

One Person Lab

OPL App 白皮书

让复杂知识工作拥有一个本地优先、结果可复查、可以持续推进的专业工作台

一个专业 AI 产品的好用，体现在用户能从真实目标出发，得到有来路的成果，知道何时继续、何时判断，并在工作环境变化后仍能接着做。

目的优先 / 成果有来路 / 状态指向行动 / 工作位置可连续

2026-07-13

Public whitepaper

目录

1	OPL App 把一次问答变成持续工作	2
2	设计一：从工作目的开始	3
3	设计二：成果必须带着来路	3
4	设计三：状态必须指向行动	3
5	设计四：工作位置变化，工作语言保持连续	4
6	一个完整场景：从本机研究到团队审阅	4
7	四层产品，一套连续体验	4
8	一个产品，一套连续体验	5
9	清楚分工如何保持体验一致	5
10	本文边界	5
11	结语	5

发布日期：2026-07-13

最近修订：2026-08-16

适用对象：希望理解 OPL App 为什么这样设计，以及这些设计怎样帮助自己完成研究、基金、演示和书稿等复杂知识工作的用户、合作者、早期采用者和技术决策者。

核心判断：一个专业 AI 产品的好用，体现在用户能从真实目标出发，得到有来路的成果，知道何时继续、何时判断，并在工作环境变化后仍能接着做。

1 OPL App 把一次问答变成持续工作

AI 已经很擅长回答问题、生成文字和修改文件。真正困难的是把这些能力放进一项会持续数天、数周甚至数月的正式工作。

一篇论文会经历材料整理、研究问题收敛、分析、作图、写作、审阅和修订；一份基金申请会反复调整创新点、技术路线和论证；一套演示要让数据、故事线、视觉和导出结果彼此一致；一本书更需要跨章节保持结构、风格和事实连续。

在这些工作里，用户需要一套贯穿全程的答案：

- 我现在应该从哪里开始？
- 当前成果由哪些材料、分析和审阅支持？
- 工作推进到了哪里，下一步可以自动继续还是需要我决定？
- 离开几天后回来，是否还能接住之前的上下文？
- 换到浏览器、服务器或云端时，是否仍在同一条工作线上？

OPL App 是 One Person Lab 面向用户的专业工作台：让目标、对话、材料、阶段、产物、审阅、回执和下一步形成一条可以继续的工作线。



图 1: OPL App 从选择工作目的到持续推进、审阅与交付的用户旅程。

2 设计一：从工作目的开始

用户打开 OPL App，首先表达“我要完成什么”，系统随后选择合适的专业能力与工作环境。内部工具和运行方式留在系统背后。

产品提供四个高频工作目的：

用户看到的入口	用户想完成的事	专业 Agent
科研	从问题、材料和数据走向分析、证据与论文交付	MAS
基金	形成有说服力的选题、论证、申请书与修订	MAG
演示	把内容、数据和故事线变成可审阅的视觉交付	RCA
写书	组织大纲、章节、风格、修订与出版交接	OBF

这四个目的入口与 OPL Meta Agent (OMA) 的“元智能体”入口，共同构成五个默认起点。它们不是能力上限：App 可以发现更多兼容的专业 Agent，用户也可以决定哪些快捷入口出现在首页。

这样的设计有两个好处：新用户可以直接从工作目的开始，熟练用户可以自由扩展快捷入口。产品可以不断增加专业能力，首屏始终回答一个问题：你今天要完成什么？

3 设计二：成果必须带着来路

正式知识工作的结果需要同时包含成果与来路。一个结论、一张图、一页演示或一个章节，能够回到它所依赖的材料、过程和审阅，才值得被继续使用。

OPL App 把工作表达为一条可以复查的链：

目标 -> 材料与资源 -> 阶段推进 -> 产物 -> 审阅与检查 -> 回执 -> 下一步

“结果有来路”意味着优先呈现与判断有关的信息：当前产物在哪里，哪些内容已经检查，哪里仍有质量债，下一步是继续修订、补充材料、等待资源，还是由人作决定。路径、标识、原始回执和运行细节在排查问题时进入诊断层。

这条原则让用户既能判断最终文件是否可信，也能在首屏聚焦真正要做的事；完整内部证据则按需展开。

OPL App 负责把已有事实翻译成用户能采取行动的状态，而不是自己复制一套状态。运行进展来自 OPL Base，专业能力说明自己的可用性，研究、基金、视觉和书稿质量来自对应专业 Agent 与人类负责人。界面说明“发生了什么”和“下一步是什么”，就绪结论仍由真正负责的人与 Agent 给出。

4 设计三：状态必须指向行动

复杂产品很容易把设置页变成内部结构目录：运行环境、集成、技能、缓存、路径和诊断信息各占一栏。这样的界面看似完整，却把理解系统的成本转嫁给用户。

OPL App 采用相反的组织方式。设置首先围绕用户要做的决定：

- 当前环境是否可以开始工作，需要我做什么？
- App、Base 或专业能力是否需要安装、更新或修复？
- 哪些专业 Agent 已可用，哪些显示在首页？
- 浏览器访问、远端资源或账号能力是否需要配置？
- 哪些数据由本机持有，清理或迁移会影响什么？

默认层只显示目的、状态、影响和主动作。标识、路径、缓存和底层运行记录等信息折叠到“高级诊断”，供需要排障的人按需展开。

这种信息架构建立了一条稳定规则：**正常状态保持安静，异常状态说清影响，所有状态都给出合法下一步**。普通用户也能知道该继续工作、等待后台完成、批准资源，还是进入修复。

5 设计四：工作位置变化，工作语言保持连续

OPL App 是本地优先的。用户可以先在自己的设备上选择工作目录，让敏感材料、早期草稿和个人项目留在自己的环境里。工作位置变化时，用户面对的是几种并列载体，而不是必须逐级迁移的路径：

