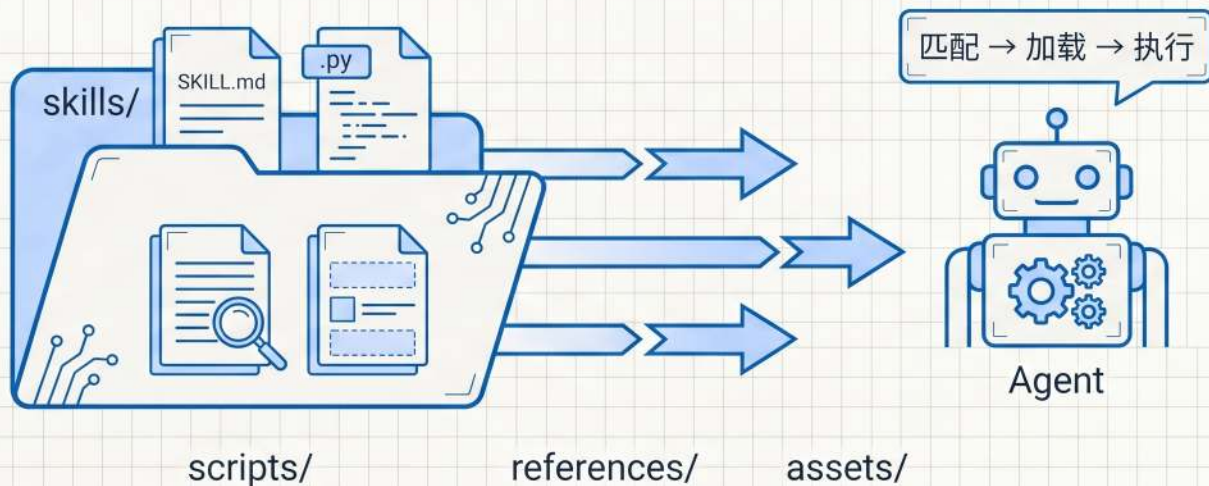


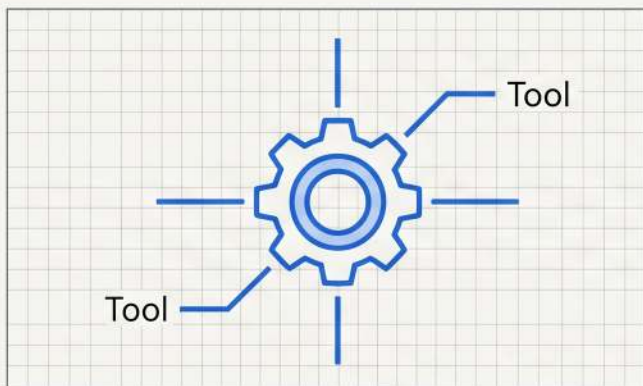
# Deep Agents 实战

## 第 9 讲: Skills (上) — 从概念到上手



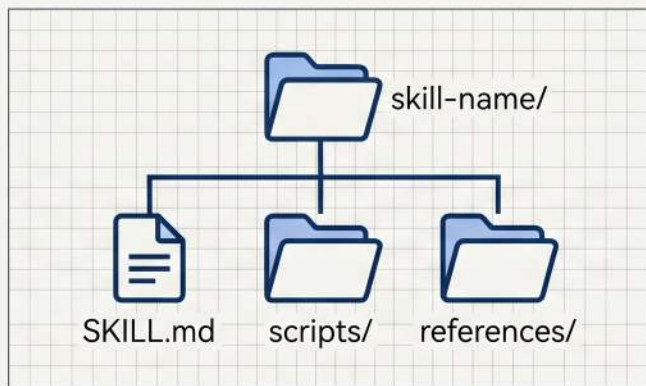
# Skills 是什么?

Tool 是"做一件事", Skill 是"按一套流程做一类事"



- 单一原子操作 (搜索、读文件、发邮件)

VS



- 多步骤工作流 + 领域知识 (SKILL.md + 参考文档 + 模板)
- 核心文件: SKILL.md (元数据 + 指令)
- 遵循开放的 Agent Skills 规范 (agentskills.io)

30+ 主流 AI 开发工具已采用 Agent Skills 规范

# SKILL.md 结构

YAML Frontmatter (元数据) + Markdown 正文 (指令)

