大数据智能体工作台 DataBuddy

数据应用

腾讯云

【联系我们】

我们致力于为您提供个性化的售前购买咨询服务，及相应的技术售后服务，任何问题请联系 4009100100或95716。

文档目录

数据应用
智能体开发
Agent 概述

最近更新时间：2026-09-11 18:24:35

本文介绍 DataBuddy 中 Agent 的产品定位、Agent 类型与典型使用流程，帮助您建立对 Agent 模块的整体认知。

产品定位

Agent 模块是 DataBuddy 数据应用层提供的智能体平台，兼顾两类用户：

- **开发者**：通过高代码开发模板（OpenAI / LangChain / LangGraph），灵活定制复杂的企业级 Agent，并与 DataBuddy 数据与 AI 能力联动。
- **业务人员/分析人员**：通过智能调优等能力，让业务人员也可以通过反馈提升 Agent 的回答质量

平台为 Agent 提供从**创建、开发、调试、部署、运行** 到**评估、优化、运维** 的全链路能力，并以「安全、被治理」的方式调用 DataBuddy 内的数据与 AI 资源。

Agent 的两种工作模式

业内通常把 Agent 划分为两类，DataBuddy 都支持：

模式	说明	适用场景
工作流模式	在 Agent 内置预定义的工作流，通过流程编排获得**可预期、稳定** 的执行结果	任务定义清晰、对结果一致性要求高（如固定流程的客服问答）
自主决策模式	基于大模型推理动态决定调用什么工具、走什么路径	需要灵活性、面对开放式任务（如周报洞察、复杂分析助手）

实际项目中两种模式经常混合使用：业务推荐「由简到繁」，先以简单 API + Prompt 解决问题，只有需要决策时才上完整 Agent。

Agent 平台支持的搭建方式

平台首页（Agent 列表）提供以下搭建入口：

类型	说明	阶段
高代码开发 Agent — OpenAI 框架	以 OpenAI Agents 框架为基础的代码模板	已上线

高代码开发 Agent — LangGraph 框架	以 LangChain / LangGraph 框架为基础的代码模板	已上线

Agent 端到端流程

```
1. 创建 Agent（选择搭建方式 / 模板）
        |
        ▼
2. 配置工具与知识库（MCP）
        |
        ▼
3. 在 Studio 中开发
        |
        ▼
4. 调试：在 ChatUI 预览体验，查看 Trace
        |
        ▼
5. 部署：从 Studio 部署到运行环境
        |
        ▼
6. 运行：通过 ChatUI / API / 被其他 Agent 调用
        |
        ▼
7. 评估与优化：评估器打分、人工反馈、智能调优（ALHF）
```

我的智能体列表

进入 Agent 模块后，列表区会展示**当前用户具备使用或管理权限**的智能体（同一主账号下跨工作空间），列表字段：

字段	说明
Agent 名称	创建时填写
Agent 类型	自定义代码 Agent
Agent 描述	创建时填写的描述信息
Owner	Agent 的所有者
调用端点	Agent 可被调用的端点名称（App 类型即为 App 名称）

最近更新	最近一次更新时间，按从新到旧排序

配套基础设施

Agent 模块运行依赖以下基础设施，已默认在平台中提供：

基础设施	作用
MCP 服务列表	Agent 调用工具的标准协议层，详见 MCP 概述
模型列表	平台托管基础模型，Agent 直接通过模型 ID 调用，无需配置 API Key
MLflow 实验	承载 Agent 的 Trace、评估器、评估运行、版本，是评估与优化的载体
Studio	Agent 代码的开发与调试环境，详见 Studio 概述
Catalog	提供 Catalog 函数、Volume、向量索引等可被 Agent 调用的资产
Studio AI 辅助	在 Studio 中提供 AI 代码生成、工具发现等辅助能力，提升 Agent 开发效率

相关文档

- Agent 概述
- 创建与管理 Agent
- MCP 概述
- 模型列表概述

配置 Agent 工具与知识库

最近更新时间：2026-09-11 18:24:35

本文介绍如何为 Agent 配置可调用的工具（MCP 服务、Catalog 函数、模型服务），让 Agent 具备调用外部能力的能力。

概述

DataBuddy 中 Agent 通过工具（Tools）机制使用外部能力：

- **工具（Tools）**：调用 MCP 服务、Catalog 函数、模型服务、SQL 查询引擎、Volume、Jobs 等已封装好的能力。
- 工具的标准协议是 **MCP（Model Context Protocol）**，详细介绍见 MCP 概述。本文聚焦「如何把工具挂到 Agent 上」。

前提条件

- 已创建 Agent，详见 创建与管理 Agent。
- 当前账号对要使用的 MCP 服务、Catalog 函数具备访问权限。
- 如需新增 MCP 服务，需具备对应工作空间的 MCP 管理权限。

工具配置（高代码）

在高代码 Agent 中调用工具

高代码 Agent 通过 **DataBuddy SDK** 在代码中调用 MCP 工具，无需开发者额外处理用户鉴权：

用途	SDK 调用方向
获取大模型实例	直接通过模型 ID（如 `wedata-gpt-5-2`）创建模型客户端
调用 MCP 工具	通过 SDK 列出 / 调用当前账号下可见的 MCP 服务
记录 Trace	自动通过 MLflow SDK 写入对应实验

具体方法签名以 SDK 文档为准。开发流程：

1. 在 Studio 中打开 Agent 代码模板。
2. 在代码中按需 `import` DataBuddy SDK，按平台 `discover_tools` 等 Skills 检索可用 MCP。
3. 在 Agent 编排逻辑中显式声明可调用的工具集合。
4. 部署后 Agent 即可在运行时按需调用工具。

> **提示**

> 可在 Studio 中通过 AI 辅助工具自动检索平台上可用的 MCP 服务，避免手工查找。

其他配置（用户授权与计费）

项目	说明
用户鉴权	Agent 调用 MCP 时使用 Agent 自身的服务账号，无需开发者在代码中处理 OAuth 鉴权
调用日志	所有工具调用、模型调用都会自动写入 Agent 关联的 MLflow 实验，详见 Agent 调试与测试
计费	当前暂不计费，正式商用上线后另行通知

常见问题

Q：MCP 工具的鉴权信息（API Key、Token）怎么管理?

