

What's Next?

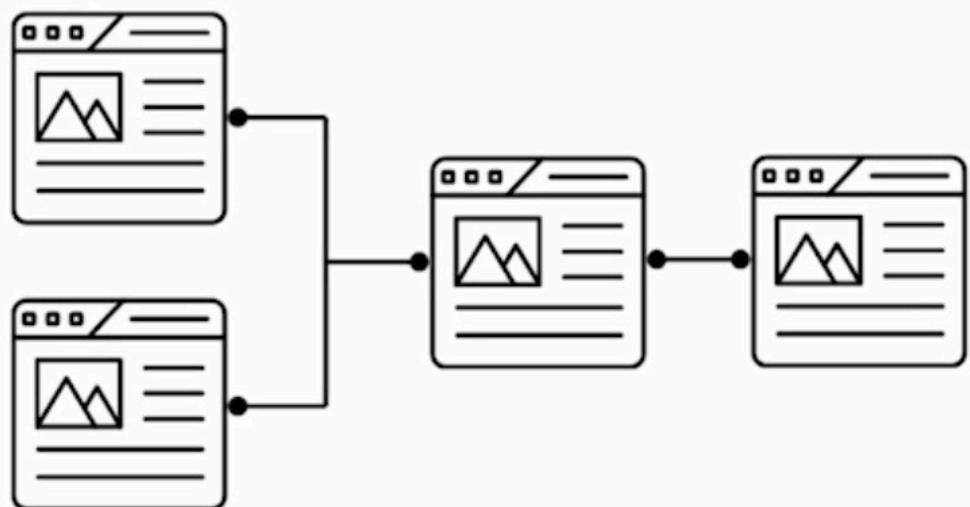


- ★ Stay in the loop by checking the GitHub repo, joining the Discord channel, and jumping into discussions on GitHub.
- ★ Contribute directly by checking out open issues labeled “help wanted”, or start a discussion if you’ve got your own ideas you’d like to explore.
- ★ Give feedback – we actively encourage it. If you run into bugs or problems, open an issue on GitHub and we’ll do our best to resolve it quickly.

Why We Need Agent Communication Standards



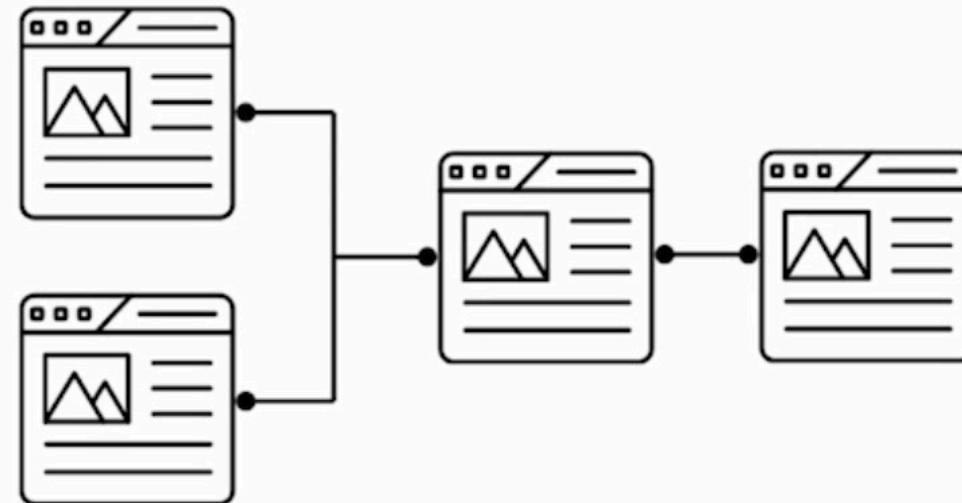
In 1989, **HTTP** unlocked the web as a transportation protocol to connect hosted pages



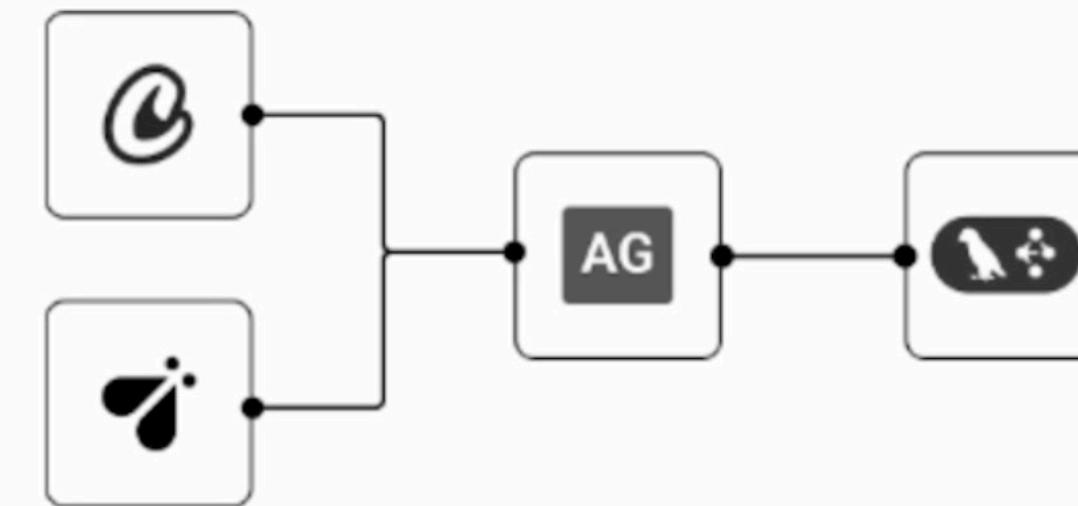
Why We Need Agent Communication Standards



In 1989, **HTTP** unlocked the web as a transportation protocol to connect hosted pages

