BladeAI Agents OS

产品白皮书

企业智能体开发 运行与持续运营平台

v0.1 技术交流版

2026 年 11 月

西安领铄智能科技有限公司



围绕业务目标组织模型、知识、工具和流程，形成可执行、可核验、可复用的智能体能力。

摘要与阅读说明

BladeAI Agents OS 是面向企业和组织的智能体开发、运行、协同与持续运营平台。平台把自然语言需求、组织知识、专业工具和业务流程连接起来，使检索、分析、材料生成、系统操作及应用开发能够在统一环境中推进，并通过权限、过程记录、评测和人工确认约束执行边界。[1][3]

企业采用智能体，关键在于完成真实任务：输入来自哪里，哪些知识可以使用，哪个工具能够执行，任务遇到异常如何继续，以及最终成果能否复核。本白皮书围绕这些问题，介绍平台能力、技术机制、部署方式和实施方法，面向信息化负责人、架构师、研发与集成团队，以及参与试点的业务人员。

产品名称与文档范围

本文按编制需求使用“BladeAI Agents OS”作为标题和产品简称。附件及当前官网主要使用“Blade Agentic OS”，参考一体机白皮书中还出现“BladeAI AgentOS”。本文将这些名称对应到同一平台产品的资料口径，不据此推断它们具有完全相同的版本、界面或交付范围。

本文沿用参考白皮书的“总体介绍、定位与价值、平台能力、技能、软件工厂、场景、部署、指标、交付”组织方式，补充软件平台需要的架构、运行机制和技术验收内容。文中的“OS”指智能体应用的组织与运行层，部署仍依赖服务器或边缘设备及其操作系统。

资料依据与技术表述

产品能力以用户提供的 47 页产品与解决方案材料、16 页一体机白皮书及当前官网为依据。多助手协作、异步工具和升级交接等说明同时参考领铄 2026 年 11 月 2 日公开技术文章。[1]—[4]

标为“参考设计”“示例”或“建议”的接口字段、状态机、数据结构和验收方法，用于解释工程机制与支持技术交流，不代表正式 API 合同或全部版本的已交付实现。外部资料用于说明通用技术原理，不构成平台采用相应开源框架、通过认证或达到某项性能的证明。

版本说明：资料检索截止 2026 年 11 月 6 日。产品功能、协议版本、模型授权、适配环境、性能和服务范围，以技术协议、版本说明和实际交付清单为准。

目录

01 总体介绍 4
02 产品定位与价值 5
03 总体技术架构 6
04 平台核心能力 8
4.1 智能体构建与运行定义 8
4.2 多助手协作与事件驱动 9
4.3 长任务与故障处理 10
4.4 知识检索与业务语义 11
4.5 上下文与分层记忆 12
4.6 模型接入与推理资源 13
4.7 工具接入与协议连接 14
4.8 过程观测与评测改进 15
05 技能开发与业务编排 16
06 软件工厂与自然语言应用开发 17
07 典型应用场景 18
08 部署方式与安全特性 20
09 产品配置与技术指标 22
10 交付价值与实施建议 24
11 结语与术语 25
附录 A 集成接口与事件示例 26
参考资料 27

阅读路径

了解产品与业务适配，可先阅读第 1—3 章及第 7 章。评估系统集成与技术可行性，重点阅读第 4—6 章、第 8 章和附录 A。组织采购试测与验收，重点阅读第 9—10 章，并核对参考资料中的能力依据。

本白皮书中的应用流程均按授权资料、授权工具和人工核验前提设计。除明确标注的公开项目进展外，场景示例用于说明使用方式。

01 总体介绍

1.1 产品概述

BladeAI Agents OS 为企业提供统一的智能体应用底座。用户从业务目标出发，说明输入材料、成果格式和操作约束，平台组织模型、知识、技能、工具与人员协作，完成多步骤任务，并保留中间文件、运行进度和最终成果。[1][2][3]

在产品层面，平台覆盖智能体构建、任务执行、能力资产管理和持续优化。在交付层面，平台既可以软件方式部署，也可以与 Blade Agent Box 组合，由一体机承载本地模型、知识、技能和运行环境。

1.2 业务能力与技术对象

业务能力	平台组织对象	典型交付结果
资料理解与知识检索	文档 知识片段 数据源及权限	带来源的回答 资料摘要
分析与内容生成	模型 分析工具 文档模板	数据分析结果 报告 图表
跨系统流程处理	工具 接口 工作流及确认节点	执行记录 待办及系统回执
自然语言应用开发	项目文件 代码 服务及预览	应用原型 代码和验证记录
能力沉淀与运营	技能 智能体 轨迹和评测集	可复用资产及版本基线

1.3 一个完整任务的基本闭环

任务首先需要明确目标和可验收成果，再匹配可以读取的知识及可执行的工具。执行过程中，平台按步骤处理检索、计算、生成和系统调用，必要时请求人工确认。任务结束后，成果与来源、工具回执和运行记录一起归档，有效流程经确认后进入技能或模板库。

这一闭环把“生成一段回答”扩展为“形成一份可核对的成果”。例如，经营月报任务需要读取指定期间数据、核对计算口径、生成图表、解释异常并输出文件；其中数值应由查询或计算工具产生，模型负责组织表达与解释。

1.4 产品组合

Blade Agentic OS 平台版承担开发、运行和治理。Blade Agent Box 提供边缘部署载体。AI 软件工厂与 AI 办公助手将平台能力组织为具体应用。三者在不同层面形成组合，不应将一体机的硬件规格视为平台版的性能上限或必需配置。[1][2]

02 产品定位与价值

2.1 产品定位

BladeAI Agents OS 面向需要将 AI 接入真实业务的组织，提供智能体开发、运行、协作和持续运营能力。平台连接现有系统与组织资料，帮助客户把岗位经验与业务方法封装为可复用的技能、工具和智能体。[1][3]

客户问题	对应平台机制	可核对的业务结果
资料分散且版本难辨	知识接入 检索与来源引用	回答关联到资料及版本
任务跨岗位跨系统	任务拆解 工具调用与协作	进度 中间产物及操作回执
试点资产难以复用	技能封装与资产统一管理	已验证的流程可重复调用
内网与敏感数据约束	本地化部署及调用范围控制	数据路径和网络边界可检查
生成结果质量不稳定	规则核验 评测与人工介入	结果能够按约定标准验收