Field	Required	Description
name	Yes	小写字母+连字符, 1-64 字符, 必须匹配目录名
description	Yes	功能描述+触发条件, 最大 1024 字符
license	No	许可证名称
compatibility	No	环境要求 (网络、CLI) , 最大 500 字符
metadata	No	任意键值对 (author, version, entrypoint)
allowed-tools	No	空格分隔的预授权工具列表 (实验性)

```
---
name: langgraph-docs
description: Use this skill for requests
  related to LangGraph in order to fetch
  relevant documentation to provide
  accurate, up-to-date guidance.
license: MIT
compatibility: Requires internet access
metadata:
  author: langchain
  version: "1.0"
allowed-tools: fetch_url
---

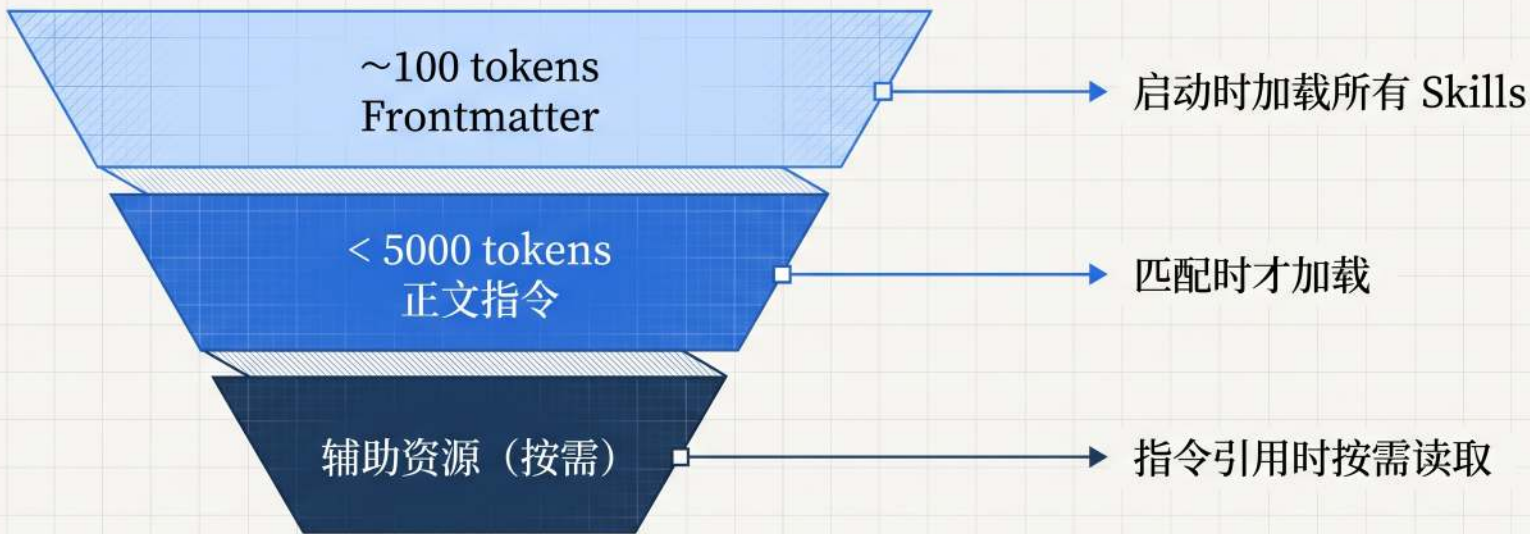
# langgraph-docs

Instructions for the agent go here.
```

