MetaOnticVideo

MetaOnticVideo 安装与使用指南

先安装插件并配置 Key，再新建会话，将模式切换为“视频创作助手”，开始制作视频。Key 配置配图来自插件真实组件的本地预览，使用模拟数据且密钥已隐藏；安装、会话与制作流程采用操作示意图，菜单名称以 Deepseek–Harness 实际界面为准。

MetaOnticVideo 是 DeepSeek Harness 的 AI 视频创作插件。使用 Seedance 2.0 制作超过15秒的视频时，会分段生成与续拍，逐段预览确认后在本地合并；完整制作流程见下文。

1. 如何安装插件

安装插件：从添加到打开

操作示意图 · 菜单名称以 Deepseek–Harness 实际界面为准

① 打开 Deepseek–Harness → 插件 → 添加插件

粘贴正式版 .tgz 附件下载链接，按提示完成安装。

② 发布页 → 复制 .tgz 附件下载链接

GitHub Releases 或 Gitee Releases → 附件
Windows 可下载 .tgz，填写文件绝对路径安装

③ 已安装插件 → MetaOnticVideo → 查看

进入设置页，选择接入方式；配置后选择“视频创作助手”开始。

1.1 安装前准备

本教程同时适用于 **Windows 与 Mac（macOS）**。MetaOnticVideo 是 DeepSeek Harness 插件，需要先安装 DeepSeek Harness 桌面端，不能作为独立软件打开。

1. 从 DeepSeek Harness 的官方发布渠道获取对应系统的桌面安装程序，按安装向导完成安装；Mac 请按电脑芯片选择官方提供的对应安装包。
2. 打开 DeepSeek Harness，确认能正常进入桌面主界面并看到「插件」入口。
3. 准备可联网的环境及 API Key；Key 的申请与填写见第 2、3 节。

安装插件成品包不需要自行安装 Git、Node.js 或 Python。多段视频本地合并所需的 FFmpeg 可在开始使用后按需配置。

项目	Windows	Mac（macOS）
推荐安装方式	正式版 `.tgz` 附件链接或 Windows ZIP	仓库地址安装；成品包作为备用
本地安装备选	Windows ZIP 解压后的 `package`，或 `.tgz` 解压后的 `package`	`.tgz` 解压后的 `package`
本地目录示例	`C:\MetaOnticVideo\package`	`/Users/你的用户名/Applications/MetaOnticVideo/package`
在线更新入口	插件详情 → 检查更新	插件详情 → 检查更新

Mac 推荐：仓库地址安装

在 DeepSeek Harness「插件 → 添加插件」中粘贴下面任意一个地址，按提示安装并启用：

```
git+https://gitee.com/hackloop/metaontic-video.git
```

或使用 GitHub：

```
github:wxsimon2022/metaontic-video
```

仓库安装需要 Git 能访问对应站点。若拉取失败，可使用下面的成品包或本地目录方式。

1.2 在线安装：Windows 与 Mac 通用

1. 打开 GitHub 正式版下载页，也可打开 Gitee 发布页。
2. 在附件中找到 `metaontic-video-数字版本号.tgz`，右键复制该附件的下载链接。不要选择 Source code，也不要复制仓库地址或发布页地址。
3. 打开 DeepSeek Harness，进入「插件 → 添加插件」，粘贴刚才复制的 `.tgz` 下载链接，按提示安装。**在线安装的地址必须以 `.tgz` 结尾**；`MetaOnticVideo-Windows-数字版本号.zip`、解压出来的 `package` 文件夹、发布页网址和 Source code 源码包都不是在线安装地址，填进去会提示「不是有效的插件包」。
4. 安装后按软件提示启用插件，进入「插件 → 已安装插件 → MetaOnticVideo → 查看」，打开设置页。
5. 继续按第 2、3 节申请 Key、填写并保存，再按第 4 节开始创作。