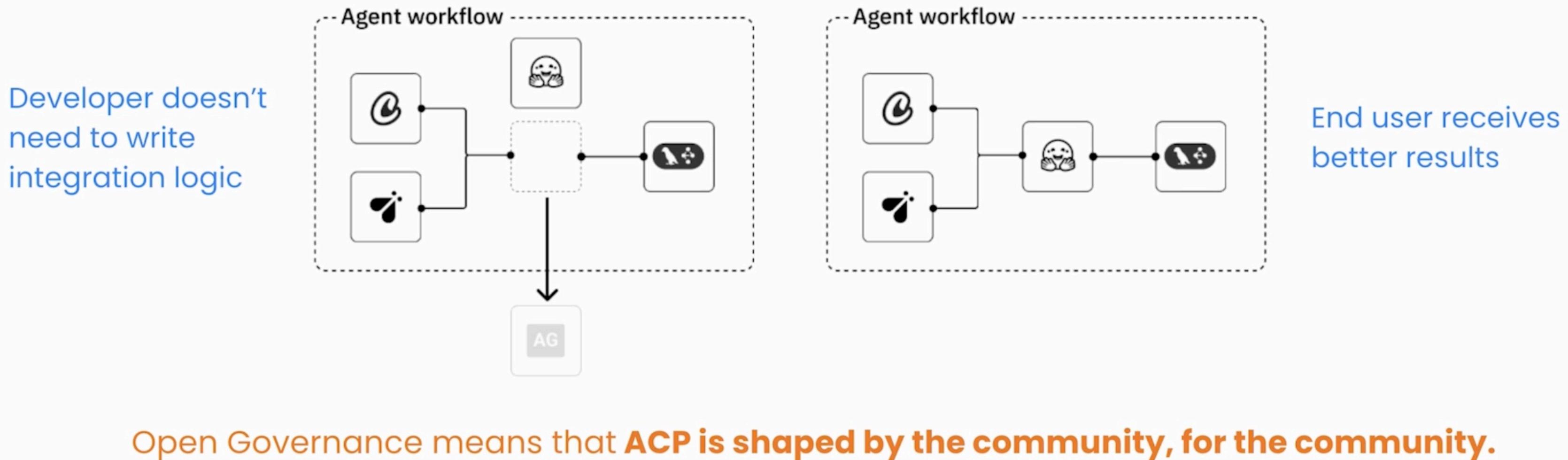


ACP aims to connect the ecosystem of stand alone agents so they can collaborate

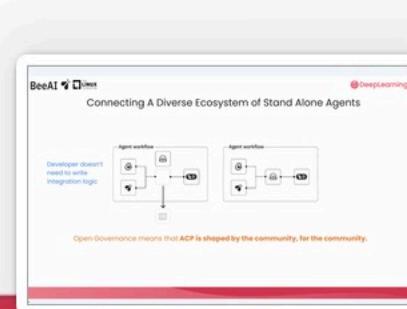


Framework and Technology Agnostic

Connecting A Diverse Ecosystem of Stand Alone Agents

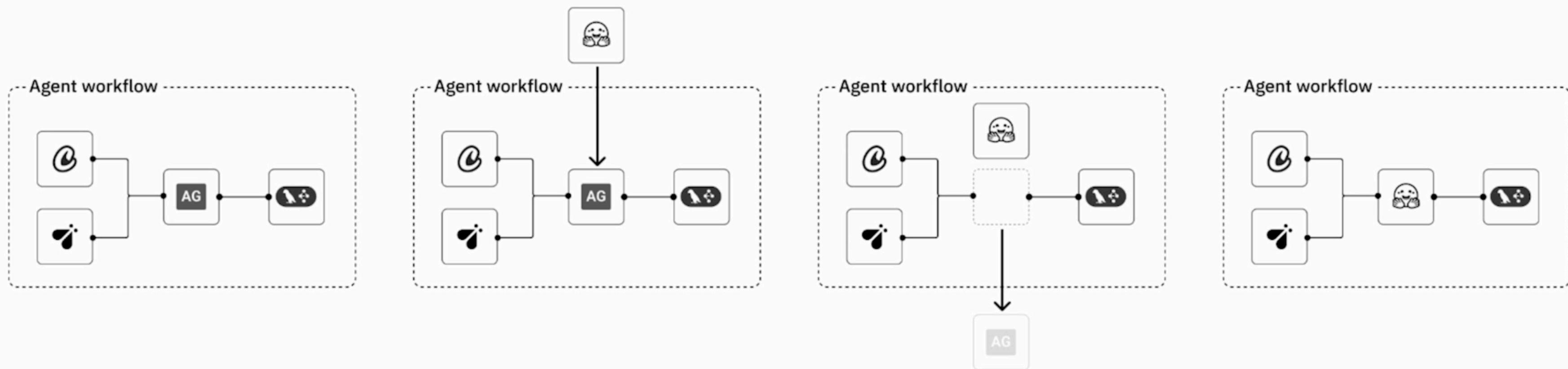


ACP provides unified interface for agents to communicate regardless of their framework



What ACP Makes Possible

Dynamic Updating: An ACP agent can be swapped for another ACP Agent, making your system truly interoperable

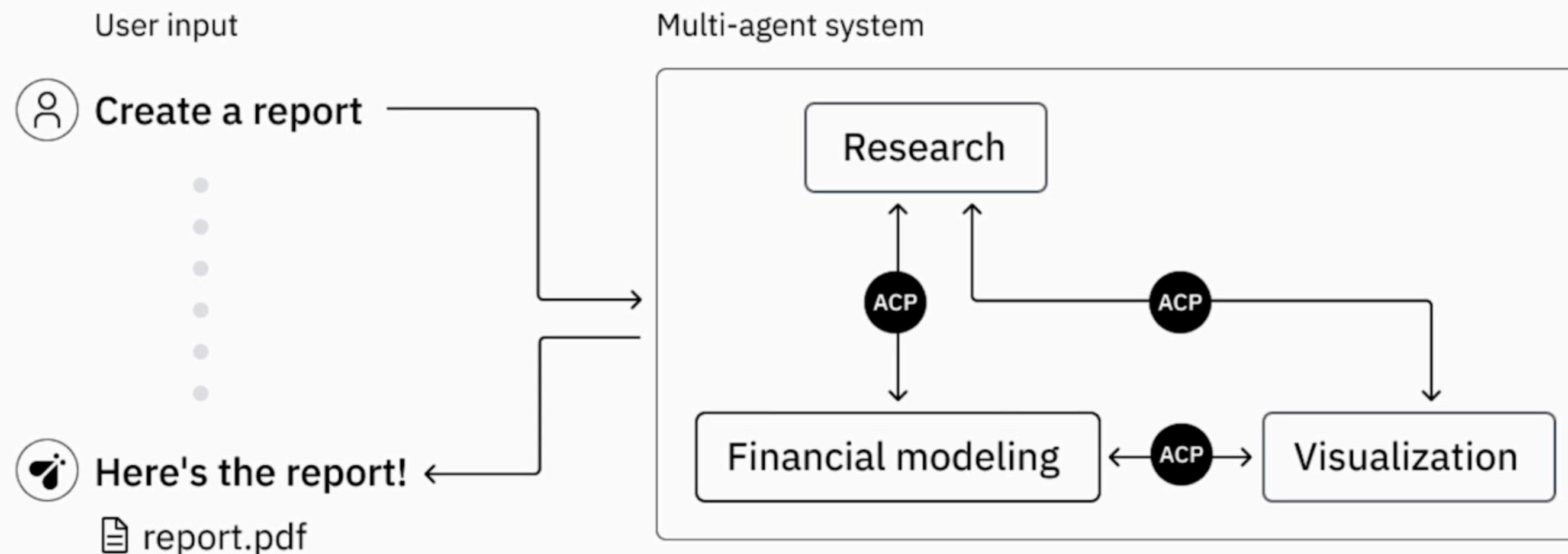


Agent which was best last month isn't best this month.

As ACP's are system compatible we can swap agents in our systems, even if new agent is build using different technology

