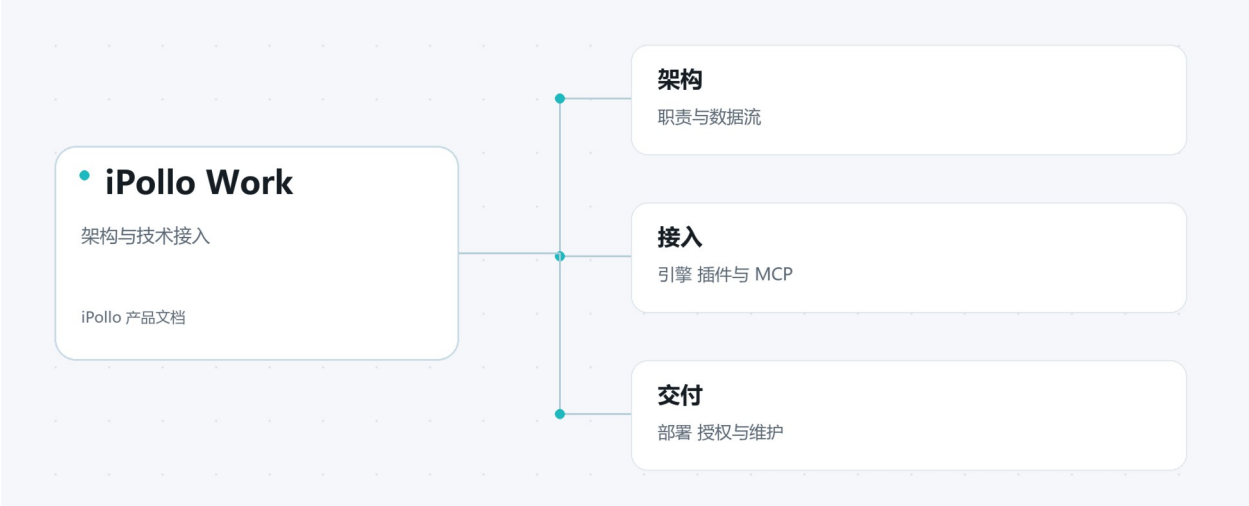
iPollo

iPollo 产品文档 / 06

iPolloWork
对外技术文档

企业评估与开发接入参考



iPollo

文档编号 IP-TECHNICAL

2026 年 11 月 2 日

技术基线 v0.50.13

摘 要

本文档面向企业技术负责人、部署与运维人员及插件开发者，介绍 iPolloWork 的产品定位、系统架构、任务数据流、多运行时协作层以及扩展接入方法，为技术评估、部署设计和开发集成提供参考。本文保留原对外技术文档 v0.50.13 的接口、示例与能力状态范围。

全书从产品定位和快速开始展开，依次说明架构、Harness 协作层、网页和视频案例、计划任务、插件开发、部署与数据管理，以及商业授权和维护。案例保留具体输入、操作、输出和限制；插件章节提供声明式包、Skill、MCP、鉴权及排错参考，附录整理术语与任务检查单。

技术评估时应将界面可见能力、已验证链路和依赖外部环境的能力分别核对。接口和示例的适用性需要结合目标版本确认；当前客户端的日常功能操作请参阅配套 Work 功能手册。涉及部署、凭据、数据外发和升级时，应结合组织要求与实际运行记录进行验收。

关键词：iPolloWork；系统架构；Harness；插件开发；部署与授权

适用读者　企业技术负责人、部署与运维人员、插件开发者

版本说明　本文接口与案例保留 v0.50.13 技术基线。当前客户端操作与设置请参阅 Work 功能使用手册。

图示说明　截图聚焦操作入口与配置位置，图下注明关注点。模型调用、远程部署和外部服务的最终结果，需要在实际环境确认。

目 录

第 1 章 产品定位与适用范围.....1
第 2 章 快速开始.....2
第 3 章 系统架构与数据流.....4
第 4 章 Harness 协作层与多运行时.....6
第 5 章 网页可视化案例.....8
第 6 章 视频可视化案例.....10
第 7 章 计划任务与自动化.....12
第 8 章 插件与扩展开发.....14
第 9 章 部署与数据安全.....22
第 10 章 商业授权与维护.....24
附录 A 资料与术语.....25
附录 B 使用者任务检查单.....26

第 1 章 产品定位与适用范围

iPolloWork 是本地优先的智能体工作台。用户以项目组织任务、会话、日程与交付文件，通过选定运行时调用模型和工具，完成后继续预览、编辑和导出成果。适合需要保留工作过程、复用输入文件并由人检查交付物的个人与团队。

工作台统一提供项目入口；OpenCode、DeepSeek Harness 与 Codex Harness 各自执行任务。网页案例展示 HTML 源码与文案修改，视频案例保留场景和时间信息，开发者案例以同一项目安装 Skill 或 MCP。可编辑的范围应按具体成果确认，不能从网页与视频能力推断所有文档或 PPT 格式均已验收。

1.1 适用场景

场景	当前做法	需要核对
个人与小团队	桌面端建立本地项目，选择模型，完成任务并编辑成果	模型账户、硬件、素材许可
企业内部	私有 Worker，管理项目与任务	授权、网络、日志、备份、身份与数据流
生态开发	制作 Skill 或插件包，连接远程 MCP	导入校验、运行时兼容与签名要求

1.2 能力状态与验证范围

能力	可用状态	当前入口与条件	本手册验证范围
本地项目与 OpenCode	已支持	项目和会话；需自备模型账户	本地项目及文件生成
网页模板与预览	已支持	创作入口与 HTML 文件	模板起稿、人工编辑和预览
视频工程与导出	已支持	视频模板；需完整素材与渲染环境	15 秒 MP4；lint 通过
计划任务	已支持	Schedule；服务在线且模型可用	保存、触发和失败反馈；摘要未完成
插件包与远程 MCP	已支持	扩展页；声明式安全校验	清单与归档验证；回归测试；远程调用

第 2 章 快速开始

2.1 下载并确认环境

从附录 A 的 v0.50.13 Releases 链接下载，与系统和芯片架构对应的安装文件如下。Windows 使用“设置 → 系统 → 系统信息”核对系统类型；macOS 在“关于本机”查看 Apple 芯片或 Intel；Linux 可运行 uname -m。

系统	架构	v0.50.13 安装文件
Windows	Intel 或 AMD 64 位	ipollowork-win-x64-0.50.13.exe
Windows	ARM64	ipollowork-win-arm64-0.50.13.exe
macOS	Apple Silicon	ipollowork-mac-arm64-0.50.13.dmg
macOS	Intel	ipollowork-mac-x64-0.50.13.dmg
Linux	x86_64	ipollowork-linux-x86_64-0.50.13.AppImage
Linux	ARM64	ipollowork-linux-arm64-0.50.13.AppImage

Windows 运行 .exe 并按向导安装；macOS 打开 .dmg，将应用拖入 Applications；Linux 为 AppImage 添加执行权限后运行（chmod +x <安装文件>）。准备可写项目目录、模型账户和相应网络。发行资料未给统一最低内存或 CPU 指标，需在目标机器试跑。本例使用 Windows 开发环境；Windows 安装器暂未签名，应核对来源。

2.2 连接一个模型提供方

仓库提供的 Anthropic API Key 路径为：Settings → Connect Provider → Anthropic → 手动输入 Key → 保存 → 在聊天中选择该提供方的模型。Key 来自用户或企业在模型提供方的账户；账户开通、API 额度和费用由该账户承担，不包含在桌面安装中。企业可按组织策略使用共享提供方配置。

1. 在提供方控制台确认 API 访问与计费已启用，创建专用 Key。
2. 在 iPolloWork 的提供方设置保存 Key；不要把 Key 放进任务输入或项目文件。
3. 在新会话中选择一个实际可用的模型，发送“只回复连接成功，不调用工具”。
4. 收到回答后再开始文件任务；若登录成功但请求提示额度不足，先处理该账户的额度或换用已授权的可用模型。

此连接路径来自 v0.50.13 的提供方指南；本文未新增账户登录实测。ChatGPT 登录、API Key 与各运行时账户不是可互换的凭据；连接方式应以当前选择的提供方和引擎为准。

2.3 完成第一个可编辑页面

创建本地项目，确认项目目录可写。新建任务，输入以下文本；本例不需要图片、数据库或外部 MCP。模型可用后，它应在工作区写出 HTML，用户再完成预览与修改。

```
在当前项目制作一个中文静态介绍页。
名称：Signal Workspace。受众：产品与运营团队。
页面包含首屏、三项能力、工作过程和联系按钮。
三项能力为统一任务、连接会话、沉淀成果。
联系按钮使用占位动作，不发送数据，不接真实表单。
输出可编辑 HTML；列出文件路径和需要人工检查的事项。
```

