

LangChain DeepAgents

Batteries included “Deep Research” framework

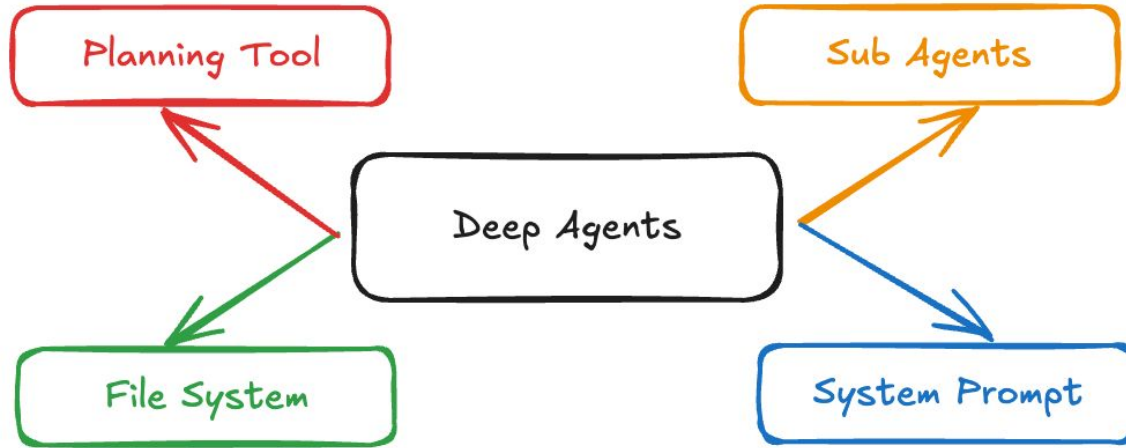
Collier King
9/10/25

Problem of “Shallow” Agents

- Simplest Agent - Using an LLM to call tools in a loop
- Doesn't cut it for complicated use-cases due to following pain points:
 - Sustained context across multi-step problems
 - Long-term planning and task execution
 - Complex workflows needing deep analysis
 - Delegation and parallel task management

Rise of “Deep” Agents

- “Deep Research”, “Claude Code”
- Effective due to implementation of **4 design components**:



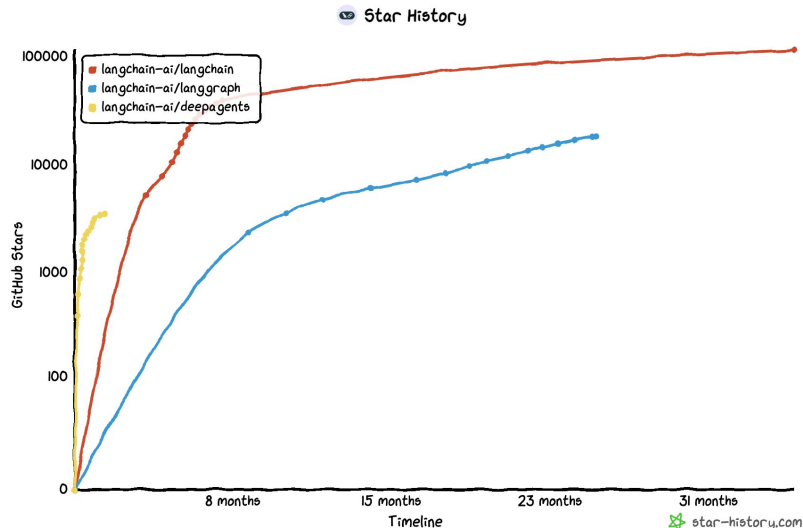
- **deepagents** implements these in a general purpose way.

