



Durable OpenAI Agents SDK with Temporal



JOIN OUR SLACK CHANNEL

Feel free to ask questions or go there
for helpful links!

<https://t.mp/nov-18-slack-channel>



Agenda

- ✓ OpenAI Agents SDK
- ✓ Temporal refresher
- ✓ Durable OpenAI Agents (Temporal + Agents SDK)
- ✓ Orchestrating Agents

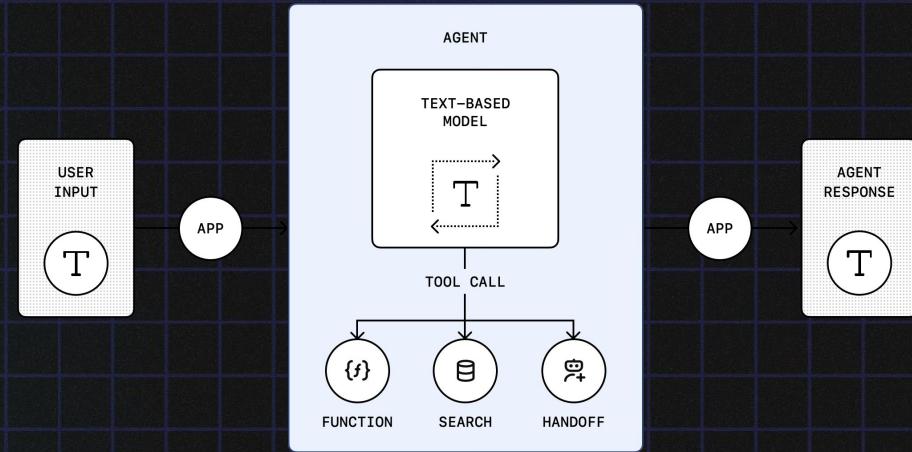




Introduction to the OpenAI Agents SDK

What's an agent?

An agent is an AI application consisting of a **model** equipped with **instructions** that guide its behavior, access to **tools** that extend its capabilities, encapsulated in a **runtime** with a dynamic lifecycle.



OpenAI Agents SDK

Based on what OpenAI learned building agents with customers.

1. Enough features to be worth using, but few enough primitives to make it quick to learn.
2. Works great out of the box, but you can customize exactly what happens.

```
1 # Python  
2 pip install openai-agents  
3  
4 # TypeScript  
5 npm install @openai/agents
```



OpenAI Agents SDK

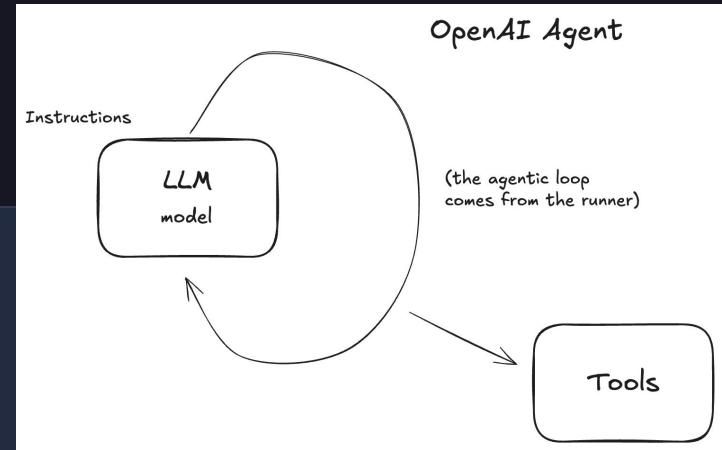
Available in both Python and TypeScript

- Works with most LLMs (even non-OpenAI)
- Handoffs
- Guardrails
- Streaming*
- Tools & MCP support
- Built-in Traces support
- Flexible session management
- Voice agent support*

```
1 from agents import Agent, Runner
2
3 async def main():
4     agent = Agent(
5         name="Assistant",
6         instructions="You only respond in haikus.",
7     )
8
9     result = await Runner.run(
10         agent,
11         "Write about recursion in programming."
12     )
13     print(result.final_output)
14     # Code within the code,
15     # Functions calling themselves,
16     # Infinite loop's dance
```



```
1 from agents import Agent
2
3 agent = Agent(
4     name="Triage Agent",
5     model="gpt-4o-mini",
6     instructions=TRIAGE_AGENT_INSTRUCTIONS,
7     handoffs=[weather_agent, local_biz_agent],
8     tools=[WebSearchTool],
9 )
10
11 result = Runner.run_sync
12     (agent, "How late is Costco open?")
13
```



