Conquering the complexity of Agent Building with Xircuits

Xpress AI

Hello. I’m Paul Dubs.

- Software developer for 20+ years
- ML practitioner for 10+ years
- CTO & Co-Founder of Xpress AI
    - Currently headquartered in Japan
    - Building a platform that allows everyone to create their own agents – in the cloud and on-prem.

What is an Agent?

agency:

“the capacity, condition or state of acting or exerting power”

agent:

“one that acts or exerts power”

Imagine something like
ChatGPT with Arms & Legs

Behold, An Agent!

AGENT?
THAT SOUNDS LIKE A CHATBOT WITH EXTRA STEPS
.com

“Can you do X for me?”

Chatbot

“**No.** But here are some instructions for how you could try to do X yourself.”

Chatbot with RAG

“**No.** But here are some instructions, *I found*, for how you could try to do X yourself.”

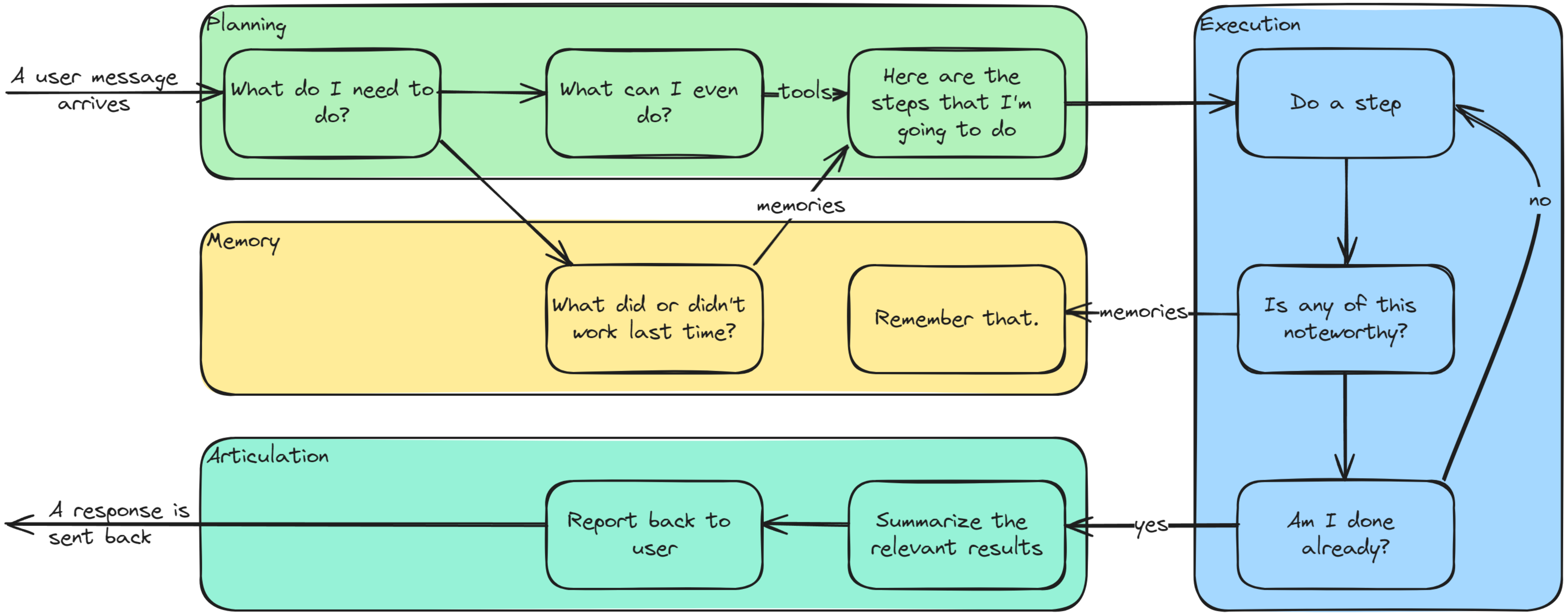
Chatbot with Function Calling

“call X(args)”