A：MCP 工具的鉴权信息在 MCP 概述 的「添加外部 MCP 工具」流程中配置，存储在工作空间的安全凭据库中。Agent 调用时由平台自动注入，开发者代码中**不需要** 直接处理 API Key。

相关文档

- 创建与管理 Agent
- Agent 调试与测试
- MCP 概述
- 模型列表概述

Agent 调试与测试

最近更新时间：2026-09-11 18:24:35

本文介绍如何在 DataBuddy 中调试与测试 Agent，覆盖 ChatUI 预览体验、Trace 追踪、评估器、评估运行和智能调优（ALHF），帮助您快速定位问题并持续优化 Agent 质量。

概述

Agent 的调试与测试通过四个层次完成：

层次	解决什么问题	入口
ChatUI 预览体验	直接对话验证 Agent 是否能正常工作	Agent 概览页右侧
Trace 追踪	排查具体一次调用为什么慢 / 出错 / 答错	MLflow 实验 > 跟踪
评估器（Evaluator）	自动化打分（相关性、安全性、正确性），规模化评估	MLflow 实验 > 评估器
智能调优（ALHF）	把人工反馈转化为 Agent 行为改进	Agent 详情页 > 智能调优

前提条件

- 已创建并启动 Agent，详见 创建与管理 Agent。
- 当前账号对 Agent 关联的 MLflow 实验具备 `CAN_EDIT` 权限。

使用限制

限制项	说明
单个实验下 Trace 数量	最多支持 100,000 条
LLM 评估器类型	支持 LLM 评估器（相关性 / 安全性 / 正确性），代码评估器后续支持
评估运行对比	支持单 Run 查看，多 Run 横向对比后续版本支持
评估反馈 / 期望卡片	单条 Trace 的反馈、期望数量上限详见平台说明

ChatUI 预览体验

入口

进入 Agent 概览页，右侧默认嵌入 ChatUI 模板预览体验窗口；也支持在新标签页独立打开。

能力清单

能力	说明
消息发送	文本输入，Enter 发送，Shift + Enter 换行；模型回答中不能发送新消息，可点击**停止**
思考过程	模型回答前显示「思考中」；思考完成后展示意图理解、执行步骤、模型输出
工具调用展示	区分 Tools / Agent 调用，调用过程显示 Loading，完成后打勾并显示耗时；点击展开查看输入 / 输出 / 耗时
流式输出	模型输出按流式渲染，支持 Markdown 富文本（代码块 / 表格 / 列表）
回答操作	点赞 / 点踩（结果会进入 ML Trace） / 复制回答内容
会话上下文	单个会话内模型自动联想上下文
新增会话	关闭当前会话开启新会话
会话历史	需指定一个可存储会话历史的数据库，详细配置详见平台说明
推荐问题	初始态展示固定推荐问题，点击可直接对话

> **提示**
> ChatUI 是 Agent 上线后用户实际看到的对话界面。建议在调试阶段就把 Agent 的「开场白、推荐问题」配置到位，提升上线后的首次使用体验。

Trace 追踪（可观测）

每次 Agent 问答都会生成一条 Trace，自动落入关联的 MLflow 实验。Trace 包含完整调用链：模型调用、工具调用、子 Agent 调用、Token 消耗、耗时等。

进入入口

1. 在 Agent 概览页点击 **追踪评测** 模块快捷入口。
2. 或从 MLflow 实验列表打开实验 → 左侧 **可观测性 > 跟踪** 。

Trace 列表字段

字段	说明
Trace ID	全局唯一 ID，点击打开 Trace 详情
Request	请求内容
Response	响应结果

Session	会话 ID
User	执行用户名
Trace name	Trace 命名
Version	Agent 版本，点击跳转到对应版本
Tokens	消耗 Token 数
Execution time	执行时长
Request Time	请求时间
Run name	关联运行名称（如来自评估运行），无则展示空
Source	产生 Trace 的入口点或脚本名称
State	状态：正常 / 错误

支持自定义字段过滤、按 Request 模糊搜索。

Trace 详情：总览

点击 Trace ID 打开详情弹窗。「总览」展示如下：

区域	说明
基础信息	状态（OK / ERROR / IN_PROGRESS）/ Trace ID / Token 消耗 / Trace 耗时
输入	用户输入内容、上下文
调用链	大模型 / 子 Agent / 工具的实际调用链；默认收起，点击展开查看每一步的输入 / 输出 / 耗时
输出	Agent 给用户的最终输出
评估项	相关性、安全性、用户反馈等评估结果（仅展示，不支持编辑）

Trace 详情：详情 & 时间线

切换到「详情 & 时间线」Tab，可看到树形结构展示的所有 Span（每个 Span 是一个独立操作单元，类型包括大模型调用、工具调用、Agent 调用等）。每个 Span 提供：

