



PyCon China 2025

张晋涛 - Kong Inc.
2025 年 9 月 20 日





AI Gateway

在 AI 应用中是否有价值



个人介绍

- Kong Inc.
- Microsoft MVP
- CNCF Ambassador
- GitHub: <https://github.com/tao12345666333>
- X: <https://x.com/zhangjintao9020>



```
curl https://api.openai.com/v1/responses \  
  -H "Content-Type: application/json" \  
  -H "Authorization: Bearer $OPENAI_API_KEY" \  
  -d '{  
    "model": "gpt-5",  
    "input": "Write a short bedtime story about a unicorn."  
  }'
```



AI 应用



```
from openai import OpenAI  
client = OpenAI()  
  
response = client.responses.create(  
    model="gpt-5",  
    input="Write a short bedtime story about a unicorn."  
)  
  
print(response.output_text)
```



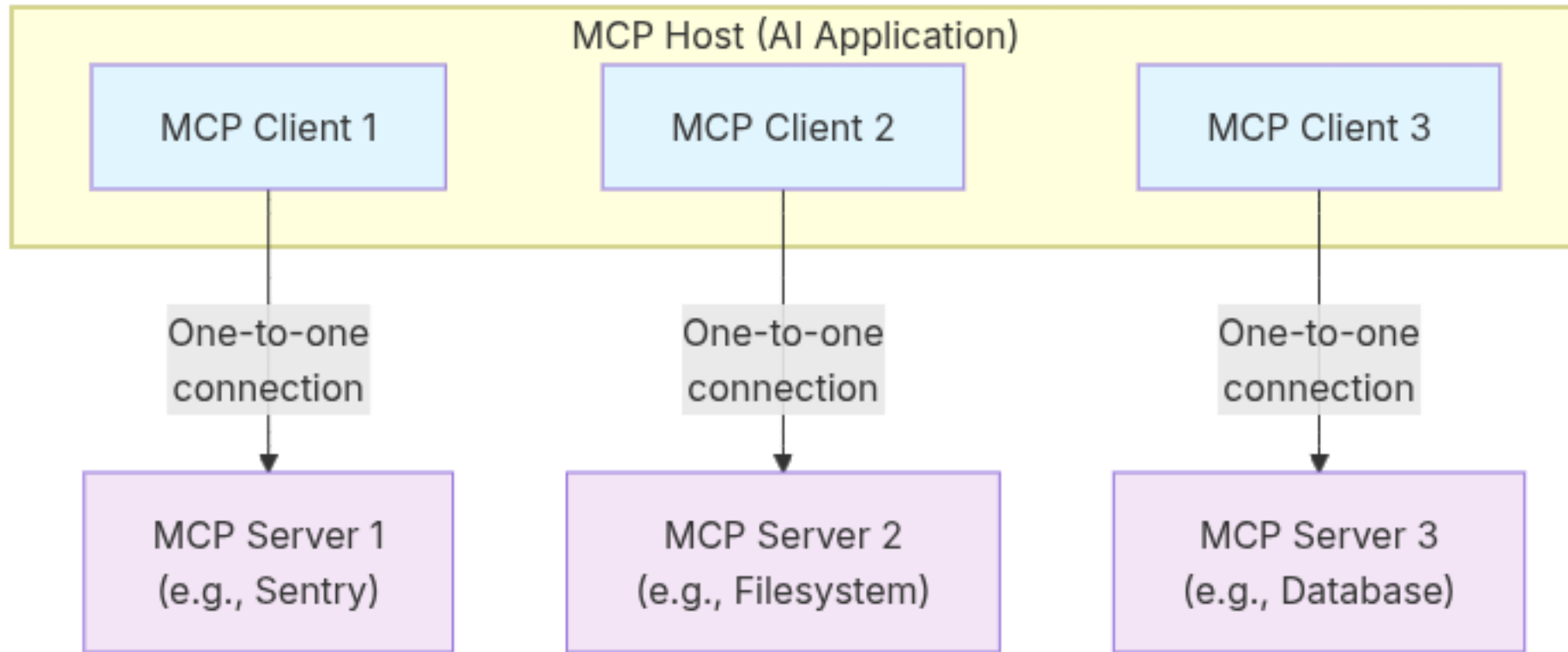
```

36 def run(self) -> int:
37     conversation: List[Message] = []
38     print("Chat with GPT (use 'ctrl-c' to quit)")
39
40     while True:
41         user_input, ok = self.get_user_message()
42         if not ok:
43             break
44
45         # Append user message to the conversation
46         conversation.append(
47             {"role": "user", "content": [{"type": "text", "text": user_input}]}
48         )
49
50         # Inference
51         assistant_text = self.run_inference(conversation)
52
53         # Append assistant reply to the conversation
54         conversation.append(
55             {"role": "assistant", "content": [{"type": "text", "text": assistant_text}]}
56         )
57
58         # Print assistant's text (yellow)
59         if assistant_text:
60             print(f"\u001b[93mGPT\u001b[0m: {assistant_text}")
61
62     return 0
63
64 def main() -> int:
65     try:
66         client = OpenAI() # Reads OPENAI_API_KEY from environment
67         agent = Agent(client, model="gpt-5", max_tokens=1024)
68         return agent.run()
69     except Exception as e:
70         print(f"Error: {e}", file=sys.stderr)
71         return 1
72
73