此方式使用已构建成品，不通过 Git 拉取源码。只有 DeepSeek Harness 明确提示需要重启时才重启软件。

1.3 Windows 本地安装

可在正式发布页的附件中下载 `MetaOnticVideo-Windows-数字版本号.zip`。适合希望随包查看教程，或在线安装不方便的用户。本地安装仍需先安装 DeepSeek Harness，生成图片与视频仍需要联网。

先记住：新安装选 `.tgz` 文件；恢复旧安装才复制 `package` 文件夹。

最简单的操作顺序：右键 ZIP → 全部解压 → 找到 `.tgz`（不用再次解压）→ 复制这个文件的完整路径 → 打开 DeepSeek Harness「插件 → 添加插件」→ 粘贴路径 → 安装。若粘贴到界面的路径前后带英文双引号，删掉这两个引号再安装；终端命令里的引号则保留。

使用 Windows ZIP 资料包：

1. 右键安装 ZIP，选择「全部解压」。不要在 ZIP 预览窗口内安装。
2. 打开包内安装教程，找到 `metaontic-video-数字版本号.tgz` 成品文件。
3. 在 DeepSeek Harness「插件 → 添加插件」中填写该 `.tgz` 文件的完整绝对路径。解压后常出现两层同名文件夹，路径形如 `D:\software\MetaOnticVideo-Windows-数字版本号\MetaOnticVideo-Windows\metaontic-video-数字版本号.tgz`；层级和盘符以你电脑上的实际位置为准，建议右键 `.tgz` 选择「复制文件地址」，不要手打。
4. 按提示安装、启用，进入设置页配置 Key。由安装器管理插件文件，避免直接依赖外部解压目录。

安装框要填的是文件，不是文件夹。如果提示「不是有效的插件包」，通常是填成了文件夹：解压后的 `MetaOnticVideo-Windows` 外层文件夹和 `package` 资料目录都不是安装对象，请改填同目录下 `metaontic-video-数字版本号.tgz` 的完整文件路径。

若拿到的旧 ZIP 只有 `package` 目录，请从正式发布页另外下载 `.tgz` 成品，填写其完整本地路径安装；不要把 ZIP 或发布页地址填进安装框。

也可以不下载到本地，直接粘贴成品地址： 复制下面这条 Gitee 成品地址，粘贴到 DeepSeek Harness「插件 → 添加插件」即可安装，效果与填写本地文件路径相同，无需安装 Git。国内网络优先使用 Gitee。在线安装只认以 `.tgz` 结尾的附件地址；ZIP、`package` 文件夹和发布页网址只能用于本地安装，不能粘贴到在线安装框。

```
https://gitee.com/hackloop/metaontic-video/releases/download/v3.13.13/metaontic-video-3.13.13.tgz
```

GitHub 同版本地址为 `https://github.com/wxsimon2022/metaontic-video/releases/download/v3.13.13/metaontic-video-3.13.13.tgz`。发布新版本后，把地址中的两处版本号换成发布页上的实际版本；也可以打开发布页，在附件中右键复制 `.tgz` 的下载链接。

- GitHub 正式版下载页：`https://github.com/wxsimon2022/metaontic-video/releases/latest`
- Gitee 发布页：`https://gitee.com/hackloop/metaontic-video/releases`

粘贴的必须是附件下载链接，不要填发布页网址，也不要选 Source code 源码包。Gitee 请使用上面的发布页地址，不要使用 `gitee.com/.../releases/latest`，该地址会直接下载源码包而不是插件成品。

只有 `.tgz` 文件时： 不必解压，直接使用本地文件完整路径安装。

已有本地目录安装，更新报 `locale` 缺失或目录变空： 先用完整成品恢复原安装路径下的整个 `package` 目录，确保 `package.json`、`index.js`、`client.js`、`locale` 和 `skills` 等内容齐全。不要只创建空的 `locale` 文件夹，不要重置 Key 或删除任务记录。恢复后核对当前安装与更新状态；迁移成品安装来源时使用软件提供的安装/修复入口，若提示已安装则停止并保留错误，不先卸载。