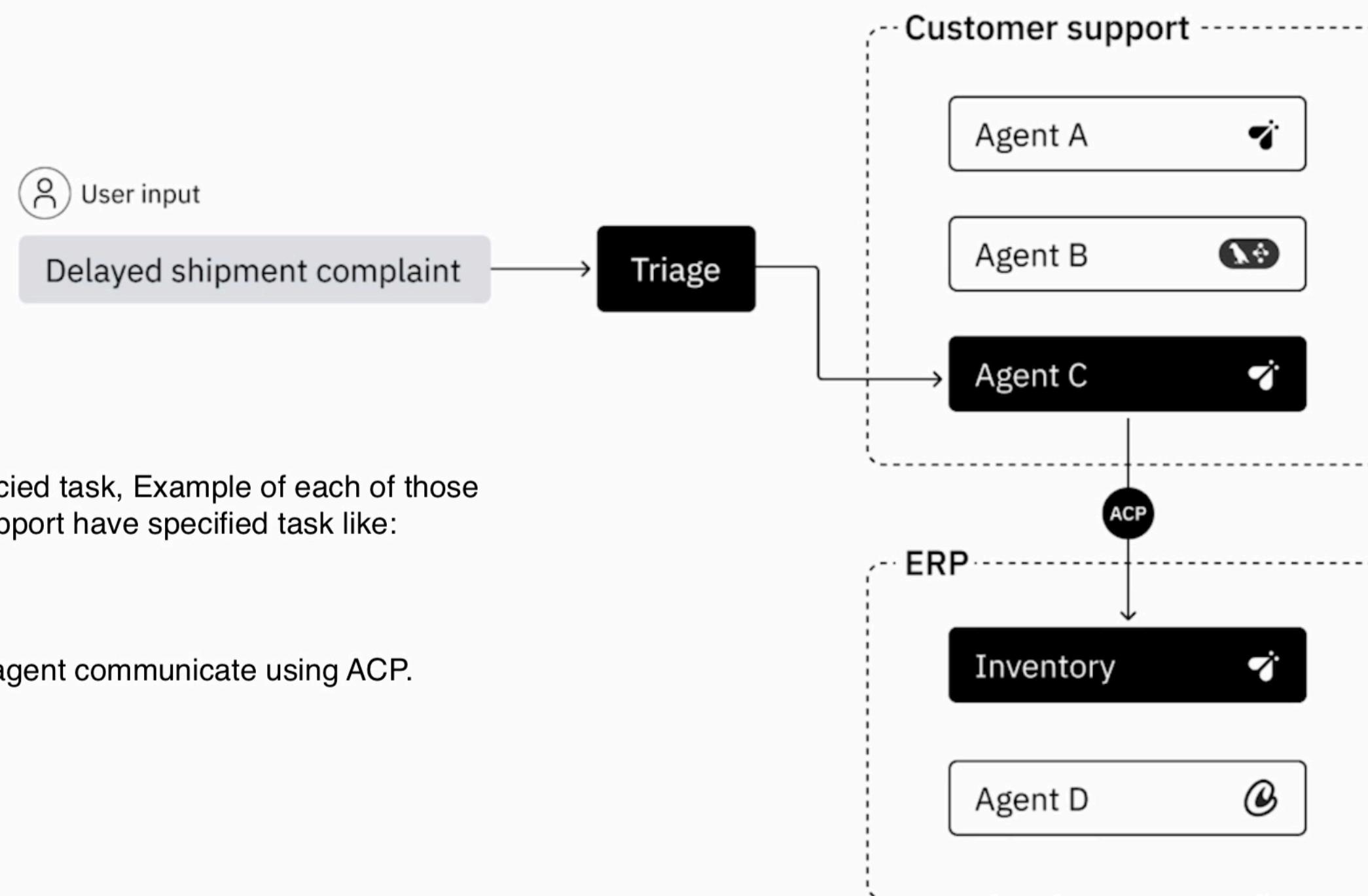
What ACP Makes Possible

Specialized Agents can work as a Team: Pre-configured Agents can collaborate without needing to build one monolithic multi-agent architecture



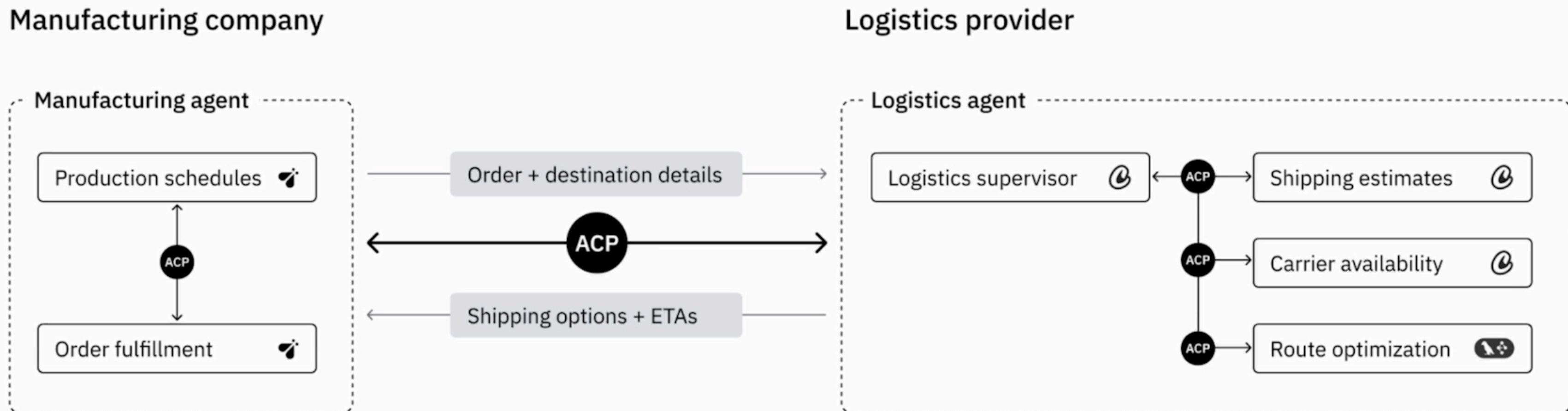
What ACP Makes Possible

Across Company Workflows: Agents that represent different systems or different parts of the company can collaborate



What ACP Makes Possible

Inter-Organizational Workflows: Agents that belong to different organizations can collaborate on inter-company tasks



