

# CIE-SF 46th Annual Conference



## Orchestrating Intelligence: Building Effective AI Agents with AG2

**Qingyun Wu**  
CEO & Founder  
AG2 (formerly AutoGen)

Assistant Professor  
Penn State University

# Build Effective AI Agents with



Qingyun Wu, AG2, 06/2025

[qingyun@ag2.ai](mailto:qingyun@ag2.ai)

434-466-4925



Source: Google Trends



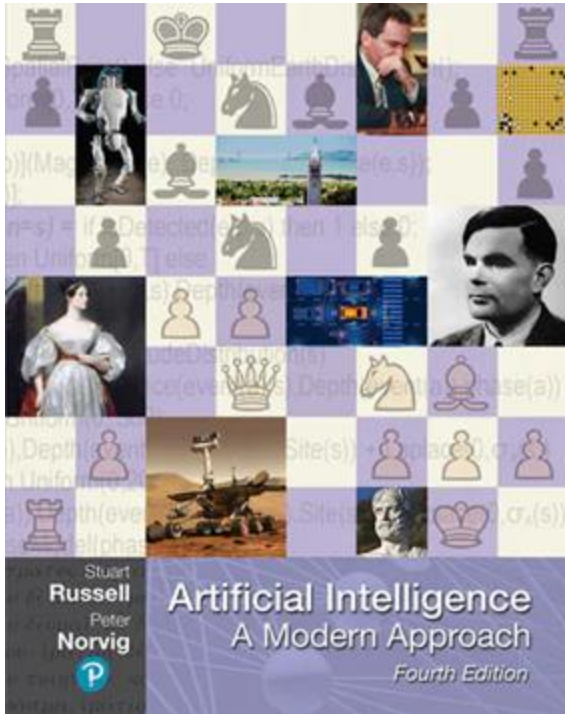
# Agenda



The WHAT, WHY, and HOW

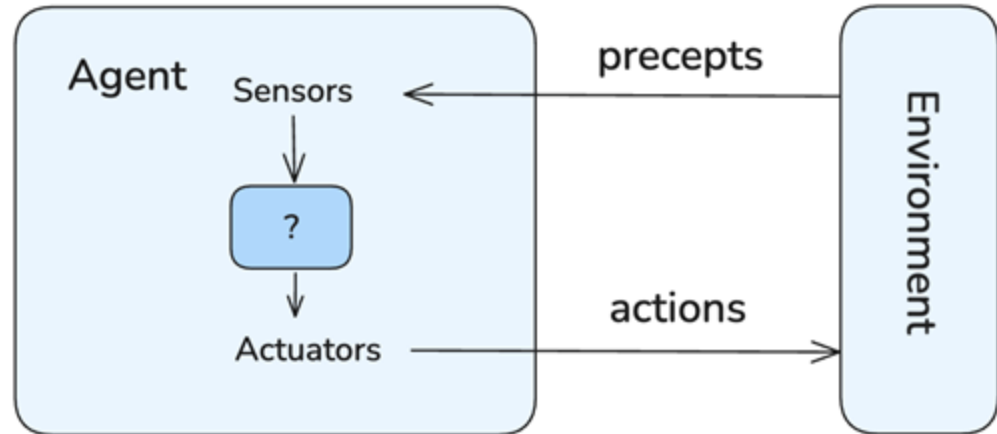


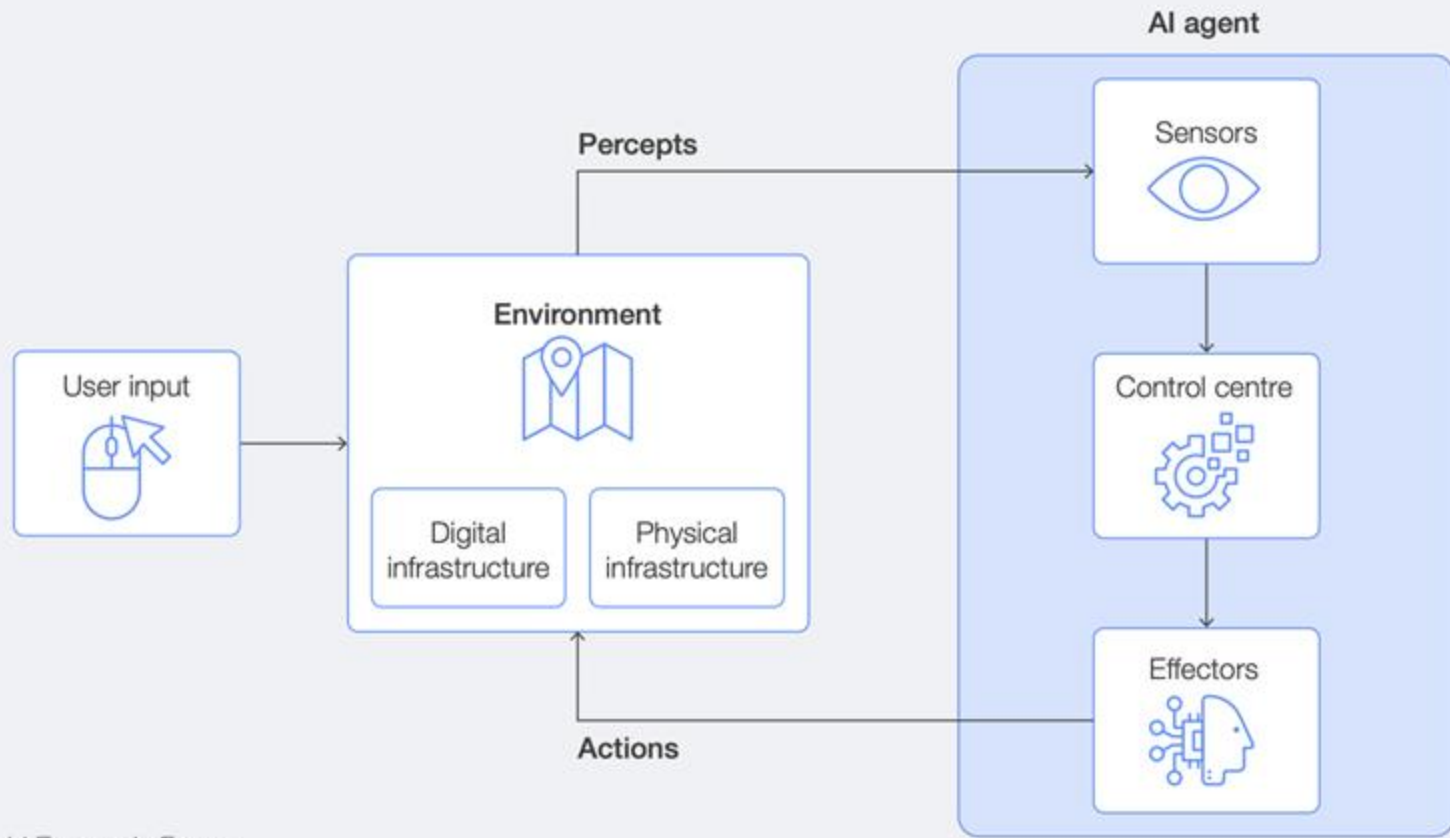
The WHAT



An **agent** is anything that can be viewed as **perceiving** its environment through sensors and **acting** upon that environment through actuators.