- **聊天：** 当 Span 输入、输出符合 OpenAI Chat Completions 消息格式时自动渲染对话视图。
- **输入、输出：** 按需展示 Span 的输入信息和输出内容（JSON 自动格式化）。
- **属性：** 键值对形式展示模型、工具等结构化属性。
- **事件：** Span 执行期间的离散事件（如异常堆栈、错误信息）。

支持搜索关键词，覆盖各节点的调试信息，过滤左侧目录树。

Trace 详情：右侧评估区

每条 Trace 的详情弹窗右侧是评估区，可对该 Trace 添加：

评估类型	用途
反馈（Feedback）	对 AI 已生成回答的质量评分（如「这次回答好不好」）
期望（Expectation）	对该问题的「理想回答」应该满足的约束（如「必须引用知识库」）

每条反馈、期望支持以下字段：

字段	取值范围
评估类型	反馈 / 期望
评估名称	⩽ 100 字符；不支持名称中含 `.` 字符
数据类型	布尔型 / 字符串 / 数字
评估值	字符串 ⩽ 500 字符
评估依据	给出该评估的原因，便于后续追溯 / 训练，⩽ 500 字符

反馈、期望卡片支持展开、收起，可编辑、删除。

评估器（Evaluator）

评估器是对 Agent 输出做**自动化打分** 的逻辑。当前支持 LLM 评估器，代码评估器后续版本支持。

进入入口

1. 进入 Agent 关联的 MLflow 实验。
2. 左侧导航 **评估 > 评估器** 。

预置 LLM 评估器

平台预置 3 类 LLM 评估器：

类型	评估目标	输出
正确性	答案是否事实正确（无需自定义 Prompt，下拉选择即可）	yes / no
相关性	答案是否回答了问题（不关注正确性 / 完整性）	yes / no

安全性	内容是否违反安全策略（仇恨言论、骚扰、暴力煽动等）	yes / no

创建 LLM 评估器

步骤 1：打开新建弹窗

在评估器列表点击 **新建 LLM 评估器** 。

步骤 2：配置通用、评估标准、自动评估

Tab	配置项
通用	评估器名称、评估描述
评估标准	选择评估模型（下拉选择已有大模型）、选择内置 LLM 评估器（正确性 / 相关性 / 安全性）
自动评估	是否开启自动评估、采样率、筛选字符串（仅对符合筛选条件的 Trace 自动跑评估）

步骤 3：保存并查看评估器卡片

保存后回到评估器列表，每张卡片显示：

- 评估器名称
- 评估器类型（LLM 评估器）
- 是否在所有未来跟踪上运行
- 采样率（基于已配置 Trace 结果计算）
- 评估跟踪开启、关闭状态
- 编辑按钮、更多（删除）

卡片支持展开、收起，编辑后即时生效。

评估运行（Evaluation Runs）

评估运行用于**展示和管理** 对 Agent 做的多次评估结果，方便对比不同版本、不同提示词配置的表现。

进入入口

进入 MLflow 实验 → **评估 > 评估运行** 。

评估运行列表

字段	说明
运行名称	自定义名称（如「人工审查 v3」）
创建于	评估创建的日期时间

数据集	关联的评估数据集名称，可点击跳转
版本	关联的 Agent 版本，无则显示 –
状态颜色标识	红 / 绿 / 黄 / 棕等彩色圆点，自定义标记
内置 LLM 评估维度	当实验配置了内置 3 类评估器时，列表显示对应维度的评估结果

操作：

- **删除**：勾选单个、多个运行删除（需确认）。
- **比较**：勾选 2 个运行进入对比视图（后续版本支持）。
- **刷新**：拉取最新数据。

单 Run 详情：会话与跟踪列表

点击某个运行后，右侧加载该运行下的所有会话与跟踪：

字段	说明
会话	仅在「按会话分组」开启时显示
跟踪 ID	唯一标识单次模型调用，可点击展开 Trace 详情
请求	用户输入
响应	模型应答
执行时间	精确到毫秒
状态	确定 / 错误

支持按 Trace ID / 请求、响应 关键词搜索。
支持通过列配置选择展示属性、评估、期望三类字段；评估、期望项默认展示 3 个，可在列配置中自定义数量。

智能调优（ALHF）

智能调优是 DataBuddy 面向业务人员的核心差异化能力，通过「人类反馈 + AI 推荐」闭环让 Agent 自动优化。

调优四阶段

```
1. 配置阶段：提供参考问答对（数据来源：Trace 历史 / 文件上传 / 手动输入）
        |
        ▼
2. 反馈阶段：对推荐问答评估"好 / 不好 / 跳过"
        |
```

```
        ▼
3. 优化阶段：系统自动提取语义记忆，生成推荐 Guidelines
        |
        ▼
4. 保存并更新：自动优化Agent的Prompt、 MCP描述等，提升Agent效果
```

入口

1. Agent 概览页 → 智能调优入口卡片 → 点击 **开始调优** 。
2. 或 Agent 详情页顶部导航 → 点击 **智能调优** Tab。

步骤 1：配置参考问答对

进入「问答配置」面板，选择问答对来源：

来源	适用场景	操作要点
从 Trace 历史导入	Agent 已有线上请求记录	穿梭框弹窗选择，至少选 3 条 Trace
从本地文件上传	问答对存在本地文件中	下载模板，上传 xlsx / csv，单文件 ≤ 200 MB
手动输入	冷启动，无历史数据	每行两个输入框（问题 + 预期答案），至少 3 组并填完整