It also supports Human in the Loop

ACP: Agent Communication Protocol

ACP Core Principles

BeeAI 

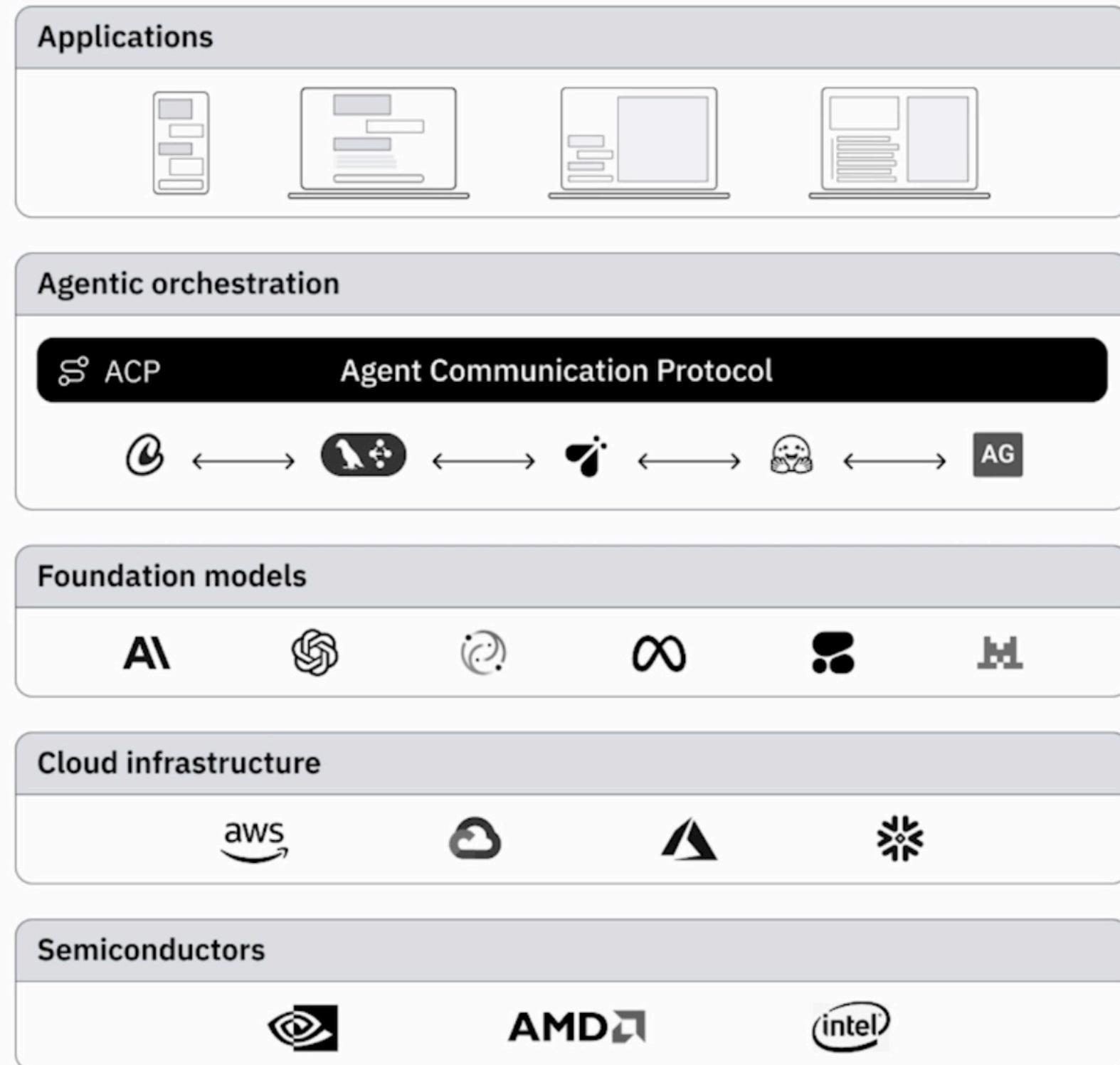


DeepLearning.AI

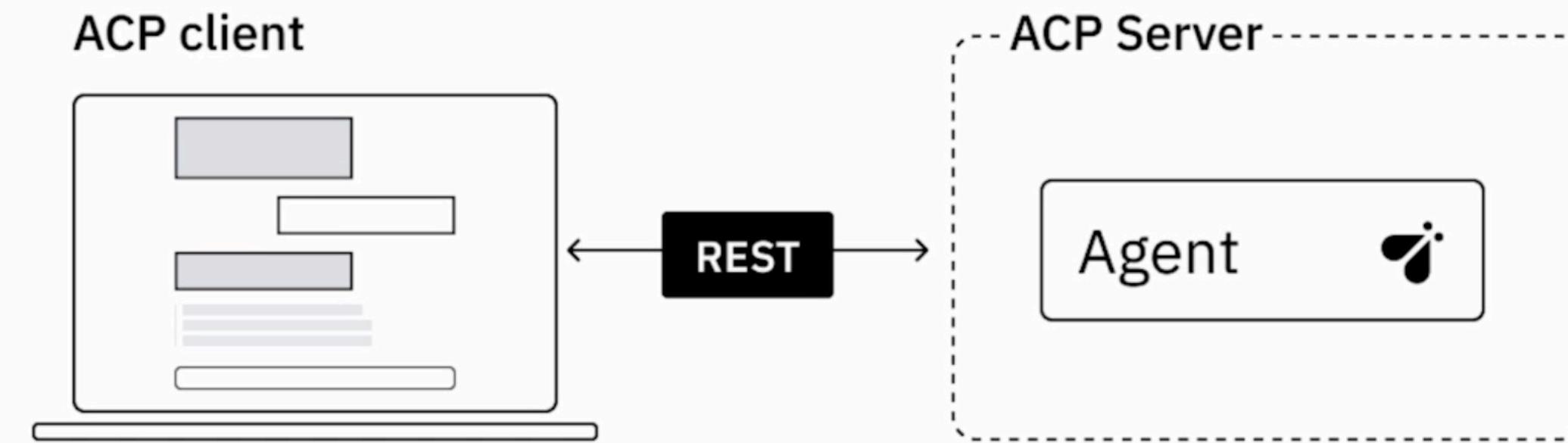


Where ACP Fits in the Tech Stack

1. ACP sits in the operation layer.
2. It uses Http and Rest Architectural style to facilitate communication between independent systems

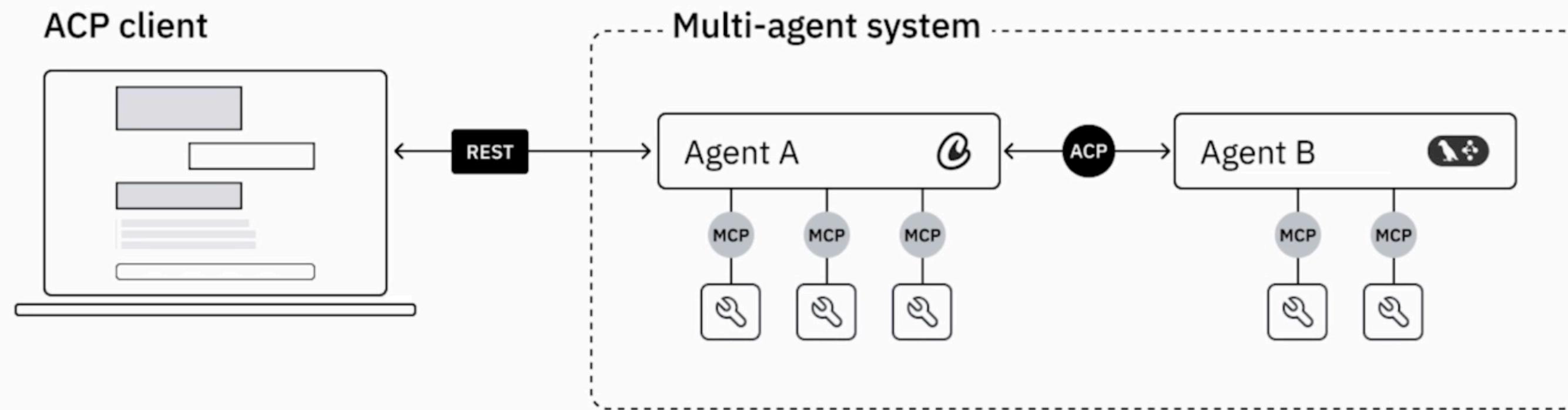


Simple ACP Client Server Transaction



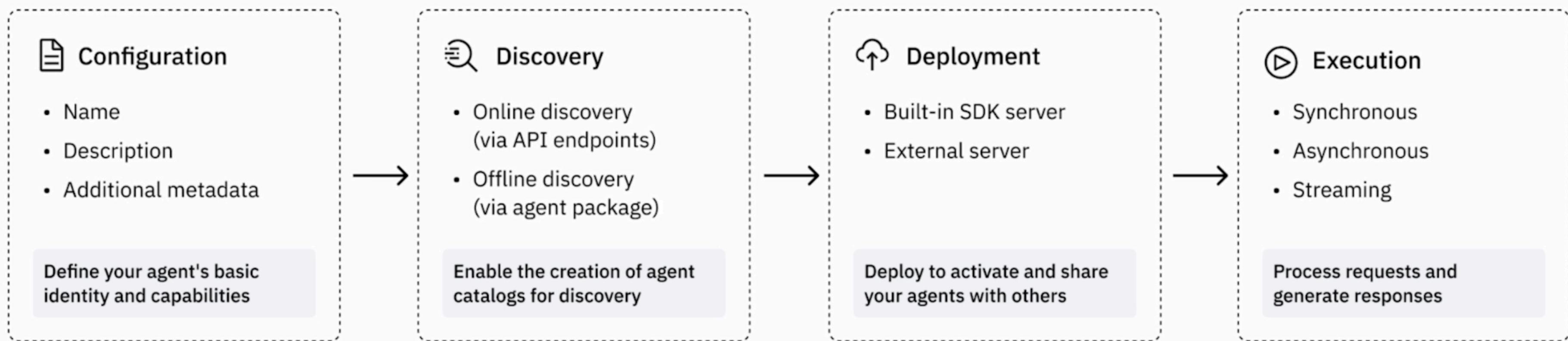
1. The ACP Client discovers available servers and then initiates a request.
 2. It sends a REST request to the ACP Server.
 3. The server, which wraps an agent, manages the request and returns a response back to the client over REST.
-
1. ACP client might be Human, AI Agent, etc
 2. Client it will always intiate the requests and server always responds to the request.

