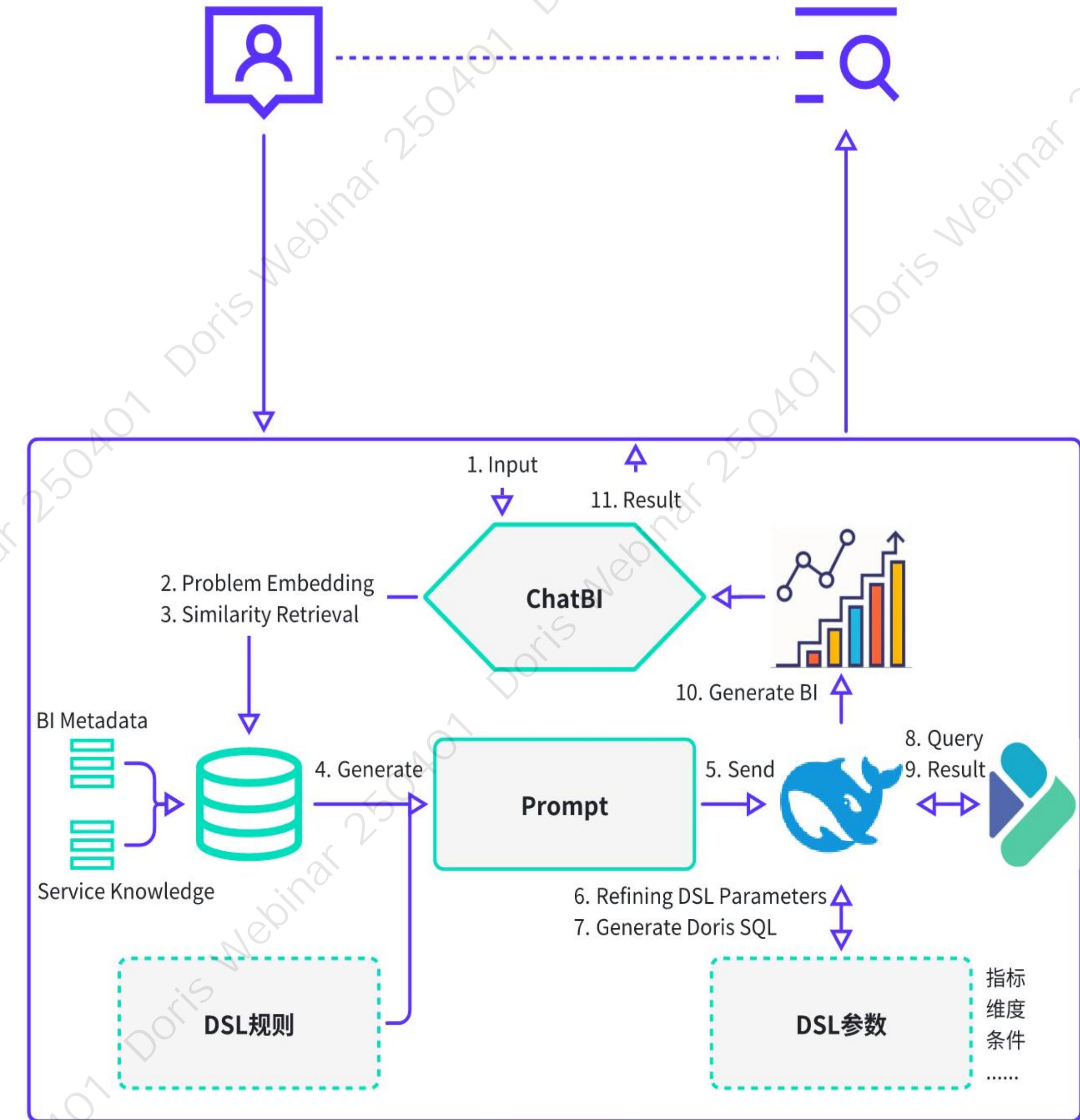
ChatBI 是一种通过自然语言对话就能让用户像聊天一样轻松提问、快速获取数据分析结果并生成可视化图表或报告的工具，无需掌握复杂的数据分析技能。