2.2 对不同使用者的价值

对业务人员，平台提供围绕目标组织任务的入口，使资料收集、分析和成果制作有连续的工作上下文。用户可以查看中间结果，补充要求，并在关键操作前确认内容。

对开发与集成团队，平台统一模型、知识和工具的组织方式，减少每个场景重复搭建的工作。技术人员可以把精力放在业务接口、数据质量、工具可靠性和验收标准上。

对平台运营团队，技能、模型、权限、任务轨迹和版本形成可管理对象，有助于定位失败原因、控制资源消耗，并把有效流程推广到更多岗位。

2.3 适用条件与投入边界

优先选择输入明确、工具可接入、输出可验证、存在重复工作量的场景。对于完全依赖隐性经验、缺少可用数据或无法核对结果的任务，应先整理业务规则与样例。

客户收益应由试点验证。评估时同时记录任务用时、人工复核投入、系统集成成本、模型费用和运维成本，形成完整基线。生成质量、执行成功率与业务收益分别度量，避免用演示速度代替整体效益。

03 总体技术架构

3.1 逻辑分层

根据附件的产品架构，平台可按交互、开发与资产、运行调度、能力服务、基础设施与治理进行归纳。下图为资料的逻辑重绘，用于解释职责和数据流，不对应固定微服务拆分或部署拓扑。[1，第12—16 页]



图 1 BladeAI Agents OS 逻辑架构示意

交互层接收目标、文件和用户补充要求。开发与资产层组织智能体、技能和流程定义。运行层维护任务计划、角色分工与执行进度，调用下层模型、知识和工具服务。治理能力贯穿身份、数据、执行、成果和版本管理。

3.2 控制与执行的职责

在工程实现中，建议把任务状态、权限策略和资产版本放在可持久化的控制范围内，将模型推理与工具执行放在受限的执行环境内。这样即使模型服务或某个工具失败，也能保留任务身份、已经完成的步骤及恢复依据。

运行信息应以任务、步骤、工具调用和成果为基本关联对象。模型输出可以提出下一步操作，具体执行仍需经过工具参数校验、权限检查和必要的人工确认。

03 总体技术架构

3.3 平台工作空间

参考白皮书展示了统一桌面、会话入口、技能市场、文件和结果预览等交互形态。这种组织方式把任务所需的材料、工具和产出放在连续工作空间中，便于用户查看处理进度并核验结果。[2，第 3、6—7 页]



图 2 附件展示的一体机内置平台桌面 界面随版本变化

3.4 能力资产的组织关系

对象	主要职责	应管理的关键信息
智能体 Agent	承担角色与任务	目标 能力范围 模型与约束
技能 Skill	封装可复用业务方法	适用条件 步骤 模板与验证规则
工具 Tool	执行确定性动作	参数结构 权限 超时与返回值
连接器 Connector	连接外部系统	认证 数据映射 速率限制
知识 Knowledge	提供事实与业务依据	来源 版本 有效期与访问权限
任务及成果	保存一次执行的上下文	输入 轨迹 文件 回执与结论

技能说明“如何完成一类工作”，工具负责实际动作，智能体依据目标组织这些能力。把三者分开管理，可以在工具接口变化时更新连接方式，同时保留经验证的业务步骤。

平台资料还将开发、运行调度、知识与模型接入、安全和评测列为能力模块。具体组件名称及可安装范围，应按交付版本核对；本文不把逻辑模块直接解释为独立授权产品。

04 平台核心能力

4.1 智能体构建与运行定义

附件描述了从自然语言、业务文档、专家访谈和参考资料抽取流程，辅助生成智能体、技能、工具和交互原型的能力。业务人员可以试用原型，补充规则并调整逻辑，技术人员随后验证工具接入和运行边界。[1，第 14 页]

自然语言构建的工程重点，是把口头需求转成可以检查的任务定义。例如，“生成供应商月度分析”还需要明确数据期间、供应商范围、统计口径、成果模板、工具权限和哪些操作必须由人确认。

定义项	内容	核验方法
任务目标	要完成的业务工作及成功条件	与业务样例逐项对照
输入与前提	文件 数据范围 依赖及缺失处理	检查资料是否齐备
输出约束	文件格式 必填字段 引用规则	模板及结构校验
工具与权限	允许调用的系统和动作	联调与越权测试
人工介入	审核节点 确认范围及接管方式	模拟拒绝 修改和恢复
资源预算	时限 调用轮次及计算资源	压测与超限测试

4.1.1 构建与验证路径

建议先从样例任务抽取输入、输出和业务步骤，生成可试用版本，再用正常、缺失、冲突和越权输入验证。模型生成的流程与工具草稿需要专业人员确认，尤其要检查单位、统计口径、外部副作用及权限边界。

对结构固定、责任明确的业务，采用预定义流程可以稳定执行路径。对步骤需要随资料变化的研究与分析任务，可以在约束内动态规划；关键计算、审批和系统写入仍保留明确规则。

4.1.2 发布基线

参考设计中，每次发布应冻结智能体定义、技能版本、工具结构、模型配置和知识快照等关联项，形成可回溯基线。线上失败才能对应到实际使用的版本，后续修改也能与原基线比较。

发布基线描述的是可复现的运行条件，不要求输出文本逐字相同。对模型的随机性，应以业务约束是否满足、事实是否正确和操作是否合规作为判定重点。

04 平台核心能力

4.2 多助手协作与事件驱动

领铄公开技术文章描述了主助手、执行助手和主动助理三类角色。主助手保持连续对话、理解补充要求并分派工作；执行助手在独立任务中处理检索、代码和测试等工作；主动助理围绕已授权目标检查变化，将有价值的信息交给主助手筛选。[4]

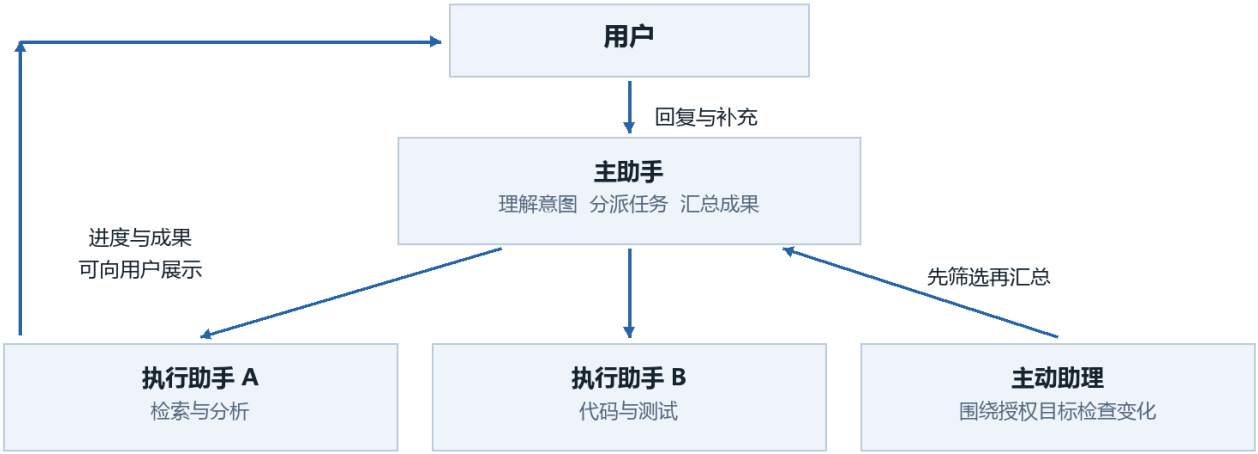


图 3 主助手 执行助手与主动助理的消息关系

4.2.1 单一对话入口与不对称发布

执行助手可以向用户展示进度、文件和结果，用户的回复回到主助手，再由主助手转交对应任务。这样用户可以同时了解多个任务，补充要求仍保留在同一交互入口。[4]