1. 出现权限提示时核对写入路径、命令与外部连接，允许范围限定在当前项目。
2. 任务结束后在输出面板找到 HTML 文件并打开预览；第 5 章实际路径格式为 design/<session-id>/entry.html。
3. 把联系按钮文案改为“预约演示”，保存文件并重新打开预览，确认修改保留。
4. 检查三项能力、链接目标和移动端布局；确认页面没有真实表单或虚构客户指标，再保留为项目成果。

完成标准：文件存在；预览能打开；修改后重新打开仍显示新文案；任务中能找到交付路径。第 5 章提供实际截图和限制。本次已有网页模板与人工编辑闭环，但未完成无研发人员陪同的首次使用验收，因此不承诺固定完成时长。

2.4 首次任务的常见失败

现象	检查与恢复	继续前的判断
模型无响应或额度不足	检查提供方、当前模型、网络和 API 额度；解决后先发短消息	短消息能返回
文件没有出现	查看任务事件与写入权限，确认当前项目及输出面板	找到实际文件
预览缺图或样式	检查 HTML 的相对资源路径，保留同目录素材	刷新后资源完整
执行中断	保存已生成文件和失败原因，在同一任务明确要求从文件继续	不重复外部写入动作

第 3 章 系统架构与数据流

下图是逻辑架构，不是固定进程清单。桌面端提供可视化入口；项目服务保存任务与文件状态；Engine Protocol 将运行状态和流式事件映射到工作台。模型、工具与代理仍由具体运行时执行。Codex 的 UI MCP 控制入口与项目内的 Codex Harness 引擎是两条不同路径。

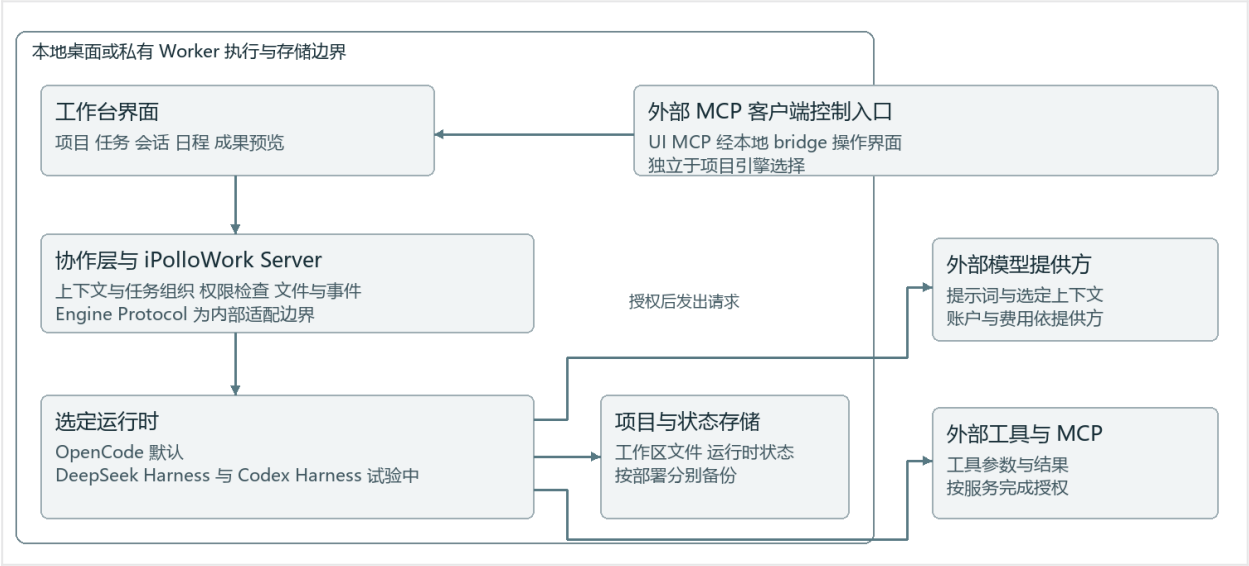


图 1 v0.50.13 逻辑架构；具体进程随桌面或私有 Worker 部署形态变化。

3.1 模型运行时与协作层

概念	负责什么	配置与共享边界
模型	理解输入、生成内容与工具调用建议	提供方账户与模型选择；费用按账户
运行时	执行代理循环、工具、权限与会话状态	OpenCode、DSH、Codex Harness 各自维护
iPolloWork	组织项目、任务、会话入口、日程和成果	可复用文件与任务描述；上下文不自动迁移
Skill	说明任务方法、输入与输出要求	随运行时工具能力逐个验证
MCP	以协议连接外部工具与资源	连接和授权决定可调用范围
插件包	管理资源、版本、授权与安装生命周期	普通包与受信特权包边界不同

Engine Protocol 是当前工作台内部的引擎适配边界。本文没有给它作对外稳定 SDK 或兼容承诺；生态开发者优先使用本版插件包、Skill、MCP 和已说明的服务接口。接口类型可见不代表获得长期稳定的公共 API。

3.2 一次任务从输入到交付

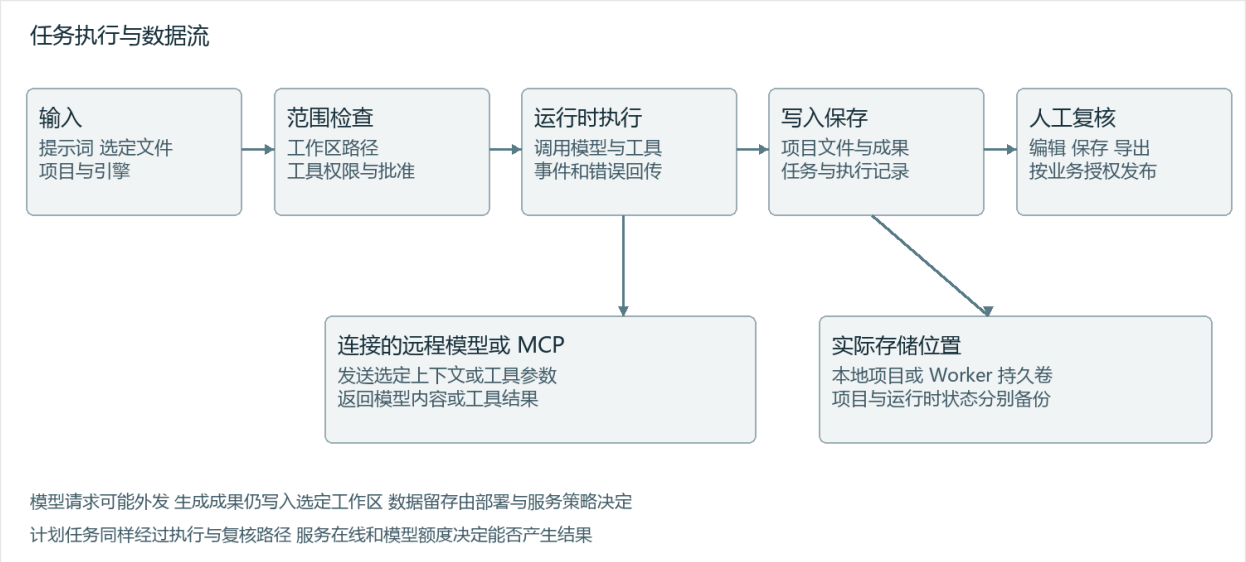


图 2 从用户输入到人工复核的数据流；外部模型和 MCP 仅在连接与授权后接收相应数据。

用户提交任务、文件和项目选择后，服务检查路径与权限，将工作交给选定引擎。引擎按配置调用模型或工具，事件回到任务视图；文件写回工作空间。网页和视频视图读取可编辑工程。人工复核决定是否继续编辑、审批、导出或发布。

3.3 企业应核对的三个数据问题

1. 哪些任务内容会发给模型提供方，是否包含客户资料、代码或个人信息。
2. 哪些插件和 MCP 具有网络连接；凭据保存在哪里；授权如何撤销。
3. 工作空间、任务记录、导出物和日志分别保存在哪里，谁负责备份。

信任边界	应做的控制
桌面渲染进程 → 本地服务	请求校验、最小权限和路径归一化
服务 → 运行时 / 模型	凭据隔离、调用范围和用户批准
插件包 → 安装目录	清单校验、资源限制和特权包签名
本地 → 远程服务	TLS、连接清单、日志去敏与合同核对