Agent

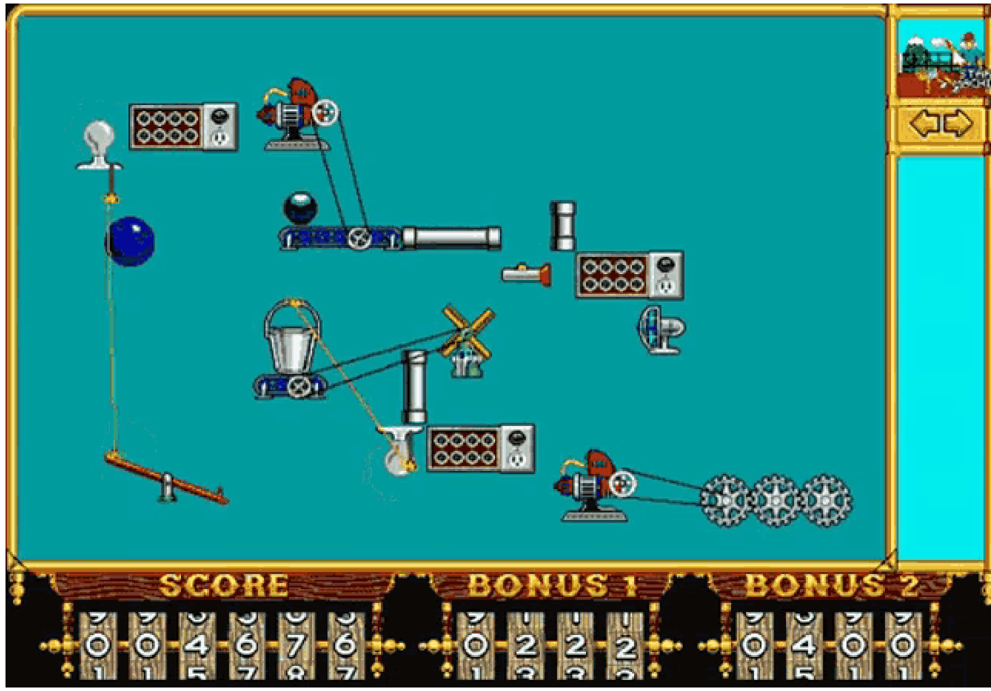
“Sure, let me get started on that.”