应用方案

- 用户问题结合 RAG 技术进行相似度召回，并结合 DSL 规则 (Text2DSL) 生成 Prompt
- 大模型 (DeepSeek) 进行语义分析和关键词识别 (N+1 轮)，并基于 Prompt 规则返回 DSL，合规检查无误后生成 Doris SQL 进行数据查询
- 组装 Doris 结果数据，并在前端进行可视化展示

应用场景

- **智能平台BI (Test2SQL)**：高度灵活性的标准数据分析场景，直接生成 SQL 语句，无需依赖预定义的业务规则，理论上可覆盖所有数据库支持的查询逻辑，但精准度不高
- **企业内部BI (Text2DSL)**：业务场景明确且已构建成熟数据资产（如指标库）的系统，将自然语言转换为业务层参数（如“销售额”“上月”“华南”），通过规则引擎生成 SQL，准确率高且响应快



Doris x MCP

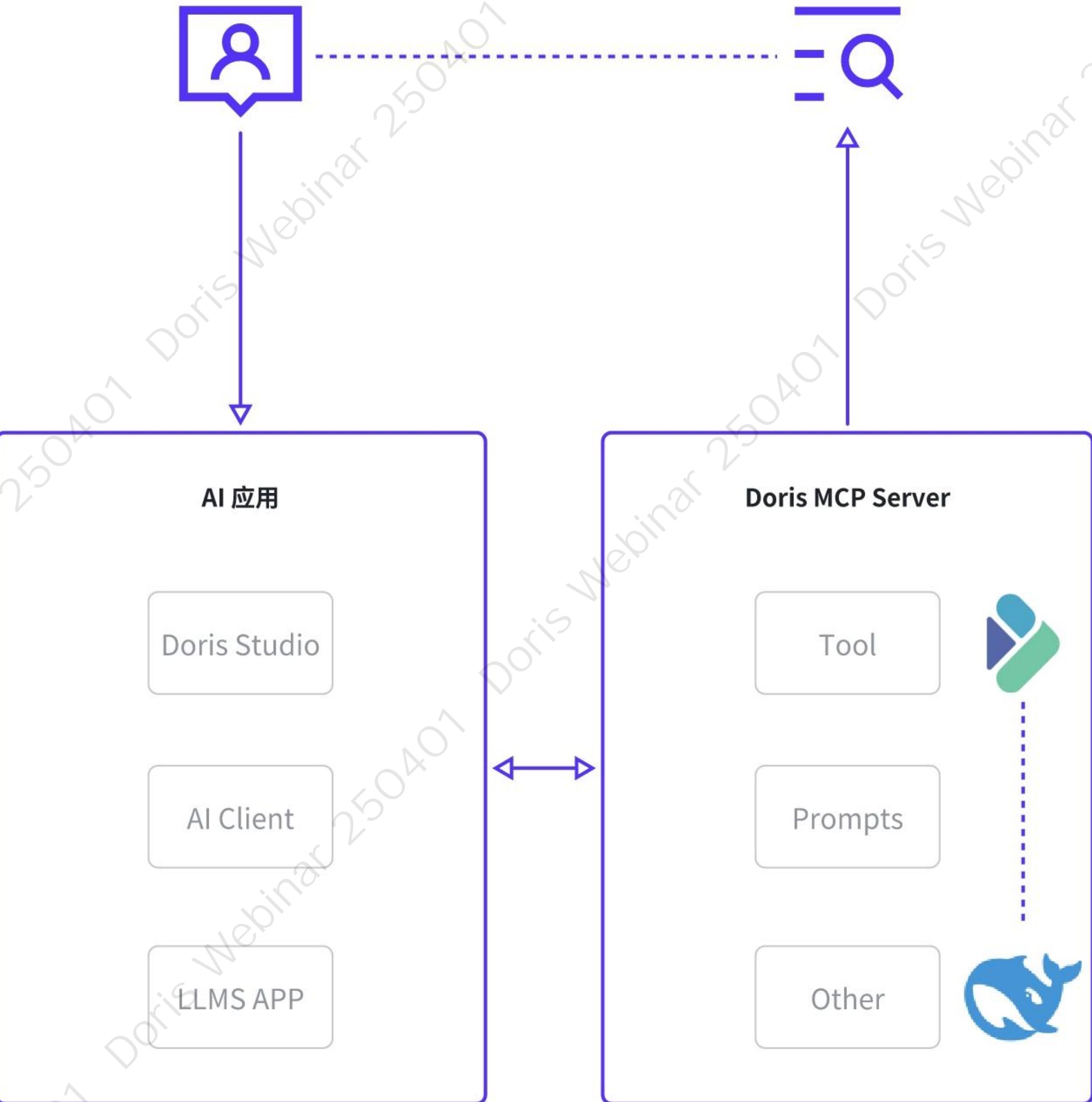
MCP (模型上下文协议) 可以理解为AI的"USB转换器"，让大模型能即插即用地连接各种工具和数据源，统一了交互标准。