NOTEBOOK SETUP

Follow along and play with our samples by accessing
our Jupyter Notebooks here:

t.mp/temporal-openai-agents-sdk



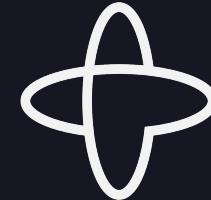


Temporal Overview

Introducing Temporal

- Technology and open source project that delivers resilience for distributed systems in a novel way.
- Supports a programming model that allows developers to code the **happy path**, while the platform provides services that compensate for a wide range of distributed system failures.
- Platform comes in the form of a service + SDKs

SDK is available for Go, Java, Python, PHP, Typescript, .Net, Ruby



Temporal

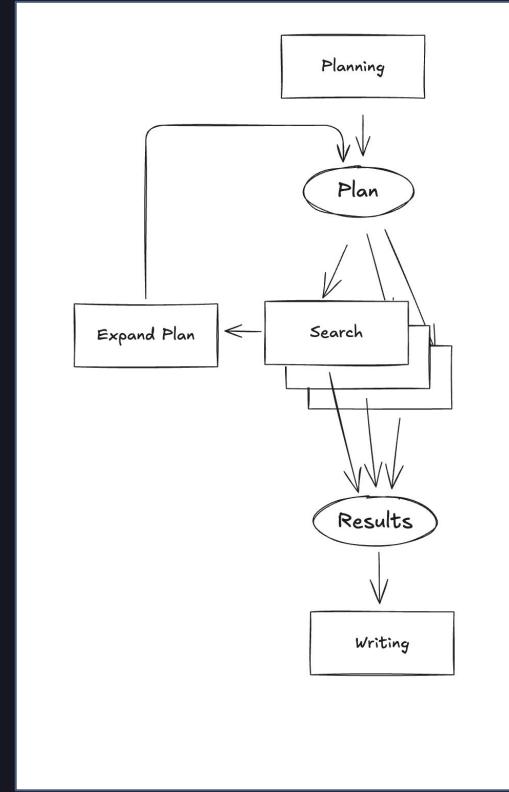
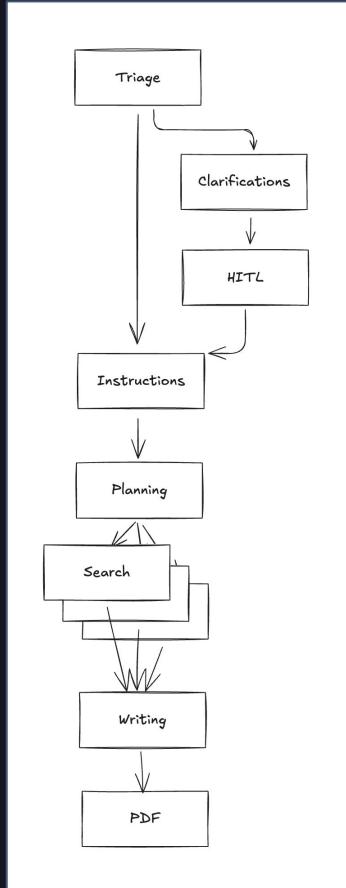
Note: The OpenAI Agents SDK + Temporal solution is currently only available for Python. Typescript coming in the future.

The MIT License

Copyright (c) 2020 Temporal Technologies Inc. All rights reserved.

Copyright (c) 2020 Uber Technologies, Inc.