Agents: a lot more than function calling



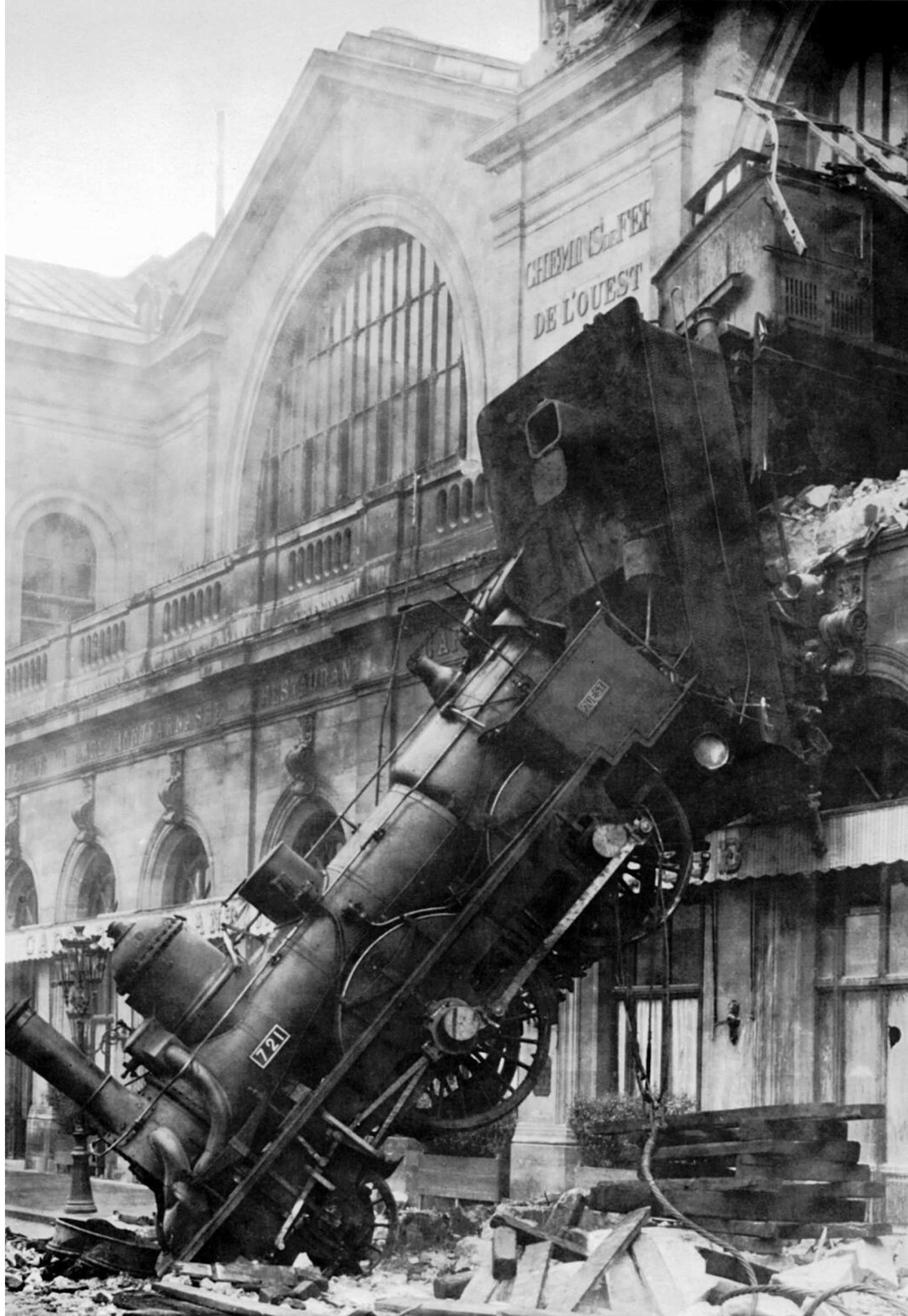
Inherent Complexity

- Infrastructure necessary to connect all those parts
- LLMs may be flaky
  - APIs may answer slowly (throttling, network isues)
  - May refuse requests
- Support for different models
  - specialized models for some tasks