1.4 Mac 本地安装

1. 从正式发布页下载 `.tgz` 附件。
2. 在 Finder 中双击解压；若得到 `.tar`，再双击解压。找到包含 `package.json`、`index.js` 的 `package` 文件夹。
3. 将安装目录放到固定位置，例如用户目录下的 `Applications/MetaOnticVideo`，避免留在会定期清理的临时目录。
4. 打开 DeepSeek Harness「插件 → 添加插件」，选择 `package` 文件夹；若使用地址输入框，填入其完整绝对路径。
5. 按提示安装、启用，进入设置页配置 Key。保留安装目录，不要随意删除或移动。

1.5 可选：终端安装

优先使用桌面端界面。以下命令仅适用于本机已经可以运行 `dsh` 的情况；提示找不到命令时，回到界面安装即可。

在线安装：将占位内容替换为实际 `.tgz` 附件下载链接，保留引号。

```
dsh plugin --profile desktop add "复制的.tgz附件下载链接"
```

Windows PowerShell 本地安装：

```
dsh plugin --profile desktop add "C:\MetaOnticVideo\metaontic-video-数字版本号.tgz"
```

Mac 终端本地安装：

```
dsh plugin --profile desktop add "/Users/你的用户名/Applications/MetaOnticVideo/package"
```

以上路径与文件名只是示例，Windows替换为实际 `.tgz` 文件路径，Mac目录安装替换为实际 `package` 路径。

1.6 安装是否成功

- 已安装插件列表中出现 MetaOnticVideo，状态为已启用。
- 点击「查看」可以打开插件设置，未提示组件启动失败。
- 配置 Key 并保存后，新建会话可以选择「视频创作助手」。

安装报错先按第 5 节处理，不要为排查安装问题直接提交付费生成任务。

2. Key 去哪里申请

需要自行申请 Key 时，目前按以下入口操作：

https://ai.bileego.com/

1. 打开网站，注册并登录账号。
2. 按网站当前页面指引获取 API Key。
3. 确认账户可用额度，以及该 Key 是否有你准备使用的视频、图片模型权限。
4. 回到插件，选择 **MetaOntic AI**，按下一节填写并保存。