第 4 章 Harness 协作层与多运行时

Harness 协作层把项目目标、任务、会话、计划、工具许可和成果文件关联起来。它统一“谁在做、做到哪里、产出什么”的工作台视图；每个引擎的内部代理、命令和模型选项仍保持原样。跨引擎委派宜提供边界清楚的输入、输出目录与验收标准。

路径	适用方式	限制
OpenCode	默认本地 Agent 执行	依赖本地 sidecar、模型和工具配置
DeepSeek Harness	可选项目引擎或委派目标	上游开发预览，插件需匹配发行线
Codex Harness	项目可选的本地引擎路径	需要 sidecar 与账户，按目标包验收

4.1 会话归属

创建项目时选择引擎。会话归属、工作目录和历史与项目关联；切换引擎不等于复制整个会话状态。团队应在任务单中记录引擎、模型、输入目录和输出目录，尤其在多人交接和跨运行时复用 Skill 时。

4.2 DeepSeek Harness 接入

桌面端可使用集成的可选运行时。若在 DSH Web UI 中单独使用创作插件，可按上游命令安装 Design、PPT 和 Video 视图；它与桌面端插件安装是不同入口。

```
npx @deepseek-ai/dsh plugin --profile web add deepseek-idesign deepseek-ippt deepseek-ivideo
npx @deepseek-ai/dsh web
```

Web UI 默认地址是 127.0.0.1:3080。由于 DSH 仍为开发预览，企业应固定上游版本、插件版本与测试工作区，验证实际需要的文件操作和审批流程。

4.3 项目中的 Codex Harness

在新项目选择 Codex Harness 后，由该引擎的 sidecar 和账户执行项目任务。先核对安装包包含匹配的 sidecar，再确认账户连接和当前可用模型，测试只读任务、文件写入和权限回复。已有会话恢复、插件投影与模型选择需要在声明支持的系统组合中单独验收；当前标为试验中。

4.4 外部客户端通过 UI MCP 控制工作台

此路径要求 iPolloWork 桌面端已运行，且本地 UI control bridge 可发现。在 MCP 客户端添加一个 STDIO 服务器：command 为 npx，args 为 ["-y", "ipollowork-ui-mcp"]。客户端配置格式依其自身文档填写。连接后先读取工作台快照，再根据用户授权执行界面动作。

```
npx -y ipollowork-ui-mcp
```

这条命令启动 UI MCP 控制服务器。它不安装 Codex Harness，也不把项目切换到 Codex 引擎。若连接失败，先检查桌面端与 UI bridge，再检查 MCP 客户端；项目引擎失败则检查 sidecar、账户和模型。

4.5 交接时具体复用什么

```
交接示例 由 OpenCode 项目交给另一个已验收运行时
输入文件：design/<session-id>/entry.html
已完成：首屏与三项能力，联系按钮为占位动作。
继续目标：只修改按钮文案和移动端排版，保存原路径。
验收：预览无溢出，文件可重新打开，列出修改项。
授权范围：当前工作区文件；不发布、不连接外部账号。
```

接手者复用的是任务说明和实际文件。原运行时的会话上下文、登录状态和工具许可不会仅因共享目录而自动迁移。本示例说明交接格式，尚未作为跨运行时成功实测。

第 5 章 网页可视化案例

案例目标：制作面向 20 至 200 人产品与运营团队的中文静态落地页，说明“统一任务、会话与成果”，展示三项能力，并保留占位联系按钮。本案例使用内置 Signal Workspace 模板，在 Windows 本地 v0.50.13 环境操作。

5.1 实际输入

```
网站名称：Signal Workspace 项目协作平台
受众：面向 20 至 200 人的产品与运营团队
内容：首屏说明统一任务、会话与成果；下方展示三个能力：
跨团队协作、可编辑网页、可编辑视频；最后放置联系按钮。
按钮先使用占位链接，不收集个人信息。
```

5.2 操作步骤

1. 在“创作 → 做网站”中选择内置 Signal Workspace 模板。
2. 填写名称、受众与页面要求，点击“生成网站”。
3. 在输出面板打开 HTML 预览，核对首屏和文件路径。
4. 自动续做中断后，打开 design/<session-id>/entry.html，人工完成三张能力卡片与页脚，刷新预览。
5. 核对联系按钮仍为占位动作，页面不提交个人信息。