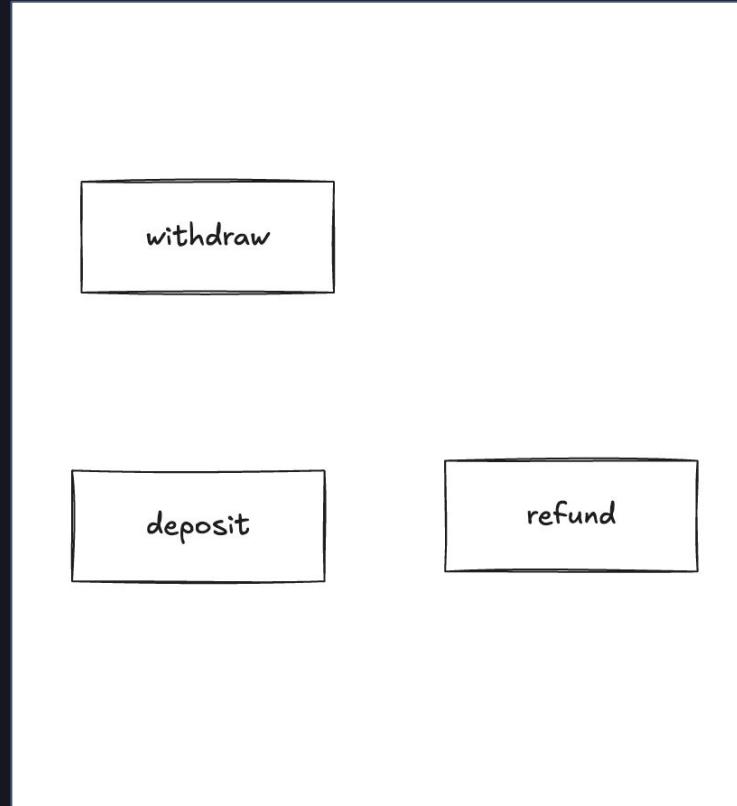
Temporal Workflow



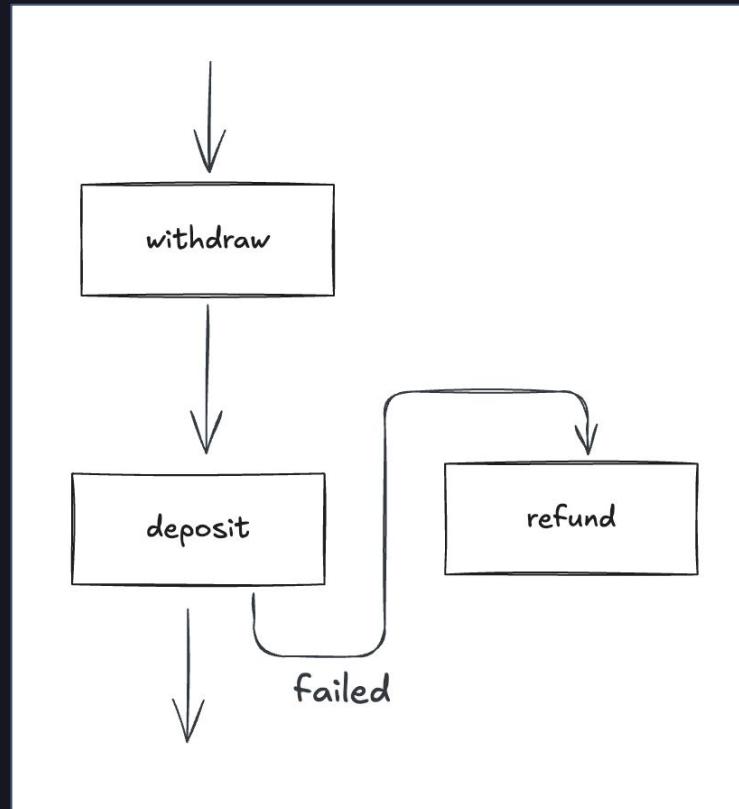
Temporal Activity



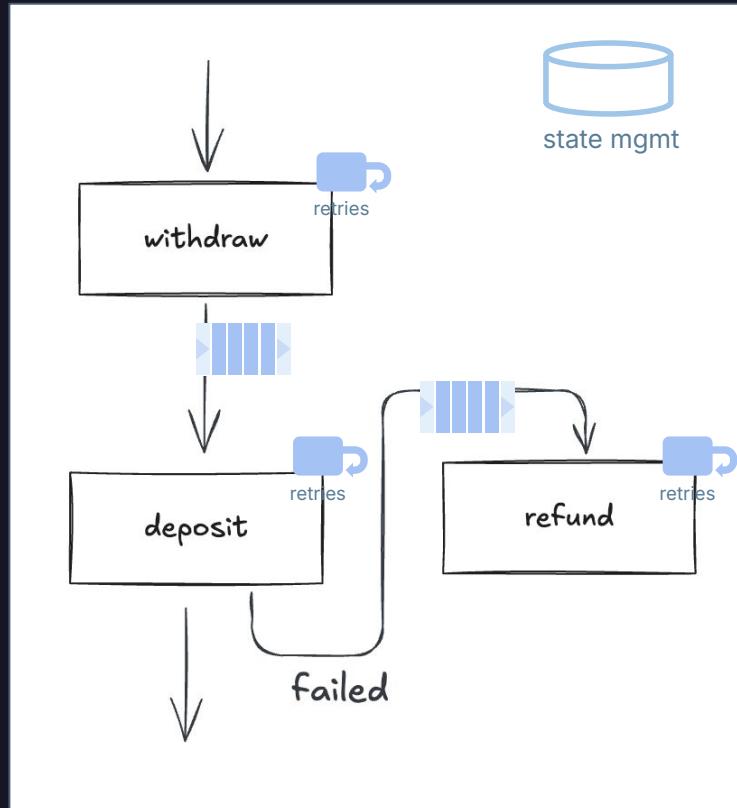
Temporal Activities



Temporal Workflows

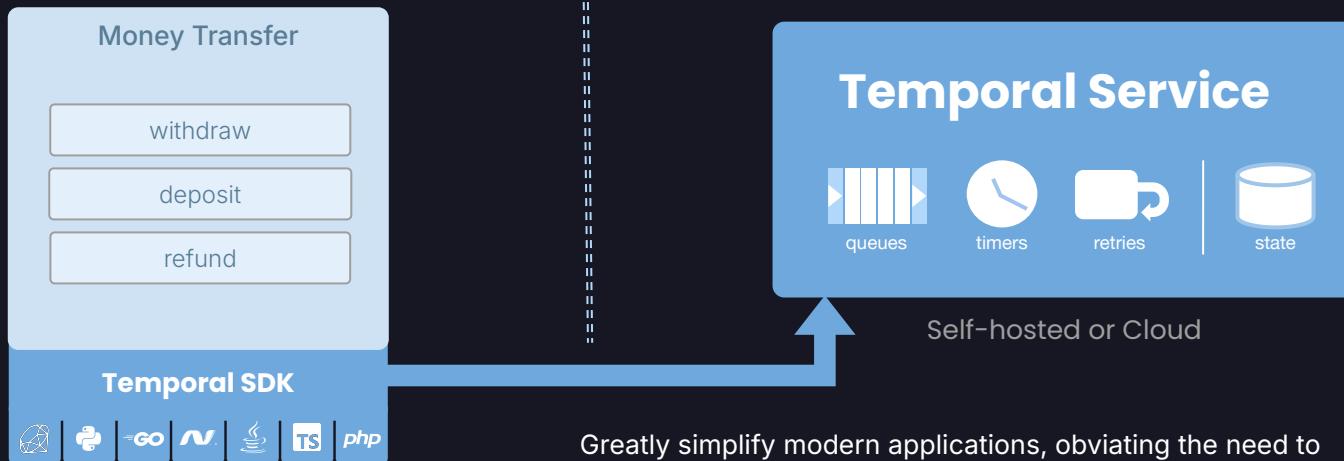


Temporal Workflows and Activities = ⚡



ENTER DURABLE EXECUTION

Define a workflow and then
maintain state and coordinate execution



Greatly simplify modern applications, obviating the need to
create queues, manage timers, publish and consume events,
implement retries and rollbacks, checkpoint state, and more.





Durable OpenAI Agents SDK

Putting the pieces together