申请到 Key 后，在插件的接入方式中选择 **MetaOntic AI**。

Key 只填写在插件密钥输入框内，不要放进脚本或发到聊天中。

3. 选择 MetaOntic AI 并配置 Key

1. 在接入方式中选择 **MetaOntic AI**。
2. 根据下面两种情况配置 Key。
3. 选择视频模型和图片模型，完成后点击保存。

切换接入方式并填好 Key 后，点击一次保存并等待保存结果；后台正在同步时会等待同步完成。若提示保存失败，草稿会保留，请按提示处理后重试。

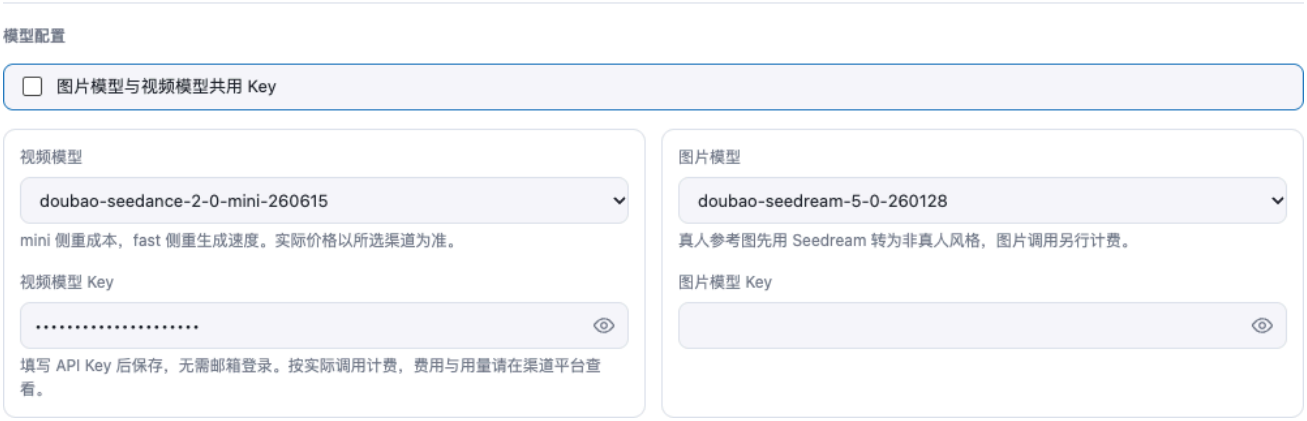
情况一：两个模型共用一把 Key



保持开启「图片模型与视频模型共用 Key」。在两个模型上方的 **共用 Key** 中填写从网站获取的 Key，再选择视频模型和图片模型。这把 Key 必须同时支持你选择的两种模型。

共用开关和共用 Key 位于两个模型上方。宽窗口中，视频模型与图片模型左右分组；窄窗口自动上下排列。

情况二：视频和图片分别使用 Key



关闭「图片模型与视频模型共用 Key」：

- 在视频模型下方的 **视频模型 Key** 中填写视频用 Key。
- 在图片模型下方的 **图片模型 Key** 中填写图片用 Key。

排列顺序：共用开关 → 视频模型 → 视频模型 Key → 图片模型 → 图片模型 Key。

独立图片 Key 未填写时，图片调用会停止并提示补填，不会自动改用视频 Key。图片提交、查询和取件均使用图片 Key；视频生成、重试和状态查询使用视频 Key。

两个模型仍沿用当前 MetaOntic AI 接入地址，分别填写 Key 不代表分别配置接口地址。MetaOntic AI 按渠道账户实际调用计费。

4. 在 Deepseek-Harness 中开始使用



安装插件并保存 Key 配置后，按以下步骤进入视频创作：

1. 在 Deepseek-Harness 中**新建一个会话**。
2. 打开会话的**模式选择**，将模式改为 **视频创作助手**。
3. 在输入框中描述你要制作的视频；需要参考图时，在会话中上传图片并说明用途。
4. 发送需求后，按助手提示确认清晰度、审核脚本，再确认是否开始制作。

这里的“视频创作助手”是会话模式，不是插件设置里的视频模型。若模式列表里没有它，先检查 MetaOnticVideo 是否已安装并启用，再重新打开模式列表；只有 Deepseek-Harness 提示需要重启时才重启软件。