应用方案

- 使用兼容 MCP Server 的 Client，例如后续 SelectDB/Doris Studio 会进行兼容
- 当 AI 应用连接到 Doris MCP Server 后，能够获取多种能力 (NL2SQL、Doris查询性能诊断、智能运维...)

应用场景

- 通过 Doris MCP Server，快速实现 Doris ChatBI 的能力
- 智能分析 SQL 执行计划、查询性能特征和数据分布特点，提供定制化的优化建议
-



Doris x AI Observability

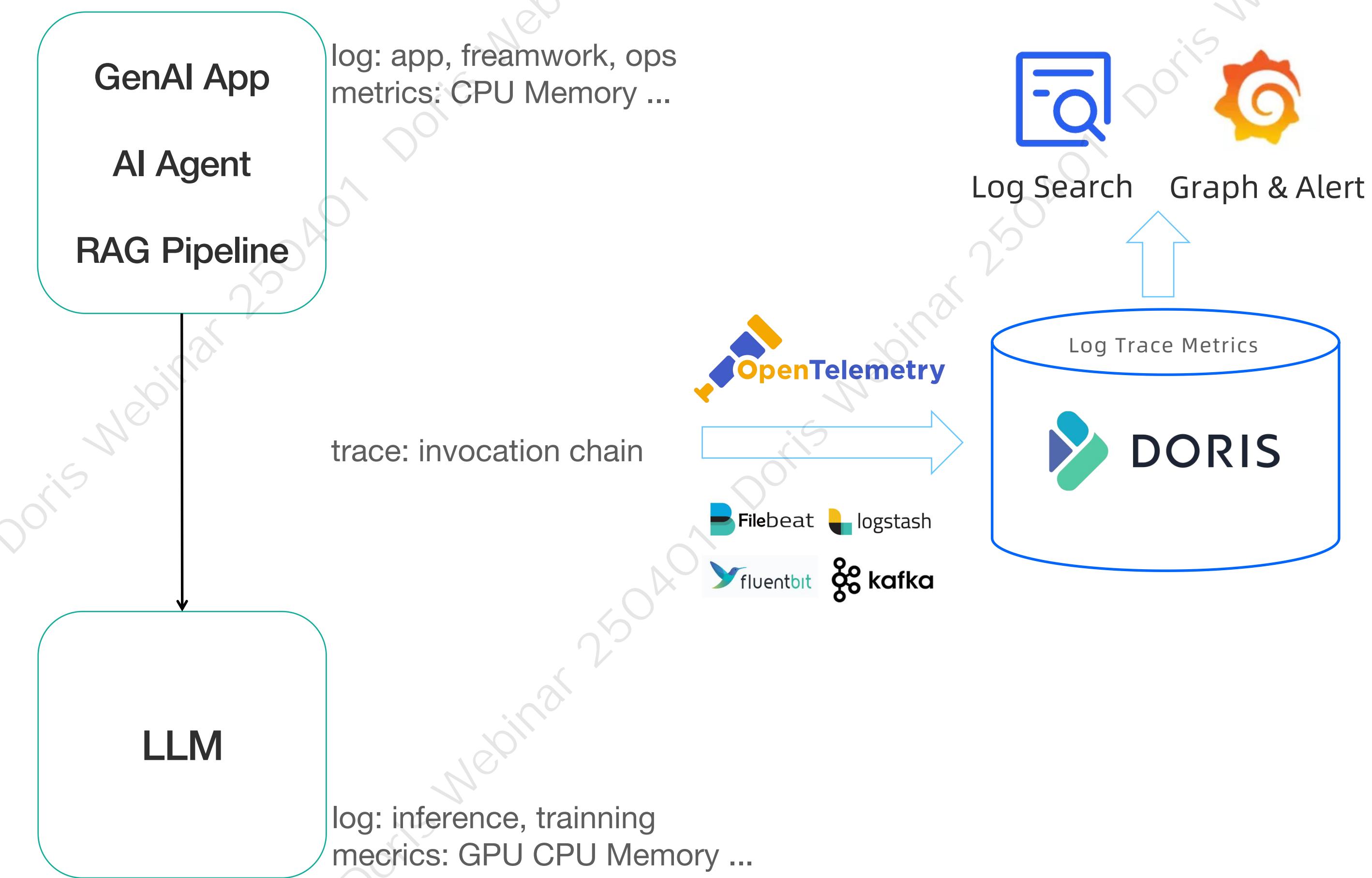
大模型应用涉及复杂的多服务和模型调用，调试和故障诊断难度高，LLM Observability 提供全链路可观测数据和可视化分析能力，为应用调试和稳定运行提供支撑。