预览页可再次确认数量、删除条目（保留至少 3 条），可手动新增问题与预期答案。

> ⓘ **提示**
> 保存覆盖时，问题没有修改的条目 ID 不变，原有的反馈记录、反馈历史不会丢失。

步骤 2：推荐反馈

系统按以下逻辑推荐让用户反馈的对话：

- 当前 Guidelines 尚未覆盖的典型场景。
- 模型置信度较低的输出。
- 冷启动阶段：直接展示前 50 条配置好的问答对。

每张卡片支持三种操作：

操作	含义
好	正向样本，存入情景记忆强化 Agent 行为
不好	负向样本，需展开文本框 + Ctrl + Enter 提交「不好的原因和期望答案」

跳过	记录但不作为优化依据；如系统认为仍需推荐，后续可继续推送

卡片评估后会有飞出动画，标题栏实时显示「剩余 X 条」。动画完成后才能操作下一张。

步骤 3：问答检查

通过「问答检查」模块查看所有已配置的问答，逐条对比 Agent 回答与预期答案：

- 支持按关键字模糊搜索（覆盖问题和预期答案）。
- 支持过滤「反馈过」的问答（已有反馈记录）。
- 单条详情：展示问题、预期答案、Agent 实际回答；可点赞、点踩，与「推荐反馈」交互一致。
- **历史反馈**：展示最近 3 次反馈内容（不含「跳过」）。

步骤 4：评估准则与提示词

评估准则（Guidelines）

定义「什么是好的回答」，每条 Guideline 同时用于：

- LLM Judge 评估打分。
- 提示词优化时的约束条件。

系统会基于以下信息生成推荐 Guidelines：

- Agent 名称与描述。
- 用户配置的问答对。
- 用户在推荐反馈和问答检查中给出的反馈。

操作：

- 逐条审查推荐 Guideline。
- 点击 **Accept** 采纳，或 **Reject** 拒绝。
- 也可点 **+ Add** 手动添加。
- 采纳后按钮变为「已采纳」（不可再次点击 Accept），但已采纳的 Guideline 在编辑区可删除。

提示词（Instructions）

定义 Agent 的高层目标和核心任务（如「你是一个专业的客服数据分析助手……」），控制回答风格、内容边界。

- Instructions 是**方向性引导**。
- Guidelines 是**具体评估和约束标准**。
- 两者互补，建议同步维护。

步骤 5：保存并更新

点击右下角 **Save and update** 按钮，平台引擎自动把反馈和规则变更拆解，应用到：

- Prompt（提示词）
- Vector Index（向量索引）

- LLM Judge（评估器）
- Evaluation Dataset（评估数据集）
- MCP Tools（工具配置）
- Agent Config（Agent 配置）

> **提示**
> 当前更新生效存在约 30 秒延迟。保存成功后会提示「已保存更新，预计将在 30 秒内生效，请稍后前往预览体验」，并提供 **前往预览体验** 快捷链接。

保存成功后，推荐反馈的内容也会异步刷新。

常见问题

Q：智能调优中「好、不好、跳过」三种操作的差别?

A：**好** 会强化当前回答模式（正向样本）；**不好** 需要给出原因和期望答案，作为负向样本和改进方向；**跳过** 不参与优化，但系统认为仍有价值时会再次推荐。

Q：评估器为什么要预置「相关性、安全性、正确性」三种?

A：这三种覆盖了 LLM 应用最常见的质量维度：相关性（是否回答了问题）、安全性（是否违反安全策略）、正确性（事实是否正确）。可以满足绝大多数初期评估需求，更细分的评估维度可通过自定义 LLM 评估器实现。

Q：如何在评估运行中对比两个版本的 Agent?

A：当前暂不支持多 Run 横向对比，后续版本会支持勾选 2 个运行进入对比视图。当前可通过为不同版本打不同的「版本标识」，再分别查看 Run 详情人工对比。

相关文档

- 创建与管理 Agent
- 配置 Agent 工具与知识库
- 发布与部署 Agent
- Studio 概述

发布与部署 Agent

最近更新时间：2026-09-11 18:24:35

本文介绍如何在 DataBuddy 中发布和部署 Agent，覆盖部署来源、部署流程、运行状态监控、版本管理与下载。

概述

Agent 在平台上以 **App 形态** 部署运行：代码、依赖、运行状态、版本和权限统一由 App 管理。当前支持从 Studio 部署，从 Git 部署后续版本支持。
部署完成后，Agent 即可通过 ChatUI、API 或被其他 Agent 调用。

前提条件

- 已创建 Agent，详见 创建与管理 Agent。
- 高代码 Agent 的代码已在 Studio 中完成开发与基本调试。
- 关联的 MLflow 实验有 `CAN_EDIT` 权限。
- 平台基础模型已在白名单中。

使用限制

限制项	说明
部署来源	当前仅支持从 Studio 部署；从 Git 部署后续版本支持
部署文件管理	支持点击下载 Agent 代码包
平台基础模型	通过白名单开通，可统计用量
计费	当前暂不计费，正式商用上线后另行通知

操作步骤

步骤 1：在 Studio 中开发、调整代码

1. 进入 Agent 概览页，点击 **代码开发** 模块快捷入口，或顶部导航 **代码开发** Tab。
2. Studio 自动打开 Agent 关联的代码模板（OpenAI / LangGraph）。
3. 在 Studio 中：
   - 编辑代码、调试。
   - 修改环境变量。
   - 通过 Studio 的 AI 辅助能力（代码生成、工具发现等）辅助开发。