第一次生成视频

第一次制作视频

操作示意图 · 清晰度在脚本阶段确认，不在插件设置页配置

① 描述需求，确认 480P 或 720P

例如：15 秒、竖屏、720P，展示一杯咖啡，不要对白和字幕。

② 看完整脚本与镜头示意图 → 审核

需要调整就直接说明；改稿后重新展示和审核。

③ 确认费用与开始制作 → 等待视频结果

有真人参考图时先同意转换与图片费用，查看转换结果，再确认用于视频。

④ 预览并保存；长视频确认后继续下一段

超时或断网先查已有任务，不直接重新生成，避免重复扣费。

切换到「视频创作助手」后，可以直接说：

> 帮我做一个 15 秒的咖啡展示视频，竖屏，720P，温暖自然，不要对白和字幕。先展示脚本，审核后再制作。

操作顺序：

1. 描述需求或提供完整方案。需要助手完善时直接说“帮我完善”；已有完整方案时可要求“只润色，不改变剧情、角色和结尾”。
2. 确认清晰度：目前支持 **480P**、**720P**，在脚本阶段选择，不在插件设置页配置。本次需求已明确指定就不重复询问；未回答或取消时不开始制作。未指定时长与画幅时，默认 15 秒、9:16。
3. 未提供图片且需要固定角色时，先选择是否生成角色参考图（图片单独计费）；同意后先看图审核，不同意则继续纯文生视频。随后查看完整脚本和正文里的镜头示意图，再确认脚本是否通过。图下会提供打开原图的链接；审核前会检查本轮是否已在正文插入分镜图；未完成时提示助手先补展示。若界面仍未显示图片，可打开原图并告诉助手补展示。示意图不是实际成片，默认不调用付费出图模型。
4. 完整脚本末尾的“费用预估”一节会展示“火山官方方案”和“MetaOnticVideo插件方案”两项人民币基础估价，以及各自模型、分段和价格依据。火山官方方案统一按 Seedance 2.5 计算，一段按 30 秒计：不足 30 秒的尾段也按 30 秒计，不按剩余秒数折半；超过 30 秒会如实按分段列出。费用明细逐段显示：首段为无视频输入，后续续拍为有视频输入；续拍按上一段完整视频时长加本段输出时长估算，并使用含视频输入单价；参考图片不算视频输入。最低用量补足尚未计入，图片处理、重试及独

立配音另计，实际以账单为准。未核实账户折扣时不默认套用优惠；不同模型效果也有差异。脚本通过后，单独确认费用与是否开始制作。按 MetaOntic AI 渠道账户实际调用计费。

5. 等待生成，预览并保存结果。完成后会显示中转站结算的实际视频费用；分段暂停时显示已生成部分，尚未结算时显示“待结算”。可按原任务ID补查，不需要重新生成。图片处理及额外重做另计。长视频分段续做沿用已确认的清晰度，例如 30 秒选择 480P，后续分段的制作确认与生成均保持 480P。改稿后需要重新展示、审核并确认制作；普通改稿保留已确认清晰度，改变清晰度须重新确认。

审核续拍过渡

续拍方式由脚本设计决定，不在插件设置中配置。请在审核时确认每段采用连续镜头还是切换近景、中景等景别，制作时按已确认的安排执行。

脚本先定义人物、道具与场景的统一称呼，再按事件顺序写清各镜头的动作、景别、视角、运镜和声音；画质与约束词按需求填写，不能代替已确认的480P或720P参数。

长视频脚本会列出各段之间的过渡画面描述：前一段人物和动作停在哪里、下一段如何接上、镜头是否切换、声音如何延续。请与脚本一起核对；不满意时在制作前修改并重新审核。

例如：男子已放下茶杯，承接其抬眼动作，切到女子近景回应，室内环境声和配乐连续。脚本会同时说明主体、动作、场景、光影、镜头、风格、画质与约束；中景、近景、视角和运镜按剧情安排，不因分段自动换地点。过渡描述有助于表达衔接意图，仍需预览实际效果。

使用参考图

上传图片并说明用途，例如“用这张图保持人物外观”或“这张作为视频第一帧”。多图需说明各自对应的人物、商品或场景；参考图与首尾帧模式不能混用，尾帧必须配首帧。

真人参考图需要先确定非真人风格，并征得你对图片处理及其费用的同意。处理完成后先展示全部转换图，等你核对并确认后再用于视频。图片生成单独计费；效果不合适或仍像真人时停止，不使用原真人图重试。

视频接口返回明确真人限制时，也遵循“同意转换 → 看转换结果 → 确认重试”流程，最多重试视频一次；图片与视频重试分别计费。部分图片已完成时会保存结果，续做不从头处理。

图片模型支持 `doubao-seedream-5-0-260128` 和 `doubao-seedream-5-0-pro-260628`，能否调用以 Key 的实际模型权限为准。