pip install deepagents

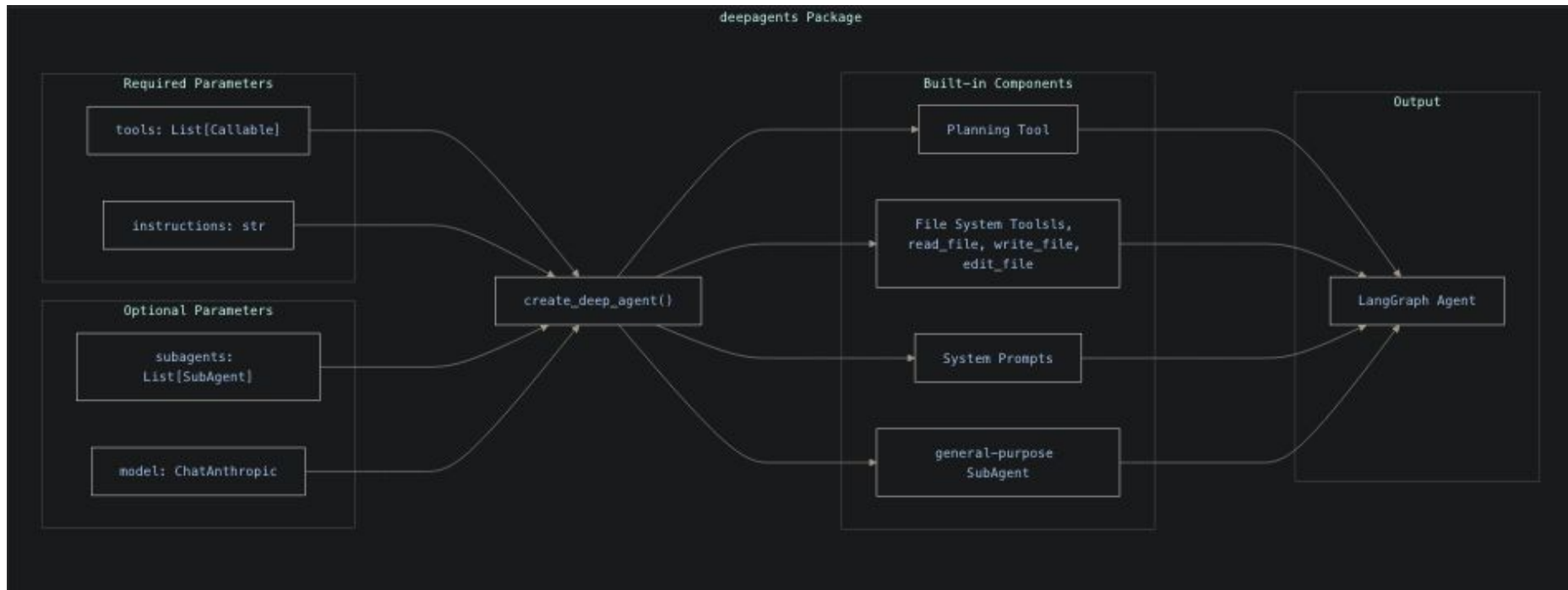
- Primarily inspired by **Claude Code**
- Creates a LangGraph graph under the hood
- Similar to “**create_react_agent**” wrapper in LangGraph
 - LLM decides whether to stop or take an action, receives feedback, continues in iterative loop

```
# Create the agent
agent = create_deep_agent(
    [internet_search],
    research_instructions,
)

# Invoke the agent
result = agent.invoke({"messages
```

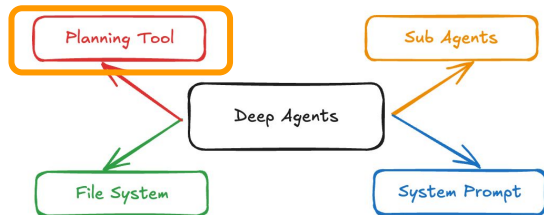


DeepAgents Design



Component - Planning Tool

- **deepagents** comes with a **built-in planning tool**.
 - Based on Claude Code's **TodoWrite** tool
- Enables Agent to “come up with a plan”
 - Create and track a structured plan in the form of a “to-do” list
 - Stay on track via persisted context



Component - SubAgents

- Based on Claude Code
- Specialized “subordinate” agents with their own prompts and tools.
- Enables "context quarantine" (avoids “context pollution” of the main agent)
- Also “general-purpose” subagent
 - Same instructions as the main agent and all the tools that it has access to.

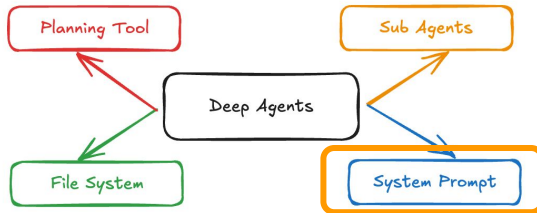


```
from deepagents import create_deep_agent

critique_sub_agent = {
    "name": "critique-agent",
    "description": "Critique the final report",
    "prompt": "You are a tough editor.",
    "model_settings": {
        "model": "anthropic:claude-3-5-haiku-20241022",
        "temperature": 0,
        "max_tokens": 8192
    }
}

agent = create_deep_agent(
    tools=[internet_search],
    instructions="You are an expert researcher...",
    model="claude-sonnet-4-20250514", # default for main
    subagents=[critique_sub_agent],
)
```