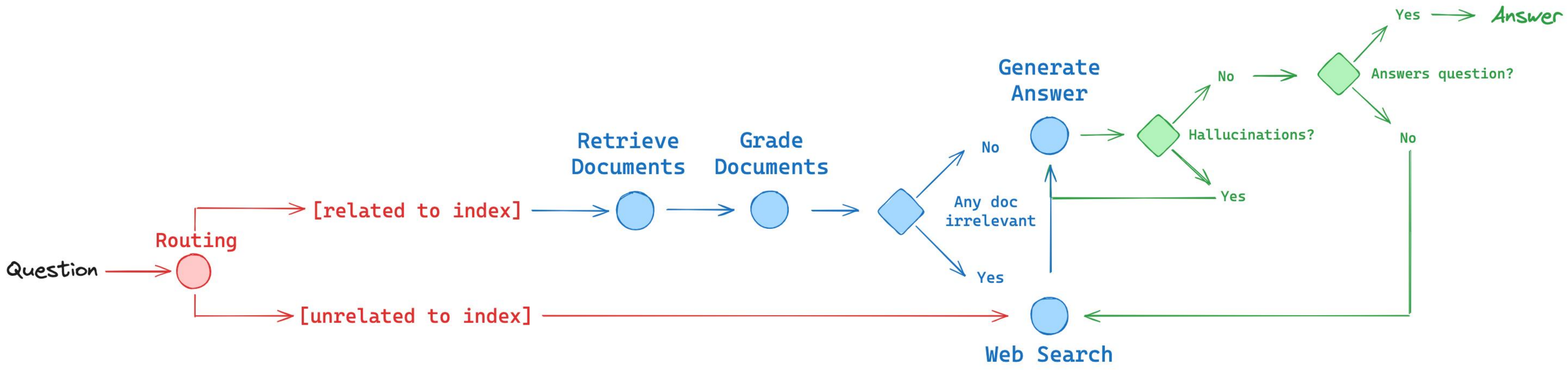


Accidental Complexity

- Quick moving field
- Libraries that make for nice tutorials but hard to customize
  - many abstractions
  - steep learning curve beyond initial tutorials
  - Execution far removed from definition

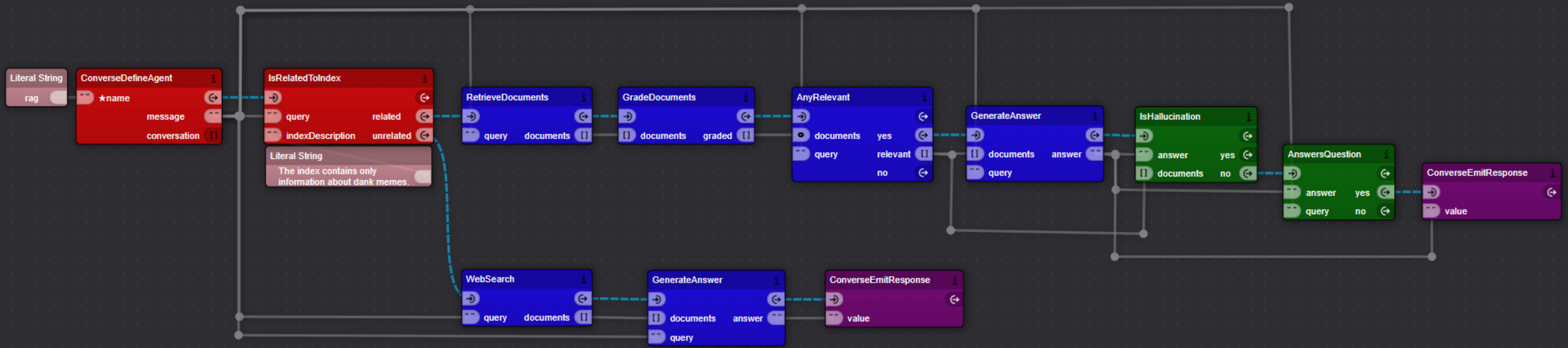


Create a Diagram to deal with Complexity



From Local RAG agent with LLaMA3: https://github.com/langchain-ai/langgraph/blob/main/examples/rag/langgraph_rag_agent_llama3_local.ipynb

Why not an executable Diagram?

