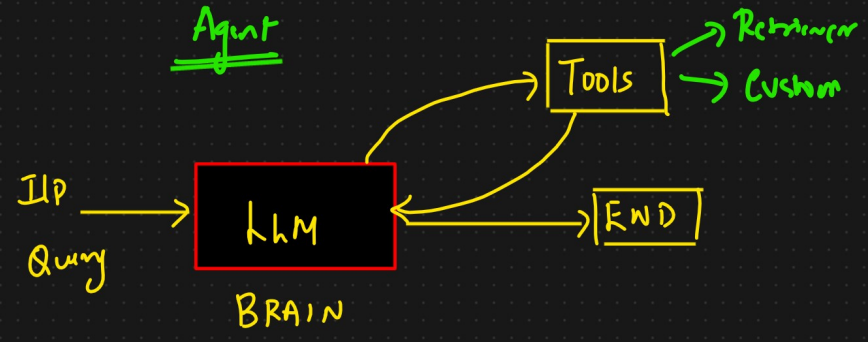
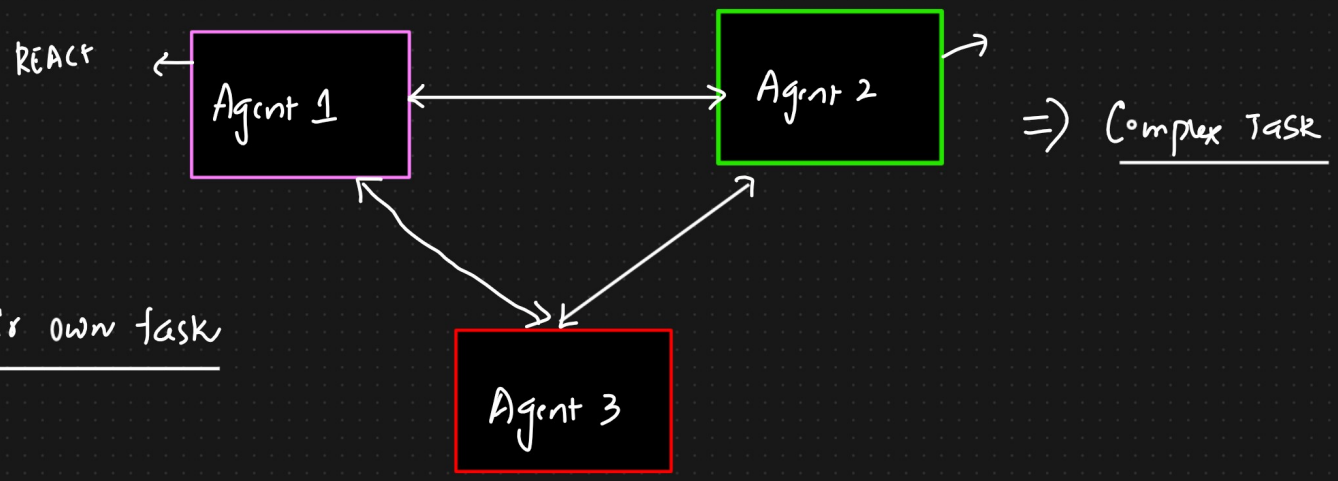
Multi Agent RAG Systems

1) COT
2) ReAct Agent Architecture
3) Self Reflection
4) Query Planning And Decomposition

} Autonomous RAG.