这要求消息与任务保留明确关联。工程参考字段包括 conversation_id、task_id、worker_id 和 event_id。主助手据此区分新增需求、对已有成果的修订以及对某次操作的确认。

4.2.2 事件优先级与合并

附件列出了用户输入、定时任务、后台任务和中断信息等事件来源，以及优先级队列、事件合并与并发控制。公开文章进一步说明：用户消息优先处理，短时间聚集的外部事件先合并筛选，主动助理遵守运行资格、周期和安静时段。[1，第 15 页][4]

具体时间窗口是版本策略参数。文章披露的线上值与已合入代码但尚未部署的值应分别理解，本文不把其中任何一个值写成全版本固定指标。任务并行度同样取决于模型、工具限额和计算资源。

04 平台核心能力

4.3 长任务与故障处理

多步骤任务需要保留中间状态，并处理工具等待、用户插话、失败重试和版本变更。附件明确描述了超时、重试、降级、上下文压缩和记忆持久化。公开技术文章还说明了工具后台运行、迟到结果回流与执行步边界的版本交接。[1，第 15 页][4]

4.3.1 用户插话与后台工具

当用户补充要求时，公开实现为运行中的工具设置收尾窗口：短时间内可完成的调用先返回结果；否则转入后台，主助手先处理新消息。工具完成后，结果关联回原始调用身份，再结合当前上下文继续处理。[4]

关键约束是保留调用身份。参考设计中，tool_call_id 标识原始动作，attempt_id 区分重试，task_version 标识规划版本。结果回流时检查任务是否已取消、计划是否已改变以及结果是否过期，避免把旧结果当成当前结论。

4.3.2 状态与恢复的参考设计

任务状态可区分排队、运行、等待工具、等待确认、成功、失败和取消。等待状态需要记录恢复条件；成功状态需要成果和验证结论；失败状态应保存原因、已完成步骤及可恢复范围。持久化内容应覆盖输入引用、计划版本、工具回执与成果地址。

失败类型	建议处理策略	必须保留的边界
临时网络故障	有限重试并增加间隔	受重试次数与总时限约束
参数错误或权限拒绝	返回结构化错误并修正或接管	不通过重复调用绕过校验
外部写入返回不明	先查询实际业务状态	避免重复创建 重复提交
模型或工具持续失败	降级为局部成果或人工处理	明确未完成部分

4.3.3 长任务升级交接

公开文章采用执行步边界交接：整理当前状态，交给新版本继续推进。该机制有特定安全交接点，不能解释为任意时刻无损切换。[4] 工程验收应加入状态格式兼容、未结束工具回包、幂等校验和失败回退测试。

04 平台核心能力

4.4 知识检索与业务语义

平台资料描述了知识接入、来源溯源、数据渐进式发现和面向智能体的本体数据接入。当前产品页强调授权文档、知识库和数据源接入，以及回答与引用的场景验证。[1，第 12 页][3]

RAG 将检索得到的外部资料加入生成过程，使回答能够依据可更新的知识内容。它可以增加事实依据，但正确性仍取决于资料、检索、权限和生成约束，不能仅凭接入向量库就保证回答正确。[7]

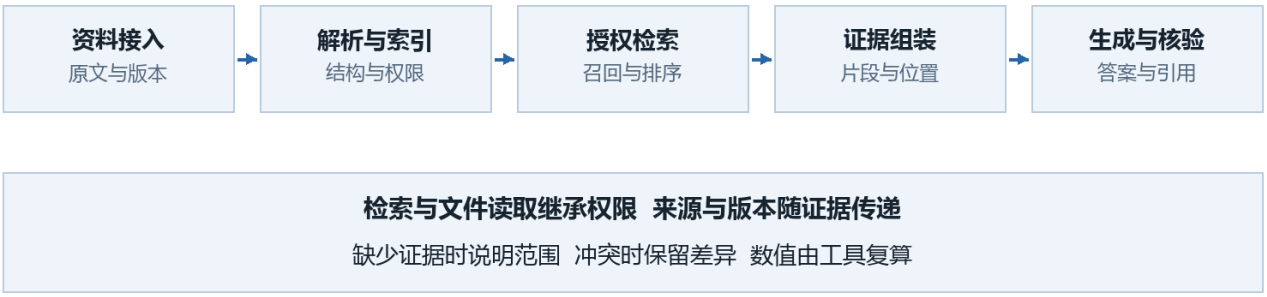


图 4 企业知识处理与检索链路参考设计

4.4.1 从资料接入到证据输出

参考设计中，接入阶段先识别文件格式、版面、表格和附件关系，保留文档标识、页码、版本和访问控制信息。切分应尽量保持章节与语义完整，避免把表头与数据、适用条件与结论拆散。

检索阶段可组合关键词与向量召回，并通过重排序选择证据。权限约束须作用于候选生成和后续读取，而非只在最终回答阶段过滤；授权改变后，索引、缓存和下载入口也应同步更新。

4.4.2 业务语义与本体

在多系统场景中，同一“项目”“客户”或“设备”可能具有不同编码和字段含义。本体建模可用于定义对象、属性、关系与可执行动作，再映射到实际接口。具体本体结构和适配深度需要按行业建设，不将其等同于自动生成完整业务知识图谱。

4.4.3 缺失与冲突处理

答案应关联实际证据，区分资料原文、计算结果和模型归纳。遇到证据缺失、版本冲突或无权限访问时，应说明可确认范围并请求补充材料。数值分析优先使用结构化查询或计算工具，并保留统计期间、单位与口径。

04 平台核心能力

4.5 上下文与分层记忆

附件将自动压缩、渐进式披露、程序化调用与上下文管理列为技术机制，并给出全局记忆、任务上下文记忆和个体记忆的分层设计。记忆的读取、存储和更新需要结合用户确认及运行反馈。[1，第12、15—16 页]

4.5.1 上下文预算

一次模型调用的上下文需要同时容纳系统约束、用户输入、任务状态、证据、工具说明、历史记录和预留输出空间。容量增加并不能代替内容选择；大量无关信息会占用资源，并增加遗漏关键条件的可能。

参考设计采用按需加载：先暴露工具与知识的简要目录，匹配后读取具体说明；长工具结果保留结构化摘要与原文件引用；历史内容按当前目标提取关键约束和已确认结论。原始证据应保存在工作区，便于核验。