Using ACP and MCP Together

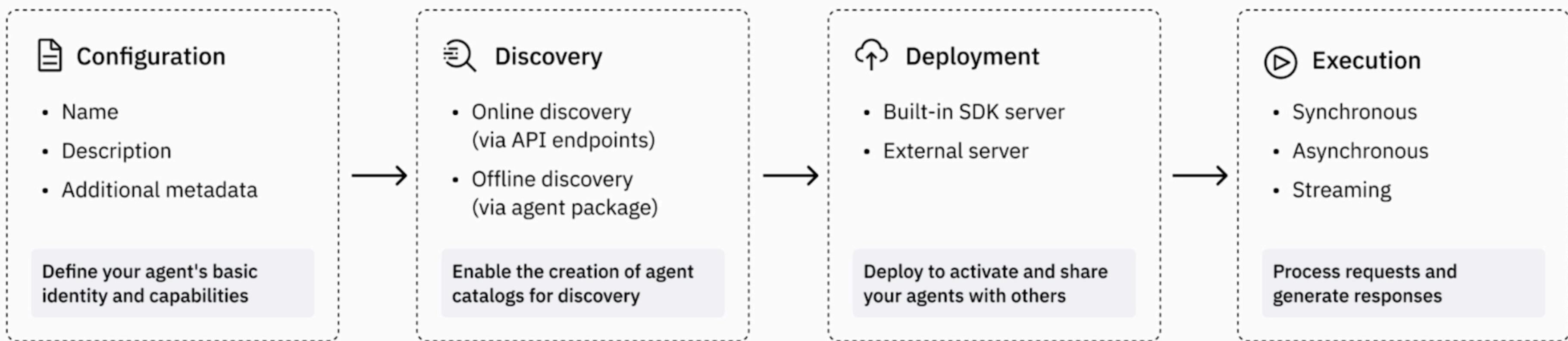


1. A client makes a request to an ACP server over REST, which wraps an agent.
2. The agent determines it needs to invoke a tool, so it sends a request to an available MCP server to execute the tool call and return the result.
3. Once the ACP agent completes its run, it returns the output to the ACP client.

ACP Agent Lifecycle



ACP Agent Lifecycle



ACP is designed with a production-grade environments in mind

Comparing ACP to MCP and A2A

MCP (Model Context Protocol)

enrich one agent's context with tools
+ prompts + resources

ACP (Agent Communication Protocol)

Openly governed agent-to-agent
communication (REST)

A2A (Agent-to-Agent Protocol)

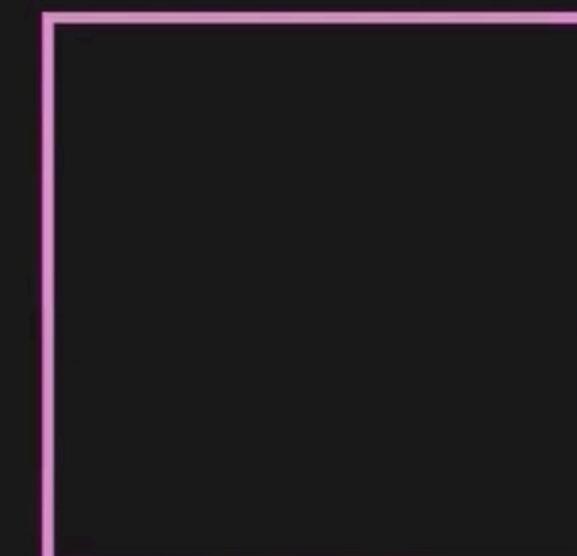
Google's Agent-to-Agent Protocol
(JSON-RPC)

MCP inside an agent → fetch tools / data → returns context

MCP → ACP → agent calls MCP for more context,
decides it needs help, then talks to peer over ACP

A2A vs ACP → same goal, different governance + architectural choices

	ACP	A2A
Governance	Openly Governed, Linux Foundation	Open-source
Hosting	Many agents/server	1 agent/server
Comm style	REST (sync / async / SSE)	JSON-RPC (async first)
Session state	Central store across servers	Agent-chosen
Artifacts + history	Unified	Separate (harder traceability)



Hospital Agent Insurance Agent



BeeAI

smolagents_server.py U X

crew_agent_server.py U

smolagents_server.py > [?] llm

```
7 import os
8
9 server = Server()
10
11 load_dotenv()
12
13 #LLM
14 llm = LiteLLMModel(
15     model_id="watsonx/meta-llama/llama-4-maverick-17b-128e-instruct-fp8",
16     api_key=os.environ['WATSONX_APIKEY'],
17     num_ctx=8162
18 )
19
20 server_parameters = StdioServerParameters(
21     command="uv",
22     args=["run", "mcpserver.py"],
23     env=None
```

PROBLEMS

DEBUG CONSOLE

TERMINAL

PORTS

OUTPUT

zsh + ~  

... ▾ ×

○ nicholasrenotte@Nicholass-MacBook-Pro-2 my_acp_project % brew install beeai

PROBLEMS

DEBUG CONSOLE

TERMINAL

PORTS

OUTPUT

> zsh + ~

...

▼

×

○ nicholasrenotte@Nicholass-MacBook-Pro-2 my_acp_project % beeai platform start

PROBLEMS

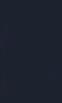
DEBUG CONSOLE

TERMINAL

PORTS

OUTPUT

> zsh + ~



○ nicholasrenotte@Nicholass-MacBook-Pro-2 my_acp_project % beeai

PROBLEMS

DEBUG CONSOLE

TERMINAL

PORTS

OUTPUT

zsh + ~ ...

--show-completion

Show completion for the current shell, to copy it or customize the installation.

--help

Show this message and exit.

Commands**version**

Print version of the BeeAI CLI and related libraries.

serve

Start server.

ui

Launch graphical interface.

env

Manage environment variables.

agent

Manage agents.

telemetry

Configure telemetry.

compose

Manage agent composition.

build**Agent commands****add | install**

Install discovered agent or add public docker image or github repository [aliases: install]

remove | uninstall | rm | delete

Remove agent

logs

Stream agent provider logs

run

Run an agent.

list

List agents.

info

Show agent details.

[PROBLEMS](#)[DEBUG CONSOLE](#)[TERMINAL](#)[PORTS](#)[OUTPUT](#)[zsh + ~](#)

Commands

version

Print version of the BeeAI CLI and related libraries.

serve

Start server.

ui

Launch graphical interface.

env

Manage environment variables.

agent

Manage agents.

telemetry

Configure telemetry.

compose

Manage agent composition.

build

Agent commands

add | install

Install discovered agent or add public docker image or github repository [aliases: install]

remove | uninstall | rm | delete

Remove agent

logs

Stream agent provider logs

run

Run an agent.

list

List agents.

info

Show agent details.