长视频与中断恢复

超过模型单次上限时，插件分段制作。每段完成先展示视频，你确认后再继续下一段。Seedance 2.0 单次 4–15 秒；2.5 保留兼容，但设置页暂不提供新候选。

暂停后可说“接着刚才的进度继续，不要重做已完成片段”。超时、断网不一定表示生成失败，先查原任务，避免重新提交。

提交结果不明时插件会阻止重复提交。保留返回的任务或提交记录，让助手协助核对渠道记录；找到原任务则继续查询，只有你在渠道确认没有创建任务后才能解锁。不要通过修改参数或清除记录绕过防重。

未完成任务继续使用原来的参数与连接配置；请保留原接入方式、地址和 Key，不要在等待期间切换。同一份暂停进度不要同时续做多次。

更新插件

Windows 与 Mac 的更新入口相同。`.tgz` 成品安装或仓库安装可使用此入口；本地目录链接安装会先停止直接替换，避免外部目录异常：

1. 等正在执行的视频任务完成。
2. 打开「插件 → 已安装插件 → MetaOnticVideo → 查看」，找到底部版本卡片，点击「检查更新」。
3. 发现更高的正式版本后，点击更新并再次确认，等待安装结果。
4. 按界面提示完成；只有明确提示需要重启时才重启，不需要先卸载。

在线更新需要能访问发布页及附件下载地址，使用对应版本 `.tgz` 成品包，不依赖 Git 拉取。下载失败时先保留现有插件和配置，核对网络后再处理，不要通过重置来解决下载问题。

插件详情底部的版本卡片提供“检查更新”。发现更高的正式版本后，点击更新并再次确认才安装。请等正在生成的视频结束后再更新，配置及已生成文件会保留，不需要先卸载。当前不提供版本选择或降级；在线安装使用对应版本成品包，当前版本不需要授权 esbuild 等构建工具的安装脚本。

5. 结果保存与常见问题

生成结果默认保存在当前会话工作目录下的 `metaontic-video/`，助手会展示视频，并在正文提供可点击的文件链接。脚本、分镜图、转换图片和恢复后交付的文件也会提供打开链接；图片还会在正文展示。超过15秒且分段生成时，先交付合并后的完整视频，再依次展示第一段、第二段及后续全部分段，合并成片的卡片排在最前；助手若把分段放在成片前面，插件会拦下这次展示并让他把成片挪到第一位重发，不会因此少给成片或分段。暂停时先交付已完成分段，合并失败时说明原因并保留分段。原始分段保留，脚本等资料通常在 `video-projects/<项目名>/`。如只返回上游链接，需在链接失效前保存。

多段本地合并需要 ffmpeg；缺少时可说“帮我检查并安装 ffmpeg”，确认安装后继续。高级设置中的合并、裁剪、等待时间通常保持默认即可；裁剪可能缩短成片，原始分段不删除。

需要重新配置时，可使用“重置插件”清除设置、密钥和登录状态；已生成的视频不会删除。

遇到的问题	如何处理
网站申请的 Key 不知道填哪里	选择 MetaOntic AI；共用时填“共用 Key”，不共用时分别填两个模型下方的 Key
图片提示缺少 Key	MetaOntic AI 关闭共用后必须填写图片模型 Key 并保存

遇到的问题	如何处理
提示无权调用模型或鉴权失败	检查对应模型的 Key 是否正确、是否开通模型权限，修改后保存
提示额度不足	到申请 Key 的平台核对对应账户额度并按平台指引充值
只生成第一段	先预览，再确认从暂停进度继续
超时、网络失败或提交结果不明	保留原任务与记录，先查询或核对，不直接重新生成
只给脚本文件就要求审核	要求先完整展示当前版脚本和镜头示意图，再重新审核
没有清晰度选项	清晰度在对话里的脚本阶段确认，支持 480P / 720P
组件启动失败或提示缺少依赖	使用正式安装包；曾配置本地开发时，检查是否仍指向已删除的本地源码目录
保存失败	输入会保留。配置冲突时核对最新设置后重试；连接中断时先重新打开设置核对是否已保存。后台取回 Key 时会等待写入完成；仅在软件明确提示需要重启时重启