记忆层级	典型内容	治理要求
全局或组织记忆	术语 模板 经确认的流程规则	明确适用组织 版本和管理员
任务记忆	当前目标 计划 约束和工具结果	绑定任务 生命周期及恢复点
个体记忆	经确认的偏好与工作习惯	用户可查看 修改与删除

4.5.2 压缩与事实保存

压缩过程应优先保留用户约束、数值及单位、来源引用、未完成事项和授权范围。重要事实以结构化字段保存，避免仅依赖自然语言摘要。摘要必须保留不确定性，不能把“待验证”压缩成“已确认”。

4.5.3 记忆与持续改进

记忆更新不等同于模型参数训练。平台可通过保存偏好、修订流程、调整提示和更新技能改善后续任务；模型微调或训练属于独立工程活动，需要相应数据、授权与评测条件。

记忆与知识库都可能包含敏感信息。参考设计中应记录来源、范围、有效期和更新人，支持撤回与删除传播。未经验证的任务结果不宜直接提升为组织级事实，用户之间也不应共享个人偏好或未授权资料。

04 平台核心能力

4.6 模型接入与推理资源

附件列出大模型接入管理、面向智能体的推理加速和上下文缓存。当前产品页支持围绕本地或私有模型进行集成评估，强调模型授权、资源和版本条件。[1，第 12 页][3]

4.6.1 按任务匹配模型

模型选择需要同时检查上下文容量、工具调用、结构化输出、中文业务表现、视觉能力、部署资源和授权条款。复杂规划、文本抽取、向量表示和重排序可以使用不同服务，但各服务的数据路径和失败策略应清楚定义。

模型返回的工具参数应经过类型和业务规则校验。对不支持所需结构或协议的模型，应通过适配与实测确认，而不能只凭“接口兼容”判断端到端任务可用。

4.6.2 推理阶段与缓存

推理通常包含处理输入上下文的预填充阶段，以及逐步生成输出的解码阶段。重复前缀的 KV 缓存可复用部分已计算内容，具体收益取决于前缀一致性、命中率和推理引擎实现。[10]

附件提出上下文缓存方向，但没有给出确定的推理引擎及缓存实现。vLLM 文档在本文中仅作为机制参考。缓存键的参考范围应考虑模型与版本、上下文前缀、模板、租户或权限域；权限敏感环境需要验证隔离与失效策略。

4.6.3 并发与容量控制

用户连接数、平台任务数、执行助手数和模型生成并发分别消耗不同资源。任务可以在等待工具时释放部分推理资源，但长上下文、多个工具和多智能体并行也会放大内存与调用压力。

容量评估建议同时观察排队时长、首个输出时延、生成吞吐、任务成功率和 P95 总耗时，并为模型、工具及任务配置资源预算。达到配额时应排队、降级或明确拒绝，避免无边界扩展。

4.6.4 本地与外部模型的数据边界

本地闭环需要模型、嵌入、重排序、知识与工具都在约定网络域内运行。接入外部模型、搜索、云端识别或遥测服务时，应单独确认传输字段、保留策略、出网控制和费用。

04 平台核心能力

4.7 工具接入与协议连接

平台资料覆盖 MCP、沙箱代码执行、浏览器控制和桌面软件操作。当前官网将 Skills、Connectors、MCP 与 HTTP/API 列为接入方式，具体协议版本、认证、配额和读写权限通过联调确认。[1，第 15 页][3]

4.7.1 MCP 的角色

MCP 采用 Host、Client 和 Server 架构，连接工具、资源及提示等能力，数据层使用 JSON-RPC。标准架构中，本地服务可使用 stdio，远程服务可使用 Streamable HTTP。平台支持的具体传输方式与协议版本需要按版本确认。[5]

MCP 解决能力发现与调用的标准化问题。连接成功之后，工具是否有权读取某个项目、是否允许写入以及结果是否可信，仍由业务授权、执行环境与校验机制承担。

接入方式	优先适用情况	需要验证的条件
HTTP/API 或 SDK	系统有稳定结构化接口	认证 字段 错误码 幂等及配额
MCP	工具需要标准发现与统一调用	协议版本 工具结构及安全策略
浏览器自动化	可通过授权页面完成业务	页面变化 登录状态和提交确认
桌面与远程环境	需要本地专业软件或遗留应用	会话隔离 外设 软件授权及回执
沙箱代码	数据处理 报表及受限计算	包依赖 文件路径 网络与资源限制

4.7.2 工具契约的参考设计

每个工具应有可验证的输入结构和结构化返回值，声明动作类型、读写范围、超时、重试条件及副作用。成功返回应能关联业务回执，失败返回应区分参数错误、权限拒绝、资源不足和暂时不可用。

对外部写入，建议携带业务幂等键，并在超时后先检查真实执行状态。浏览器或桌面操作完成后，使用系统回执或重新读取验证结果，不将“点击成功”直接等同于“业务办理成功”。

4.7.3 跨平台协作的扩展边界

A2A 面向智能体之间的发现、消息、任务和成果交换，可作为未来跨平台集成评估方向；现有资料不足以确认平台已经兼容 A2A。本白皮书不把 MCP 工具调用与 A2A 智能体通信等同。[11]

04 平台核心能力

4.8 过程观测与评测改进

附件描述了轨迹记录、输出溯源、评测环境、程序化评估、大模型打分和评估优选，并通过反馈生成智能体或技能更新建议。当前官网要求固定任务集、知识版本、运行条件和人工参与范围后开展试测。[1，第 12、16 页][3]

4.8.1 任务轨迹

应把任务、计划、模型调用、知识检索、工具执行与成果关联起来。参考设计采用 trace_id 贯穿任务，span_id 标识一次模型或工具执行，父子关系表达调用链；该组织方式可参考 OpenTelemetry 的 Trace 与 Span 概念，不代表平台已完成对应集成。[9]

记录对象	建议记录内容	主要用途
模型调用	模型版本 输入输出用量 耗时和状态	分析费用 时延与异常
知识检索	查询 资料版本 引用及权限域	核验回答依据
工具调用	参数摘要 回执 错误与重试次数	定位执行失败
人工操作	确认内容 审核者 时间与任务版本	划定授权和责任
成果产物	文件版本 校验结论及来源	验收和后续复用

日志应按需脱敏，并限制原始内容访问。轨迹记录不要求保存模型的全部内部推理文本，业务检查重点是可观察行为、引用依据、操作参数、结果和人员确认。

4.8.2 分层评测

首先检查结构、必填项、计算口径和接口结果，再由业务人员判断事实、完整性和适用性。模型评审可用于初筛和解释性反馈，需要与规则及人工抽检结合，避免同源模型偏差。

评测集应覆盖正常任务、信息缺失、无答案、资料冲突、工具故障、权限拒绝、用户插话和中途取消。每次发布与基线版本比较，同时报告效果、人工介入和资源消耗。

4.8.3 受控改进闭环

失败轨迹产生优化建议后，先在隔离环境修订技能、提示或工具，再跑回归集并审查权限变化。通过验证的候选版本逐步发布，保留回退入口。持续改进应以可测量的变更推进，避免未经评测的自动覆盖。