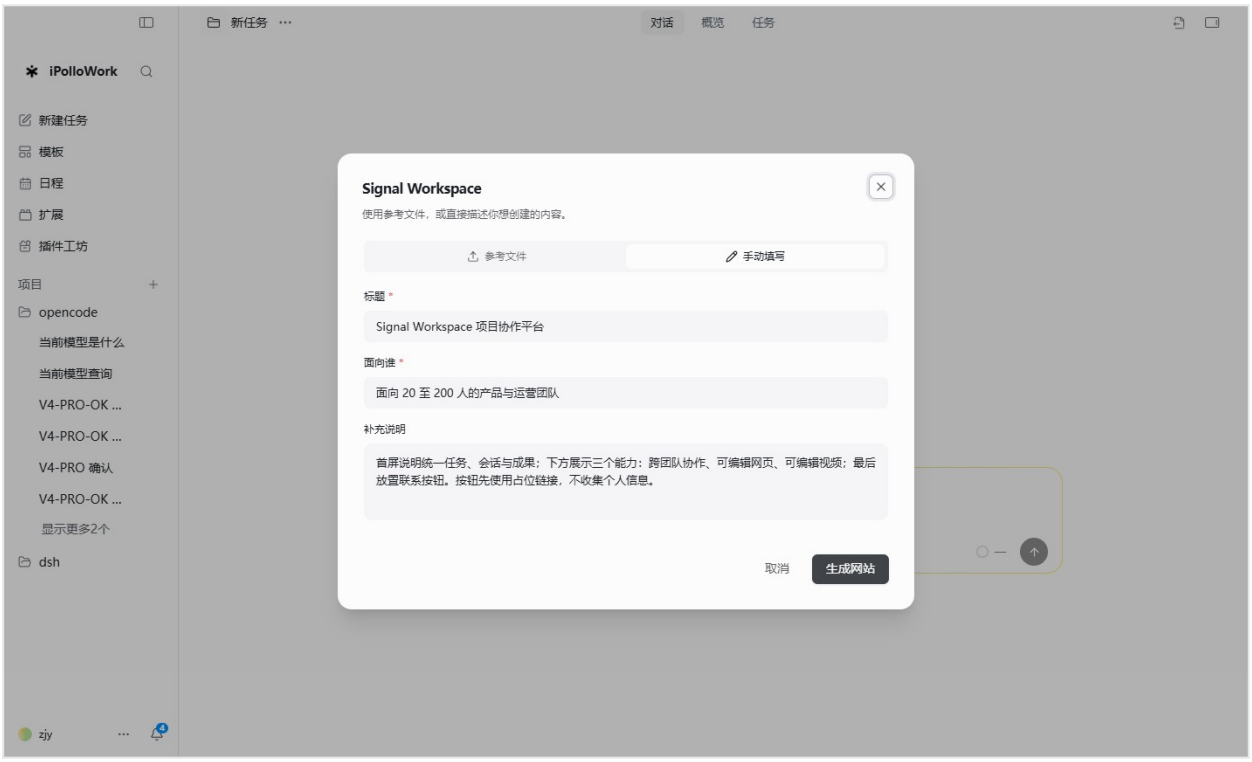


图 3 网站任务的实际输入界面，2026 年 10 月 5 日。

5.3 输出与限制

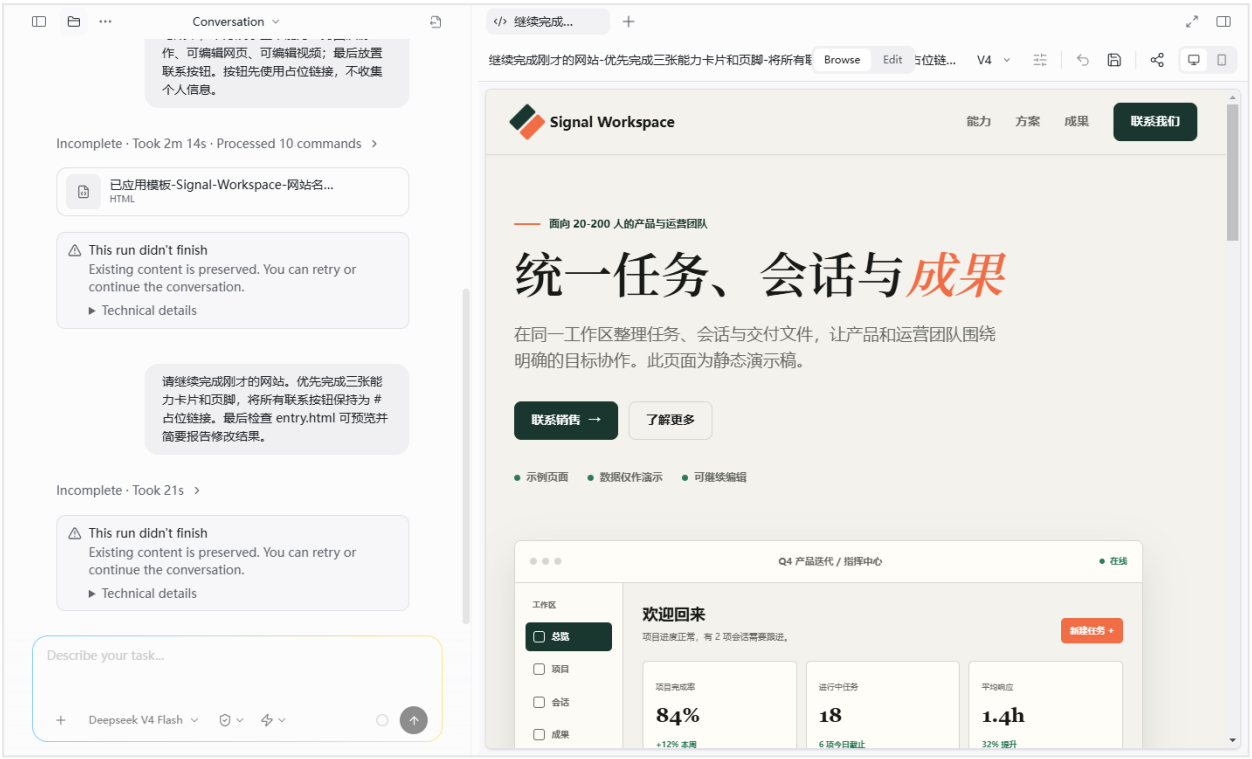


图 4 工作台中的实际 HTML 预览。左侧保留自动运行中断记录，右侧显示人工补完后的首屏。

输出文件为 design/<session-id>/entry.html，可继续编辑。页面包含中文首屏、三项能力卡片、工作过程、页脚和联系占位按钮。这个案例验证了模板、文件写入、人工编辑和工作台预览的闭环；并不能证明一次提示可以稳定生成完整页面。

检查项	本次结果	上线前仍需做
结构与文案	已补齐并在预览中打开	移动端逐宽度检查、品牌审校
按钮	占位动作，无真实表单	接入真实联系路径与隐私告知
示例数据	页面保留演示数字	替换为经核验数据或删除
自动生成	续做两次中断，文件保留	定位模型与工具故障，人工复核

适用判断：适合以模板快速起稿，并允许设计或前端人员继续修订。若项目要求无需人工干预自动上线，应另外做端到端验收。

第 6 章 视频可视化案例

案例目标：制作 15 秒、16:9 的内部趋势简报。工程保留 HTML 场景和时间轴供继续编辑；本次在本机完成 MP4 导出。

6.1 实际输入

```
模板：AI 热点拆解
标题：AI 每周趋势简报
受众：产品和运营团队
要求：15 秒、16:9。开头显示标题；中段展示智能体协作、
可编辑网页、可编辑视频；结尾显示“了解更多”。
只生成本地工程，不发布。
```

6.2 操作步骤

1. 打开“模板 → 视频”，选择“AI 热点拆解”并使用模板。
2. 填写标题、受众与完整要求，提交后查看写入文件。
3. 检查 video/<session-id>/index.html 的三场景结构与 15 秒时长。
4. 在工程目录运行下方 lint 和导出命令；导出后播放 MP4，检查首尾和中段。