> **提示**
> 代码同步规则：
> - Studio 中的代码会同步到 COS。
> - 在 Agent 详情页点击下载按钮时，平台会调用 Studio 接口拉取**最新代码** 打包下载。
> - 部署始终以 Studio 中**最新代码** 为准；用户在本地修改后回传，应同步到 Studio。

步骤 2：触发部署

在 Agent 详情页顶部状态卡片中，点击 **部署** 按钮（仅在状态允许部署时可点击，详见 状态流转 章节）。
平台执行以下动作：

1. 从 Studio 拉取 Agent 最新代码。
2. 安装代码依赖、应用环境变量。
3. 启动 Agent 运行实例。
4. 自动注册调用端点（Endpoint），可被 ChatUI、API 等调用。

部署进度通过进度条实时展示。

> **注意**
> 部署过程中 Agent 不可对外提供服务。如需保证可用性，建议错峰部署或提前通知用户。

步骤 3：在概览页观察运行状态

部署完成后回到 Agent 概览页，可查看：

指标	说明
今日调用	当日 0:00 至当前的总请求次数（含成功 / 失败），点击跳转到追踪评测
成功率	成功响应数 / 总请求数 × 100%（成功 = HTTP 2xx 且正常返回），点击跳转到运行监控
平均延迟	成功请求的端到端响应时间算术平均值，点击跳转到运行监控
当前版本	线上运行的部署版本号，点击跳转到部署状态
今日 Token 消耗	当日所有 Trace 的 Token 消耗之和，按千分位（k / m / b）展示
近 30 日 Token 消耗	近 30 日所有 Trace 的 Token 消耗之和，按千分位展示

步骤 4：测试部署后的 Agent

通过以下任一方式测试：

方式	说明
ChatUI 预览体验	Agent 概览页右侧直接对话
API 调用	通过 Endpoint URL（在 Agent 概览页 info 浮层中查看 / 复制）发起 HTTP 请求

步骤 5：必要时下载代码包

在 Agent 概览页或代码开发模块点击 **下载**，平台会从 Studio 拉取最新代码并打包为 zip 文件下载到本地。

状态流转

Agent 在生命周期中存在以下状态：

当前状态	可执行操作	下一状态
已停止	启动	启动中
启动中	—	部署中（仅首次创建完智能体后）/ 运行中
部署中	部署	运行中（成功）/ 上一状态（失败）
运行中	停止 / 部署	停止中 / 部署中
停止中	—	已停止
出错	启动 / 停止	启动中 / 停止中

不同状态下页面显示对应的运行视图与可执行操作；当 Agent 处于「出错」状态时，点击错误信息会展示具体的错误原因。

部署进度展示

部署中页面会展示进度条，分阶段标注：

- 拉取代码
- 安装依赖
- 启动实例
- 注册端点

每个阶段失败时，会回退到部署前的上一状态，并保留错误信息便于排查。

权限管理

通过 Agent 详情页右上角配置部署相关权限：

权限点	含义

可使用	可对话调用、可查看 Trace
可管理	可触发部署 / 启动 / 停止、可下载代码、可修改配置

授权范围覆盖**整个主账号下跨工作空间**的成员。

版本管理

Agent 部署后，每次重新部署会生成一个新的版本：

- 概览页「当前版本」展示线上运行版本号。
- 点击进入部署状态页可查看历史版本列表。

> **提示**
> 当前版本管理主要用来追溯「当前线上运行的是哪一次部署」。完整的版本回滚、多版本灰度等能力以后续版本上线时间为准。

运行监控

进入 Agent 详情页 **部署运维** Tab，可查看：

模块	说明
运行监控	调用量趋势、成功率趋势、平均延迟趋势
部署状态	当前部署进度、历史部署记录

调用方式

部署成功后，Agent 提供以下调用方式：

方式	适用场景	说明
ChatUI	业务用户直接对话	平台预置的前端模板，开发者可二次开发
HTTP API	集成到自有应用	通过 Endpoint URL + 鉴权调用
模型服务节点	在 Agent 之间互相调用 / 在 SQL Editor 中调用	通过 MCP 协议调用，需对应配置

常见问题

Q：部署失败如何排查?

A：常见原因有：① 代码依赖安装失败（查看部署进度条上的错误提示）；② MLflow 实验权限不足（确认账号对实验有 `CAN_EDIT` 权限）；③ 平台基础模型未开通白名单（联系管理员）；④ 调用的 MCP 服务不可用（在 MCP 概述 列表确认服务状态）。

Q：如何回滚到上一个版本?

A：当前主要支持「重新部署」。如需回滚，建议：① 在 Studio 中通过 Git 切换到对应版本的代码；② 或下载之前版本的代码包，本地修改后回传 Studio 再部署。完整版本管理与回滚能力以后续上线时间为准。

Q：部署时是否支持灰度?

A：当前暂不支持灰度发布。建议在测试阶段使用一个独立的「测试 Agent」，验证完成后再部署到生产 Agent。后续版本会引入多版本与灰度能力。

Q：Agent 处于「运行中」状态但调用没有响应怎么办?

A：先在概览页查看「今日调用」和「成功率」是否异常下降；再到 Trace 列表查看最近的 Trace 状态，定位具体失败原因（多数是工具调用失败、模型不可用）。如果状态指标异常，建议先停止 Agent 再重新启动。

Q：从 Git 部署什么时候支持?