PROBLEMS

DEBUG CONSOLE

TERMINAL

PORTS

OUTPUT

zsh + ~



NAME	ST...	DESCRIPTION	LOCATION
agent_docs_...	no...	The agent analyzes ...	official/beeai-framework/agent-docs-creato
aider	no...	An AI pair programm...	community/aider:agents-v0.1.3
chat	no...	The agent is an AI-...	official/beeai-framework/chat:agents-v0.1.
gpt_researc...	no...	The agent conducts ...	community/gpt-researcher:agents-v0.1.3
literature_...	no...	This agent automate...	community/literature-review:agents-v0.1.3
marketing_s...	no...	The agent performs ...	community/marketing-strategy:agents-v0.1.3
ollama_deep...	no...	The agent performs ...	community/ollama-deep-researcher:agents-v0
podcast_cre...	no...	The agent creates s...	official/beeai-framework/podcast-creator:a
sequential_...	no...	The agent orchestra...	official/sequential-workflow:agents-v0.1.3

PROBLEMS

DEBUG CONSOLE

TERMINAL

PORTS

OUTPUT

zsh + ~



NAME	ST...	DESCRIPTION	LOCATION
agent_docs_...	no...	The agent analyzes ...	official/beeai-framework/agent-docs-creato
aider	no...	An AI pair programm...	community/aider:agents-v0.1.3
chat	no...	The agent is an AI-...	official/beeai-framework/chat:agents-v0.1.
gpt_researc...	no...	The agent conducts ...	community/gpt-researcher:agents-v0.1.3
literature_...	no...	This agent automate...	community/literature-review:agents-v0.1.3
marketing_s...	no...	The agent performs ...	community/marketing-strategy:agents-v0.1.3
ollama_deep...	no...	The agent performs ...	community/ollama-deep-researcher:agents-v0
podcast_cre...	no...	The agent creates s...	official/beeai-framework/podcast-creator:a
sequential_...	no...	The agent orchestra...	official/sequential-workflow:agents-v0.1.3

⚡ smolagents_server.py U

⚡ crew_agent_server.py U ●

⚡ crew_agent_server.py > {} Message

```
1  from collections.abc import AsyncGenerator
2  from acp_sdk.models import Message, MessagePart, Metadata
3  from acp_sdk.server import RunYield, RunYieldResume, Server
4
5  from crewai import Crew, Task, Agent, LLM
6  from crewai_tools import RagTool
7  import nest_asyncio
8  import os
9
10 nest_asyncio.apply()
11
12 server = Server()
13 llm = LLM(
14     model='watsonx/meta-llama/llama-4-maverick-17b-128e-instruct-fp8',
15     base_url=os.environ['WATSONX_URL'],
16     project_id=os.environ['WX_PROJECT_ID'],
17     max_tokens=1024)
```

smolagents_server.py U

crew_agent_server.py U

smolagents_server.py

```
25
26 @server.agent(metadata=Metadata(ui={"type": "hands-off"}))
27 async def health_agent(input: list[Message]) -> AsyncGenerator[RunYield, RunYield]:
28     "This is a CodeAgent which supports the hospital to handle health based questions"
29     agent = CodeAgent(tools=[DuckDuckGoSearchTool(), VisitWebpageTool()], model=model)
30
31     prompt = input[0].parts[0].content
32     response = agent.run(prompt)
33
34     yield Message(parts=[MessagePart(content=str(response))])
35
36 @server.agent()
37 async def doctor_agent(input: list[Message]) -> AsyncGenerator[RunYield, RunYield]:
38     "This is a Doctor Agent which helps users find doctors near them."
39     with ToolCollection.from_mcp(server_parameters, trust_remote_code=True) as tool_collection:
40         agent = ToolCallingAgent(tools=[*tool_collection.tools], model=model)
41         prompt = input[0].parts[0].content
```

smolagents_server.py 1, U

crew_agent_server.py U

smolagents_server.py > ...

```
26     ;-off","user_greeting":"Ask your health question!"}) )  
27     \syncGenerator[RunYield, RunYieldResume]:  
28         ital to handle health based questions for patients. Current or prospective patie  
29         L(), VisitWebpageTool()), model=model)  
30  
31  
32  
33  
34     (response)) )  
35  
36     ;-off","user_greeting":"Find ["}) )  
37     \syncGenerator[RunYield, RunYieldResume]:  
38         id doctors near them."  
39         `s, trust_remote_code=True) as tool_collection:  
40             [lection.tools], model=model)  
41  
42
```

smolagents_server.py U

crew_agent_server.py U

smolagents_server.py > ...

```
26     /er.agent(metadata=Metadata(ui={"type":"hands-off","user_greeting":"Ask your health care provider."}))
27     : def health_agent(input: list[Message]) -> AsyncGenerator[RunYield, RunYieldResult]:
28         'This is a CodeAgent which supports the hospital to handle health based questions'
29         agent = CodeAgent(tools=[DuckDuckGoSearchTool(), VisitWebpageTool()], model=model)
30
31         prompt = input[0].parts[0].content
32         response = agent.run(prompt)
33
34         yield Message(parts=[MessagePart(content=str(response))])
35
36     /er.agent(metadata=Metadata(ui={"type":"hands-off","user_greeting": "Hello! How can I help you today? I'm here to answer your medical questions."}))
37     : def doctor_agent(input: list[Message]) -> AsyncGenerator[RunYield, RunYieldResult]:
38         'This is a Doctor Agent which helps users find doctors near them.'
39         with ToolCollection.from_mcp(server_parameters, trust_remote_code=True) as tool_collection:
40             agent = ToolCallingAgent(tools=[*tool_collection.tools], model=model)
41             prompt = input[0].parts[0].content
42             response = agent.run(prompt)
```

BeeAI

localhost:8333/agents/agent_docs_creator/run

agent_docs_creator

Agents

- agent_docs_creator
- aider
- chat
- gpt_researcher
- literature_review
- marketing_strategy
- ollama_deep_researcher
- podcast_creator

agent_docs_creator

Provide source code of the AI agent you want to document

Write the agent prompt here

meta-llama/llama-4-maverick-17b-128e-instruct-fp8

Run ►

⚙️

smolagents_server.py U

crew_agent_server.py U X

crew_agent_server.py > {} Message

```
1 from collections.abc import AsvncGenerator
```

PROBLEMS DEBUG CONSOLE

TERMINAL PORTS OUTPUT

> uv + v □ ✖ ... ^ X

marketing_s... no... The
agent performs ... co
mmunity/marketing-str
tegy:agents-v0.1.3

ollama_deep... no... The
agent performs ... co
mmunity/ollama-deep-re
searcher:agents-v0

podcast_cre... no... The
agent creates s... of
ficial/beeai-framework
/podcast-creator:a

sequential_... no... The
agent orchest

nicholasrenotte@Nichol
ass-MacBook-Pro-2 my_a
cp_project %

INFO: Uvicorn runn
ing on **http://127.0.0.**
1:8000 (Press CTRL+C t
o quit)

INFO: 127.0.0.1:65
421 - "GET /agents HTT
P/1.1" 200 OK

INFO: HTTP Request
: POST http://127.0.0.
1:8333/api/v1/provider
s/register/unmanaged "
HTTP/1.1 200 OK"

INFO: Agent regist
ered to the beeai serv
er.

INFO: HTTP Request
: GET http://127.0.0.1
:8333/api/v1/variables

nicholasrenotte@Nichol
ass-MacBook-Pro-2 my_a
cp_project % uv run cr
ew_agent_server.py

smolagents_server.py U

py crew_agent_server.py U X

crew_agent_server.py > {} Message

```
1 from collections.abc import AsyncGenerator
2 from acp_sdk.models import Message, MessagePart, Metadata
3 from acp_sdk.server import RunYield, RunYieldResume, Server
4
5 from crewai import Crew, Task, Agent, LLM
6 from crewai_tools import RagTool
7 import nest_asyncio
8 import os
9
10 nest_asyncio.apply()
```