应用方案

- Doris 提供高性能、低成本的可观测性数据（Log Trace Metrics）存储和分析
- Doris 与开放的可观测性生态集成，接收来自 OpenTelemetry、ELK 等采集器的数据
- Doris 提供标准 SQL 查询接口，支持 Grafana 可视化和告警

应用场景

- **AI 应用：**调用框架和服务产生的调用链 Trace
- **AI Agent & RAG：** 框架的日志和 CPU 内存监控数据
- **LLM 推理和训练：** 大模型推理和训练日志，GPU CPU 等资源监控数据



目录

Apache Doris x AI 关键特性

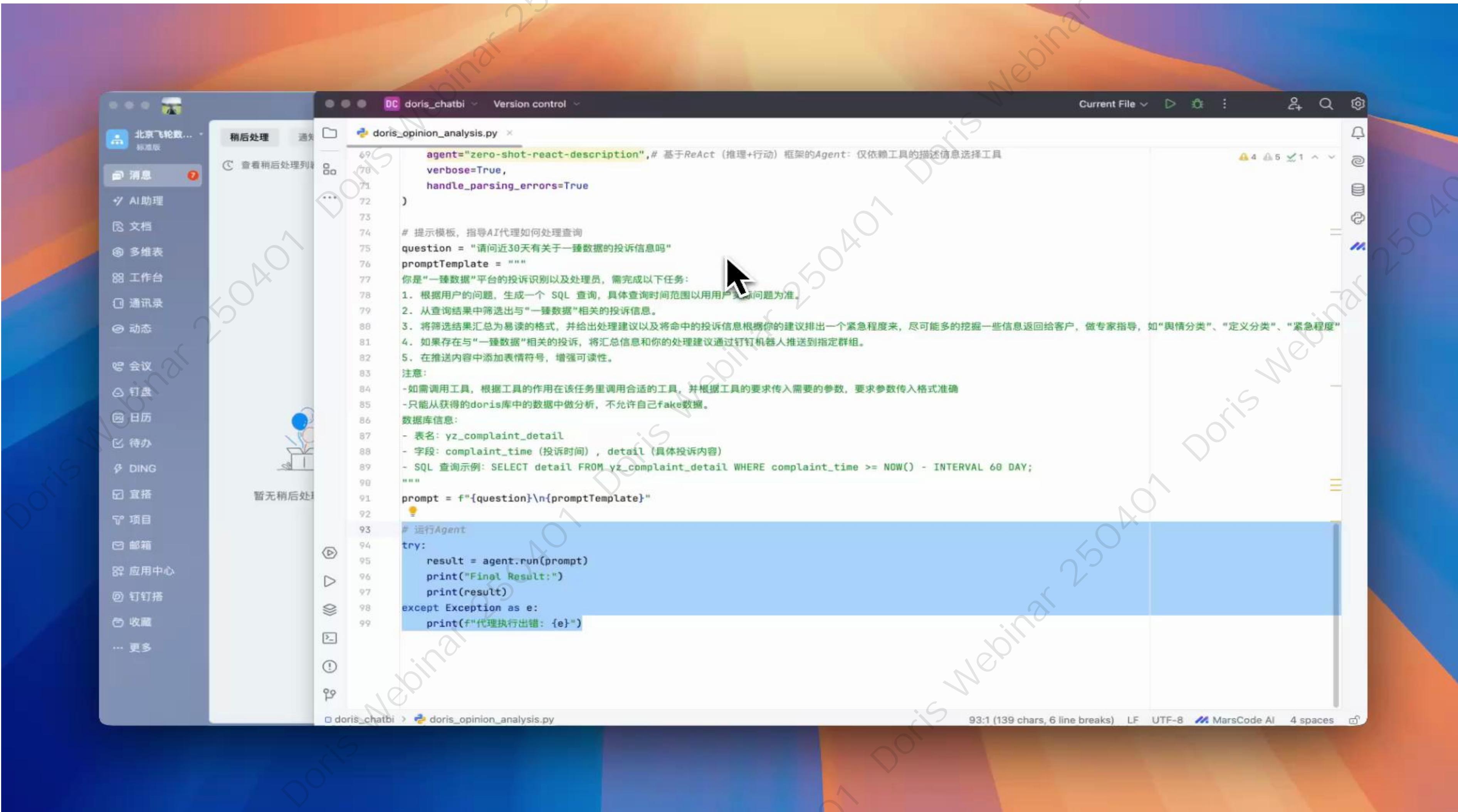
Apache Doris x AI 应用场景

Apache Doris x AI 案例演示

Apache Doris x AI 未来规划



Doris x DataAgent AI 舆情分析

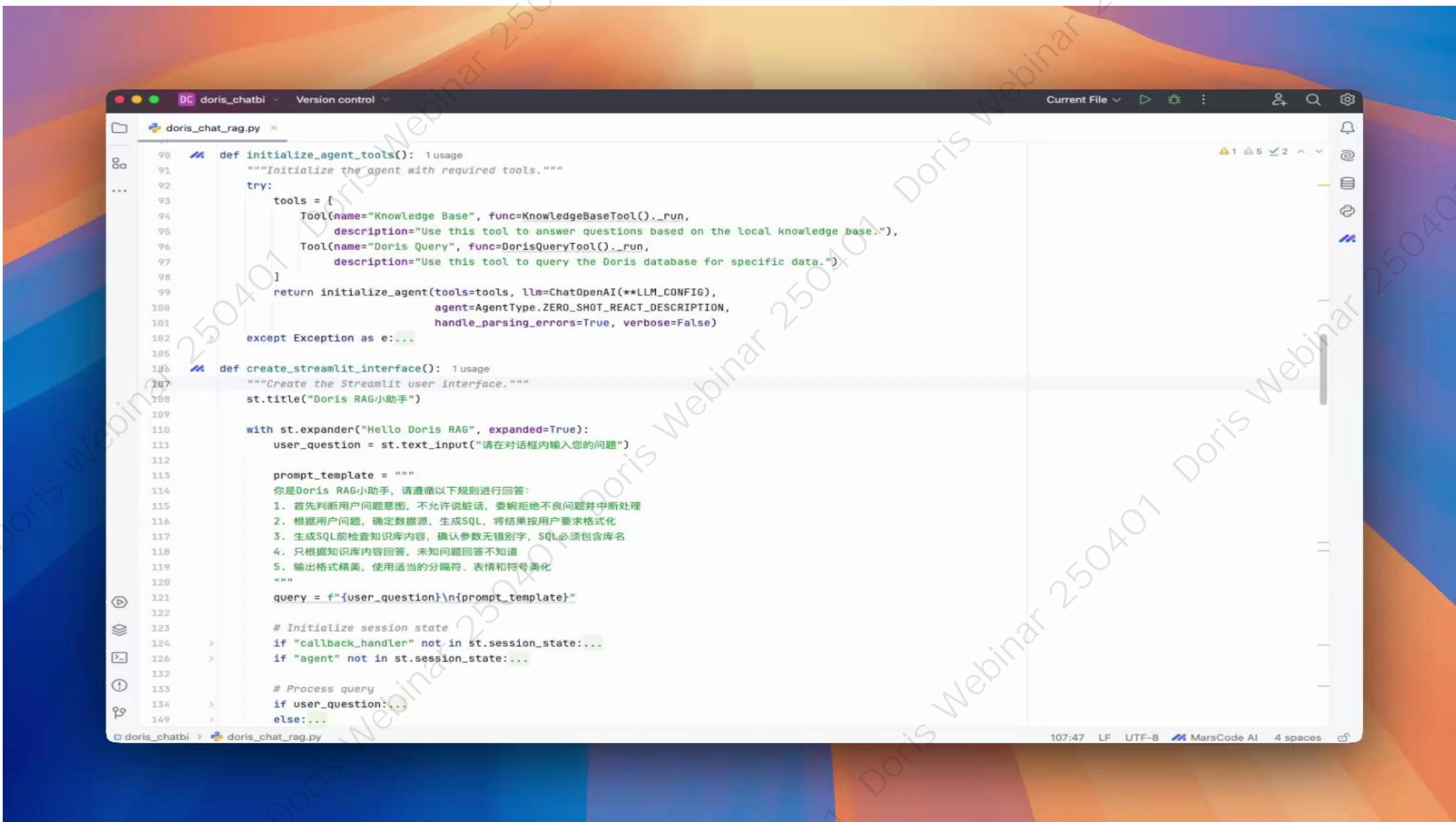


The screenshot shows a Mac desktop environment. On the left, there is a sidebar menu for the application 'doris_chatbi'. The menu items include: 消息 (highlighted with a red dot), AI助理, 文档, 多维表, 工作台, 通讯录, 动态, 会议, 钉盘, 日历, 