05 技能开发与业务编排

5.1 技能的定义

技能是对一类业务方法的封装，可包括适用条件、输入输出、步骤、提示、模板、脚本、工具约束和验证要求。参考白皮书明确描述了从制度、文档和案例生成技能草稿，并把单点能力组合为工作流或多智能体流程的方式。[2，第 7—9 页]

5.2 技能定义参考示例

下面以“供应商月度分析”为例展示建议字段，用于说明封装粒度，不是平台正式文件格式。

```
id: supplier_monthly_review
version: 0.1.0
inputs: [period, supplier_scope, source_files]
outputs: [report_docx, metrics_csv, evidence_index]
tools: [procurement.read, table.compute, document.render]
constraints:
  source_access: inherit_user_permissions
  missing_data: request_clarification
  external_write: require_confirmation
checks: [period_consistency, totals, source_coverage]
```

5.3 从技能到业务流程

流程可以把资料接入、口径校验、指标计算、异常归纳、报告生成和人工审核串联起来。涉及业务逻辑的分支应明确触发条件，例如采购数据缺失时返回补充清单，统计总额不一致时停止形成正式结论。

参考设计中，可把独立的资料检索并行执行，再在汇总节点等待必要结果；对有顺序依赖的计算与审批，则保持显式前后关系。多智能体分工服从流程约束，不能用角色数量代替业务完整性。

5.4 版本与供应链管理

技能发布应同时检查文档、脚本、工具依赖和权限声明。外部引入的技能需要来源确认、代码检查和隔离试运行，不能仅凭描述即可获得生产权限。

更新时应比较输出结构、依赖和权限是否变化。参考发布流程包括创建、测试、审核、发布与停用；具体实现按交付版本确认。预置技能数量以可用版本清单为准，数量不直接代表某个行业的适配深度。

06 软件工厂与自然语言应用开发

6.1 能力定位

软件工厂支持用户以自然语言描述页面、系统、数据与交互要求，辅助生成代码和应用原型，并在同一工作空间中修改、运行和预览。参考白皮书展示了项目文件、开发对话、服务管理与预览能力。[1][2，第 9—12 页]

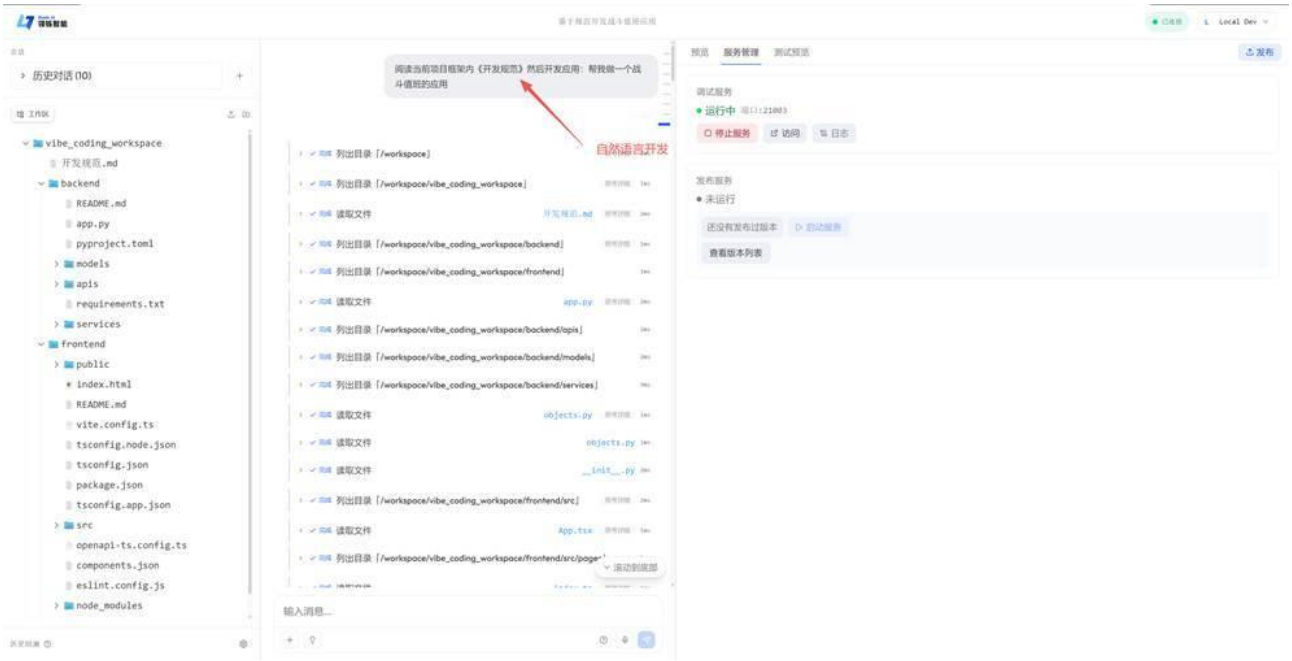


图 5 附件中的软件工厂界面 展示文件 对话与服务管理

6.2 从需求到可验证成果

需求阶段需要明确角色、页面、数据对象、权限和验收样例。生成阶段形成代码、配置与项目说明。验证阶段应检查运行、功能、权限、数据处理与依赖，依据用户反馈持续修改，最终再进入受控发布流程。

生成代码可以缩短原型形成过程，但生产系统仍需要专业工程验证。重点检查身份认证、服务端授权、输入校验、数据库变更、文件访问、错误处理、敏感信息及开源依赖授权。

6.3 工程化交付建议

建议将生成的变更纳入版本管理，保留需求与代码、测试和发布记录的关联。预览环境与生产环境分开，密钥通过专用机制配置；发布前检查构建、依赖、接口和回退方案。

软件工厂可与技能建设组合：经验证的报表工具、分析脚本或业务接口可以封装为后续任务可调用的能力。应用发布、数据库修改和外部业务写入应根据风险配置人工确认。

07 典型应用场景

7.1 知识办公与材料生成

平台可围绕授权制度、项目档案、报告模板和历史材料组织知识问答、资料梳理及材料编写。成果应保留引用位置、资料版本与待核实项。审核人员可以直接检查结论与来源是否一致。