PROBLEMS DEBUG CONSOLE TERMINAL PORTS OUTPUT

zsh + v □ ⌫ ... ^ ×

NAME	STATUS	DESCRIPTION	UI	LOCATION	MISS
doctor_agent	running	This is a Doctor Agent w...	han...	http://127.0.0.1:8000/	<non
health_agent	running	This is a CodeAgent whic...	han...	http://127.0.0.1:8000/	<non
policy_agent	running	This is an agent for que...	han...	http://127.0.0.1:8001/	<non
agent_docs_creator	not_ins...	The agent analyzes AI so...	han...	official/beeai-framework/agent-docs-creator:agents-v0.1.3	<non
aider	not_ins...	An AI pair programmer th...	han...	community/aider:agents-v0.1.3	<non
chat	not_ins...	The agent is an AI-power...	chat	official/beeai-framework/chat:agents-v0.1.3	<non
gpt_researcher	not_ins...	The agent conducts in-de...	han...	community/gpt-researcher:agents-v0.1.3	<non
literature_review	not_ins...	This agent automates dee...	han...	community/literature-review:agents-v0.1.3	<non
marketing_strategy	not_ins...	The agent performs compr...	han...	community/marketing-strategy:agents-v0.1.3	<non
ollama_deep_researcher	not_ins...	The agent performs AI-dr...	han...	community/ollama-deep-researcher:agents-v0.1.3	<non
podcast_creator	not_ins...	The agent creates struct...	han...	official/beeai-framework/podcast-creator:agents-v0.1.3	<non
sequential_workflow	not_ins...	The agent orchestrates a...	<no...	official/sequential-workflow:agents-v0.1.3	<non

nicholasrenotte@Nicholass-MacBook-Pro-2 my_acp_project %

smolagents_server.py

⌘ crew_agent_server.py U ×

crew_agent_server.py >

```
1 from collections.abc import AsyncGenerator
2 from acp_sdk.models import Message, MessagePart, Metadata
3 from acp_sdk.server import RunYield, RunYieldResume, Server
4
5 from crewai import Crew, Task, Agent, LLM
6 from crewai_tools import RagTool
7 import nest_asyncio
8 import os
9
10 nest_asyncio.apply()
```

PROBLEMS DEBUG CONSOLE TERMINAL PORTS OUTPUT

Python + ⌂ ⌂ ⌂ ⌂ ⌂ ⌂ ⌂ ⌂

gpt_researcher	not_ins...	The agent conducts in-de...	han...	community/gpt-researcher:agents-v0.1.3	<non
literature_review	not_ins...	This agent automates dee...	han...	community/literature-review:agents-v0.1.3	<non
marketing_strategy	not_ins...	The agent performs compr...	han...	community/marketing-strategy:agents-v0.1.3	<non
ollama_deep_researcher	not_ins...	The agent performs AI-dr...	han...	community/ollama-deep-researcher:agents-v0.1.3	<non
podcast_creator	not_ins...	The agent creates struct...	han...	official/beeai-framework/podcast-creator:agents-v0.1.3	<non
sequential_workflow	not_ins...	The agent orchestrates a...	<no...	official/sequential-workflow:agents-v0.1.3	<non

>>> Type your message (/? for help, /q to quit)

smolagents_server.py U

crew_agent_server.py U X

crew_agent_server.py > ...

```
1  from collections.abc import AsyncGenerator
2  from acp_sdk.models import Message, MessagePart, Metadata
3  from acp_sdk.server import RunYield, RunYieldResume, Server
4
5  from crewai import Crew, Task, Agent, LLM
6  from crewai_tools import RagTool
7  import nest_asyncio
8  import os
9
10 nest_asyncio.apply()
```

PROBLEMS

DEBUG CONSOLE

TERMINAL

POROS

OUTPUT

Python + ▾ □ ⚡ ⌂ ⌃ ⌄ ⌅ ⌆ ⌇

gpt_researcher	not_ins...	The agent conducts in-de...	han...	community/gpt-researcher:agents-v0.1.3	<non
literature_review	not_ins...	This agent automates dee...	han...	community/literature-review:agents-v0.1.3	<non
marketing_strategy	not_ins...	The agent performs compr...	han...	community/marketing-strategy:agents-v0.1.3	<non
ollama_deep_rese...	not_ins...	The agent performs AI-dr...	han...	community/ollama-deep-researcher:agents-v0.1.3	<non
podcast_creator	not_ins...	The agent creates struct...	han...	official/beeai-framework/podcast-creator:agents-v0...	<non
sequential_workflow	not_ins...	The agent orchestrates a...	<no...	official/sequential-workflow:agents-v0.1.3	<non
○ nicholasrenotte@Nicholass-MacBook-Pro-2 my_acp_project % beeai run doctor_agent					
Find a doctor, pass your query and state!					

```
>>> I'm based in Atlanta, GA. Are there any Cardiologists near me?
```

 smolagents server.py

⚡ crew agent server.py ⚡

`py` `crew agent server.py > ...`

```
1  from collections.abc import AsyncGenerator
2  from acp_sdk.models import Message, MessagePart, Metadata
3  from acp_sdk.server import RunYield, RunYieldResume, Server
4
5  from crewai import Crew, Task, Agent, LLM
6  from crewai_tools import RagTool
7  import nest_asyncio
8  import os
9
10 nest_asyncio.apply()
```

PROBLEMS DEBUG CONSOLE TERMINAL PORTS OUTPUT

zsh + ⌘ ⌘ ... ^ x

```
nicholasrenotte@Nicholass-MacBook-Pro-2 my_acp_project % beeai run doctor_agent
Find a doctor, pass your query and state!
```

>>> I'm based in Atlanta, GA. Are there any Cardiologists near me>

Yes, there are several cardiologists near you in Atlanta, GA. One of them is Dr. Sarah Mitchell, who is board-certified and has 15 years of experience. She is affiliated with Emory University Hospital and Piedmont Atlanta Hospital, and accepts several insurance providers including Blue Cross Blue Shield, Aetna, Cigna, and UnitedHealth. You can contact her at (404) 555-2847 or via email at s.mitchell@atlantaheart.com.

nicholasrenotte@Nicholass-MacBook-Pro-2 my acp project %

smolagents_server.py

crew_agent_server.py U >

py crew_agent_server.py > ..

```
1  from collections.abc import AsyncGenerator
2  from acp_sdk.models import Message, MessagePart, Metadata
3  from acp_sdk.server import RunYield, RunYieldResume, Server
4
5  from crewai import Crew, Task, Agent, LLM
6  from crewai_tools import RagTool
7  import nest_asyncio
8  import os
9
10 nest_asyncio.apply()
```

[PROBLEMS](#) [DEBUG CONSOLE](#) [TERMINAL](#)

>>> I'm based in Atlanta, GA. Are there any Cardiologists near me>

Yes, there are several cardiologists near you in Atlanta, GA. One of them is Dr. Sarah Mitchell, who is board-certified and has **15** years of experience. She is affiliated with Emory University Hospital and Piedmont Atlanta Hospital, and accepts several insurance providers including Blue Cross Blue Shield, Aetna, Cigna, and UnitedHealth. You can contact her at **(404) 555-2847** or via email at s.mitchell@atlantaheart.com.

- nicholasrenotte@Nicholass-MacBook-Pro-2 my_acp_project % beeai run policy_agent
Got a question about your policy? Ask here!

>>> what is the waiting period on physiotherapy?

The waiting period on physiotherapy is 2 months. Group physiotherapy is covered with a limit of \$35 per visit.

○ nicholasrenotte@Nicholass-MacBook-Pro-2 my_acp_project %

smolagents_server.py U

crew_agent_server.py U X

crew_agent_server.py > ...

```
1  from collections.abc import AsyncGenerator
2  from acp_sdk.models import Message, MessagePart, Metadata
3  from acp_sdk.server import RunYield, RunYieldResume, Server
4
5  from crewai import Crew, Task, Agent, LLM
6  from crewai_tools import RagTool
7  import nest_asyncio
8  import os
9
10 nest_asyncio.apply()
```

PROBLEMS DEBUG CONSOLE TERMINAL PORTS OUTPUT

Python + ▾

>>> I'm based in Atlanta, GA. Are there any Cardiologists near me?