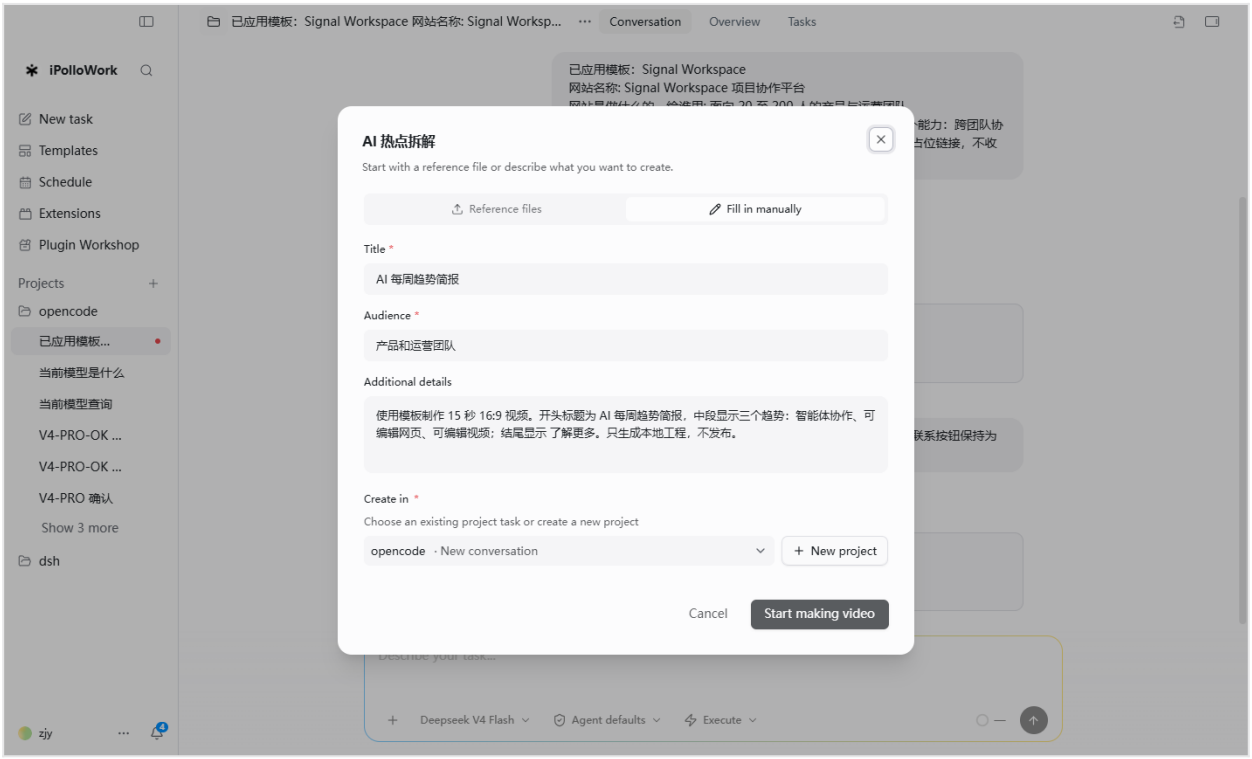


图 5 视频任务的实际输入界面。

6.3 输出与限制

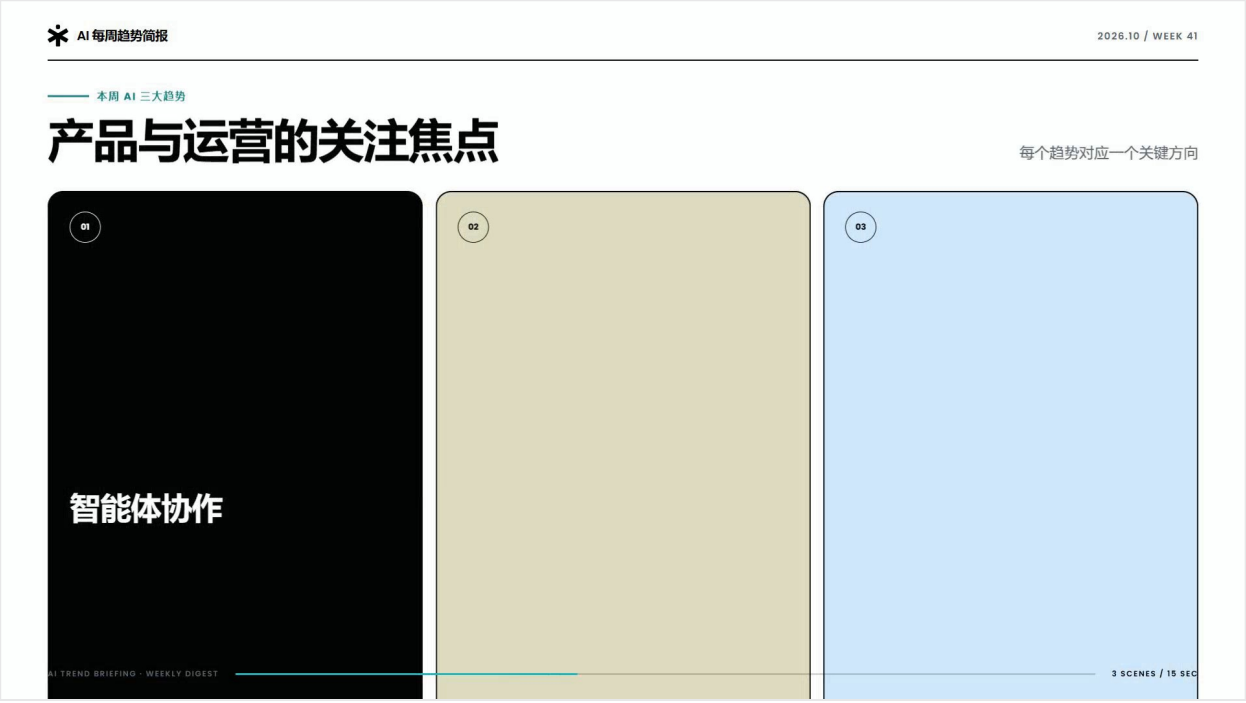


图 6 本次导出 MP4 的第 6 秒画面，来自实际生成的工程。

输出为 video/<session-id>/index.html 和本地 MP4。工程包含 3 个场景，分辨率 1920×1080、帧率 30 fps、时长 15.0 秒；lint 为 0 错误、0 警告，草稿质量 MP4 导出成功。

6.4 可复现的命令与编辑范围

在 video/<session-id> 工程目录执行以下命令。v0.50.13 仓库中 HyperFrames CLI 为 0.7.60；本次复核使用同版本仓库 CLI 完成 lint。npx 首次运行需要下载包；本地部署可预装并锁定版本。导出需要浏览器运行时、字体、完整 assets 与编码环境。

```
npx --yes hyperframes@0.7.60 lint
npx --yes hyperframes@0.7.60 render --fps 30 --quality draft --workers 1 --output
weekly-brief.mp4
```

画面文字在 HTML 中修改，时长需同时核对工程时间属性和动画时间线。保存后重新 lint、预览与导出，检查首尾帧、文字安全区和声音。本例未验证独立字幕轨、TTS 或任意素材替换。10 月 8 日复核 lint 通过，15.0 秒成片于 10 月 5 日导出。

第 7 章 计划任务与自动化

案例目标：建立每日运营摘要任务，数据限于项目任务记录，结果供内部复核。本次完成计划保存与定时触发，因模型额度耗尽未生成摘要。计划任务是智能运营的执行基础；外部数据、渠道发布和效果回写需另行接入并验收插件或 MCP。

7.1 实际输入与触发

```
任务标题：每日运营摘要（示例）
任务说明：每天汇总前一日已完成任务、失败任务和待人工确认事项；
仅生成内部摘要，不发布或发送给外部人员。
数据来源：本项目任务记录。
输出：三段式摘要和失败任务列表。
运行：每日，跟随项目 Agent；示例开始时间 16:30（Asia/Shanghai）。
```

7.2 操作步骤

1. 打开 Schedule，点击 New schedule，选择项目 opencode。
2. 填写标题和说明，设置开始及截止时间。
3. 打开 Run automatically，将 Repeat 设为 Daily，执行模型保留 Follow project Agent。
4. 保存后返回日程板，确认出现任务卡片与时间。