保存配置时，插件会一并保存接口地址、模型和请求路径；未填写地址时使用 MetaOntic AI 默认地址。密钥仅用于所选连接，图片独立密钥不会替代视频密钥。

排查处理缓慢时，可查看 `~/.dsh/logs/metaontic-video/日期/requests.jsonl` 中的阶段耗时；设置了 `DSH_HOME` 时日志位于该目录下。插件会记录调用、生成等待、下载、合并和费用查询耗时；脚本设计、示意图与等待用户确认由助手标记开始和结束，其中脚本保存为 `script.md` 时插件会自动结束脚本设计这一段，并另记一条 `script_saved`（文件名、新建或改写、字数与脚本版本）。这些是流程耗时，并非模型纯推理时间；只有开始记录时表示该阶段未完整结束。写脚本、画分镜本身是助手和模型在做事，不属于插件耗时。日志不包含密钥或完整提示词。

设置页按接入与密钥、模型配置、画面与声音分组；画面比例和声音开关在宽窗口并排显示，窄窗口上下排列。服务端登录表单位于接入方式下方，重置入口位于设置页底部。

“更多设置”展开后分为画面、视频合并、查询与等待、本地工具四组；本地工具路径单独占一行，便于查看和填写完整路径。



续拍时，插件会持续带上最初已审核的角色参考图，帮助保持人物外形。未单独指定角色图时沿用最初参考图或首帧；暂停后继续也保留同一套图。若图片已按你的确认转换为非真人风格，后续沿用获审并成功使用的转换图，不重复处理。纯文字起步且没有图片时不会自动生成角色图；需要固定形象时，请在制作前提供并审核角色图。

纯文生视频需要固定人物外形时，助手会先询问是否生成角色参考图，并说明图片单独计费。选择直接文生视频就不生成图片；选择生成时先确认角色与画面风格，出图后先展示给你审核，满意后才用于视频首段与后续续拍。角色图费用与视频费用分开，生成角色图不代表同意开始制作视频。超时先查询已有图片任务，不重复出图；重画需再次确认费用。

提交新的纯文生视频时，插件会强制显示角色图选择。选择“直接文生视频”后再确认视频费用；选择“先生成角色图”则先出图、审核后再制作。未回答、关闭或取消不会自动开始；相同任务的明确选择会保留，已有参考图或续拍恢复不重复询问。

安装与更新故障排查（Windows / Mac）

现象	处理方法
`ERR_PNPM_GIT_RESOLVE_FAILED`	检查是否仍使用 Git 仓库地址。改用正式版 `.tgz` 附件链接或本地 `package` 目录。无需全局 Git 地址重写或配置 SSH。

现象	处理方法
`Lockfile passes supply-chain policies` 后又报错	这句表示供应链检查通过，实际失败原因在后面的错误。保留完整错误信息。
找不到本地插件或 `package.json`	确认已经解压，并选择包含 `package.json`、`index.js` 的 `package` 目录。
附件下载失败或超时	核对网络，换同一版本的另一个官方发布源；也可通过浏览器下载后本地安装。不要使用 Source code 代替成品包。
在线安装提示「不是有效的插件包」	在线安装的地址必须以 `.tgz` 结尾，请用 `.tgz` 附件下载链接；`MetaOnticVideo-Windows-数字版本号.zip`、解压后的 `package` 文件夹、发布页网址和 Source code 都不是在线地址，本地安装请另按 1.3 节填写文件路径。
在线或本地安装报 `ERR_PNPM_MISSING_TARBALL_INTEGRITY`	插件安装目录里早先安装留下的 `pnpm-lock.yaml` 缺少校验值，新版 pnpm 会拒绝安装，与插件包本身无关。插件已安装时会每 2 秒检查一次并自动清理这条记录，更新失败后也会等清理完成自动重试；仍不行或还没装上插件时，按本节「安装报 ERR_PNPM_MISSING_TARBALL_INTEGRITY」手动处理后再装。
提示找不到 `dsh`	使用 DeepSeek Harness 桌面端的插件安装入口。
组件启动失败、`ERR_MODULE_NOT_FOUND` 或 Schema 错误	保留完整错误及插件版本，确认安装的是成品包；曾配置本地开发加载路径时需核对是否仍覆盖正式安装目录。不要先清除 Key 或任务记录。
`ERR_PNPM_IGNORED_BUILDS` / esbuild 脚本提示	核对是否误用了源码或旧 Git 安装来源，改用成品包；不要关闭供应链检查或直接批准未知构建脚本。
更新失败	保留现有插件、Key 及已生成文件，记录目标版本与完整错误，检查附件可访问性；不要先卸载或重置。