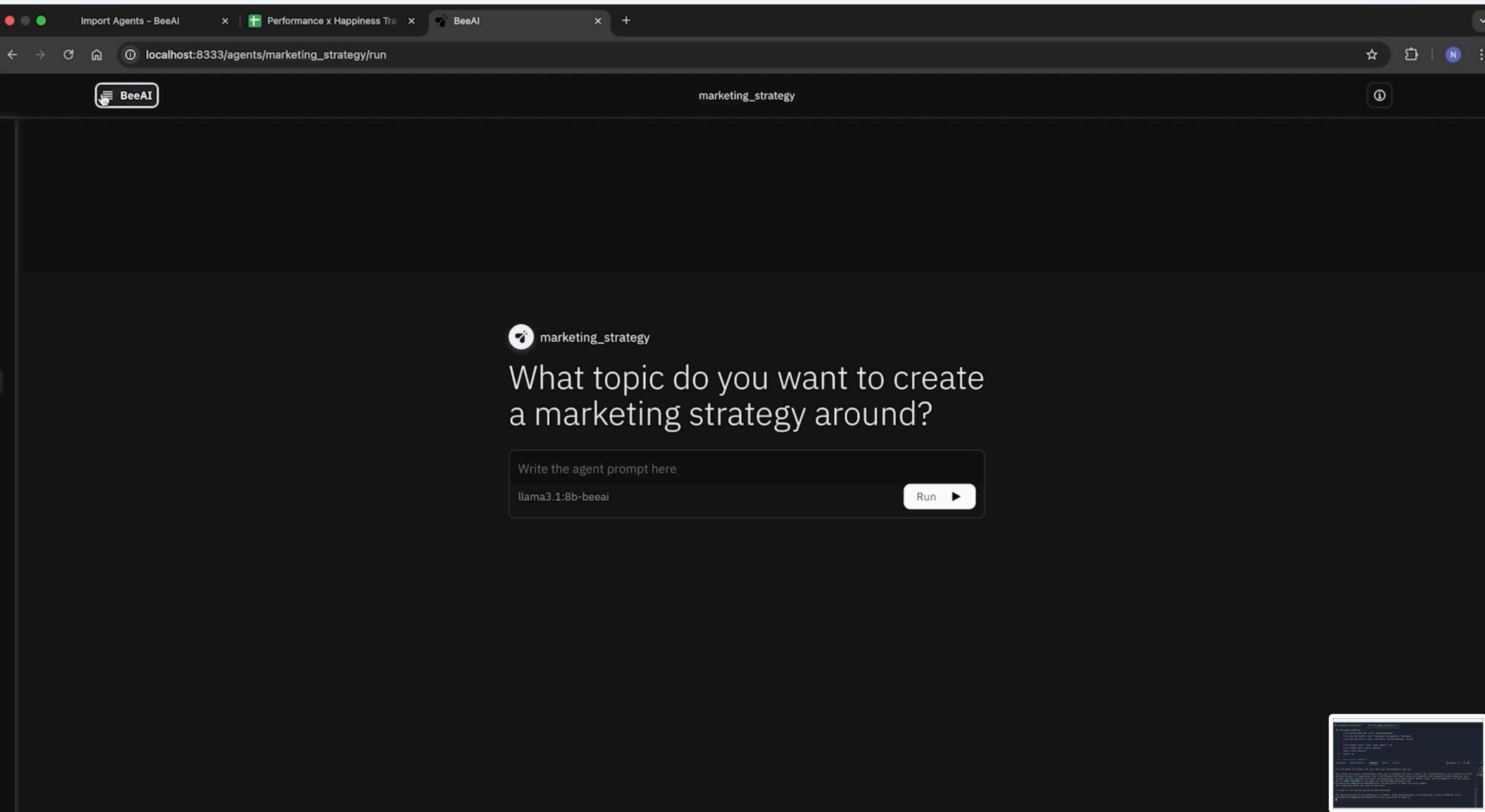
Yes, there are several cardiologists near you in Atlanta, GA. One of them is Dr. Sarah Mitchell, who is board-certified and has 15 years of experience. She is affiliated with Emory University Hospital and Piedmont Atlanta Hospital, and accepts several insurance providers including Blue Cross Blue Shield, Aetna, Cigna, and UnitedHealth. You can contact her at (404) 555-2847 or via email at s.mitchell@atlantaheart.com.

● nicholasrenotte@Nicholass-MacBook-Pro-2 my_acp_project % beeai run policy_agent
Got a question about your policy? Ask here!

>>> what is the waiting period on physiotherapy?

The waiting period on physiotherapy is 2 months. Group physiotherapy is covered with a limit of \$35 per visit.

○ nicholasrenotte@Nicholass-MacBook-Pro-2 my_acp_project % beeai ui



Agents

agent_docs_creator

aider

chat

doctor_agent

gpt_researcher

health_agent

literature_review

marketing_strategy

ollama_deep_researcher

podcast_creator

policy_agent



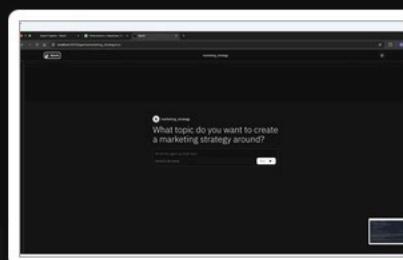
marketing_strategy

What topic do you want to create a marketing strategy around?

Write the agent prompt here

llama3.1:8b-beeai

Run



Import Agents - BeeAI

Performance x Happiness Tra

BeeAI

localhost:8333/agents/policy_agent/run

BeeAI

policy_agent

Agents

- agent_docs_creator
- aider
- chat
- doctor_agent
- gpt_researcher
- health_agent
- literature_review
- marketing_strategy
- ollama_deep_researcher
- podcast_creator
- policy_agent

policy_agent

Got a question about your policy? Ask here!

what is the waiting period on d

llama3.1:8b-beeai

Run ►

A screenshot of a web browser window showing the BeeAI platform. The URL in the address bar is 'localhost:8333/agents/policy_agent/run'. The title bar shows three tabs: 'Import Agents - BeeAI', 'Performance x Happiness Tra', and 'BeeAI'. The 'BeeAI' tab is active. The main content area has a dark background. On the left, there is a sidebar titled 'Agents' with a list of agent names: 'agent_docs_creator', 'aider', 'chat', 'doctor_agent', 'gpt_researcher', 'health_agent', 'literature_review', 'marketing_strategy', 'ollama_deep_researcher', 'podcast_creator', and 'policy_agent'. The 'policy_agent' item is highlighted with a dark gray background. The main content area has a title 'policy_agent' at the top right. Below it is a large text input field containing the placeholder 'Got a question about your policy? Ask here!'. A smaller text input field below it contains the text 'what is the waiting period on d'. To the right of this is a button labeled 'Run' with a right-pointing arrow icon. At the bottom left of the main content area is the text 'llama3.1:8b-beeai'. The bottom left corner of the entire interface has a small gear icon.

Import Agents - BeeAI

Performance x Happiness Tra

BeeAI

localhost:8333/agents/policy_agent/run

BeeAI

policy_agent

policy_agent

Task input:

what is the waiting period on rehabilitation?

The waiting period for rehabilitation, which falls under Hospital Substitution Programs, is generally 2 months of continuous cover. However, if the rehabilitation is related to a pre-existing condition, the waiting period is 12 months. It's also important to note that a referral from a treating doctor is required to be eligible for these programs. Additionally, continuity of cover is provided for individuals transferring from another registered Australian health fund or changing from another RBHS product, provided that an equivalent or higher level of cover was held and a Transfer Certificate is provided.

Task completed in 7s