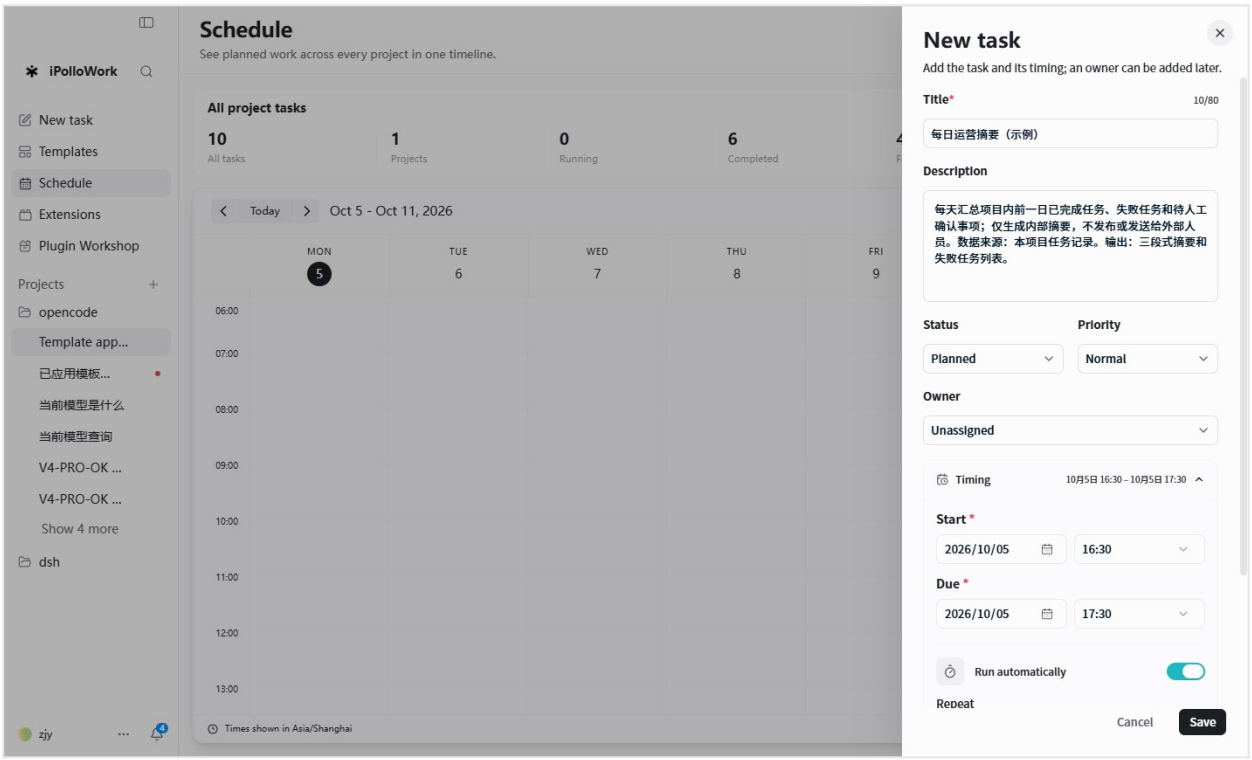


图 7 每日自动运行配置，包含重复周期和执行模型。

7.3 输出、审核与失败处理

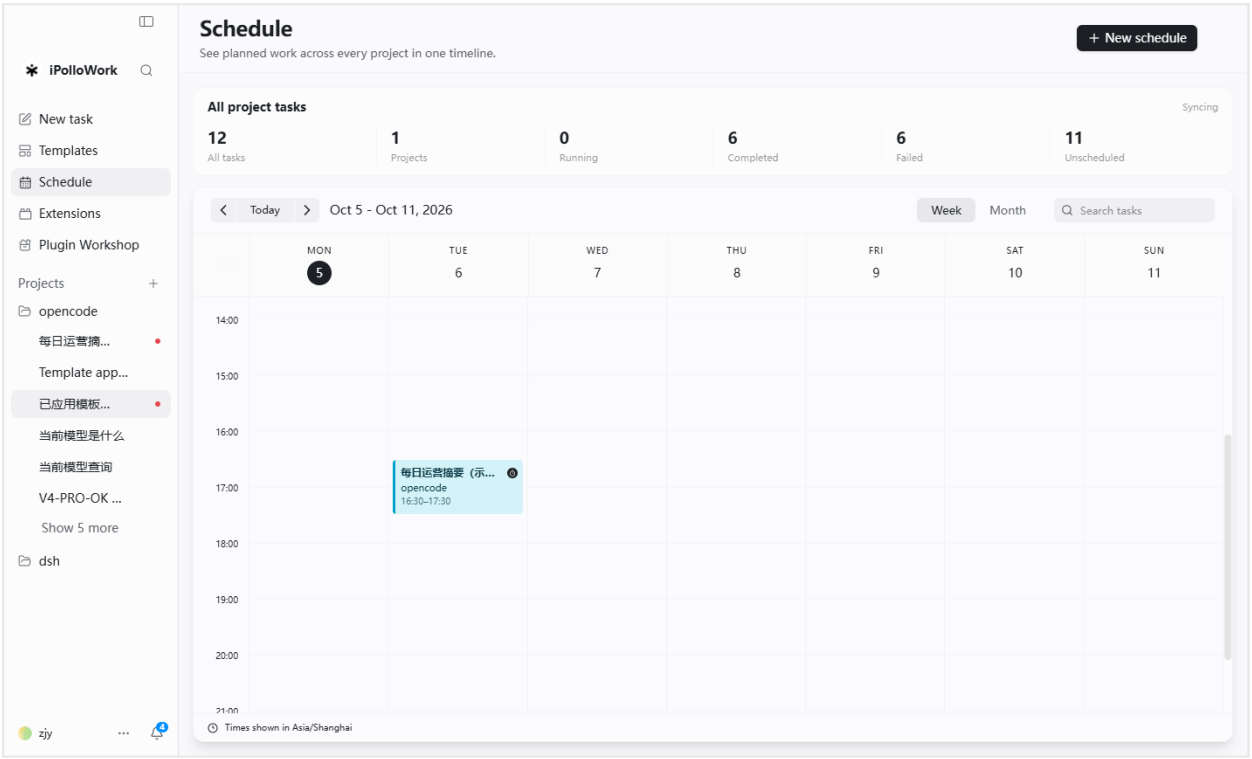


图 8 保存后的重复计划进入下一次日程；实际触发的会话另显示模型额度不足。

本次得到工作项、触发会话和失败提示，没有摘要；重复计划已排入下一次日程。代码测试覆盖一次、每日、每周推进和失败重试，实际产出仍依赖 Agent、模型额度和服务在线状态。

补足额度或切换模型后，先手动复跑并核对摘要，再继续重复计划。渠道发布、账号授权和效果回写需先安装并验收相应插件或 MCP；本例没有外部发布动作。

7.4 固定输入后重跑摘要

示例包 inputs/ops-tasks.csv 含三条虚构任务。放入当前项目同一路径，先手动运行以下任务，再用于日程。此输入供后续验收，尚无成功摘要实测。

```
只读取 inputs/ops-tasks.csv，不访问外部系统。
按已确定工作、临近期限、待确认分类，保留负责人与日期。
缺失日期写待确认，不推测进度；保存 reports/daily-summary.md。
不外发、不修改 CSV，返回输出路径供复核。
```

通过条件：到时触发，读取 CSV 并保存无编造摘要，记录可定位时间、引擎和输出。成功路径仍待验收。

第 8 章 插件与扩展开发

本章配套文件为 iPolloWork_开发示例_v0.50.13_2026-10-08.zip，随本手册交付。包含两个插件的源码与成品、打包脚本、MCP 调用脚本和虚构运营 CSV；解压后先读 README，再执行示例。

扩展分三个层次。Skill 是指导 Agent 何时、怎样完成一类工作的说明；MCP 是把外部工具或数据源接入运行时的协议；iPolloWork 插件包负责资源、版本、安装和授权生命周期。一个插件可包含多个 Skill 和远程 MCP。

对象	解决的问题	普通导入边界
Skill	规定输入、步骤与输出	目录含 SKILL.md；安装时不自动执行脚本
远程 MCP	调用外部工具与资源	普通包要求 HTTPS，按服务完成 OAuth
插件包	打包、更新、启停、兼容	声明式资源可普通导入；特权能力需审核签名
引擎原生绑定	调用特定运行时能力	按引擎和签名流程单独验收

8.1 最小声明式插件包

下例只整理用户提供的文本，不依赖外部服务。目录内只有清单和 SKILL.md；压缩包扩展名为 .ipollowork-plugin。以下清单已在 v0.50.13 的校验器中验证，压缩包也通过归档读取。

```
acme-summary/
  ipollowork.plugin.json
  skills/acme-summary/SKILL.md
{
  "schemaVersion": 2,
  "id": "acme-summary",
  "name": "Acme 文档整理",
  "description": "整理用户给定资料，生成摘要与行动项。",
  "source": {"format":"ipollowork-extension-
manifest","origin":"local","trusted":false},
  "package": {"version":"1.0.0","publisher":{"id":"acme","name":"Acme Team"},
              "updateId":"acme/acme-summary"},
  "resources": [{"type":"skill","id":"acme-summary","label":"文档整理",
                 "path":"skills/acme-summary","required":true}]
}
```

8.2 Skill 内容、打包与验收

```
---
name: acme-summary
description: 当用户要求整理文档、提取重点或生成行动清单时使用。
---
# 文档整理
输入：用户提供的文本，以及希望给谁阅读。
步骤：识别事实与来源；写 150 字摘要和行动清单；
    对缺失信息标注“待补充”，不得编造。
输出：摘要、关键事实、行动项、待补充。
依赖：无外部服务。
```

准备 Python 3.9 或更高版本。在 acme-summary 的父目录保存以下脚本为 pack.py，运行 python pack.py，生成 acme-summary-1.0.0.ipollowork-plugin。归档根目录须有唯一清单；不要包含源码仓库、node_modules、密钥或用户资料。

```
from pathlib import Path
from zipfile import ZipFile, ZIP_DEFLATED
root = Path("acme-summary").resolve()
out = Path("acme-summary-1.0.0.ipollowork-plugin").resolve()
assert (root / "ipollowork.plugin.json").is_file()
assert not out.is_relative_to(root)
with ZipFile(out, "w", ZIP_DEFLATED) as z:
    for file in sorted(root.rglob("*")):
        if file.is_symlink(): raise ValueError(file)
        if file.is_file(): z.write(file, file.relative_to(root).as_posix())
```

在交付安装包前，用 ZIP 查看器确认只有清单与 SKILL.md 两个文件，且清单位于根目录。下节说明在软件中安装和使用；真正可用还取决于目标项目的模型、运行时和插件安装状态。

8.3 添加 MCP 与特权边界

普通包只允许声明远程 HTTPS MCP，不允许在清单中放固定 headers、env 或本地 command。需要登录时优先使用远程服务支持的 OAuth。local-service、进程型 MCP、engineBindings、原生程序、自定义授权或更高权限，需先由维护方审核并进入受信发布者签名流程；自行签名不等于软件信任。

```
// 清单 resources 中增加
{"type":"mcp","id":"acme-search","path":"mcp/acme-search.json","required":true}

// mcp/acme-search.json
{"type":"remote","url":"https://mcp.example.com/mcp","enabled":true,"oauth":{}}
```

示例域名仅表示配置格式，不能直接连接。实际接入时需替换为已部署的 HTTPS MCP 服务，安装后完成该服务要求的 OAuth，并测试至少一次工具调用。

安装器有体积、文件数、路径和未知字段限制；以目标版本的插件交付说明和导入错误为准。跨 OpenCode、DSH 与 Codex Harness 复用 Skill 时，应逐个验证工具名称、路径和输出行为。

8.4 在桌面端安装并使用插件

以下步骤面向收到 .ipollowork-plugin 文件的用户，与“插件工坊 → 导入插件源码”不同。后一入口用于继续开发源码，不用于安装本节的成品包。

1. 打开 iPolloWork，选择一个本地项目，进入 Extensions 或插件库。
2. 点击“导入插件 → 文件”，选择 acme-summary-1.0.0.ipollowork-plugin。
3. 在预览中核对名称、发布者、版本和唯一的 Skill 资源；本例不应要求账户授权或本地执行权限。
4. 点击“安装插件”；如界面提示重新加载，完成重新加载后确认 Skill 已启用。
5. 在项目会话中粘贴下面的示例输入，检查回答是否包含摘要、关键事实、行动项和待补充。

```
请使用 acme-summary 整理下面的会议纪要，面向项目负责人。
原文：10 月 5 日确认首页文案由产品组修改；设计组在 10 月 8 日前
交付移动端稿。发布日期尚未确定。
```

期望输出格式：摘要说明两项已确定工作；关键事实包含负责人和 10 月 8 日期限；行动项分别指向产品组与设计组；“待补充”写明发布日期未确定。模型生成的具体措辞可能变化，不能凭安装成功推断内容质量。

8.5 更新、停用和排错

操作或现象	处理方法
更新	修改清单或 Skill 后递增 package.version，保持 id、updateId 与发布者身份稳定；重新打包并在插件库执行更新
停用 / 卸载	先停用 Skill 并确认新任务不再调用；卸载后核对插件资源移除，项目中已有成果另行保留
导入失败	确认扩展名为 .ipollowork-plugin，清单位于包根目录且只有一份，schemaVersion 为 2
安装后不可见	检查项目和引擎选择、插件启用状态与重新加载提示；确认资源路径覆盖 SKILL.md
调用失败	先检查模型与会话，再检查 Skill 文案中是否引用了不存在的工具、文件或外部依赖