排查时可以提供系统类型、插件版本、安装来源和完整错误文字；不要发送 API Key、登录验证码或包含密钥的配置文件。

安装报 `ERR_PNPM_MISSING_TARBALL_INTEGRITY`

报错原文形如：

```
Cannot install package "metaontic-video@https://…/metaontic-video-版本号.tgz":
its lockfile entry has no "integrity" field, so pnpm cannot verify the downloaded
tarball.
```

这条报错与插件包本身无关。DeepSeek Harness 的插件安装目录里保留了早先安装生成的 `pnpm-lock.yaml`，其中这条 `.tgz` 地址的记录没有写入校验值；当前 pnpm 为防安装被篡改会直接拒绝，因此在删掉这条记录之前，换成在线地址、换成本地 `.tgz` 路径或重装旧版本都会同样失败。

处理步骤：

1. 完全退出 DeepSeek Harness。
2. 删除插件安装目录里的 `pnpm-lock.yaml`。Windows 在文件管理器地址栏输入 `C:\Users\你的用户名\.dsh\profiles\desktop` 打开该目录；Mac 打开 `~/.dsh/profiles/desktop`。设置了 `DSH_HOME` 时改到该目录下的 `profiles/desktop`。
3. 重新打开 DeepSeek Harness，按 1.2 节或 1.3 节再安装一次。安装器会重新生成 `pnpm-lock.yaml` 并写入校验值。

插件运行期间还会每 2 秒检查一次这个文件：一旦发现里面留着本插件缺少校验值的记录（包括旧版本留下的、已经没人引用的记录），会自动删掉本插件这条并把原文件另存为同目录下的 `pnpm-lock.yaml.metaontic-backup-时间`，不需要你手动处理。更新时出现这条报错，插件会等清理完成自动重试一次，多数情况下接着就能继续；只有当插件本身还没装上、后台还没有这个清理动作时，才需要按上面的步骤手动删除。

只删除 `pnpm-lock.yaml`，不要删除同目录的 `package.json`。该文件只记录依赖版本与校验值，删除后重新安装即可，已保存的插件设置不受影响。

本地目录安装点击更新后目录变空

如果报错仍指向原先解压的 `package` 目录，请停止重复点击更新。用完整安装包恢复原路径下的整个目录，不只补一个空locale文件夹，也不要重置Key或删除任务记录。

插件更新前会读取安装来源：本地目录链接、来源无法核实或元信息异常时停止，不调用安装器。它不会自动修复此前已变空的目录，也不会静默迁移旧安装。新装推荐使用 `.tgz` 成品文件；旧目录安装迁移需要先核对实际加载路径和配置保留情况，不能只把目录恢复后再次点击旧版“立即更新”，也不要直接卸载。

自动检查覆盖成品安装及成品升级，不等于旧本地目录迁移或桌面界面的实机验收。遇到拦截请保留错误与安装来源，进一步排查，不绕过保护。