— Stuart Russell and Peter Norvig, in Chapter 2.1





Source: World Economic Forum



The WHY



... because Bill Gates and Andrew Ng Said So!!!



(Angel Investor of AG2)



What Future AI Applications Look Like?



# What Future AI Applications Look Like?

GenAI-Intensive, LLM Inference-Intensive

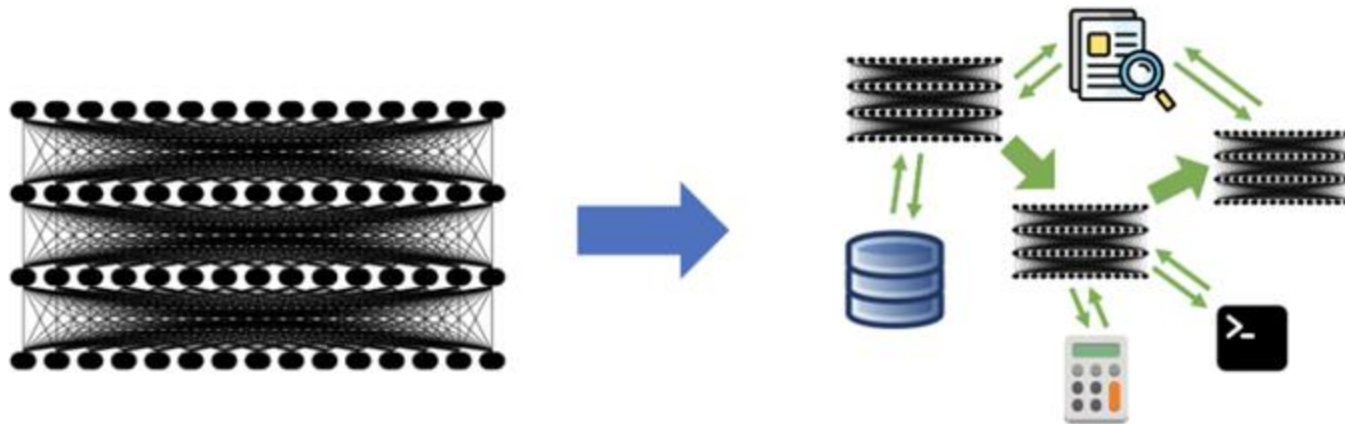


Data-Intensive



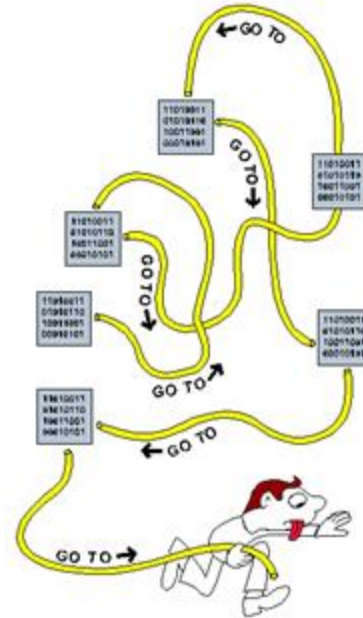
Compute-Intensive

# The Shift from Models to Compound AI Systems



*Increasingly many new AI results are from compound systems.*

# Complex Static Logics Compounded with the Dynamic Nature of LLMs



# The Dawning Age of Agents

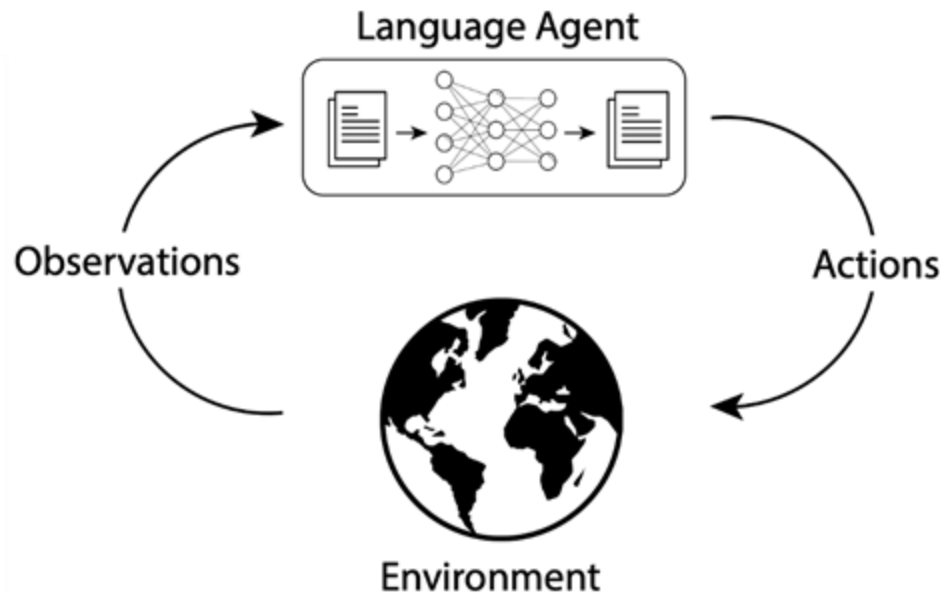
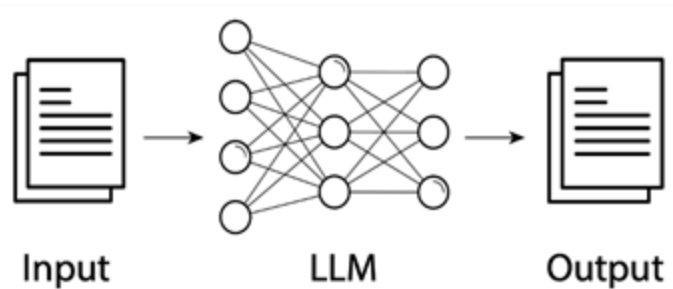
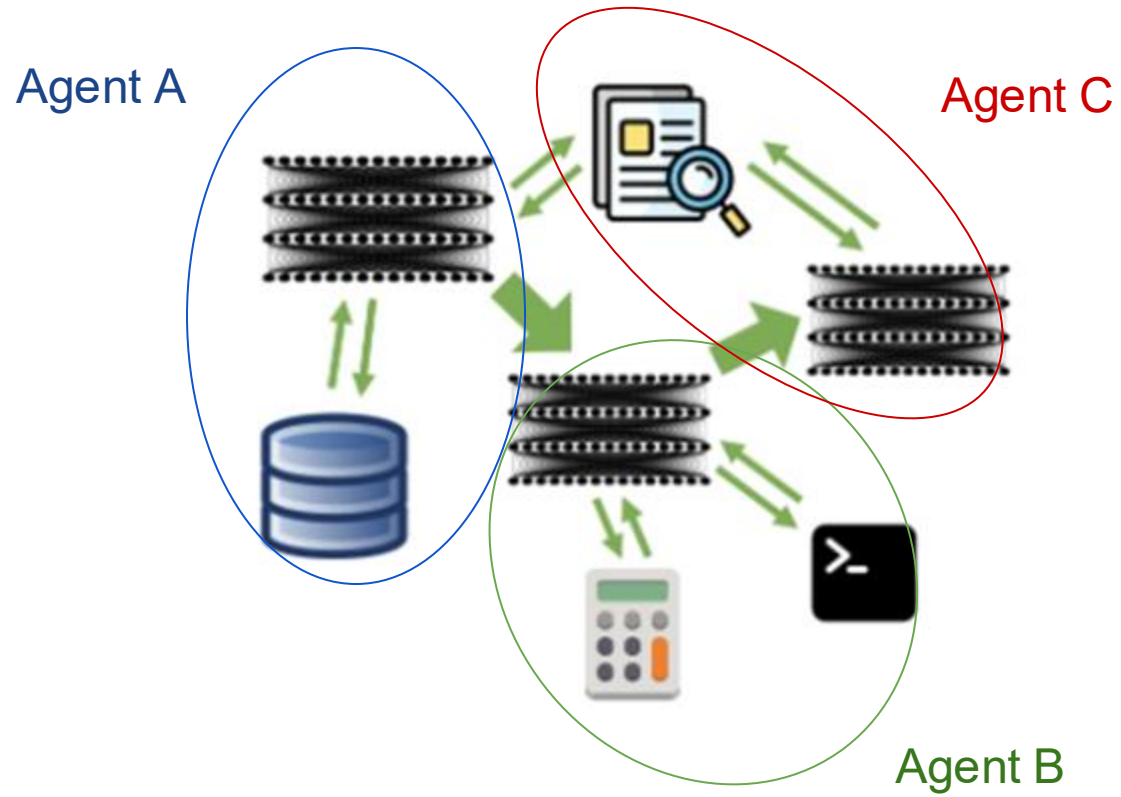


Figure source: Summers, T. R., Yao, S., Narasimhan, K., & Griffiths, T. L. (2023). Cognitive architectures for language agents. *arXiv preprint arXiv:2309.02427*.



The HOW







To The Rescue

AG



AG2

### AutoGen: Enabling Next-Gen LLM Applications via Multi-Agent Conversation Framework

Qiyao Wu\*, Qipeng Shao\*, Jinyi Zhang\*, Yuxi Wu\*, Shaojun Zhang\*,  
Xinrui Zou\*, Botao Li\*, Li Ding\*, Xinyuan Zhang\*, and Qian Wang\*

\*Phosphorus State University

\*Microsoft

\*University of Washington



Figure 1. Overview of the AutoGen framework. The framework enables multi-agent conversations using multi-agent conversational LLMs. Multiple agents are interconnected and can be used to solve tasks, generate and refine a code snippet, or write a document. Agents can interact with each other in a flexible way. (Bottom-right) The framework supports agent collaboration and flexible conversation patterns.

**08.2023:**  
Research paper

**03.2023:** Initial Prototype  
- Flexible multi-agent  
conversation framework  
- Code/function execution

**10.2023:** Top  
trending on  
GitHub

**12.2023:**  
5 favorite AI  
papers by The  
Sequence

**Top 100**  
Open source  
achievements

**5K**  
Forks

**37K**  
Stars

Forbes, The  
Economist,  
WIRED...

**20K**  
@Discord

**600K**  
Downloads  
/month

**Top 1**  
SWE-Bench Lite

Best Paper  
at ICLR 2024  
LLM Agents  
Workshop

**COLM**

In our Data Science department  
AutoGen is helping us develop  
a production ready  
multi-agents framework