OpenAI Agents SDK with Temporal

Agents

Tools

Guardrails

Tracing

Agent Framework: OpenAI Agents SDK

Durable Execution: Temporal

AI Inference: OpenAI Models

Temporal Activity



LET'S BUILD THE ACTIVITY!

Setting it up for tool use is easy

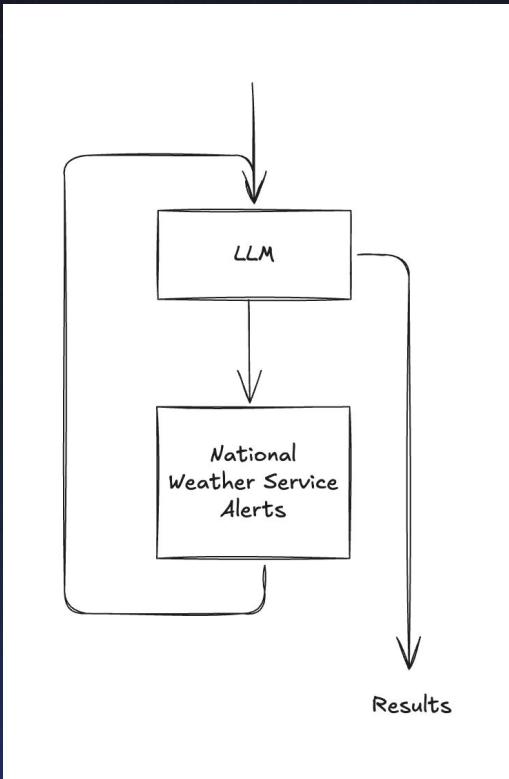


1. Name the function appropriately or use a decorator (`@activity.defn(name="get_weather")`)
2. Docstrings!!
3. Consider letting the LLM process API outputs

This is about giving the LLM enough context to make smart choices



NOW, LET'S ORCHESTRATE IT!



This comes from a combination of Temporal and the OpenAI Agents SDK.