SKILL.md 文件上限 10 MB, 超出的文件在加载时跳过

# Progressive Disclosure

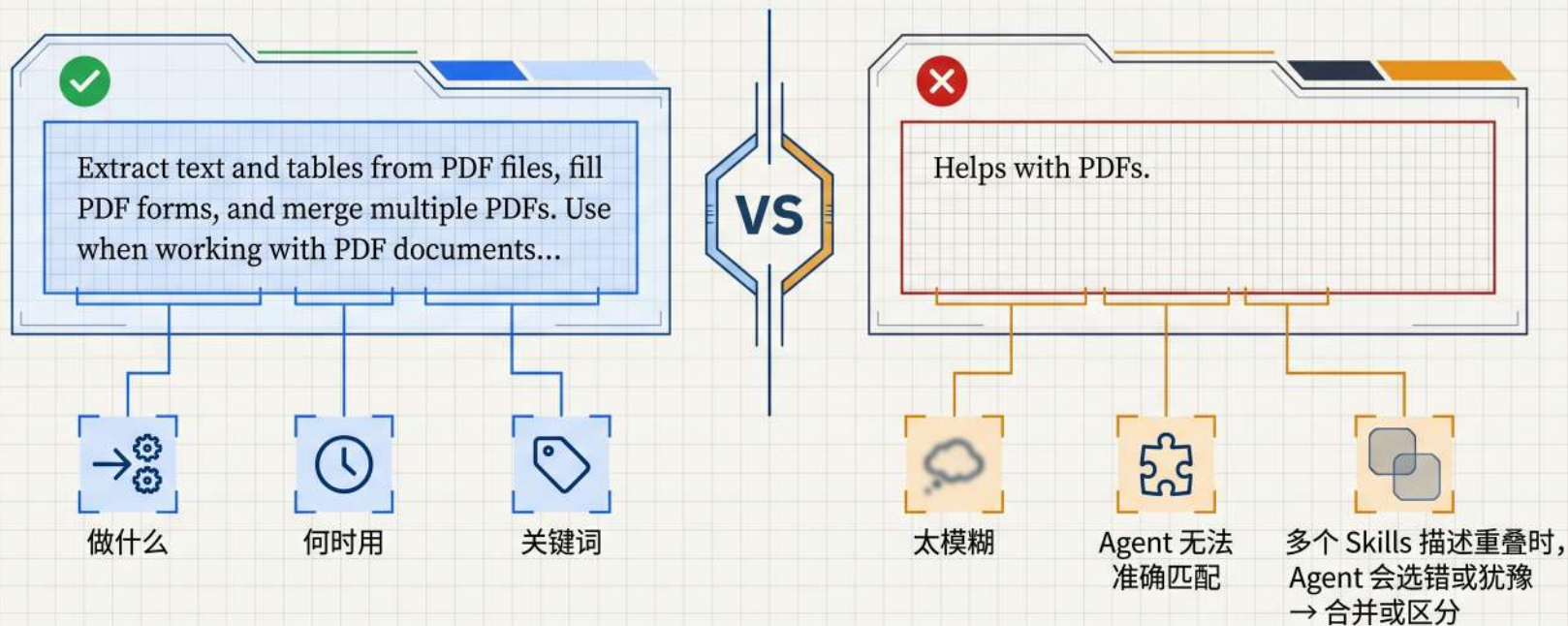
三层加载 — 越往下越详细，加载越晚



- 好处：节省 token、精准匹配、可扩展

# 写好 description 的关键

description 是 Agent 在启动时唯一能看到的 Skill 信息



# 在 Deep Agents 中使用 Skills

通过 `create_deep_agent` 的 `skills` 参数指定 Skill 目录

```
create_deep_agent(  
    model="gpt-4",  
    backend="local",  
    skills=["/skills/"]  
)
```

/skills/

langgraph-docs/

code-review/

- `skills=["/skills/"]` 指定包含 Skill 目录的父路径
- 路径使用正斜杠，相对于后端根目录
- Agent 自动扫描并发现所有 Skills
- 启动时加载 Frontmatter，匹配时读取完整指令

# 三种后端的 Skills 加载

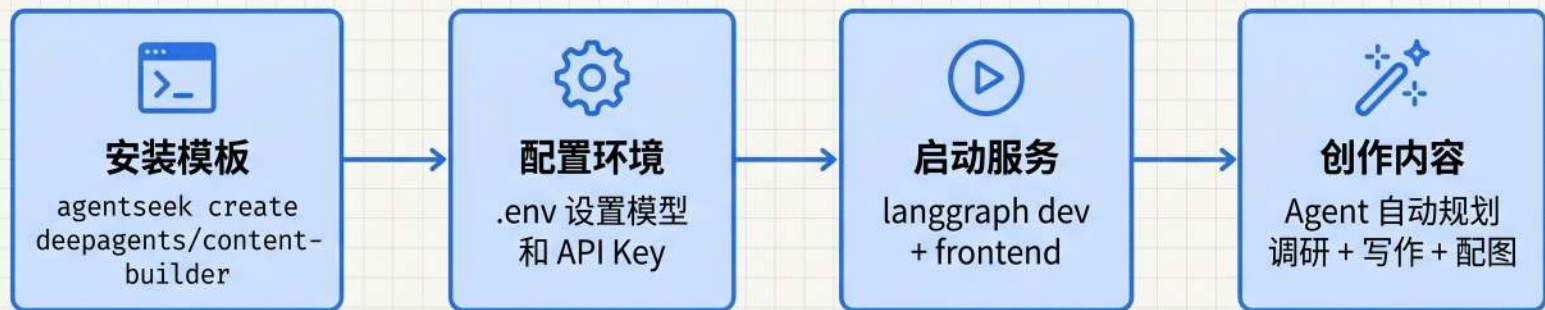
StateBackend / FilesystemBackend / StoreBackend