Sam  
Khalil

How  
**BETTERFUTURELABS**  
Uses AutoGen

Justin Tappan  
Co-founder & CEO of BetterFuture Labs

emerge

Multi-Agent AI Enables Emergent  
Cognition and Real-Time Knowledge  
Synthesis in Science and Engineering

Andrew Ng | 100  
University of Amsterdam, AI, Managing Director of AI...

Best Agents AI short course AI Agents: Design Patterns with AutoGen\*,  
taught by Microsoft's Qiyao Wu and Phyllis Kollar's Qiyao Wu, shows  
you how to use AutoGen to implement agents design patterns like  
multi-agent collaboration, interdependent and nested if/else, self-reflection, tool  
use, and planning. Learn how to build and combine multiple specialized  
agents - such as researchers, planners, coders, writers, and artists -  
that interact to generate complex workflows, like generating financial  
reports, that would otherwise have taken extensive manual  
effort.

This course starts with key agents design strategies with many fun  
exercises. For example, you'll build a conversational chess game  
using two agent agents, each of which use a tool to generate moves  
and update the board state, while engaging in busy banter about the  
game!

For anyone using AutoGen, and those who will use AutoGen to get  
started now: <https://github.com/microsoft/autogen>

For anyone using AutoGen, and those who will use AutoGen to get  
started now: <https://github.com/microsoft/autogen>

PyPI downloads 686k/month

pypi package 0.9.1

AG2 (formerly AutoGen)

21813 MEMBERS



## Contributors Wall



tylersuard 3/23/25, 8:44 PM

Hey Autogen/AG2 Community,

I've spent the last year and a half building an enterprise-level RAG system for a Fortune 500 manufacturer - 50 million records, 12 databases, and 100K+ PDFs, all answered in 10-30 seconds using Autogen to manage concurrency, agent logic, and question rewriting. This project was such a success that I ended up writing a book about it, which covers everything from data ingestion to orchestrating agents at scale. If you're interested in how Autogen fits into a real-world, large-scale RAG pipeline, or just want to see code samples and best practices, I'd love to share details. Feel free to ask questions here or check out my book in Manning.com's Early Access on March 27th!