验证记录：acme-summary 和 acme-public-docs 清单通过 v0.50.13 校验器；前者归档读取成功。10 月 8 日仓库回归样例完成归档预览、安装、卸载、更新与回滚，2 项测试通过。回归样例与教学包的桌面验证是不同范围，教学包的安装、模型调用、更新与卸载界面全流程仍待验收。

8.6 源码开发与成品安装是两条路径

开发者可把解压后的目录放在工作区 plugins/acme-summary，进入“设置 → 扩展 → 插件包 → 开发者：安装本地插件包”，填写工作区相对路径并先点“验证”。这里读取源码目录；第 8.4

节的“导入插件 → 文件”读取 .ipollowork-plugin 成品。选择路径前确认当前工作区，避免把包放到另一个项目。

阶段	开发者应检查	使用者看到的结果
验证	清单、资源路径、权限和将写入的文件	预览名称、版本、资源与授权要求
安装与授权	先安装最小能力；需要服务时再接入授权	插件可见，按需连接外部账户
调用	在新任务中触发 Skill 或声明的动作	获得业务输出或明确的失败原因
更新与回滚	递增语义版本，保留 ID 与 updateId；勿改安装目录	可选择更新，失败时检查回滚记录
停用与卸载	确认资源、进程和授权清理结果	新任务不再调用该插件

已发布的同一版本应视为不可变。安装器记录包拥有文件的哈希；若有人在安装目录手改文件，更新或卸载可能因冲突停止。开发时应修改原始目录、提高版本后重新验证。

8.7 服务接口与授权边界

v0.50.13 的工作区服务提供插件包生命周期接口，设置页已经使用这些能力。下表用于开发者平台或 CLI 集成；路径中的 :id 是工作区 ID，:pluginId 是清单 ID。调用写接口必须经过服务器鉴权、工作区权限和可能出现的用户审批。

用途	方法与路径
列出与验证	GET /workspace/:id/plugin-packages；POST /workspace/:id/plugin-packages/validate
安装与更新	POST /workspace/:id/plugin-packages；POST /workspace/:id/plugin-packages/:pluginId/update
状态与恢复	PATCH /workspace/:id/plugin-packages/:pluginId；POST /workspace/:id/plugin-packages/:pluginId/rollback
卸载	DELETE /workspace/:id/plugin-packages/:pluginId

本地目录验证、安装和更新请求使用 JSON：{"packageRoot":"plugins/acme-summary"}；停用请求使用 {"enabled":false}。文件导入另走 /plugin-packages/import/validate 与 /plugin-packages/import，不能把压缩包路径当作 packageRoot。接口示例只说明请求形状，不替代目标部署的访问令牌和审批流程。

需要 API Key 的插件应声明授权方法，由宿主的插件设置页保存凭据。Skill 不应要求用户在聊天中粘贴 Key；服务动作只返回业务结果，不返回密钥。远程 MCP 的 OAuth 授权也应在连接界面完成。原生引擎插件与其运行时共享进程，不能把存储隔离解释为恶意代码沙箱。

8.8 兼容与发布验收

检查维度	最低验收样本	记录内容
系统与版本	声明支持的 Windows / macOS / Linux 与 v0.50.13	安装包版本、平台和架构
运行时	逐个验证声称支持的 OpenCode、DSH 或 Codex Harness	Skill 触发、工具名和输出路径
生命周期	安装、调用、重启、更新、停用、回滚、卸载	每一步结果与失败截图
安全与隐私	权限预览、网络目标、凭据保存和卸载清理	数据流、日志去敏与撤销方式

对外发布页至少写清包 ID、发布者、版本、兼容范围、权限理由、数据去向、变更记录和支持渠道。商店上架与受信签名涉及平台审核；本文只说明现有本地包开发与导入，不把未来上架流程作为已交付入口。

8.9 建议随包提供的说明

最小示例故意只含清单和 Skill。面向第三方用户分发时，开发者还应在包外提供一页安装说明，写明以下信息，便于采购、管理员和使用者核对同一版本。

字段	建议写法
适用范围	插件 ID、版本、发布日期、支持的 iPolloWork 版本与运行时
安装与撤销	导入入口、所需权限、首次连接、停用和卸载步骤
数据处理	会发送哪些字段到哪个域名，是否留存，如何删除与撤销授权
验收证据	安装、调用、重启、更新、失败恢复的测试环境与结果
变更与支持	已知限制、更新内容、问题反馈入口和维护周期

示例 acme-summary 当前只有代码级包验证记录，因此说明页应如实写“桌面界面流程待验收”。不应把开发机上的单次运行结果写成对所有系统、模型或引擎的兼容保证。

8.10 清单字段参考

字段依据 v0.50.13 的 schemaVersion 2 校验器。“无”表示未声明默认值；未知字段和非法路径可能被拒绝。

字段	类型	要求与默认	规则
schemaVersion	整数	必填，无	只能为 2
id 与 name	字符串	必填，无	id 使用小写稳定 ID；name 用于显示
description	字符串	必填，无	说明业务能力，允许空字符串但不建议
source	对象	必填，无	format 与 trusted 必填；第三方 trusted 为 false
resources	数组	必填，无	资源 ID 唯一，path 限在包内
package	对象	可选；成品示例提供	version 为语义版本，updateId 为稳定更新标识
package.publisher	对象	可选，无	声明身份不等于受信签名
package.compatibility	对象	可选，无	示例固定 ipollowork 为 0.50.13
package.engines	字符串数组	可选，无	省略可移植包；提供则逐个验收
permissions	数组	可选，无	按行为说明权限理由；不能替代批准

schema 校验不授予运行时权限。成品包仍须遵守版本、哈希与冲突规则，特权能力还需受信审核。

8.11 可真实调用的 HTTPS MCP 示例

示例 acme-public-docs 包含远程 MCP 和 Skill。第三方 DeepWiki 服务无需 Key，使用 Streamable HTTP，只发送指定的公开 owner/repo 名称。

```
mcp/deepwiki.json
{"type":"remote","url":"https://mcp.deepwiki.com/mcp","enabled":true}

resources 中的 MCP 项
{"type":"mcp","id":"deepwiki","path":"mcp/deepwiki.json","required":true}
```

从扩展设置安装后，在会话中输入“使用 DeepWiki 的 read_wiki_structure，读取公开仓库 modelcontextprotocol/python-sdk 的目录，不发送本地文件”。若工具不可见，检查 MCP 启用、项目引擎和网络；未索引仓库会返回对应错误。

10 月 8 日运行 python mcp_public_check.py，完成 initialize、tools/list 和 tools/call；返回 Overview 等目录项，isError 为 false。结果见示例包 JSON，仅验证远程协议调用，桌面安装与 Agent 调用仍待验收。

8.12 接口鉴权请求响应与排错

工作区服务接口按部署取得 Bearer token；用 Authorization: Bearer <token> 请求目标 iPolloWork server。令牌应从受控环境读取，不能硬编码在源码、Skill 或日志。查看者令牌不可执行写操作；安装、更新和卸载需可写服务、collaborator 范围及宿主审批。

```
POST /workspace/<workspace-id>/plugin-packages/validate
Authorization: Bearer <token>
Content-Type: application/json
{"packageRoot":"plugins/acme-summary"}

HTTP 200 响应结构节选
{"preview":{"manifest":{"id":"acme-summary"},
            "safety":{"level":"declarative","localCode":false}}}
```

preview 还包含资源投影、版本和写入计划等字段；上面的节选用于说明结构。先验证再安装，不能把验证 HTTP 200 当成模型调用成功。输入路径必须位于当前工作区；请求中的 workspace-id 与相对包路径应对应同一项目。

HTTP 状态与代码	含义	处理
401 unauthorized	令牌缺失或不匹配	核对目标服务与访问令牌
403 forbidden 或 read_only	权限不足或服务器只读	使用已授权范围；不绕过审批
400 plugin_package_import_unsafe	普通导入含不允许的特权能力	改为声明式资源或申请受信审核
400 plugin_package_archive_format_invalid	成品扩展名或归档格式不匹配	使用 .ipollowork-plugin，根目录放清单
409 plugin_package_downgrade_confirmation_required	降级需要明确确认	先备份，按界面确认版本回退
400 plugin_package_id_mismatch	更新包 ID 与安装对象不同	核对稳定 ID 与 updateId

错误响应采用 code、message、details 结构；排障时保留 HTTP 状态、code、时间、版本和脱敏请求。本文接口说明固定到 v0.50.13，未提供跨版本不变的公共 API 保证。升级前重放集成测试，Engine Protocol 继续按内部接口处理。

8.13 调试与贡献

先检查导入预览和错误响应，再看相应任务事件及本地或自建服务日志。实际日志路径由启动配置与部署平台决定，仓库未提供统一日志导出契约；支持材料应附运行方式和已找到的日志位置，并去掉 token、Key、账户与客户资料。