```



MCP



Apps & Agents



AI Gateway

Support for all Kong plugins and Konnect capabilities, plus:

AI Governance	AI Observability	AI Security
AI Credentials Store	AI Traffic Control	AI Load Balancing
Multi-LLM Routing	MCP Gateway	LLM Catalog

LLM Gateway

 Multi-LLM	 Guardrails	 AI Observability
 Semantic Routing	 Semantic Caching	 Auto-RAG
 PII Removal	 Semantic Caching	+ More

MCP Gateway

MCP AuthN/Z	MCP Routing	Autogeneration
MCP Analytics	Semantic tools	+ More

APIs

Service Meshes

Microservices

Service Meshes

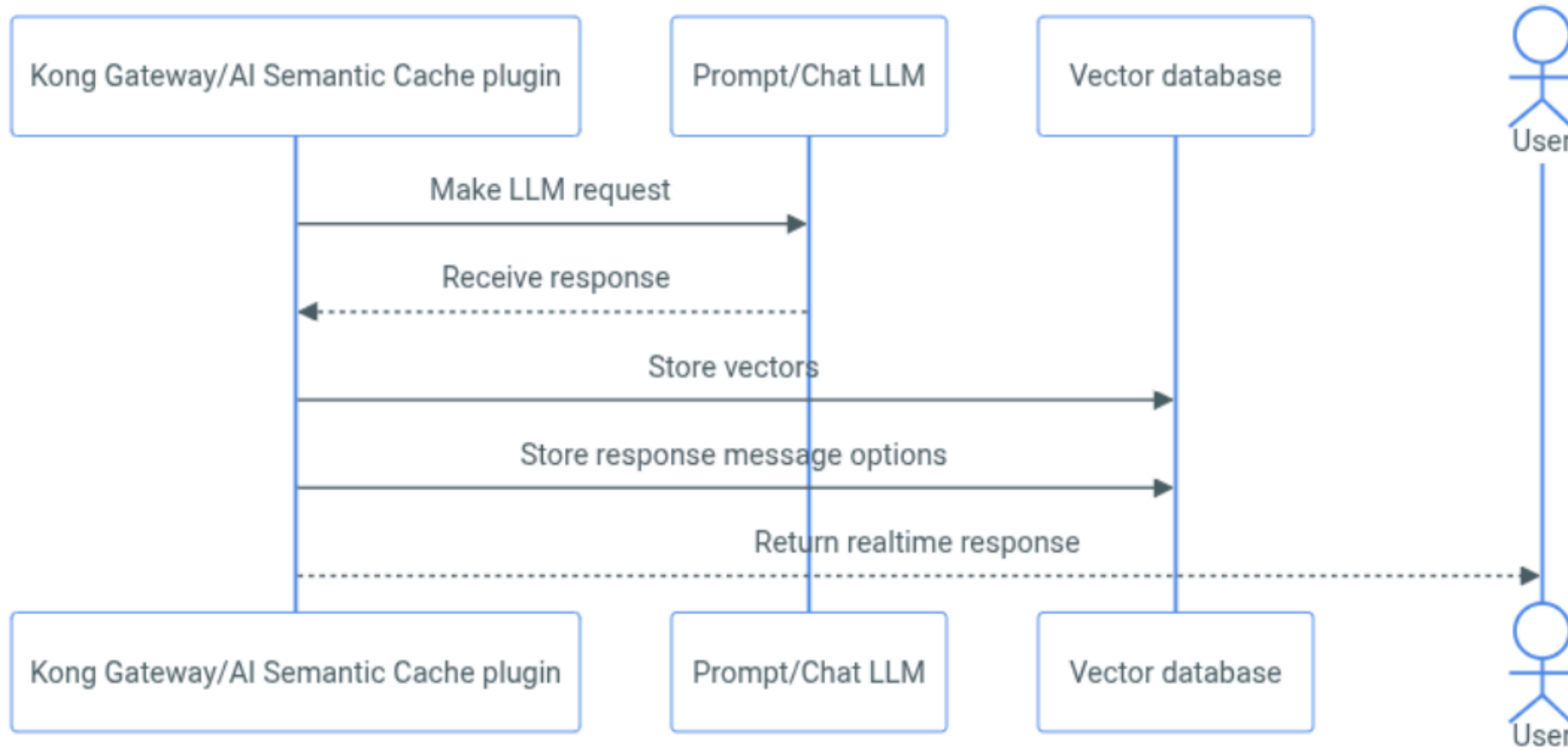
MCP Servers

Events

LLMs

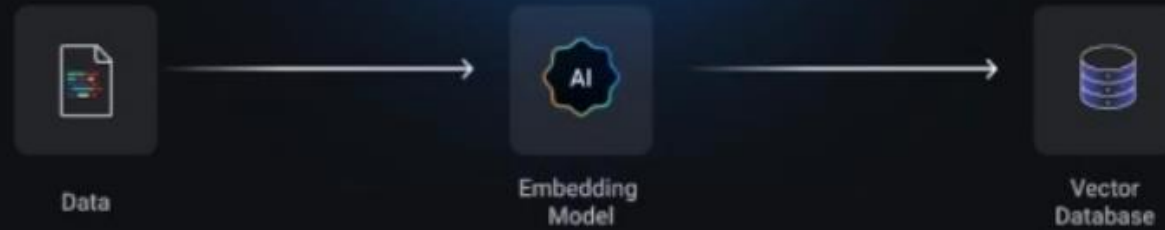
 Cohere	 Anthropic	 LLaMA
 OpenAI	 Azure	 Mistral
 AWS Bedrock	 Vertex	 Hugging Face
 Databricks	 Grok	 More

Semantic Cache

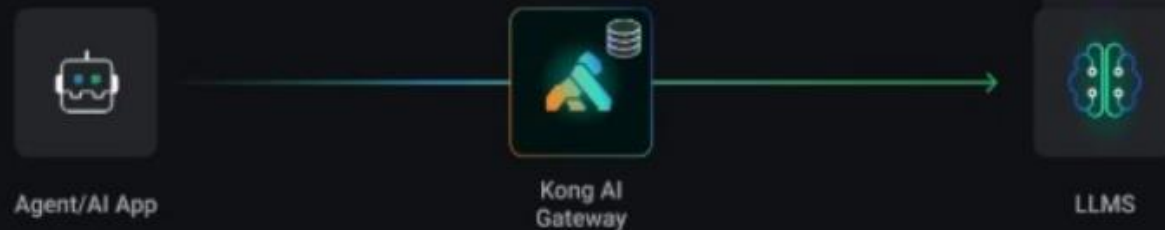


Automated RAG with Kong

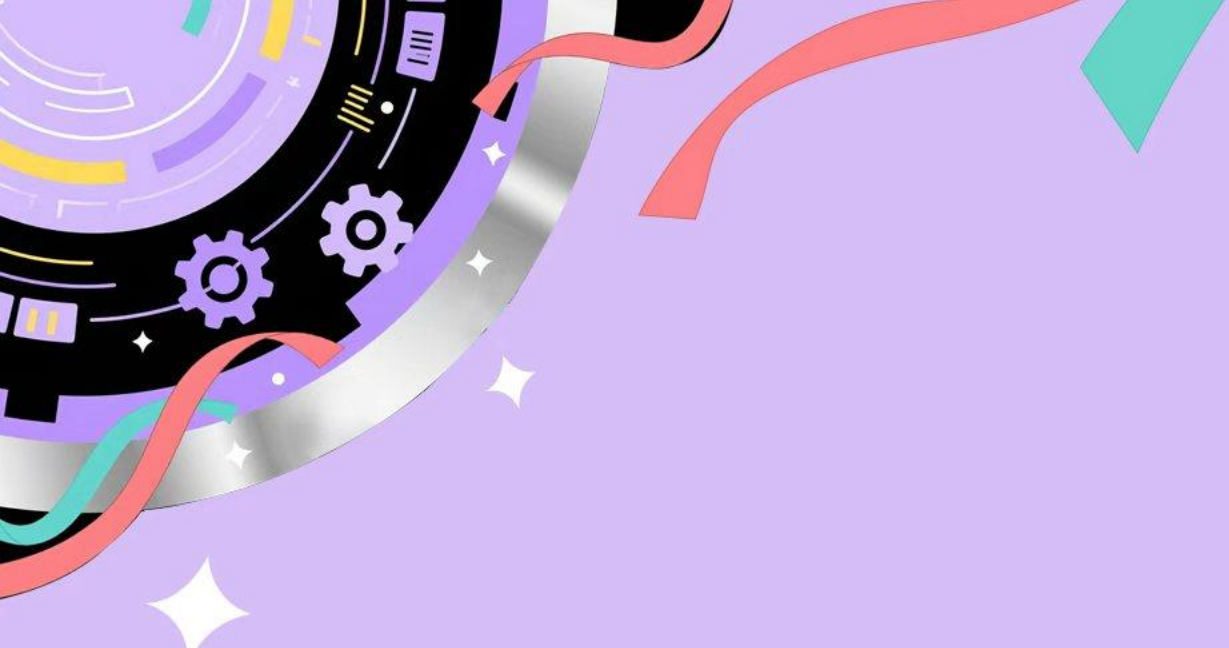
1. Processing Data



2. Associating relevant data to prompt



- ✓ Lower LLM hallucinations
- ✓ Improve dev productivity
- ✓ Grow secure



Thanks!