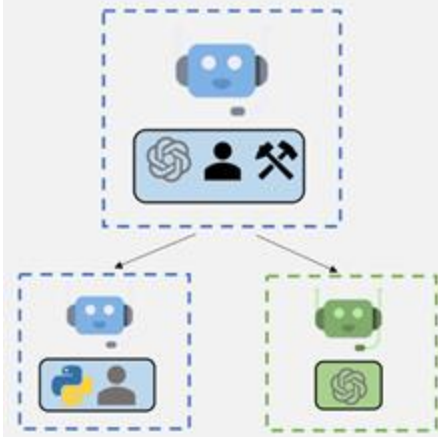
# AG2's Perspective on What's Essential in Supporting AI Agent Development



Agentic Abstraction

Multi-Agent Orchestration

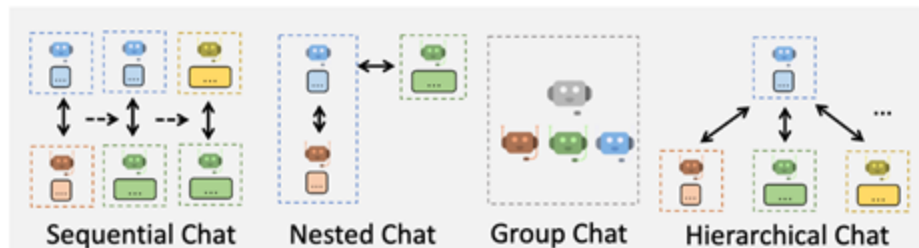
Conversable agent



Agent Customization



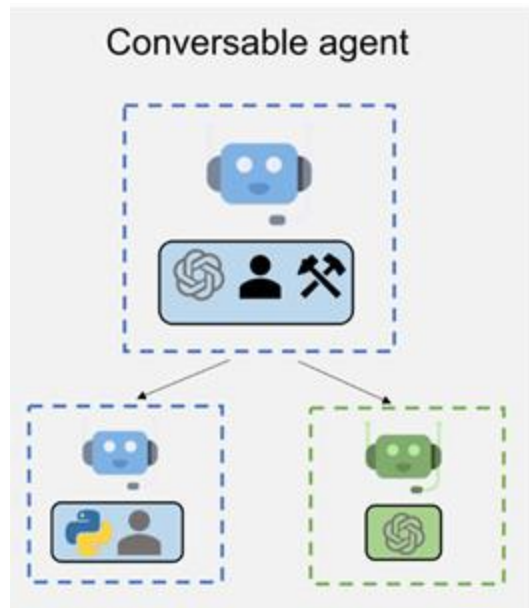
Multi-Agent Conversations



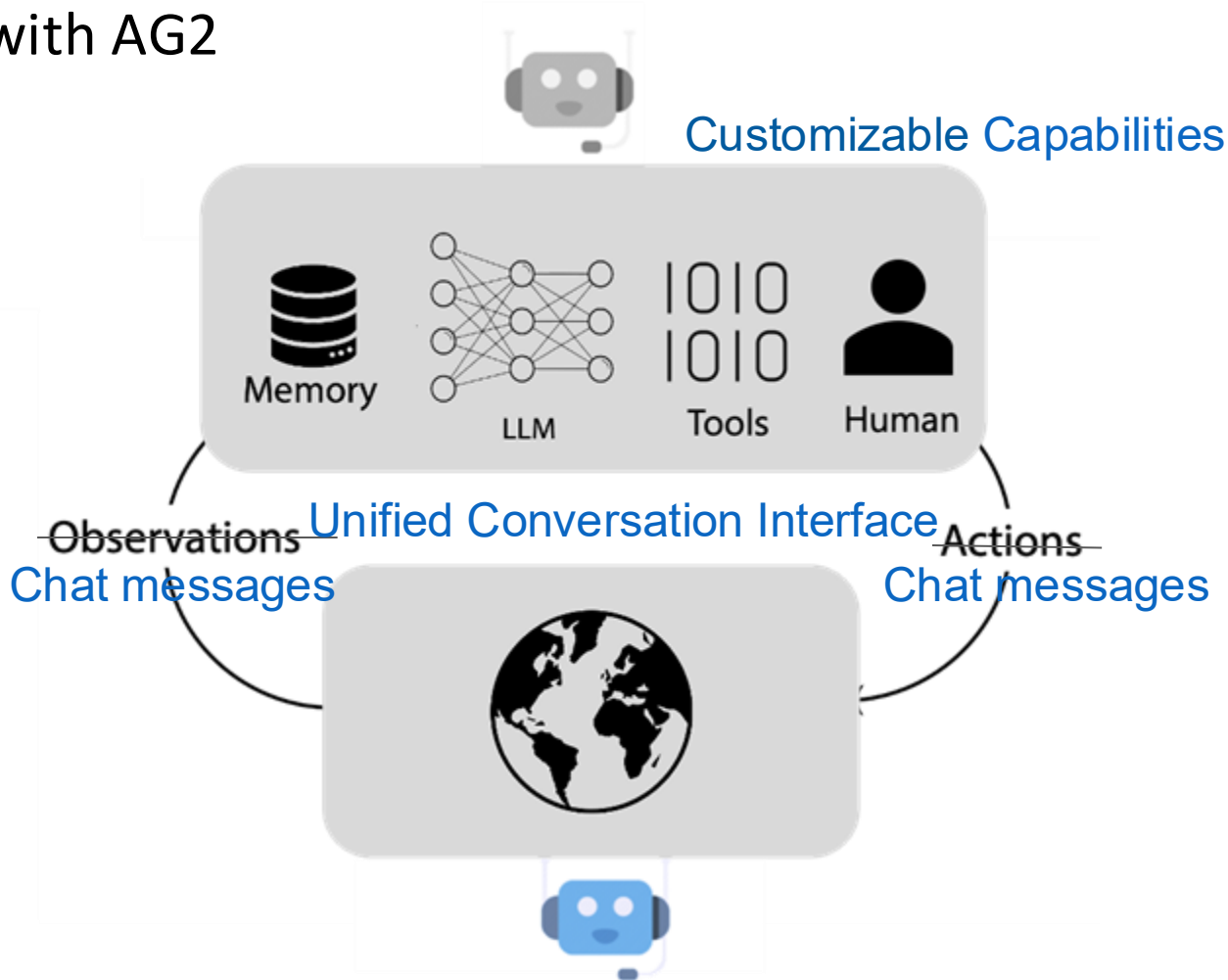
Flexible Conversation Patterns

# Build Effective Agents with AG2

**Step 1.** Define AG2 agents:  
Conversable & Customizable



Agent Customization





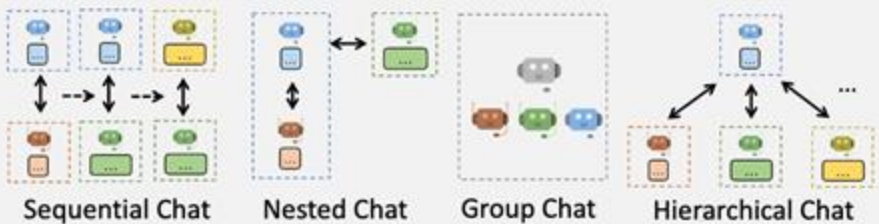
# Build Effective Agents with AG2



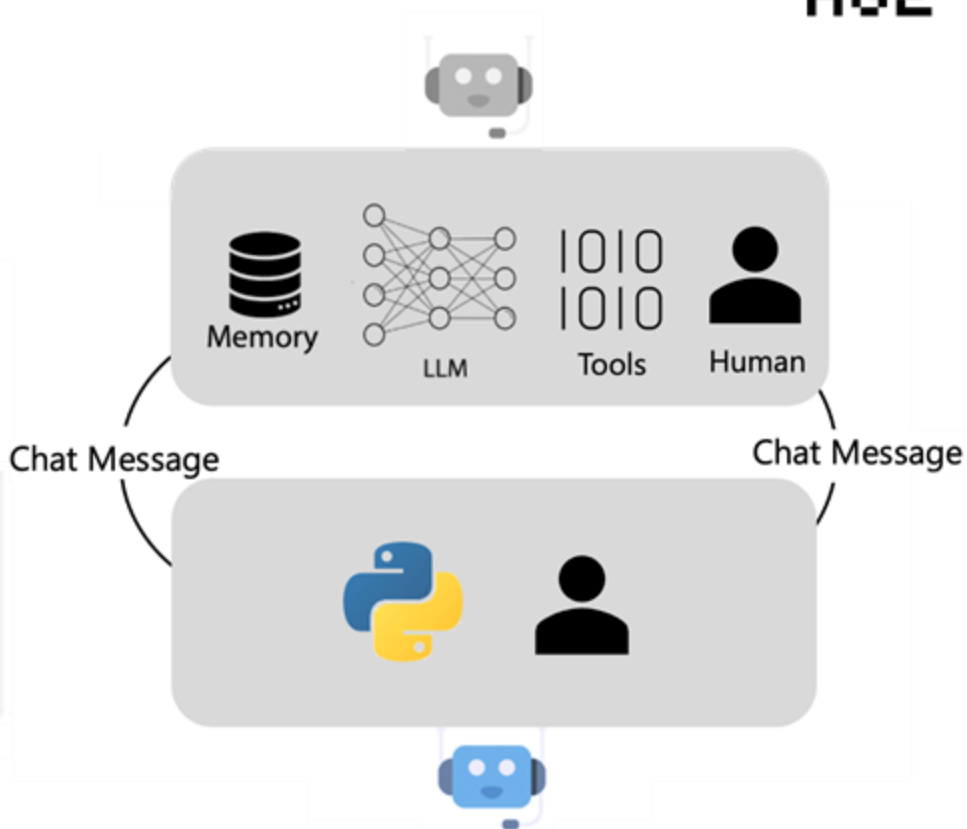
## Step 2. Get them to talk: Conversation Programming



Multi-Agent Conversations



Flexible Conversation Patterns