7.2 数据分析与经营洞察

在表格或数据库授权范围内，组织清洗、指标计算、异常分析和可视化。数值由查询与计算工具形成，模型解释分析结果，成果保留查询条件、计算过程及统计口径。

7.3 跨系统事务与流程协同

通过开放接口、MCP、浏览器或桌面工具，组织信息读取、材料准备、表单填充和事项跟进。对提交、审批、外部发布或资金相关动作配置明确确认，并核对实际业务回执。

7.4 科研研发与软件生产

围绕文献、实验材料、专业计算和代码组织研发支持任务。软件工厂用于业务应用原型与工具生成。涉及研究结论和正式系统发布时，分别由研究与工程团队验证。

场景	关键依赖	建议验收成果
知识问答	有效资料 权限与检索质量	正确答案 来源与无答案处理
经营月报	指标口径 数据库或表格工具	可复算数据 图表和报告
业务办理	接口开放 账号与确认机制	业务回执与操作记录
研发与应用原型	环境 依赖和专业样例	可运行成果及测试记录

7.5 公开项目进展的引用边界

2026 年 10 月 27 日官网文章披露，领铄智能被确定为一家地方国有投资管理机构统一投资业务管理系统的承建方，建设涉及 Blade Agentic OS、AI 软件工厂及智能尽调场景。[12]

文章描述了建设范围与分阶段推进计划，没有提供已完成验收或量化收益的证明。因此，该信息可作为产品与行业流程组合的公开项目进展，不能据此宣称全部模块已投产或实现特定提升比例。

07 典型应用场景

7.6 经营月报任务的完整流程示例

本节以企业经营月报为技术示例，展示平台如何组织知识、工具和人工审核。该流程不对应已确认客户项目，也不提供收益承诺。

用户提出：“基于本月采购与交付数据，按部门生成经营月报，说明异常，输出 Word 和可复算表格。”平台首先确认月份、组织范围、指标口径和模板，检查数据及工具访问权限。

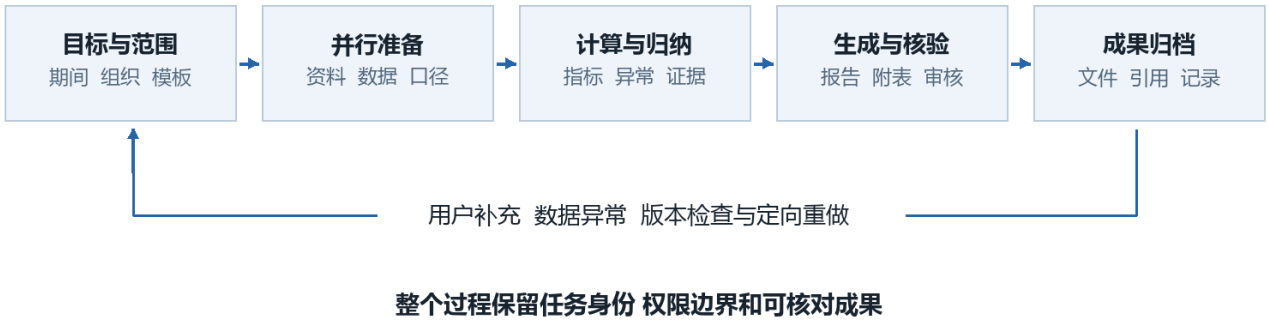


图 6 从资料确认到报告归档的流程示例

7.6.1 并行准备与口径核对

资料助手检索有效制度与报告模板，数据助手读取授权数据并整理缺失项，计算工具形成指标。不同分支通过任务标识关联，汇总前检查期间、币种、单位和统计范围是否一致。

7.6.2 生成与审核

分析助手依据计算结果归纳异常，引用原始表格位置和相关业务记录。报告助手生成正文、图表与附表。规则检查核对合计、必填字段和引用，业务人员确认解释是否合理，再形成正式成果。

7.6.3 中途变更与失败处理

若用户把范围改成“仅华东部门”，主助手更新当前约束，并标记需要重算的步骤。旧范围的后台回包仍关联原调用，经过版本检查后再决定是否复用。若数据库不可用，可先返回资料清单和未完成说明，恢复后继续授权范围内的步骤。

7.6.4 应保存的成果

最终保存月报、指标表、证据索引、计算口径、资料版本、工具回执与审核记录。评估时同时观察完成率、事实与数值正确性、人工修订量及总耗时，才能判断该场景是否具备推广条件。

08 部署方式与安全特性

8.1 部署形态

平台支持围绕本地化、私有化及边缘环境开展部署。参考一体机白皮书给出单机、局域网共享访问和场景化集成方式。平台版的操作系统、数据库、浏览器与资源容量需要按版本和项目确认。[1][2][3]

部署形态	适用范围	主要确认事项
单机或边缘一体机	场景试点 现场节点	本地资源 模型和离线依赖
内网共享服务	部门及多终端使用	用户权限 并发及文件隔离
私有化服务器	组织级平台与系统集成	容量 运维 身份及备份
本地与外部服务组合	经授权使用外部模型或数据	数据路径 出网与费用

8.2 服务与网络的参考组织

参考部署中，访问入口负责身份与流量控制，任务服务维护状态，执行环境处理代码和工具，模型与知识服务提供推理和检索，文件与记录存储保存成果及轨迹。组件可按规模合并或拆分，具体数量与依赖以交付拓扑为准。

网络规划应明确终端到入口、执行环境到模型、工具到业务系统、运维到管理服务等路径。对外部访问采用允许列表及受控出口，独立验证下载、搜索、模型 API 和遥测等依赖。

8.3 本地闭环与离线使用

“数据不出域”成立的条件，是模型推理、嵌入、知识检索、文件处理和所需工具均在约定域内完成。离线运行还需要事先准备模型、依赖包、技能、许可证及必要文档。依赖公网的数据搜索与云服务不能自动获得离线能力。

8.4 运维与恢复

建议分别备份资产配置、知识原文与索引元数据、任务状态、文件、审计记录和密钥配置，验证恢复顺序及一致性。高可用、容灾、RPO 与 RTO 属于部署设计及验收目标，不在缺少测试的情况下承诺固定等级。

升级前检查状态和接口兼容，长任务按安全步边界交接，保留失败回退。恢复演练应覆盖服务重启、模型不可用、索引重建和外部工具回包等场景。

08 部署方式与安全特性

8.5 安全控制面

附件描述了数据隔离、权限管控和智能体运行沙盒。当前产品页也要求通过角色测试知识访问、工具授权、异常中断和人工确认。[1，第 12 页][3] 下表为面向落地的控制与验证建议，具体实现需逐项核对。

风险对象	控制建议	验证方式
知识与文件	继承用户权限 下载与引用同样受控	越权检索 直链访问和撤权测试
工具与外部系统	最小权限 参数校验与动作分级	伪造参数 越权动作及失败回执
代码与执行环境	隔离目录 限制网络及资源	路径越界 出网与资源超限测试
密钥和凭证	受控注入 脱敏与轮换	日志泄露和过期凭证测试
审批与人工确认	绑定动作 参数 任务版本和有效期	修改参数后重用旧确认测试