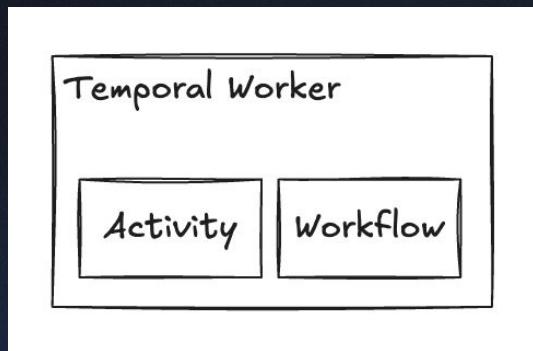
1. Agentic loop is implicit in the **Agent**
2. **activity_as_tool** creates the **dataclass** supplied to the agent.
3. Durability configurations (i.e. **start_to_close_timeout**) supplied as a part of tool definition

And then you need to start the agentic loop!!



WE NEED TO CONFIGURE THE INTEGRATION

The `OpenAIAgentsPlugin` is used to configure the integration



1. Handles serialization of Pydantic types
2. Propagates context for OpenAI Agents tracing
3. Handles LLM calls through activities
 - o Including durability configurations
4. Sets up some Python sandboxing

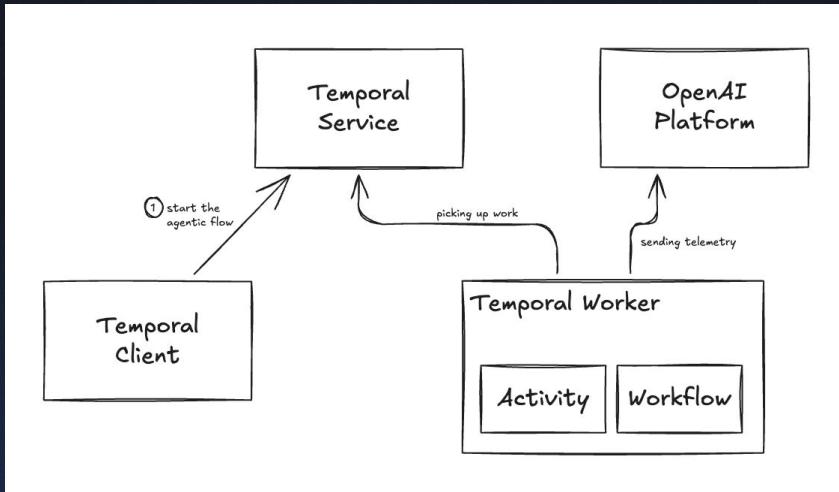
You still need to register your tool activities with the worker (but not the LLM activity)!



OKAY, NOW LET'S RUN IT!

The Temporal Client is loosely coupled from the agent implementation

1. Once accepted by the Temporal Service, this agent is durable
2. Workers can come and go





Durable!!!

let's take a closer look



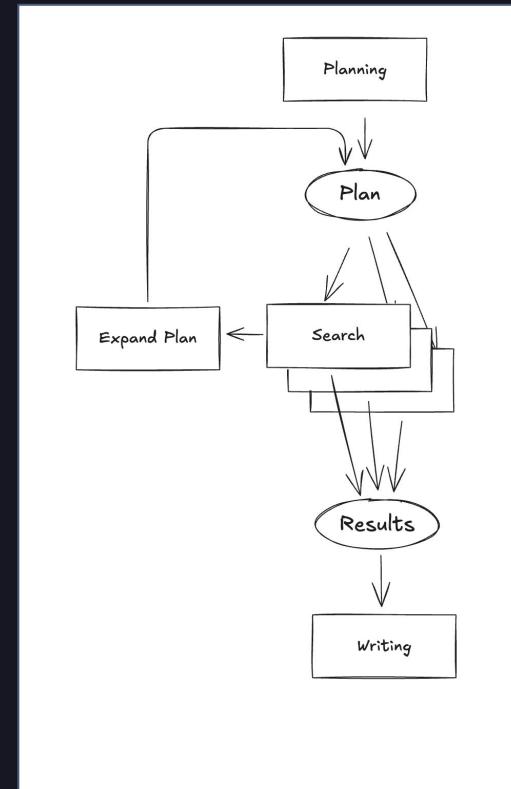
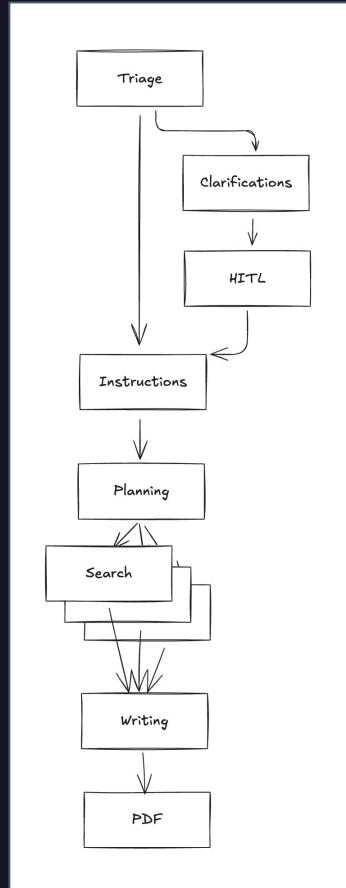
DEMO TIME!