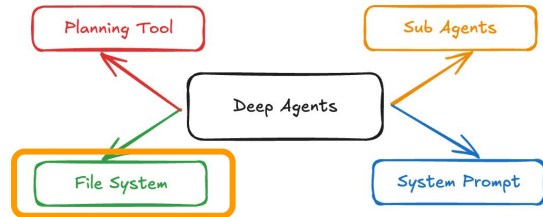
Component - System Prompt



- Comprehensive set of instructions to **deepagents** has a **built in system prompt!**
 - Based on attempts to replicate Claude Code's system prompt but "more general purpose"
- Contains detailed instructions how to use the built-in planning tool, file system tools, and sub agents...guides the LLM's core behavior.
- User can customize only part of the system prompt via **instructions=....** param

*"Without this default system prompt - the agent **would not be nearly as successful** at going as it is. The importance of prompting for creating a "deep" agent **cannot be understated**." - DeepAgents README.md*

Component - Virtual File System



- Simulated, in-memory file system using LangGraph “State” object
 - Represented as a Python dict
- Can have “files” that user passes in, or agents create/edit
- Can run many agents on the same machine...won’t edit the same files.

```
agent = create_deep_agent(...)

result = agent.invoke({
    "messages": ...,
    # Pass in files to the agent using this key
    # "files": {"foo.txt": "foo", ...}
})

# Access any files afterwards like this
result["files"]
```

Built in Tools

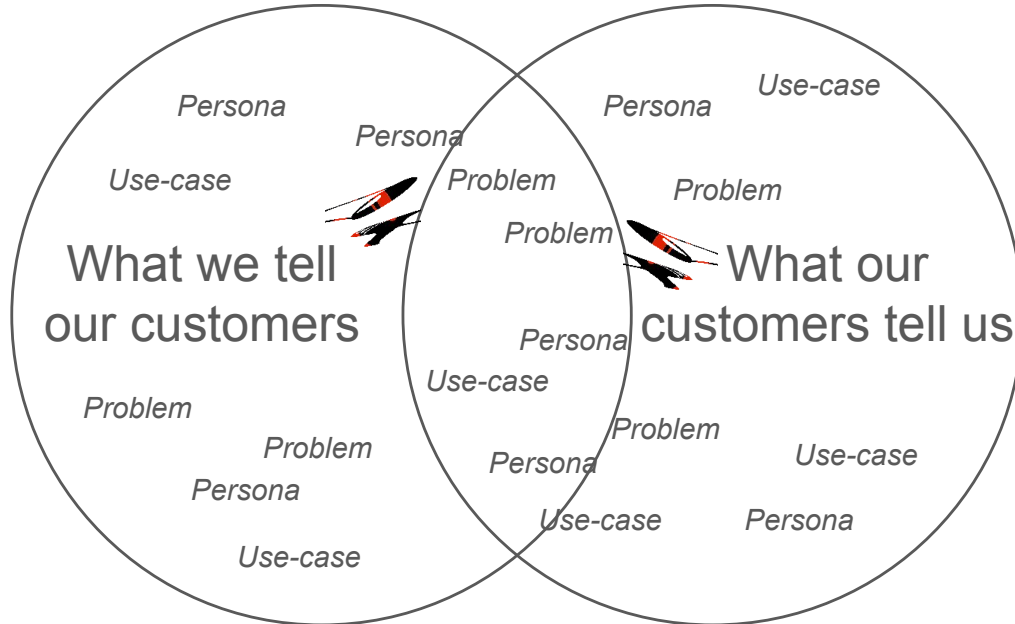
- deepagents comes with 5 built-in file system tools:
 - **write_todos**: Tool for writing todos
 - **write_file**: Tool for writing to a file in the virtual filesystem
 - **read_file**: Tool for reading from a file in the virtual filesystem
 - **ls**: Tool for listing files in the virtual filesystem
 - **edit_file**: Tool for editing a file in the virtual filesystem

******These can be disabled via the builtin_tools parameter.

Demo - “Product Manager” Deep Research Agent

- Who are our customers
- What are their use cases?
- Are we solving their problems?

<https://github.com/CollierKing/langchain-deepagents-examples>



Questions?



Resources

<https://github.com/langchain-ai/deepagents>

<https://deepwiki.com/langchain-ai/deepagents>

What are Deep Agents?

<https://www.youtube.com/watch?v=433SmtTc0TA>

Implementing deepagents: a technical walkthrough

<https://www.youtube.com/watch?v=TTMYJAw5tiA>

Using `deepagents` to Build Deep Research (Python)

<https://www.youtube.com/watch?v=geTtqyFnyHA>

Deep Agents UI

https://www.youtube.com/watch?v=0CE_BhdnZZI

Coln's Writeups

<https://colinmcnamara.com/blog/deep-agents-part-1-beyond-shallow>