# Agent: ConversableAgent - The Basics

## CONVERSABLEAGENT

ConversableAgent is at the heart of all AG2 agents while also being a fully functioning agent.

Let's *converse* with ConversableAgent in just 5 simple steps.

```
# 1. Import our agent class
from autogen import ConversableAgent

# 2. Define our LLM configuration for OpenAI's GPT-4o mini
# Put your key in the OPENAI_API_KEY environment variable
llm_config = {"api_type": "openai", "model": "gpt-4o-mini"}

# 3. Create our agent
my_agent = ConversableAgent(
    name="helpful_agent",
    llm_config=llm_config,
    system_message="You are a poetic AI assistant, respond in rhyme.",
)

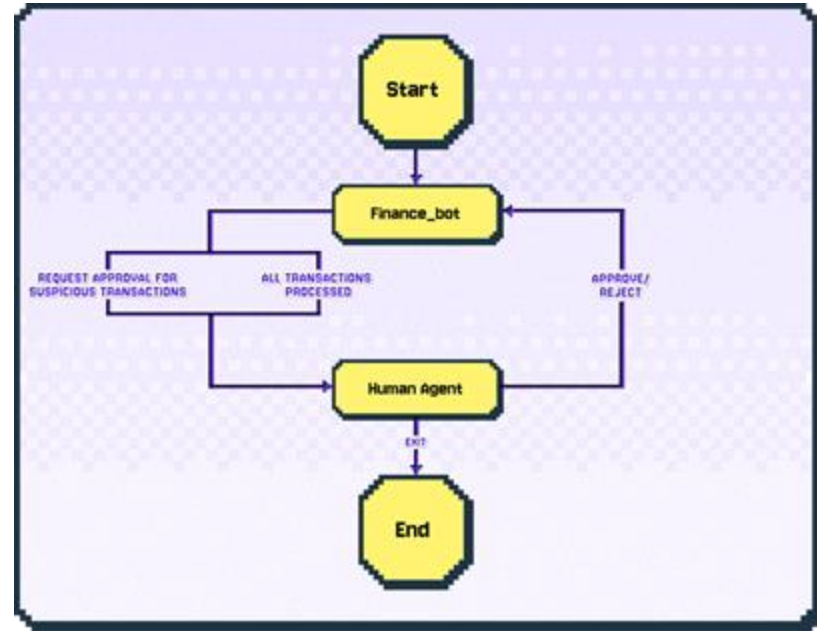
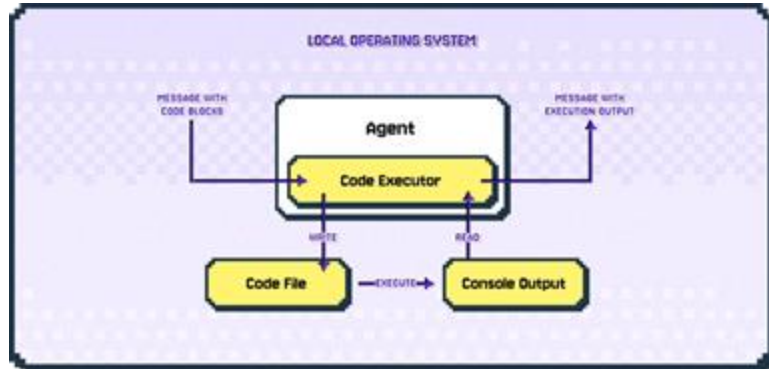
# 4. Chat directly with our agent
my_agent.run("In one sentence, what's the big deal about AI?")
```