A：当前仅支持 Studio 部署。Git 部署在后续版本中提供，具体时间以平台 Roadmap 公告为准。

相关文档

- 创建与管理 Agent
- 配置 Agent 工具与知识库
- Agent 调试与测试
- Studio 概述

MCP
什么是 MCP

最近更新时间：2026-09-11 18:24:35

MCP（Model Context Protocol，模型上下文协议）是一种用于将外部工具、数据源、业务能力以**标准化协议**暴露给大模型 / Agent 调用的开放协议。在 DataBuddy 中，MCP 是 Agent 调用平台数据资产的**统一桥梁**，Agent 即可在对话或任务中按需发现、调用这些工具，无需为每种数据源或服务单独编写适配代码。
举一个典型例子：一个智能问数 Agent 需要查询销售明细表、调用 Catalog 中已注册的 `compute_gmv()` 函数、再通过外部 API 获取实时汇率。借助 MCP，您只需要把销售明细的 Catalog 函数、外部 HTTP API 等注册为 MCP 服务，Agent 即可像调用本地函数一样统一调用它们。

核心能力

能力	说明
协议统一	不同来源的工具（Catalog 函数 / 外部 HTTP / 第三方插件）均通过同一套协议暴露给 Agent，Agent 侧无须了解底层差异。
多来源接入	同时支持平台内置工具、用户自定义工具、外部 MCP / API、MCP 插件市场，覆盖企业常见集成场景。
凭证集中管理	外部 MCP / API 的身份认证通过 Connection 统一托管。
与 Agent 无缝集成	MCP 服务可直接挂载到 Agent 调试会话中，调用记录会同步进入 MLflow Tracing。

MCP 服务类别

DataBuddy 一期支持以下几种 MCP 服务来源，您可以在「Agents > MCP」列表中集中管理。

类别	适用场景	创建方式
Catalog 函数（UC Function）	把 Catalog 中已注册的 SQL / Python 函数（如指标计算、数据库查询）作为工具调用。	在 SQL Editor 或 Notebook 中创建函数后自动暴露为 MCP 工具。
外部 MCP / API 服务	接入企业自有 MCP Server 或第三方 HTTP API。	在 MCP 列表点击**添加外部 MCP 工具**，通过 `mcp.json` 配置。
合作伙伴 MCP 插件市场	一键安装官方 MCP 插件。	在 MCP 列表点击 **从市场发现更多 MCP 服务**，进入插件市场安装。

关键概念

理解 MCP，需要熟悉以下核心概念：

- **MCP 服务（MCP Server）**：一组对外暴露的工具集合，对应一个可调用的 MCP 服务地址。
- **工具（Tool）**：MCP 服务下的最小调用单元，包含名称、描述、输入参数 Schema 与输出结构 Schema。一个 MCP 服务最多可包含 **1000 个工具** 。
- **Connection**：用于管理外部 MCP / API 的连接配置与凭证。
- **MCP 服务地址**：技术侧生成的、可被 Agent / SDK 调用的 URL，支持复制。
- **MCP 服务状态**：标识当前 MCP 是否可用。列表中显示的是上次查看时的状态，**不会实时刷新**；进入详情页或点击**刷新** 时才触发最新状态校验。

> **提示**
> 完整术语表请参考 核心概念与术语表 。

适用场景

场景	适用方式	推荐 MCP 类别
Agent 查询业务数据库 / 计算指标	把已有 SQL / Python 函数注册为 MCP 工具	Catalog 函数
Agent 调用企业内部已有的 API / MCP Server	自定义 MCP	外部 MCP / API 服务
让 Agent 具备文档抽取 / 联网搜索 / 第三方数据接入能力	一键安装，自动完成参数配置	MCP 插件市场

使用流程

一个典型的 MCP 使用闭环包含以下步骤：

1. **创建或安装 MCP 服务**：在 MCP 列表中创建 Catalog 函数、外部 MCP，或从插件市场安装合作伙伴 MCP。
2. **查看 MCP 详情**：进入详情页确认服务状态、复制服务地址、浏览工具及参数定义。
3. **绑定到 Agent**：在高代码 Agent 中通过 SDK 调用 MCP 工具。
4. **运行与观测**：Agent 调用 MCP 的全过程会记录到 MLflow Tracing，可在实验模块查看每次调用的输入、输出、耗时与状态。

子功能导航

子功能	说明	文档

MCP 服务配置	创建、查看、管理各类 MCP 服务	配置 MCP 服务
MCP 与 Agent 集成	在 Agent 中使用 MCP 工具	MCP 与 Agent 集成

推荐学习路径

如果您是首次使用 MCP，建议按以下顺序阅读：

1. **了解核心概念：** 本文。
2. **创建第一个 MCP 服务：** 配置 MCP 服务，从 Catalog 函数或安装一个 ADP 插件开始最为简单。
3. **在 Agent 中调用工具：** MCP 与 Agent 集成。
4. **了解 Agent 总体能力：** Agent 概述。

相关文档

- Agent 概述
- 创建与管理 Agent
- 配置 Agent 工具与知识库
- Model List 概述

配置 MCP 服务

最近更新时间：2026-09-11 18:24:35

概述

本文介绍如何在 DataBuddy 中查看、创建与管理 MCP 服务，包括三种典型来源：基于 Catalog 函数自动暴露、添加外部 MCP / API、从合作伙伴插件市场一键安装。配置完成的 MCP 服务可被高代码 Agent 直接调用。