- 同一套 OPL 产品语言
 - 本机工作台：保留个人控制与敏感材料
 - 浏览器工作台：从其他设备继续同一项工作
 - 在线 Workspace：按需获得协作与远端资源

手机可以通过浏览器访问同一工作台；消息入口只负责传递内容，不创建第二套任务与会话历史。如果系统无法确认两个入口是否对应同一项工作，就保留原入口，不隐藏或误合并任务。

项目、任务、阶段、产物、审阅、阻塞、证据和下一步组成一套连续的工作语言。不同载体不必在每个细节上完全相同，但都应让用户理解自己正在做什么、成果在哪里、接下来由谁负责。

6 一个完整场景：从本机研究到团队审阅

设想一位研究者准备一篇需要重新分析数据的论文。

从目标开始。她在本机打开 OPL App，选择自己的项目目录和“科研”入口，用普通语言说明研究问题、数据位置和希望形成的交付物。App 将任务交给 MAS，由 OPL 与 MAS 协同组织所需模型、方法和工具。

让过程可见。MAS 围绕问题、数据、分析、图表、论断和稿件推进。App 持续呈现当前阶段、最新产物、需要补充的材料和等待她决定的问题。她可以打开结果，也可以回到支持结果的材料和审阅意见。

在正确的时刻让人判断。如果分析已经形成可消费的增量，但某张图仍需重画，工作可以带着清楚标注的质量债继续；如果要声称结果达到投稿标准，则必须回到 MAS 的质量门和负责人确认。App 准确区分“有进展”与“已就绪”。

延展工作位置。当本机资源不足时，她可以批准使用服务器或云端资源；需要合作者参与时，可以在浏览器或 Workspace 中继续。项目仍以相同方式呈现任务、产物、审阅和下一步，而资源、账号和组织策略留在对应管理面。

把下一轮接住。一周后回来，她直接看到当前稿件、最近一次审阅、未关闭的问题和可以继续的动作。产品为专业判断准备材料、上下文和清楚的交接点，最终学术判断仍由她作出。

这个场景也适用于基金、演示和书稿：专业内容不同，可信工作的结构相同。

7 四层产品，一套连续体验

OPL 把运行基础、用户工作台、专业能力和在线服务组织成四层产品。用户看到的是一套连续体验，每一层都由明确的产品负责：

对象	用户应如何理解
OPL Base	后台运行基础，负责让任务、阶段、专业能力和证据可靠工作
OPL App	用户看到和操作的工作台，负责入口、交互、状态翻译与产品体验
OPL Packages	可安装和管理的专业智能体、能力与工作流，决定“能完成哪类专业工作”
OPL Cloud	在线工作空间、资源连接、协作与服务治理，让工作可以跨越本机继续推进

具体执行工具、工作流配置、辅助工具和底层集成都归入这四层产品。用户数据与产物拥有独立的存储、保留和清理边界；软件与在线服务各自按照适合的交付方式持续更新。

这个模型让产品边界更容易理解：Base 负责运行，App 负责工作体验，Packages 负责专业能力，Cloud 负责在线工作与治理。界面汇总它们的真实状态，不要求用户理解多套运行器、状态来源与恢复机制。

8 一个产品，一套连续体验

OPL App 用一份产品定义统筹导航、页面、状态、操作、首次使用引导和能力管理。用户始终面对一套一致的工作语言，不需要理解底层能力来自哪里，也不需要多个控制面之间判断哪个状态才是真的。

专业能力会根据当前任务动态进入工作台。App 把它们组织成可理解的页面、操作和进度，同时保持任务、文件、运行状态和专业判断各自的真实来源。新增或升级能力时，用户已有的项目和工作方式继续有效；只有经过兼容性、安装、安全和发布验证的变化才会进入正式产品。

界面设计可以持续改进，但每次正式发布只呈现一套完整验证的产品体验。产品升级不会复制用户工作、能力安装状态或操作规则，也不会让显示结果取代运行事实和专业结论。

9 清楚分工如何保持体验一致

OPL App 通过清楚分工建立专业性：

OPL 产品体验

- = OPL Base：阶段运行、恢复、证据与交接
- + OPL Packages：专业能力、领域判断、审阅与交付
- + OPL App：入口、交互、状态与下一步
- + OPL Cloud：在线 Workspace、托管资源与组织治理

App 负责入口、交互、状态翻译和下一步；Framework 负责运行事实；MAS、MAG、RCA、OBF 或 OMA 负责领域质量判断；云端管理面负责账号、资源和组织策略。App 把这些事实组织成普通用户可以理解和操作的工作台。

清楚边界直接建立用户信任：跨界面的状态保持一致，专业结论按权威路径给出，资源和权限变化始终经过用户知情与授权。

10 本文边界

本文聚焦 OPL App 的设计理念和用户价值。安装说明、设置手册、功能清单、版本与可用状态由正式产品页面提供；具体专业成果是否合格，由对应专业 Agent 和人类负责人判断。

11 结语

OPL App 希望让一个人拥有类似专业团队的工作界面，并让复杂系统保持易于理解和持续维护。

它让用户从真实目的开始，让成果带着来路，让状态指向下一步，让设置围绕用户决策组织；它让工作从本机自然延伸到浏览器和 Cloud，同时清楚区分设计目标与发布事实；它把专业能力放进可扩展 Packages，让 Base、App、Packages 和 Cloud 各自承担清楚而互补的责任。

产品把复杂性放在正确的位置：用户看到目标、成果和决定，专业 Agent 保留判断权，运行系统保留事实，诊断细节在需要时出现。OPL App 的一致性与可靠性，应由这些可观察、可复查的结果持续证明。