待办, DING, 宜搭, 项目, 邮箱, 应用中心, 钉钉播, 收藏, and 更多. In the center, a code editor window titled 'doris_opinion_analysis.py' is open. The code is written in Python and uses the MarsCode AI plugin. The code defines a function to analyze complaints based on a template and an AI agent. The code editor interface includes tabs for 'Current File', 'Search', and 'File', along with various toolbars and status bars at the bottom.

```
agent="zero-shot-react-description",# 基于React（推理+行动）框架的Agent: 仅依赖工具的描述信息选择工具
verbose=True,
handle_parsing_errors=True
)
# 提示模板，指导AI代理如何处理查询
question = "请问近30天有关于一臻数据的投诉信息吗"
promptTemplate = """
你是“一臻数据”平台的投诉识别以及处理员，需完成以下任务：
1. 根据用户的问题，生成一个 SQL 查询，具体查询时间范围以用户提问问题为准。
2. 从查询结果中筛选出与“一臻数据”相关的投诉信息。
3. 将筛选结果汇总为易读的格式，并给出处理建议以及将命中的投诉信息根据你的建议排出一个紧急程度来，尽可能多的挖掘一些信息返回给客户，做专家指导，如“舆情分类”、“定义分类”、“紧急程度”
4. 如果存在与“一臻数据”相关的投诉，将汇总信息和你的处理建议通过钉钉机器人推送到指定群组。
5. 在推送内容中添加表情符号，增强可读性。
注意：
- 如需调用工具，根据工具的作用在该任务里调用合适的工具，并根据工具的要求传入需要的参数，要求参数传入格式准确
- 只能从获得的doris库中的数据中做分析，不允许自己fake数据。
数据库信息：
- 表名: yz_complaint_detail
- 字段: complaint_time (投诉时间) , detail (具体投诉内容)
- SQL 查询示例: SELECT detail FROM yz_complaint_detail WHERE complaint_time >= NOW() - INTERVAL 60 DAY;
"""
prompt = f"{question}\n{promptTemplate}"
# 运行Agent
try:
    result = agent.run(prompt)
    print("Final Result:")
    print(result)
except Exception as e:
    print(f"代理执行出错: {e}")
93:1 (139 chars, 6 line breaks) LF UTF-8 MarsCode AI 4 spaces
```