- LLM-Driven Dynamic Multi-Turn Chat
- Built-in Code Execution
- Built-in Tool Usage
- Built-in Human-in-the-Loop

[ConversableAgent](#)

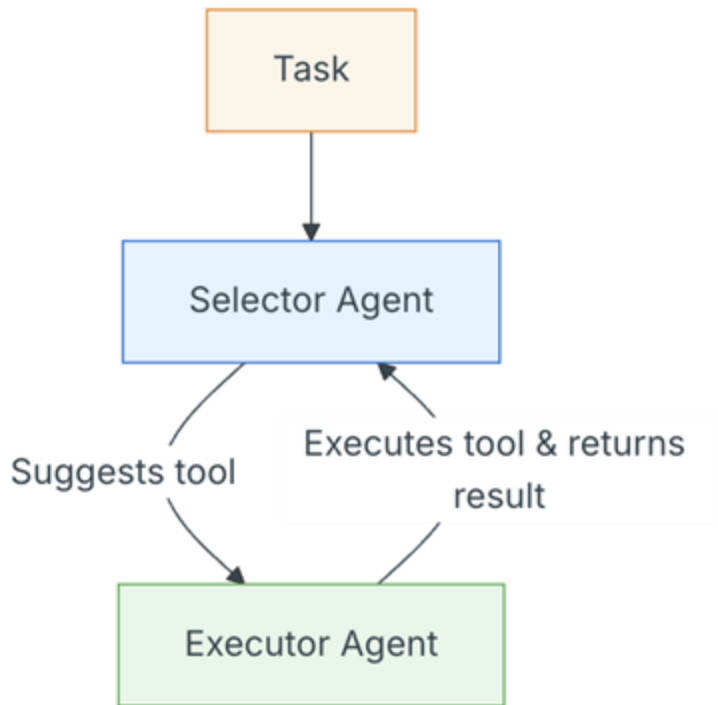
## Built-in Human-In-The-Loop Support

### Built-in Code Execution





## Sophisticated Tool Usage



Search for the latest news on AG2  
AI

# Reference Agents

## Reference Agents



CaptainAgent

Communication Platforms >

DocAgent

DocAgent Performance

ReasoningAgent

DeepResearchAgent

WebSurferAgent

WikipediaAgent



Captain Agent



Real-time Voice Agent



Doc Agent



Web Surfing & Deep Research Agent

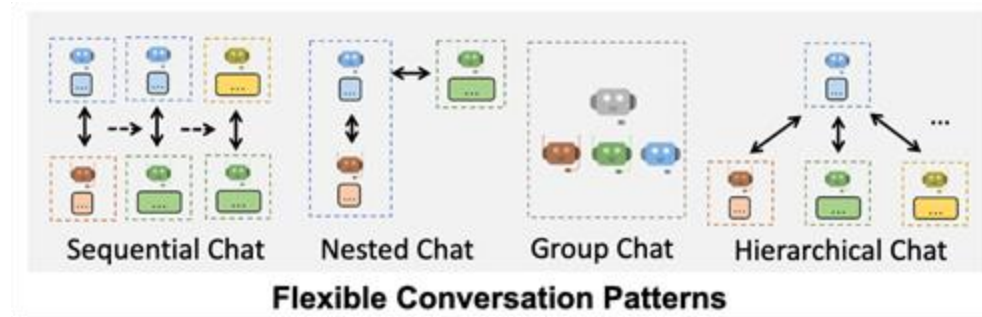


Reasoning Agent



Comms Agent

# Agentic Workflow // Multi-Agent Orchestration // Conversation Programming



## Overview

Many hands make for light work and orchestrating workflows containing many agents is a strength of the AG2 framework.

## Super Power: Composability

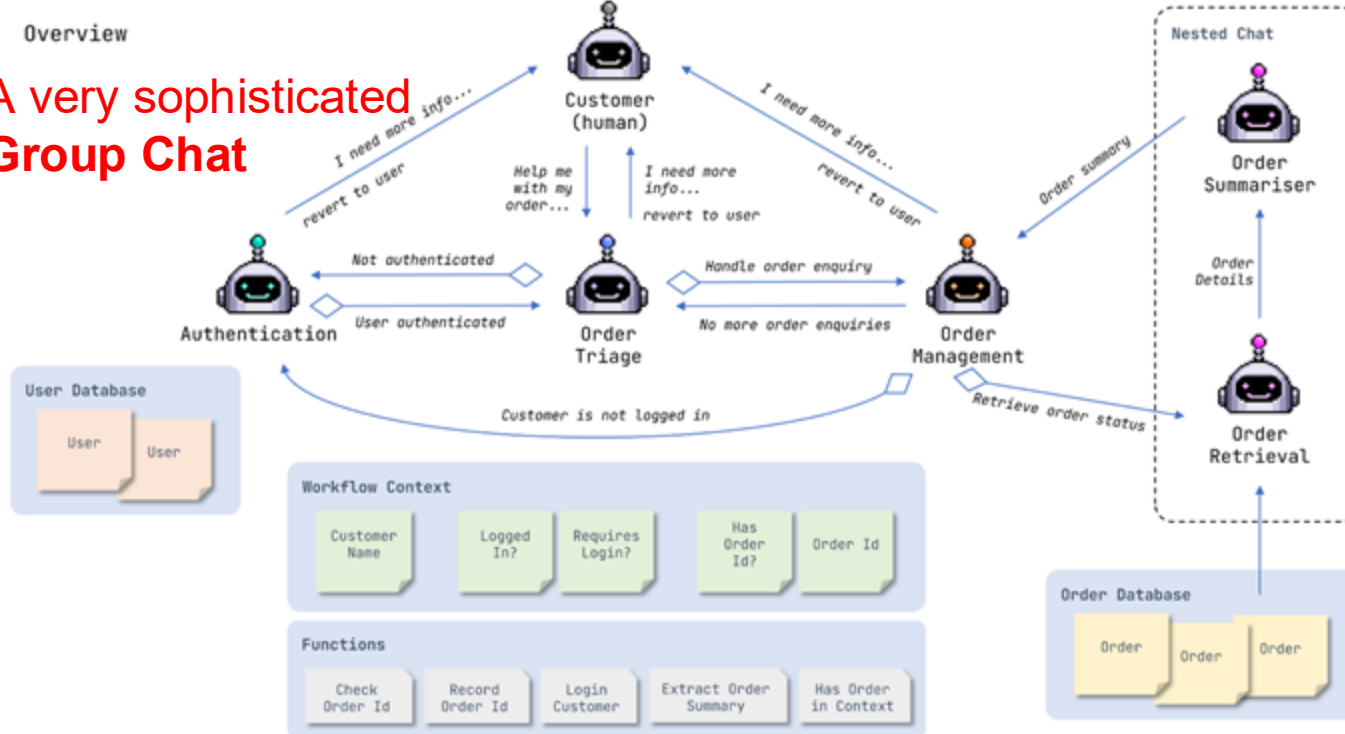
1. **Two-agent chat:** The simplest form of conversation pattern where two agents chat back-and-forth with each other.
2. **Sequential chat:** A sequence of chats, each between two agents, chained together by a carryover mechanism (which brings the summary of the previous chat to the context of the next chat). Useful for simple sequential workflows.
3. **Group chat:** A chat with more than two agents with options on how agents are selected.