前提条件

在开始本操作前，请确保满足以下条件：

- 已开通 DataBuddy 服务并完成空间初始化。
- 当前账号在目标 Workspace 中拥有 **Agent 开发者** 或更高角色（角色权限说明详见 成员与权限管理 ）。
- 根据要创建的 MCP 类别，准备相应资源：
  - **Catalog 函数**： 已在 Catalog 中创建可用的 UC Function（在 SQL Editor 或 Notebook 中创建）。
  - **外部 MCP / API**： 已知外部服务的地址、认证方式和凭证。
  - **合作伙伴插件**： 根据插件需要，准备腾讯云 AK/SK、API Key 或用户名密码等凭证。

使用限制

限制项	说明
单个 MCP 服务工具数	最多 1000 个
外部 MCP 添加方式	一期支持通过 `mcp.json` 录入
MCP 服务状态刷新	列表展示上次查看时的状态，**不实时刷新** ，需进入详情页或点击 **刷新** 触发最新校验

进入 MCP 列表

1. 登录 DataBuddy 控制台 。
2. 在顶部切换到目标地域和 Workspace。
3. 在左侧导航栏选择 **数据应用 > Agents** ，进入 **MCP** 标签页。

MCP 列表展示当前 Workspace 内所有可用的 MCP 服务，包括服务名称、类别、作者、工具数、更新时间和服务状态。您可以在列表上方按名称模糊搜索、按类别过滤，或点击 **刷新** 触发选中 MCP 的状态校验。

操作步骤

方式一：基于 Catalog 函数创建 MCP 服务

Catalog 函数创建后会自动以 MCP 服务的形式暴露，无需在 MCP 页面单独新建。

1. 进入 **数据工程 > Studio**，打开任意 Notebook 或 SQL 脚本。
2. 编写并执行 `CREATE FUNCTION` 语句注册函数到指定的 Catalog 与 Schema 下，例如：

```
CREATE OR REPLACE FUNCTION my_catalog.sales.compute_gmv(
  start_date STRING COMMENT '统计起始日期，格式 yyyy-MM-dd',
  end_date STRING COMMENT '统计结束日期，格式 yyyy-MM-dd'
)
RETURNS DOUBLE
COMMENT '计算指定时间区间内的 GMV'
RETURN (
  SELECT SUM(amount) FROM my_catalog.sales.orders
  WHERE order_date BETWEEN start_date AND end_date
);
```

1. 函数创建成功后，回到 MCP 列表，类别为 **Catalog 函数** 的 MCP 中即可看到该函数对应的工具。
2. （可选）点击列表项进入详情页，确认工具名称、参数描述与服务地址。

> **提示**
> 工具名称、参数说明、必填性来自函数定义中的参数名与 `COMMENT`。建议在创建函数时为每个参数填写清晰的中文或英文 `COMMENT`，Agent 才能更好地理解何时调用该工具。

方式二：添加外部 MCP / API 服务

如果您已有自建的 MCP Server 或希望接入第三方 HTTP API，可以将其作为外部 MCP 服务录入。

1. 在 MCP 列表点击 **添加外部 MCP 工具**。
2. 选择录入方式：

- **mcp.json 录入**：在文本框中粘贴标准 MCP JSON 配置，平台保存后会进行连通性校验，校验状态可在详情页中查看。

1. 配置 **是 MCP 连接**：在连接详情步骤中默认勾选且不可取消。
2. 点击 **添加**，提交后界面会提示添加成功或失败。
3. 添加成功后，新 MCP 出现在列表中（类别为 **外部 API**），点击进入详情页查看校验状态、服务地址与工具列表。

> **注意**
> 对于通过 `mcp.json` 录入的服务，校验失败时 MCP 状态会标记为不可用，但记录仍会保留。请进入详情页查看错误原因，修正配置后重新校验。

方式三：从插件市场一键安装 MCP

MCP 插件市场提供开箱即用的 MCP 工具，可以快速接入各类第三方能力。

1. 在 MCP 列表点击 MCP 市场。
2. 在插件详情页右上角点击 **安装**：
   - **无需认证的插件**：直接完成安装，安装后即出现在 MCP 列表中。
   - **需身份认证的插件**：弹出安装弹窗，按提示填写凭证（详见下方 插件身份认证），勾选 **同意用户协议和隐私政策** 后点击 **安装**。
1. 安装成功后会有 toast 提示，列表中可看到新增的 MCP 服务。

查看 MCP 详情

在 MCP 列表中点击任意一项进入详情页，可查看以下信息：

区块	内容
头部信息	MCP 名称、类别、描述、作者（官方 MCP 显示为「Wedata 官方」）、工具数、更新时间
服务状态	当前 MCP 是否可用，进入详情页或点击 **刷新** 时触发最新校验
服务地址	用于 Agent / SDK 调用的 URL，点击复制按钮提示「复制成功」
工具列表	每个工具展示名称、描述、输入参数 Schema 与输出结构 Schema

参数说明

外部 MCP 身份认证

外部 MCP 当前支持 `mcp.json` 录入。

验证结果

操作完成后，您可以通过以下方式确认 MCP 服务可用：

1. 在 MCP 列表中找到刚创建或安装的 MCP，**服务状态** 显示为可用。
2. 进入详情页，点击 **刷新** 后状态保持可用，工具列表与参数 Schema 正常显示。

常见问题

Q：为什么 MCP 列表里看到的状态和实际不一致?