8.6 提示注入与过度授权

外部文档、网页、邮件和工具结果应被视为业务数据，其中出现的“忽略规则” “上传文件”等内容不自动成为有效指令。参考设计将不可信内容与系统规则分开，工具动作仍经过独立授权和参数校验。[6][8]

OWASP 列出的提示注入、敏感信息泄露、不当输出处理、过度代理、向量与嵌入风险及无界资源消耗，均与智能体业务落地相关。防护需要权限、隔离、预算、审计和业务复核共同承担，不能仅依赖提示词约束。[8]

8.7 协议接入安全

MCP 官方安全资料讨论了混淆代理、令牌透传及 SSRF 等问题。工具服务接入时应核对身份、令牌受众与授权范围，控制动态目标地址和出口，并保留用户或管理员的明确授权。[6]

8.8 人工确认的有效范围

确认界面应让用户看到真实动作、目标对象和关键参数。参考设计以任务、参数摘要和有效期绑定授权；参数改变后重新确认。对已发出的外部操作，取消任务不能视为外部系统已撤销，必要时采用补偿或人工核对。

09 产品配置与技术指标

9.1 平台版配置的确认维度

平台软件没有脱离模型、数据和业务工具的统一容量值。项目评估应分别确定模型服务、知识处理、任务运行、执行环境、文件存储与管理服务的资源需求，并记录实际组件版本。

配置维度	需要明确的项目
产品与组件	平台版本 功能模块 资产与接口版本
模型服务	模型名称 授权 量化 上下文 推理后端
知识服务	文件格式 规模 更新频率 嵌入与检索方式
运行与工具	任务规模 工具数量 超时 并发和执行隔离
基础设施	CPU GPU 内存 存储 网络 操作系统
治理与运维	身份 权限 日志 备份 监控与支持范围

9.2 Blade Agent Box 的参考配置

下表引用附件中的一体机参数，仅用于说明平台的一种承载环境；不作为平台版容量承诺。[1，第 21 页][2，第 15 页]

项目	附件参考口径
CPU 与 GPU	20 核 ARM 及 GB10
AI 算力	1 PFLOPS FP4 理论规格
内存与存储	128 GB 内存 1 TB 存储基准
模型与上下文	35B 或 122B 级配置 128K 上下文口径
并发与技能	典型 4 路并发 300+ 技能口径

产品介绍 PPT 另列 1 TB / 4 TB 存储选项，应按实际型号确认。模型规模、量化、上下文长度和并发必须在同一配置下核验，不能将各项标称最大值直接叠加为同一运行条件。

理论算力不能直接换算为任务完成速度。典型 4 路并发也不代表任意模型、任意上下文和任意多智能体任务均可同时达到相同响应。最终规格、模型授权、技能数量和适配范围以技术协议与实际交付为准。

09 产品配置与技术指标

9.3 场景试测与验收指标

当前产品官网未提供统一口径的量化性能报告，明确建议按固定任务、知识和配置组织测试。[3] 本节给出指标定义与记录方法，目标值由双方在试点前约定。

指标	统计口径	测试注意事项
任务完成率	满足全部验收条件的任务数 / 总数	人工介入后完成与自主完成分别统计
事实正确率	核对正确的事实项 / 被核对事实项	数值 单位 期间与资料版本一致
引用正确率	支撑对应结论的引用数 / 被检查引用数	另记应引用内容的覆盖情况
工具调用成功率	获得有效业务结果的调用数 / 总数	记录重试 首次成功和外部故障
端到端耗时	接收任务至成果通过检查的时间	报告 P50 P95 及人工等待时长
人工介入	每任务确认 修订 接管次数及用时	区分必要审批与失败补救
资源与成本	每任务调用量 算力时间及费用	本地资源 外部 API 与运维分别记录

9.4 可复现的测试条件

测试记录至少包含平台与技能版本、硬件、操作系统、模型与量化、推理参数、知识快照、工具权限、输入输出规模、并发和重复次数。冷启动、冷缓存与热缓存结果分开，工具等待和人工等待分别报告。

任务集应来自真实业务的脱敏样本，包含典型、边界和失败情况，保留样本数量与结果分布。知识问答加入无答案和越权问题；长任务加入取消、恢复、用户改需求与迟到回包。

9.5 安全与可靠性的专项验证

建议测试撤权后知识访问、工具越权、参数变化后审批失效、重复提交、注入内容、敏感日志、模型超时及资源超限。对未通过项目记录阻断范围和修复计划，再做定向回归。

在采购或验收材料中，演示验证、场景实测、正式验收和长期运行数据应分别标注。对外收益表述应引用实际样本与测试条件，不把单次演示推广为普遍性能结论。

10 交付价值与实施建议

10.1 四阶段实施路径

参考白皮书建议从少量高价值场景开始，随后导入知识、沉淀技能、验证原型并逐步扩展。平台版实施可沿用该路径，把每阶段的业务、技术和验收成果明确下来。[2，第 15 页]

阶段	建设重点	阶段交付物
场景与基线	选择任务 明确输入输出和约束	场景说明 数据清单 验收口径
环境与能力接入	部署模型 知识 工具和权限	环境记录 接口联调 权限矩阵
流程与应用验证	构建技能 智能体及应用原型	可用流程 测试结果 问题清单
发布与持续运营	试运行 培训 版本与监控	交付基线 运维手册 回退与培训

10.2 交付物建议

除安装与账号外，建议交付部署拓扑、配置与版本清单、知识处理规则、技能和智能体清单、接口说明、权限矩阵、测试报告、运维手册及备份恢复方案。涉及软件工厂的项目，还应交付约定的代码、构建说明、依赖信息与发布记录。

10.3 协作职责

业务负责人确认规则、样例、成果标准和最终结论。信息化与安全团队确认网络、身份、系统接口和数据范围。平台交付团队完成部署、接入、流程建设和问题整改。运营团队负责知识更新、反馈收集、权限复核与版本维护。

客户需要为数据质量和接口开放提供配合，项目组共同定义不可执行或需要人工处理的情况。模型输出、业务审核和系统操作的责任应在试点阶段明确，避免上线后才补充边界。

10.4 推广条件

扩展到新部门前，应验证数据权限、流程差异、工具稳定性和资源容量。可复用的是经验证的技能结构与工程方法，行业规则、字段映射和审批要求仍需适配。

评估长期价值时，持续观察任务使用率、有效完成率、人工修订、资产复用和维护成本。平台建设的成果，应体现为稳定可用的业务能力，以及组织能够持续维护和改进这些能力的方法。

11 结语与术语

11.1 结语

BladeAI Agents OS 把智能体构建、协同执行、知识与工具连接、技能沉淀和持续评测组织为统一平台。其产品价值需要在具体任务中体现：能够按权限读取资料，调用合适工具，处理长流程与异常，并输出可核验成果。