# Use Case Example - Customer Service

## Customer Service

Overview

A very sophisticated  
**Group Chat**



**Nested Chat**



## Example Use Cases








emergence

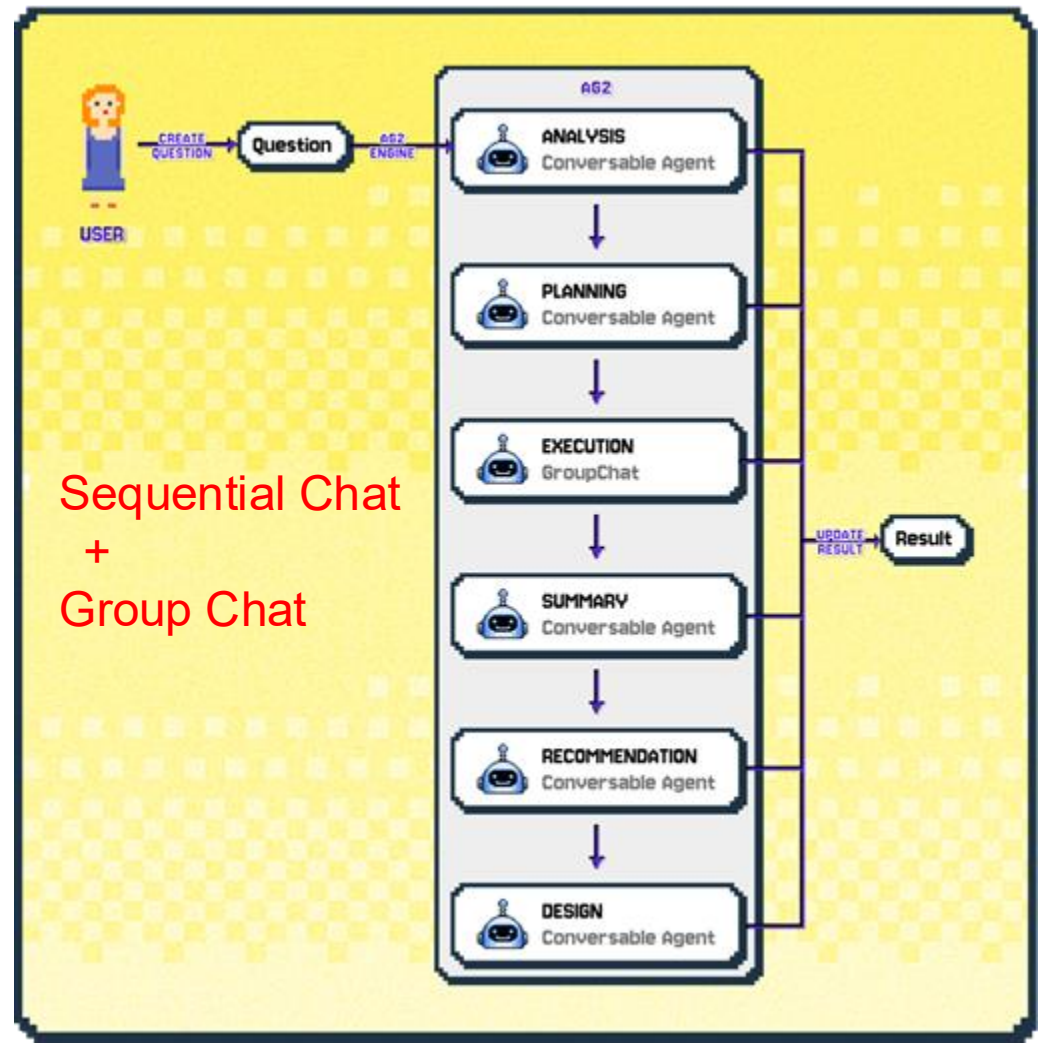




### Financial Services Related Use Cases:

-  Processing expenses
-  Paying invoices
-  Chasing customer debts
-  Collecting employee payroll details, analyzing financial KPIs
-  Generating reports in various formats