MCP 服务状态在列表中**不实时刷新**，显示的是上一次查看时的状态。请进入详情页或点击 **刷新** 触发最新校验。

Q：通过 `mcp.json` 添加外部 MCP 后状态显示不可用怎么办?

请检查以下几点：

1. JSON 格式是否合法、必填字段是否完整。
2. 服务地址是否可从 DataBuddy 平台所在网络访问（注意内网/VPC 隔离）。
3. 凭证是否正确、是否过期。
4. 校验失败的具体错误在详情页中可见，按提示修正后重新校验。

相关文档

- 什么是 MCP
- 配置 MCP 服务
- 创建与管理 Agent
- 配置 Agent 工具与知识库

模型列表
模型列表概述

最近更新时间：2026-09-11 20:57:31

模型列表（Model List）是 DataBuddy 数据应用模块下统一汇集**大语言模型（LLM）端点**的入口。在这里您可以查看平台托管的开箱即用基础大模型，并以统一的方式被 Agent、MCP、Notebook、SQL Editor、Buddy 等模块调用。

模块定位

模型列表解决的是「LLM 在哪里、怎么调」这个基础问题：

- **对 Agent 开发者**：在 Agent 代码中只需写一行 `model="wedata-gpt-5"`，无需在工程中维护 API Key、Endpoint URL、超时与重试逻辑。
- **对数据分析师**：在 Notebook 或 SQL Editor 中直接调用 `ai_query()` 等函数，把 LLM 当成一个普通的 SQL UDF。
- **对管理员**：所有大模型调用经平台托管 Endpoint 路由，方便统一监控调用量、审计调用记录、控制权限与配额。

> **提示**
> **模型列表 vs. 数据科学模型管理**
> 模型列表（7.3）面向的是 **Agent / 应用调用的大语言模型（LLM）端点**，是数据应用模块的资产。
> 数据科学的 模型管理 面向的是 **MLflow 注册的机器学习模型（ML Model）**，由数据科学家训练并注册到 Catalog。
> 二者在产品中是相互独立的两个列表，本文仅描述前者。

可用模型

平台预置并托管主流 LLM，用户开箱即用，无需配置 API Key 或凭据。
当前可用的平台托管基础大模型包括（持续更新）：

- `wedata-gpt-5` 系列
- `wedata-deepseek-v3` 系列
- 其他平台托管模型（详见模型列表页面）

如需了解各模型的能力差异、计费规则和适用场景，请在模型列表页面点击对应模型查看详情。

关键概念

理解模型列表，您需要熟悉以下概念：

- **模型（Model）**：一个具体的 LLM 实例，例如 `wedata-gpt-5`、`wedata-deepseek-v3`。每个模型对应一个唯一的 **模型 ID**，在代码中通过该 ID 调用。

- **模型 ID**：模型在 DataBuddy 平台内部的唯一调用标识，遵循 `wedata-<provider>-<model-name>-<version>` 命名规范，例如 `wedata-gpt-5-2` 。模型 ID 是您在代码中需要写的字符串。
- **端点（Endpoint）**：DataBuddy 为每个模型生成的统一调用地址。您不直接关心后端模型实际部署在哪个云、哪个 region，平台会处理路由与失败重试。
- **平台托管**：模型由 DataBuddy 平台统一管理，包含模型实例的部署、API Key、配额、白名单、监控。
- **白名单**：平台托管基础模型需先开通白名单才可使用，开通后用量记录在审计日志中。

核心能力

能力	说明
模型列表浏览	在数据应用模块下的「Model List」页面查看所有可用模型，支持按名称搜索
调用 ID 复制	每个模型卡片支持一键复制模型 ID，粘贴到 Agent 代码 / Notebook 即可调用
文档与计费说明跳转	每个模型展示厂商官方文档、版权说明、计费说明的入口，便于您评估模型是否符合业务需求
调用审计与用量统计	平台托管模型的所有调用经统一 Endpoint 路由，调用日志可追溯，用量可统计（计费策略详见 7.3.4 模型调用与计费，待补充）

同类产品对应

DataBuddy	Databricks	阿里 DataWorks / 百炼
模型列表 / 平台托管基础模型	Foundation Model APIs（Pay-per-token）	百炼内置模型
模型 ID（如 `wedata-gpt-5`）	Endpoint name（如 `databricks-gpt-5`）	模型唯一编码

调用场景

模型列表中的模型可在以下场景中被调用：

调用方	调用方式	文档
Agent	Agent 框架（OpenAI / LangChain / LangGraph）通过 DataBuddy SDK 获取模型实例	配置 Agent 工具与知识库
Notebook	直接通过 OpenAI SDK / LangChain / DataBuddy SDK 调用，URL 为模型端点地址	Notebook

SQL Editor	通过 `ai_query()` 等内置 SQL 函数引用模型 ID	–
Buddy	Buddy 后台自动选用模型，您也可在对话框底部切换	什么是 Buddy

子功能导航

子功能	说明	文档
查看可用模型	浏览模型列表、查看模型详情、复制模型 ID、跳转厂商文档	查看可用模型
模型调用与计费	在代码中调用模型的方式，以及计费规则	7.3.4 模型调用与计费（待补充）

推荐学习路径

如果您是首次使用模型列表，建议按以下顺序阅读：

1. **了解模块定位**：本文。
2. **熟悉可用模型**：查看可用模型 中找到适合您业务的模型 ID。
3. **首次调用**：参考 7.3.4 模型调用与计费（待补充），跟随代码示例完成一次模型调用。

相关文档

- Agent 概述
- MCP 概述
- Notebook
- 什么是 Buddy