完整文字教程：<https://mp.weixin.qq.com/s/lk1d9eWogf6WzcC8RgyFow>

Doris x RAG



The screenshot shows a code editor window titled "doris_chatbi" with the file "doris_chat_rag.py" open. The code implements a RAG (Retrieval-Augmented Generation) system for the Doris database. It defines two main functions: `initialize_agent_tools()` and `create_streamlit_interface()`. The `initialize_agent_tools()` function initializes an agent with required tools, including a Knowledge Base tool and a Doris Query tool. The `create_streamlit_interface()` function creates a Streamlit user interface where users can input questions and receive answers generated by the agent. The code uses MarsCode AI for syntax highlighting and includes comments explaining the logic and rules for generating responses.

```
98     def initialize_agent_tools(): 1usage
99         """Initialize the agent with required tools."""
100        try:
101            tools = [
102                Tool(name="Knowledge Base", func=KnowledgeBaseTool().__run,
103                     description="Use this tool to answer questions based on the local knowledge base."),
104                Tool(name="Doris Query", func=DorisQueryTool().__run,
105                     description="Use this tool to query the Doris database for specific data.")
106            ]
107            return initialize_agent(tools=tools, llm=ChatOpenAI(**LLM_CONFIG),
108                                    agent=AgentType.ZERO_SHOT_REACT_DESCRIPTION,
109                                    handle_parsing_errors=True, verbose=False)
110        except Exception as e:...
111
112    def create_streamlit_interface(): 1usage
113        """Create the Streamlit user interface."""
114        st.title("Doris RAG小助手")
115
116        with st.expander("Hello Doris RAG", expanded=True):
117            user_question = st.text_input("请在对话框内输入您的问题")
118
119            prompt_template = """
120                你是Doris RAG小助手，请遵循以下规则进行回答：
121                1. 首先判断用户问题意图，不允许说脏话，委婉拒绝不良问题并中断处理
122                2. 根据用户问题，确定数据源，生成SQL，将结果按用户要求格式化
123                3. 生成SQL前检查知识库内容，确认参数无错别字，SQL必须包含库名
124                4. 只根据知识库内容回答，未知问题回答不知道
125                5. 输出格式精美，使用适当的分隔符、表情和符号美化
126            """
127
128            query = f"{user_question}\n{prompt_template}"
129
130            # Initialize session state
131            if "callback_handler" not in st.session_state:...
132            if "agent" not in st.session_state:...
133
134            # Process query
135            if user_question:...
136            else:...
```

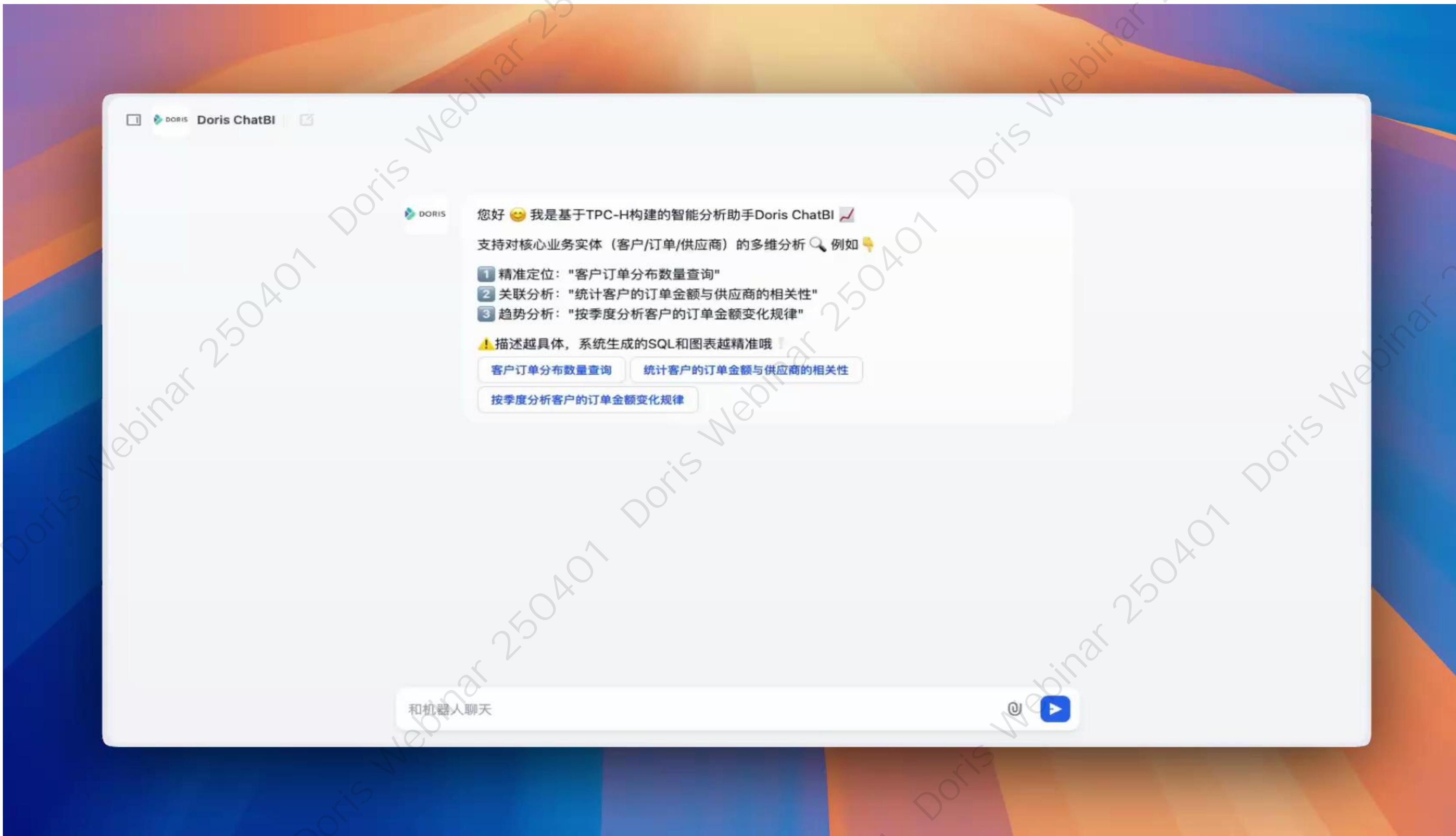
完整文字教程：https://mp.weixin.qq.com/s/GWXYAkooiBZUTmd_2Ovp4Q