平台建设应从明确任务开始，用真实样本验证知识质量、工具执行和人工协作，再将通过验证的能力沉淀为资产。随着业务范围扩展，版本治理、资源控制、数据边界和持续评测成为稳定运行的基础。

对于希望将 AI 接入既有系统的企业与组织，平台版和 Blade Agent Box 提供不同的部署载体，软件工厂和业务技能提供不同的应用入口。实际建设范围由业务目标、环境条件与交付协议共同确定。

11.2 常用术语

术语	本白皮书中的含义
Agent 智能体	围绕目标组织模型 知识 工具和流程的执行单元
Skill 技能	可复用的业务方法及其说明 模板 脚本和约束
Tool 工具	具有确定输入输出与权限边界的实际执行能力
RAG 检索增强生成	将外部检索证据加入生成过程的技术方式
MCP	连接 AI 应用与外部工具和上下文的标准协议
A2A	面向智能体之间任务与消息交换的协议方向
本体	对业务对象 属性 关系和动作的统一语义定义
上下文	一次模型调用可见的输入 约束 证据和历史信息
KV 缓存	推理过程中保存并复用注意力键值计算的机制
幂等	同一业务请求重复执行仍维持约定业务效果
Trace 与 Span	描述任务调用链及其中单次执行的关联记录

附录 A 集成接口与事件示例

本附录为技术沟通参考，不是已发布 API 规范。真实端点、认证方式、字段、错误码与版本兼容性应使用实际交付接口文档。[3]

A.1 任务创建的参考结构

```
{
  "request_id": "req_20261106_001",
  "agent_ref": "monthly_report@0.1.0",
  "input": {"period": "2026-10", "scope": "east"},
  "resources": ["file_001", "dataset_002"],
  "policy": {"external_write": "require_approval"},
  "budget": {"max_steps": 30, "timeout_seconds": 900}
}
```

示例中的步骤和时限仅用于展示字段。服务端必须从身份与权限系统确定真实权限，不能因客户端传入 policy 或资源标识就扩大访问范围。重复 request_id 的处理需与业务幂等策略一致。

A.2 工具结果回流的参考结构

```
{
  "event_id": "evt_102",
  "task_id": "task_001",
  "task_version": 2,
  "step_id": "step_04",
  "tool_call_id": "call_021",
  "attempt_id": "attempt_1",
  "type": "tool.completed",
  "artifact_ref": "artifact_017"
}
```

回流处理先检查调用归属、事件重复、任务状态、规划版本和成果权限，再决定继续执行或保留为历史结果。事件排序与重放规则由实现定义，业务写入仍依赖外部系统的幂等或状态核对机制。

A.3 集成动作清单

建议接口评估覆盖任务创建与查询、进度订阅、取消、人工确认、成果获取、资产版本查询和工具健康检查。确认操作绑定目标动作与参数摘要。取消与外部操作撤销分别定义，所有成果下载继续执行访问控制。

参考资料

产品资料与公开说明

[1] 西安领铄智能科技有限公司．《【综合】领铄智能产品与解决方案_1021》．用户提供的 PPTX，共 47 页，文件名未明确标注年份。主要引用第 11—17 页平台定位、架构与机制，第 20—21 页一体机规格，以及第 25、28—31 页产品组合与应用方式。

[2] 西安领铄智能科技有限公司．《智刃边端 Blade Agent Box 白皮书 v0.1》．2026 年 4 月，共 16 页。用于白皮书结构、平台界面、技能、软件工厂、部署与参考规格。图 2、图 5 及封面工作环境图取自该附件；图 1、3、4、6 为资料归纳或参考设计重绘。

[3] 领铄智能．Blade Agentic OS 企业智能体开发与运行平台．当前产品页，检索日期 2026-11-06。用于核对知识、工具、部署、测试方法和交付边界。

https://www.bladeai.com.cn/products/blade-agentic-os/

[4] 领铄智能．《当 DeepSeek 画出 Super Agent 的蓝图时，领铄智能已经在生产环境实践出 4 条技术选择》．2026-11-02，检索日期 2026-11-06。用于主助手与执行助手消息关系、主动助理、事件合并、工具收尾、后台回包和步边界升级机制。本文引用领铄自身公开描述，不将文章中的其他产品比较作为独立验证事实。

https://www.bladeai.com.cn/news/super-agent-production-architecture-20261102/

[12] 领铄智能．《一家地方国有投资机构的投资业务管理系统，确定由领铄智能承建》．2026-10-27，检索日期 2026-11-06。用于公开项目进展与工程化交付范围；资料未提供最终验收与量化收益证明。

https://www.bladeai.com.cn/news/lingshuo-investment-management-system-20261027/

资料使用原则

当前官网与用户附件共同支撑产品描述。附件中的收益数字、未核验资质及未获公开确认的客户信息未用作本白皮书的性能或交付承诺。文中的工程示例用于解释机制，具体能力按交付清单与试测记录逐项核对。

参考资料

外部技术资料

[5] Model Context Protocol．Architecture overview．官方文档，检索日期 2026-11-06。用于 Host / Client / Server、JSON-RPC 与传输机制说明。

https://modelcontextprotocol.io/docs/learn/architecture

[6] Model Context Protocol．Security Best Practices．官方文档，检索日期 2026-11-06。用于混淆代理、令牌透传和 SSRF 等接入风险说明。

https://modelcontextprotocol.io/specification/latest/basic/security_best_practices

[7] Lewis P．等．Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks．2020，arXiv:2005.11401。用于 RAG 的外部检索与生成结合原理，检索日期 2026-11-06。

https://arxiv.org/abs/2005.11401

[8] OWASP GenAI Security Project．Top 10 for LLM Applications 2025．官方风险清单，检索日期 2026-11-06。用于提示注入、信息泄露、输出处理、过度代理与资源风险说明。

https://genai.owasp.org/llm-top-10/

[9] OpenTelemetry．Traces．官方概念文档，检索日期 2026-11-06。用于调用链、Trace、Span 与上下文关联的参考设计。

https://opentelemetry.io/docs/concepts/signals/traces/

[10] vLLM．Automatic Prefix Caching．官方设计文档，检索日期 2026-11-06。用于前缀 KV 复用和隔离机制的技术说明；该链接为动态版本文档，不代表平台使用该推理后端。

https://docs.vllm.ai/en/latest/design/prefix_caching/

[11] A2A Project．A2A Protocol Specification．官方规范，检索日期 2026-11-06。用于界定智能体之间的消息、任务与成果交换。本文仅列为扩展评估方向。

https://a2a-protocol.org/latest/specification/

上述外部资料提供通用技术依据。与这些概念一致，不等于完成协议认证、开源组件集成或性能验证；具体实现应通过版本文档、接口联调和测试报告确认。