功能问题通过附录 A 的 Issues 链接反馈，至少提供产品版本、系统与架构、项目引擎、插件 ID 与版本、最小输入、期望结果、实际结果和脱敏证据。安全漏洞应通过维护方确认的私密渠道报告，不在公开 Issue 中披露利用细节。该发行版 SECURITY.md 未列出可核验的收件地址，正式发布前仍需维护方补齐私密报告入口。

第 9 章 部署与数据安全

9.1 两种部署选择

形态	组成	适用条件
本地桌面	桌面 UI、本地服务、项目文件与运行时	个人或小规模内部使用；核对授权人数
私有远程 Worker	持续在线 Worker、持久化目录、令牌与 TLS	需要远程访问与持续运行

私有 Worker 的网络入口应使用 HTTPS，不要暴露底层 OpenCode 端口。Worker 的数据目录与项目文件应持久化，并落实访问控制、备份和恢复责任。

9.2 上线前核对清单

1. 固定发行包和依赖版本，记录来源、签名或未签名状态。
2. 配置 TLS、可信来源、访问令牌、提供方凭据和最小网络出口；密钥不写入项目文件或日志。
3. 持久化工作区、任务记录、数据库和对象存储；明确备份保留期、恢复目标与演练负责人。
4. 为模型、MCP、浏览器自动化和第三方插件设置审批与撤销流程。
5. 用真实企业身份、目标模型和插件做权限、失败、恢复及审计验收。

9.3 数据驻留与凭据

本地优先只说明默认工作空间在本机；连接外部模型、MCP 或插件服务后，数据可能跨出本地。Worker 可使用 IPOLLOWORK_TOKEN 和 IPOLLOWORK_HOST_TOKEN；生产环境应落实传输加密、凭据轮换与日志去敏。

安全模型明确残余风险：用户批准的原生命令、第三方插件、浏览器自动化、外部模型及配置不足的自建部署不会自动被平台完全沙箱化。生产验收应针对实际启用的能力。

9.4 数据位置与恢复责任

数据或凭据	典型位置	需要的控制
项目文件与生成成果	本地或 Worker 挂载目录	访问权限、版本备份、恢复演练
运行时状态	Worker 数据目录和 OpenCode SQLite	持久卷与一致性备份
插件和模型凭据	操作系统、密钥服务或授权存储	最小授权、轮换、日志去敏
外部模型 / MCP 请求	所连接的第三方服务	发送字段、驻留地和保留期确认

恢复时先停止新的写入并保留事故现场，再按数据库、Worker 状态和项目文件的依赖关系恢复；恢复完成后复跑一项只读查询、一项文件写入和一项插件调用。具体恢复点与时间目标应写入企业运维方案，不能从本手册推断产品 SLA。

9.5 联网条件

能力或路径	需要什么	企业评估范围
本地成果编辑	已存在的文件和素材	可在无模型请求时编辑；全离线模式未验收
远程模型	提供方网络、账户与额度	提示词和选定上下文可能外发
远程 MCP	目标 HTTPS 服务与按需授权	工具参数和结果按该服务策略处理
首次安装与更新	发行包或预先缓存依赖	内网需提供安装包与依赖镜像

商业授权授予的权利与部署里实际启用的功能应分别确认。合同应列清部署组件、已启用功能、责任人和验收证据。

9.6 具体数据如何保存与外发

提示词与会话状态由所选运行时及项目服务按配置保存；参与模型调用时会将任务输入和选定上下文发给提供方。用户选择的文件仅在工具读取或上传动作需要时进入相应处理路径，但本手册没有覆盖每一种工具的发送字段，应在真实工作流中核对。成果文件保存在项目工作区；连接远程 Worker 时，该目录位于 Worker 持久卷。

服务日志可能包含任务与工具元数据，应按实际日志配置去敏和设定留存。模型和插件凭据由提供方配置或插件授权存储处理，不能与项目成果一起公开导出。数据留存期限、删除方式和处理地点由本地配置、企业运维与第三方政策共同决定，本版未给出统一保留期。

9.7 导出卸载与升级

1. 导出前记录项目目录与引擎，保存任务说明和已生成文件；停止新的写入，再复制项目文件与必要素材。
2. 需要恢复会话和计划时，还应备份对应运行时状态和 Worker 数据；仅复制 HTML 或 MP4 不能恢复全部任务状态。
3. 卸载插件先停用，再检查拥有文件、服务和授权清理；卸载桌面程序前自行保存项目和用户数据，不假定所有目录会自动导出或清除。
4. 升级前备份，核对迁移与插件兼容，使用预生产环境完成登录、任务、导出和失败恢复，再开放生产使用。

以上为运维执行顺序。统一项目导出、所有运行时会话迁移和卸载后数据清除的端到端行为仍需目标环境验收。

第 10 章 商业授权与维护

v0.50.13 根目录采用 iPolloWork Source Available License 1.0。本文统一称为源码可见许可，不称为 OSI 批准的开源许可，也不以代码可读推断免费商业使用权。权利以该版本 LICENSE 与有效书面授权为准。

该版本 LICENSE 的版权声明与 Licensor 定义为 Different AI, Inc. 或其授权继承者。iPolloWork 品牌、实际销售方与合同签约主体之间的授权关系不能仅由本手册确定；企业签约前应取得许可方、签约方及必要授权链条的书面确认。本文保留历史与第三方版权声明，不推断其权属异常。

使用方式	许可证中的边界
个人非商业使用	一名自然人的非商业工作、学习、研究、评估或测试可在许可范围内进行
少于 3 人的小规模内部使用	限内部、非商业；不得向外部客户提供软件或功能
3 人及以上	无论本地、托管、商业或非商业，均需事先书面授权
付费交付、托管、SaaS、白标或客户服务	需事先书面授权
品牌与再分发	保留品牌和许可声明；修改版标明变更；商业再分发须授权

第三方组件和早期 MIT 许可部分仍按各自许可处理；历史授权不因当前许可证自动失效。企业采购应把人数、部署形态、外部交付、数据处理、支持响应和升级范围写入合同。

10.1 版本与升级

本文以 v0.50.13 为准。升级前备份项目目录与数据库，核对插件 schema、运行时 sidecar、模型登录方式和浏览器扩展。先在预生产环境重放第 5 至 7 章的关键任务，再批准生产升级；特别验证数据库迁移、MCP 授权和视频导出。

10.2 故障排查顺序

1. 确认发行版本、操作系统和项目引擎；区分 UI MCP、Codex Harness、DSH 与 OpenCode。
2. 检查工作区是否可写、模型是否可用、任务服务是否在线、外部服务是否可达。
3. 查看任务事件与错误信息，记录可复现输入、时间、项目类型和插件清单。
4. 恢复时先保全工作空间和输出文件，再重试单次任务；不要盲目重复外部发布动作。

公开问题可通过仓库 Issues 与发布说明跟进；企业部署和商业授权应以仓库所有者及合同约定渠道为准。

附录 A 资料与术语

发行安装包与发布说明

安装与系统架构

插件包说明

插件字段定义

信任与安全边界

商业使用许可证

问题反馈

安全政策

DeepWiki MCP 服务说明

Harness：智能体的执行与工具协作层。Engine Protocol：工作台与具体执行引擎之间的适配边界。Skill：供 Agent 遵循的任务说明。MCP：连接工具和资源的协议。HyperFrames：视频工程使用的 HTML 合成与渲染能力。

仓库资料链接固定到 v0.50.13；Issues 与外部服务文档以访问当日为准。

附录 B 使用者任务检查单

B.1 网页任务

- 输入中写明受众、内容结构、品牌素材、移动端要求、交互行为和占位内容。
- 检查 HTML、CSS 和资源文件实际存在；打开预览并手动修改至少一处内容。
- 发布前确认图片许可、链接目标、表单后端、隐私声明、无障碍与响应式效果。

B.2 视频任务

- 先约定时长、画幅、场景脚本、素材路径、字幕语言、音频来源和输出格式。
- 检查 video/<session-id>/index.html 与资源目录；逐段预览文字、过渡和时间线。
- 渲染后重新播放导出文件，检查黑帧、溢出、音画错位、编码和结尾。

B.3 智能运营任务

- 明确触发时间、时区、数据源、只读或写入权限、报告格式和人工复核人。
- 先手动运行，检查空数据、重复运行和连接失效时的表现。
- 如需外发，列明目标平台、审核规则、失败补偿与发布回执。

这些检查项适用于本地和远程项目；具体入口随客户端与插件版本变化。