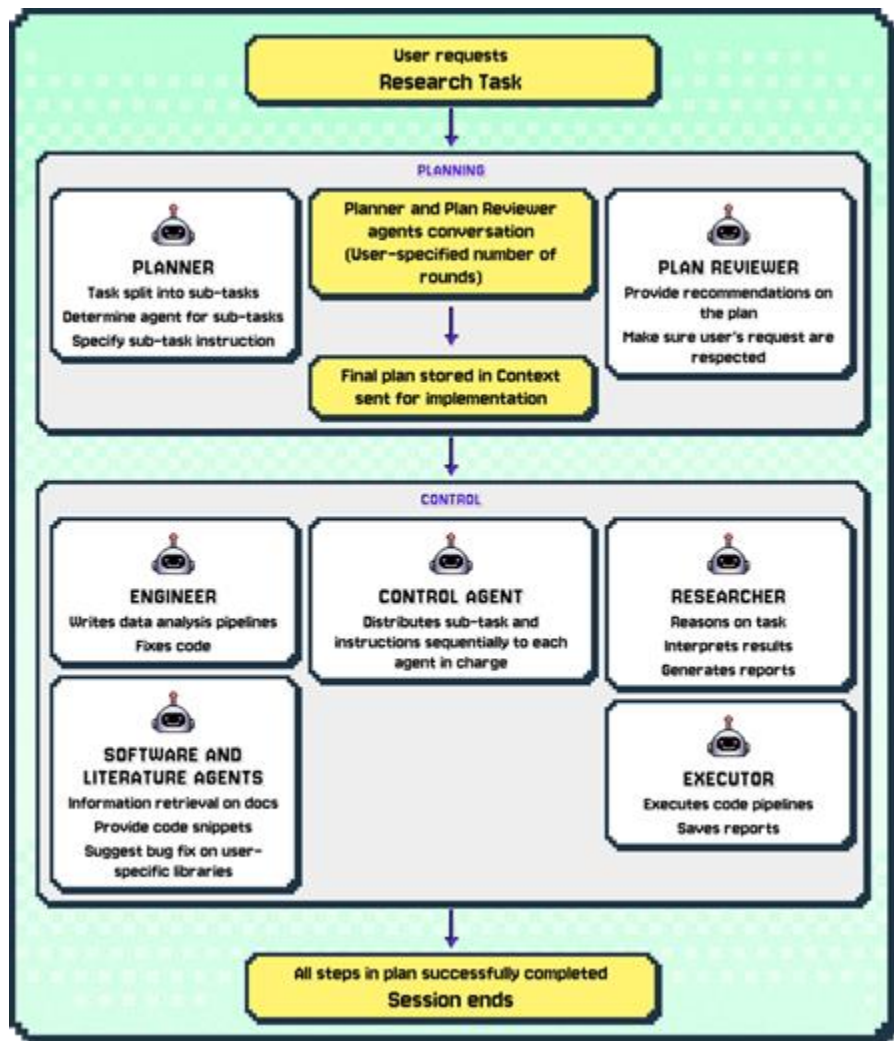
Sequential Chat  
+  
Group Chat





## Features That Helped:

- 🤖 Tool Calls
- 🤖 Multi-LLM Calls
- 🤖 Group Chat
- 🤖 Structured Output
- 🤖 Agentic RAG







Dynamic content generation with **high accuracy** and **following brand guideline**

**walmart.com** scale

**Key Designs:**

- A planner agent as the group chat manager
- Agent selection policy: which ones are most likely to disagree with each other & sequence the debate
- A stepwise plan

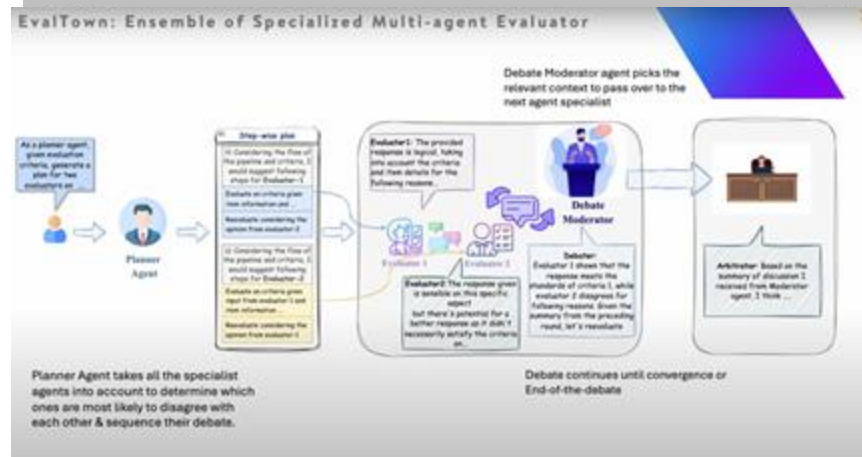
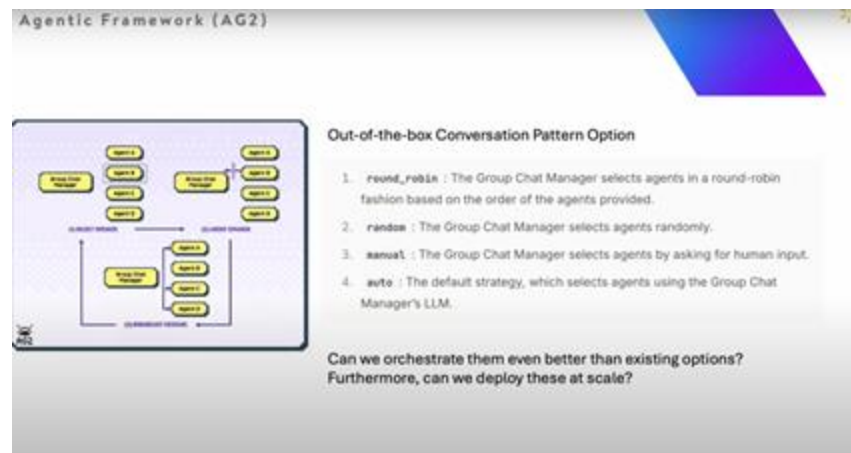
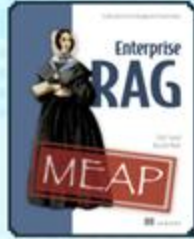


Figure Sources: Screenshots of Google Cloud Next 2025 Official Youtube Videos





×



🤖 Fortune 500 RAG Chatbot Scales to 50M+ Records in Under 30 Seconds

🤖 Handling thousands of product queries a day across 12 databases used to mean 5-minute wait times and brittle infrastructure.

🤖 Avoids hallucinations by **rewriting queries**, streaming **structured outputs**, and running **parallel database calls**



Build Together with The AG2 Community

# Recent Updates: OSS Weekly Releases



3 days ago

 marklysize

 v0.9.2

 ce3531f 

Compare 

v0.9.2

Latest

## Highlights

- 🛠️ **ReliableTool** - Ensure your tools do what you need them to do!
  - 📄 [Documentation](#)
  - 📓 Notebook examples: [Basic](#), [Group Chat](#), [Google Search](#)
- 🧠 MCP Examples: [arXiv](#), [file system](#), [Wikipedia](#)
- 📖 Documentation and notebook corrections and updates
- 🐛 Bug fixes

❤️ Thanks to all the contributors and collaborators that helped make the release happen!





# Thank You



*AG2.AI*



Example Apps Built  
with AG2



*AG2 Info Agent*