Orchestrating (micro) Agents

Agents as Activities

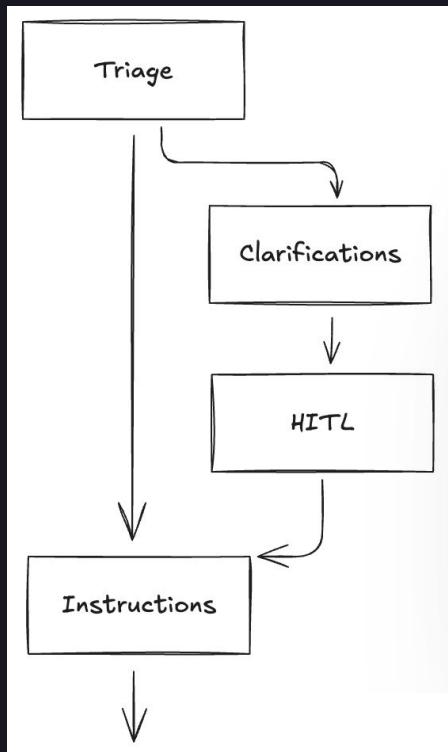


Two ways to orchestrate (micro)agents

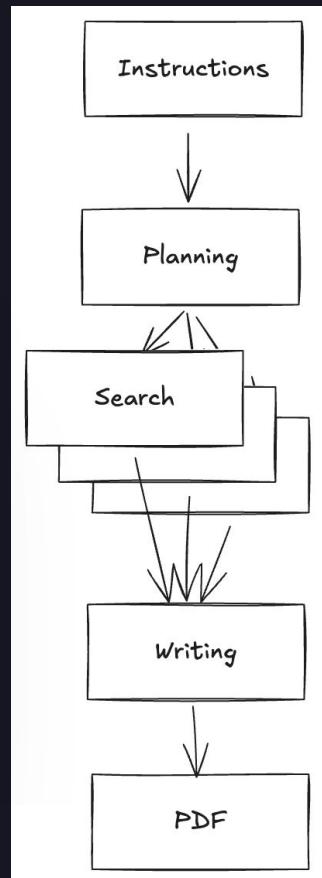
Just Code

Handoffs

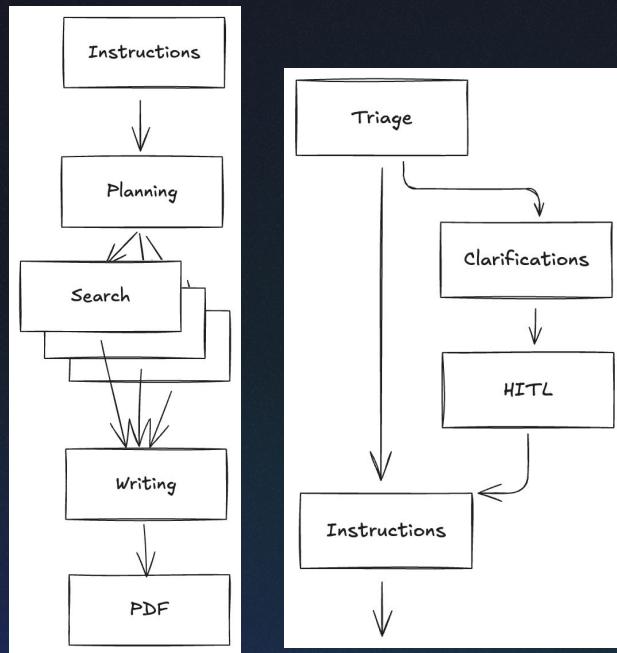
ORCHESTRATING AGENTS JUST CODE



```
1 triage_agent = Agent(...)
2 instructions_agent = Agent(...)
3
4 result = Runner.run_sync(triage_agent, ...)
5 if <user input needed>:
6     # Gather user input
7
8 ready_for_research = Runner.run_sync
9     (instructions_agent, result, ...)
```



ORCHESTRATING AGENTS JUST CODE



You can/will:

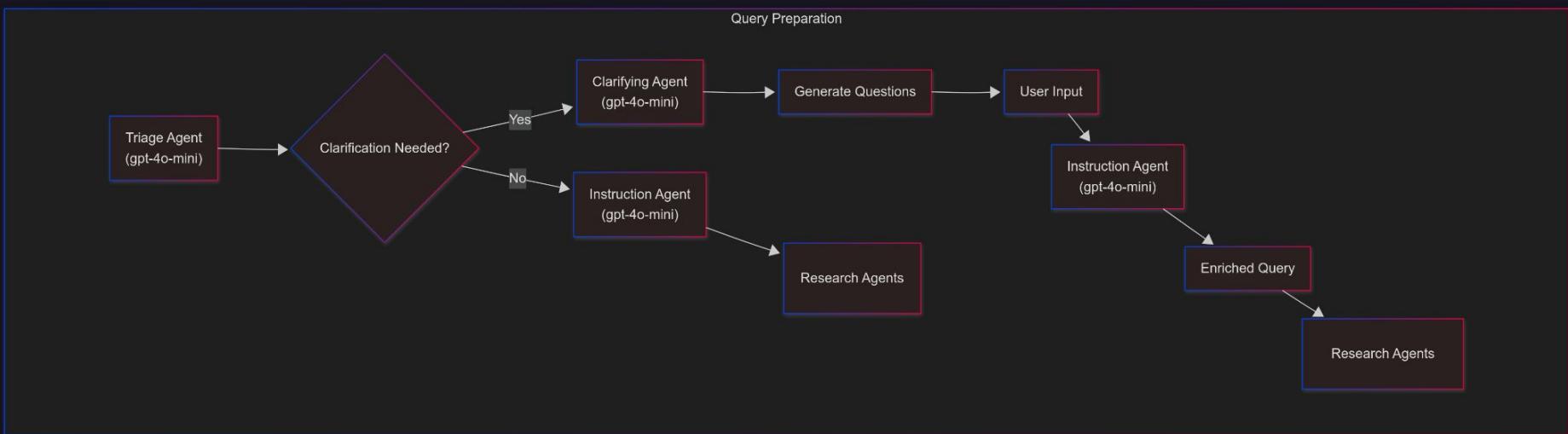
1. Each micro-agent has its own agentic loop
2. Parallel execution
3. Waits
4. HITL
5. ...

Have I mentioned? It's just code!



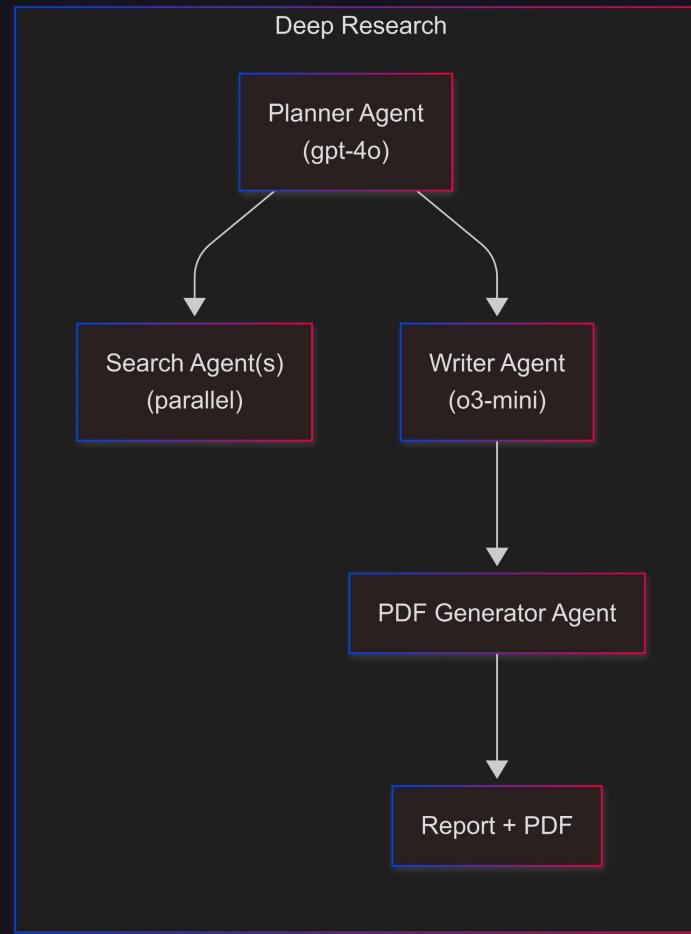
Deep Research Clarifying Questions

User inputs a topic...



Deep Research

Query prompt to research agents...