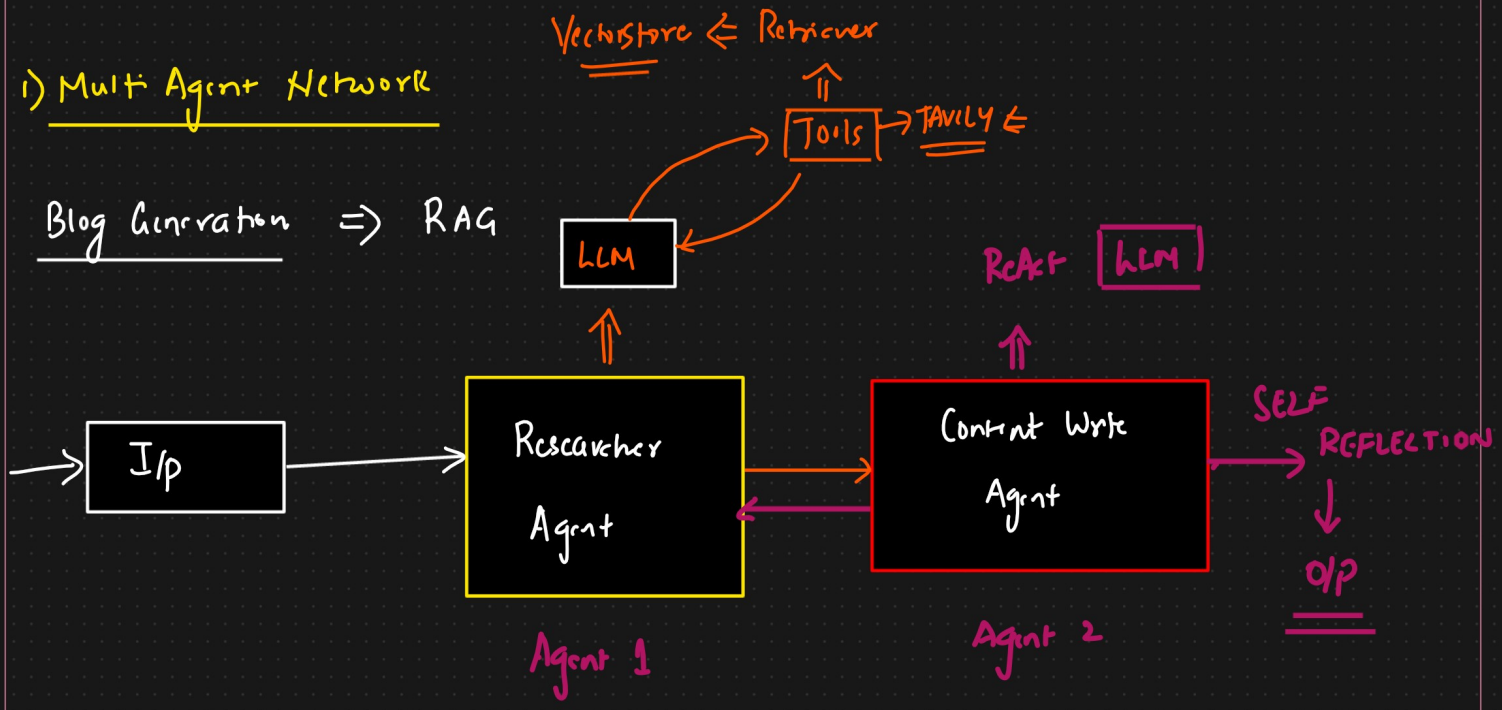


⇒ Complex Task

Their own task

1) Multi Agent Network

Blog Generation ⇒ RAG



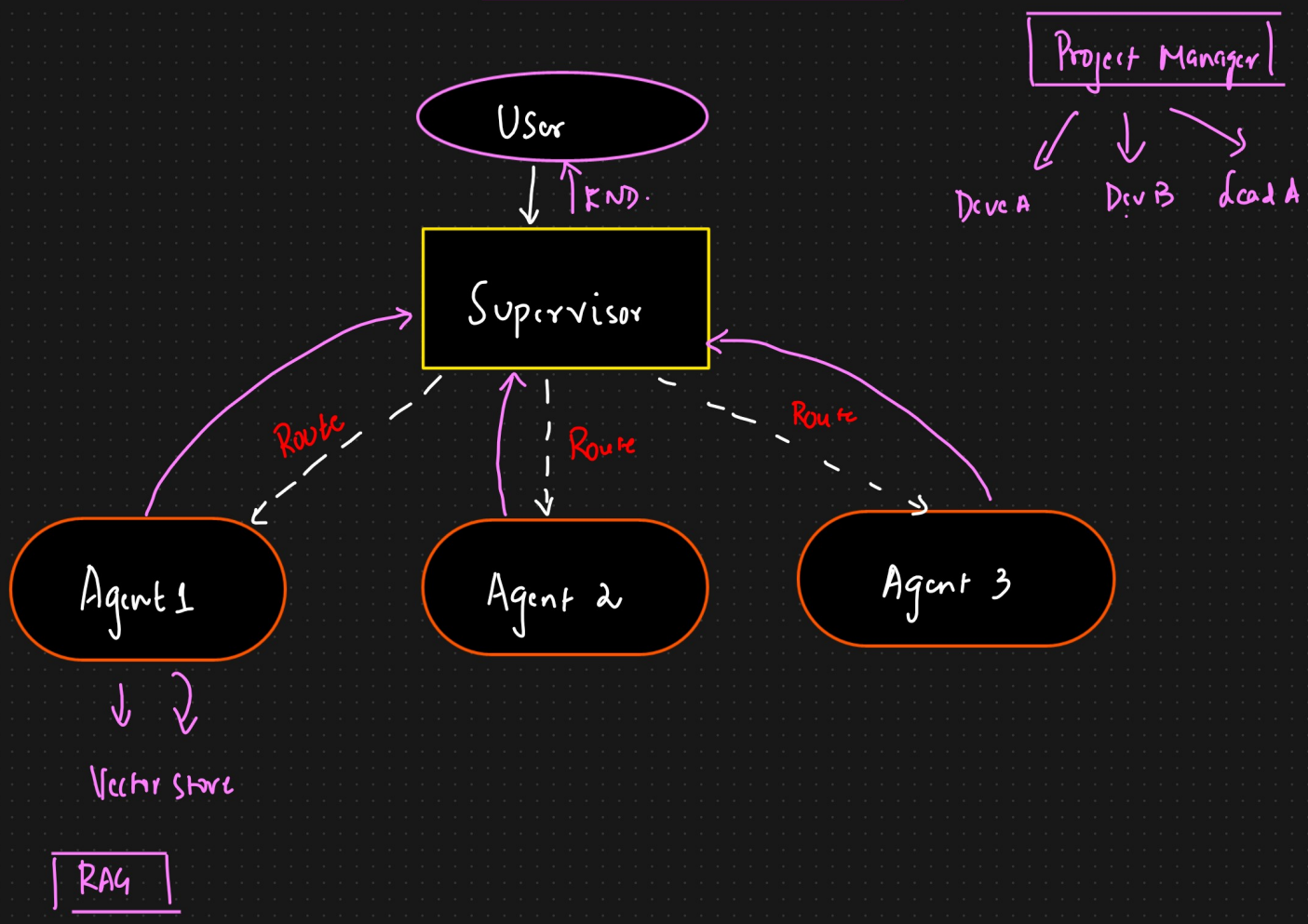
Multi-agent network

A single agent can usually operate effectively using a handful of tools within a single domain, but even using powerful models like gpt-4, it can be less effective at using many tools.

One way to approach complicated tasks is through a "divide-and-conquer" approach: create a specialized agent for each task or domain and route tasks to the correct "expert". This is an example of a multi-agent network architecture.

2) Supervisor Multi Agent RAG

Supervisor is a multi-agent architecture where specialized agents are coordinated by a central supervisor agent. The supervisor agent controls all communication flow and task delegation, making decisions about which agent to invoke based on the current context and task requirements.