Doris x ChatBI



1. 基于 TPC-H 数据进行实体构建
2. Apache Doris 2.1.8 作为数据底座
3. DeepSeek V3 LLM 进行 Text2SQL 及分析可视化

Doris x ChatBI



完整文字教程：https://mp.weixin.qq.com/s/ot9CZQo0iJvMhyB22_eAqQ

目录

Apache Doris x AI 关键特性

Apache Doris x AI 应用场景

Apache Doris x AI 案例演示

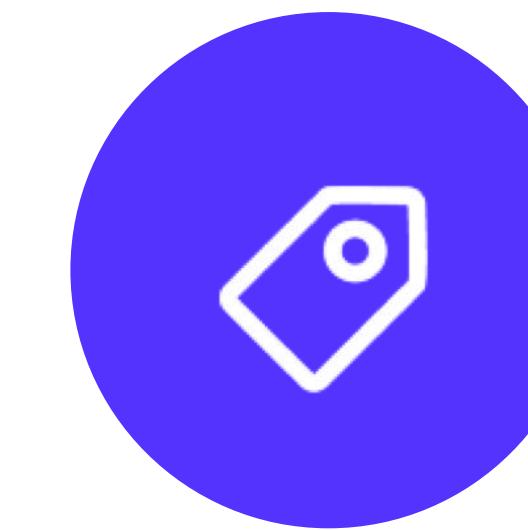
Apache Doris x AI 未来规划



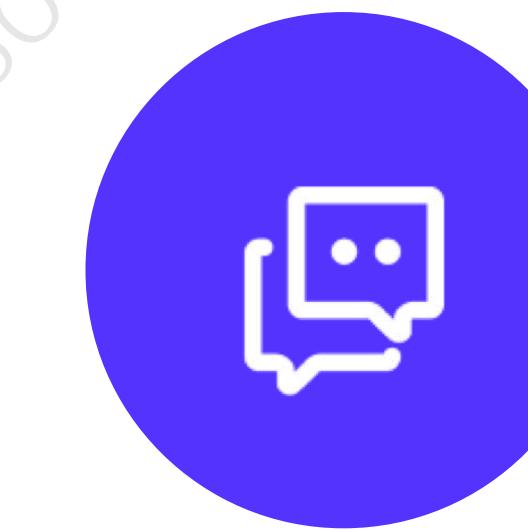
Doris x AI Overview



**Vector semantic
search**



**Data Preparation
& Feature Store**



Lakehouse Integration



ChatBI & Agent

Welcome to Doris Community

Subscribe

Subscribe to our mailing list and join our discussion:

dev@doris.apache.org

Get technical support

- Slack: apachedoriscosmunity.slack.com
- Wechat Group: Scan the QR code on the right.

Doris 官方小助手



Doris x AI 专项交流群



Thanks !