Deep Research

OpenAI Cookbook

Jun 25, 2025

Introduction to deep research in the OpenAI API

 Glory Jain, Kevin Alwell (OpenAI)  Open in GitHub  View as Markdown

https://cookbook.openai.com/examples/deep_research_api/introduction_to_deep_research_api

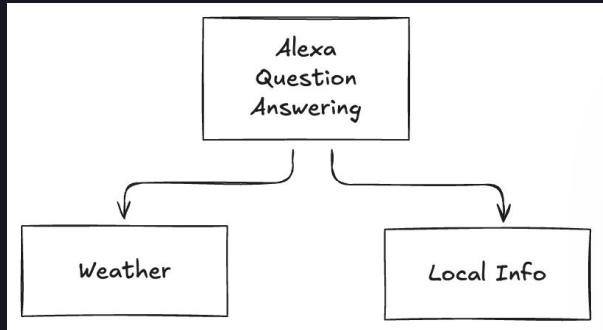
Some samples

OpenAI Agents SDK Github Repo



t.mp/github

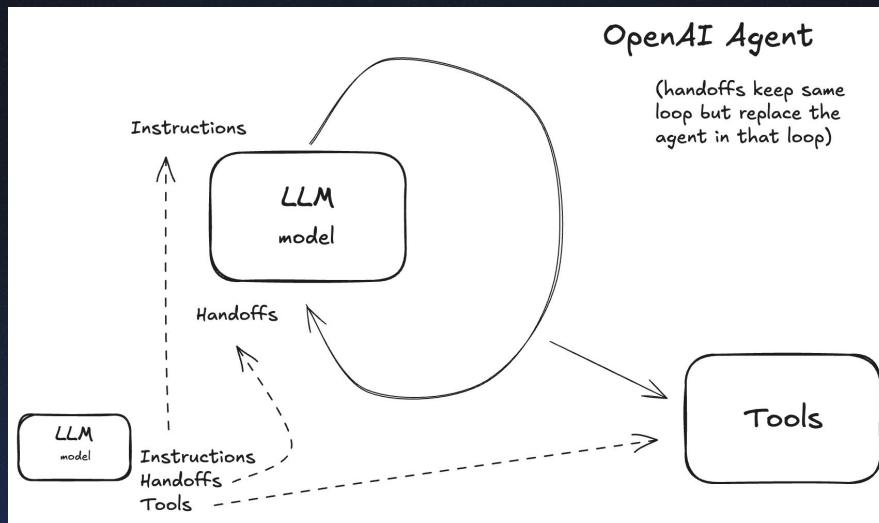
ORCHESTRATING AGENTS: HANDOFFS



```
1  from agents import Agent  
2  
3  agent = Agent(  
4      name="Triage Agent",  
5      model="gpt-4o-mini",  
6      instructions=TRIAGE_AGENT_INSTRUCTIONS,  
7      handoffs=[weather_agent, local_biz_agent],  
8      tools=[WebSearchTool],  
9  )  
10  
11 result = Runner.run_sync  
12     (agent, "How late is Costco open?")  
13
```

ORCHESTRATING AGENTS: HANDOFFS

These are:



1. Handoff happens within same agentic loop
2. An abstraction for context engineering
3. Carefully understand the semantics (i.e. not a sub-agent)
4. Temporal is handoff-aware

Agent takes on a different identity



DEMO TIME!



Summarizing



Takeaways

- ✓ Temporal Activities for LLM and Tool invocations - the LLM invocations are done for you automagically! 😊
- ✓ Temporal workflows to orchestrate agents (in code and handoffs)
- ✓ OpenAI Agents SDK native (i.e. you'll find info in traces)
- ✓ Use Pydantic AI? We've got you covered!



Check this out:

OpenAI Agents SDK Blog



t.mp/agents-blog

Pydantic Blog



t.mp/pydantic

WE WELCOME YOUR FEEDBACK!



T.MP/AI-WORKSHOP-FEEDBACK



GET STARTED WITH TEMPORAL!

- Join the community!
- Read the documentation
- Follow a tutorial
- Take a free course online and
more



Replay 2026

📍 Moscone Center, SF
📅 May 5-7, 2026
👤 1500+ attendees



Use Code
LAUNCHANDLEARN75 for
75% off

	TEMPORAL	01		TEMPORAL	02		REPLIT	03
	SAMAR ABBAS			MAXIM FATEEV			AMJAD MASAD	
	NVIDIA	04		NVIDIA	05		FIVETRAN	06
	JEREMY COOK			MATT TESCHER			ZAINAB MERCHANT	
	PYDANTIC	07						
	SAMUEL COLVIN							

- // JUST ANNOUNCED

MAY 05-07, 2026

MOSCONC CENTER, SAN FRANCISCO



REPLAY