③ Hierarchical Agent

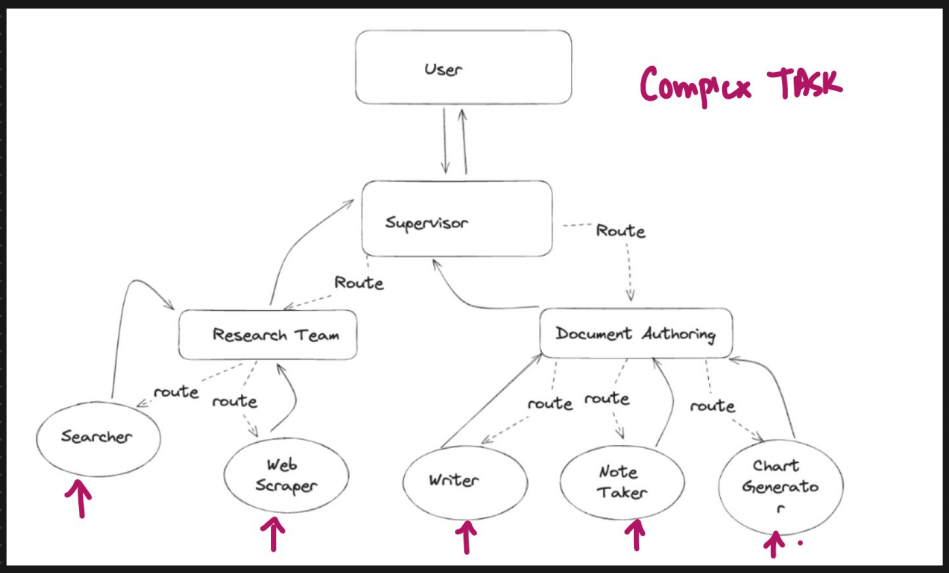
In our previous example (Agent Supervisor), we introduced the concept of a single supervisor node to route work between different worker nodes.

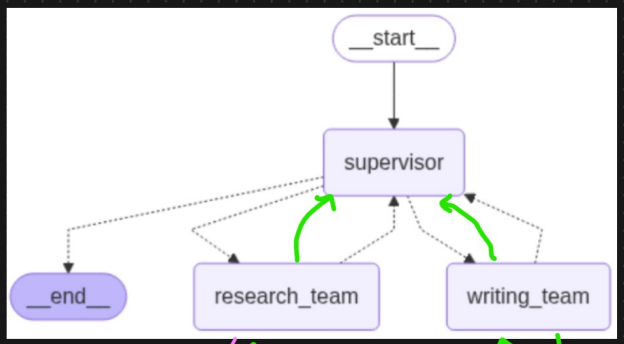
But what if the job for a single worker becomes too complex? What if the number of workers becomes too large?

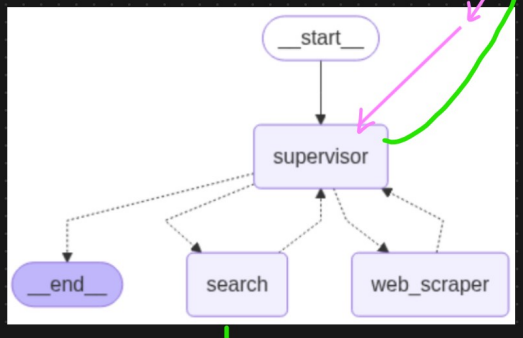
For some applications, the system may be more effective if work is distributed hierarchically.

You can do this by composing different subgraphs and creating a top-level supervisor, along with mid-level supervisors.

To do this, let's build a simple research assistant! The graph will look something like the following:







Web Search. Vector Store