# 实战：content-builder 模板

## AgentSeek 模板应用 — 内容创作 Agent

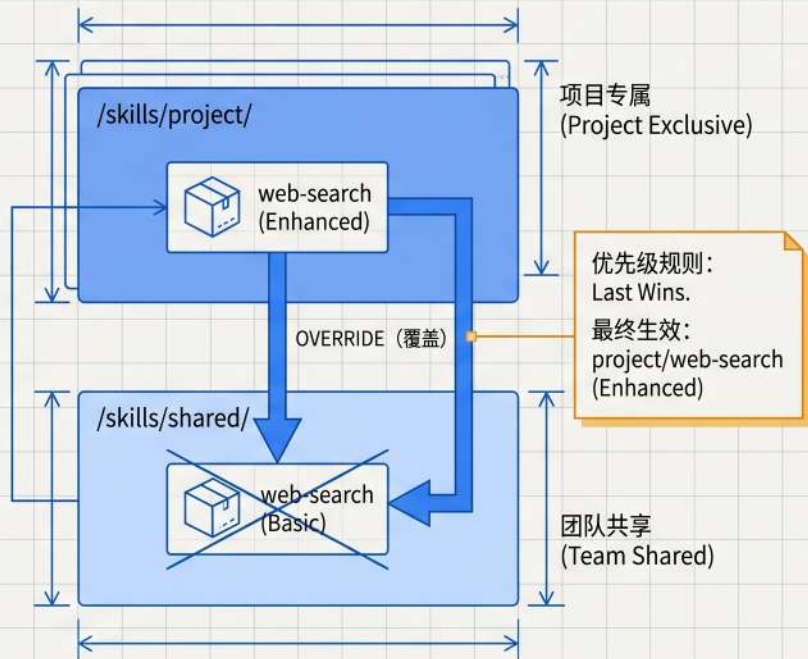
- 内置 Skills: blog-post (博客写作流程) + social-media (社交媒体内容)
- 品牌记忆: AGENTS.md 定义品牌声音与写作风格
- 子 Agent: researcher (Tavily 搜索) 协助素材调研
- 图片生成: Gemini 生成封面和社交配图
- 安装: `agentseek create deepagents/content-builder`



# 多源 Skills 与优先级

后面的覆盖前面的 (last wins)

- skills 参数接受多个路径：  
["/skills/shared/", "/skills/project/"]
- 同名 Skill 出现在多个目录时，后面的优先
- 分层管理：团队共享（基础）+ 项目专属（覆盖）
- 示例：两个目录都有 "web-search" Skill, project/ 下的生效



# 本讲回顾

📦 Skills = 目录 + SKILL.md + 可选资源，遵循开放的 Agent Skills 规范

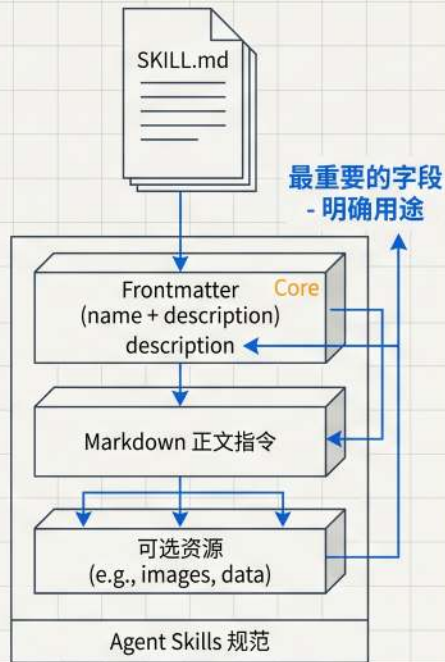
📄 SKILL.md = Frontmatter (name + description) + Markdown 正文指令

📖 Progressive Disclosure: 三层加载，节省 token、精准匹配

📢 description 是最重要的字段 — 要具体说明做什么和何时使用

📦 三种后端: StateBackend / FilesystemBackend / StoreBackend

➔ 下一讲: Skills 高级用法 — 解释器技能、子 Agent、权限控制



下一讲预告



高级用法: 解释器技能、子 Agent、权限控制