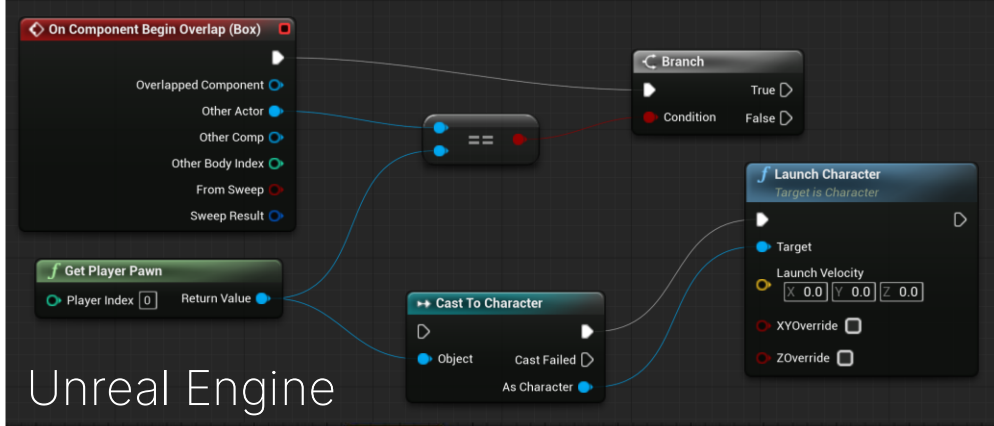


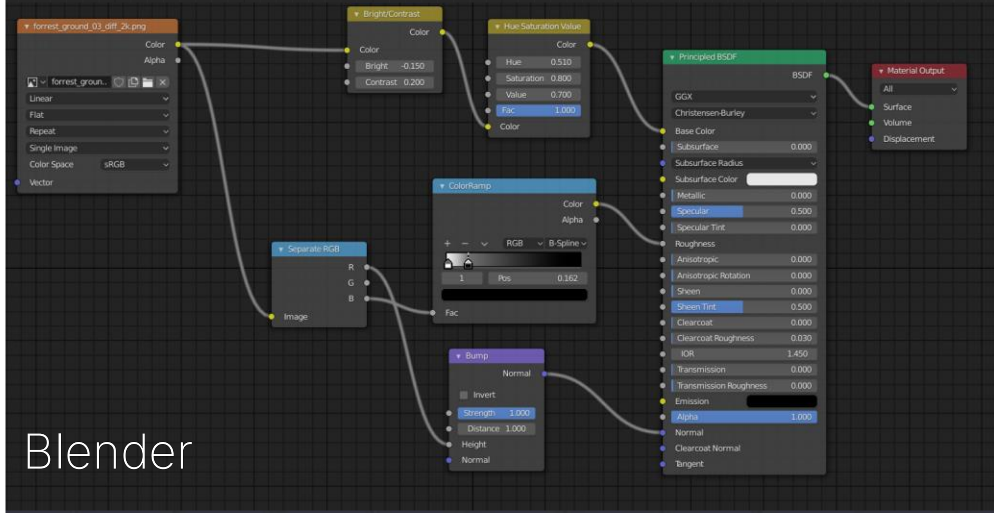
Xircuits: Visual programming in JupyterLab

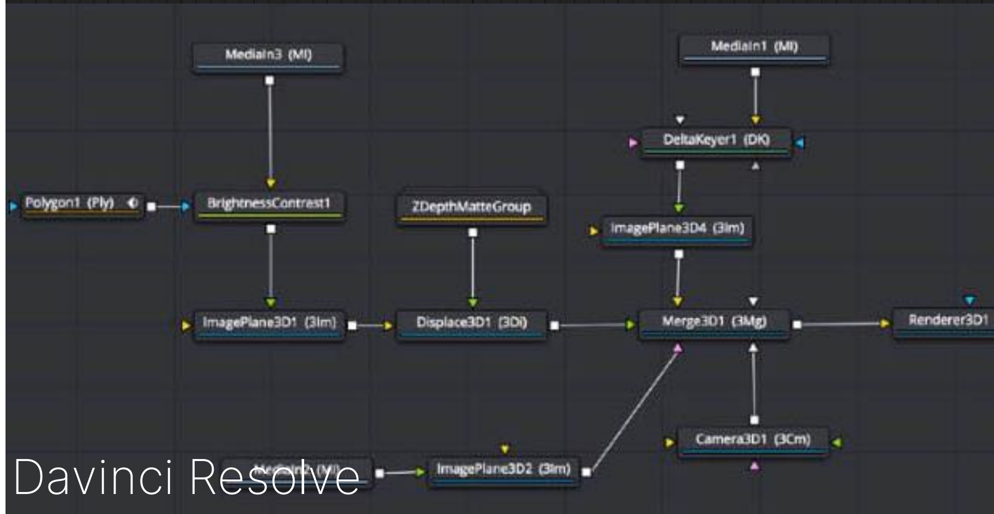
- Graphical programming approach to create executable diagrams
- Compiles to regular python
- Allows you to choose your own abstraction level
- Extendibility and modifiability are primary concerns
- Apache 2.0 License

Why Xircuits?

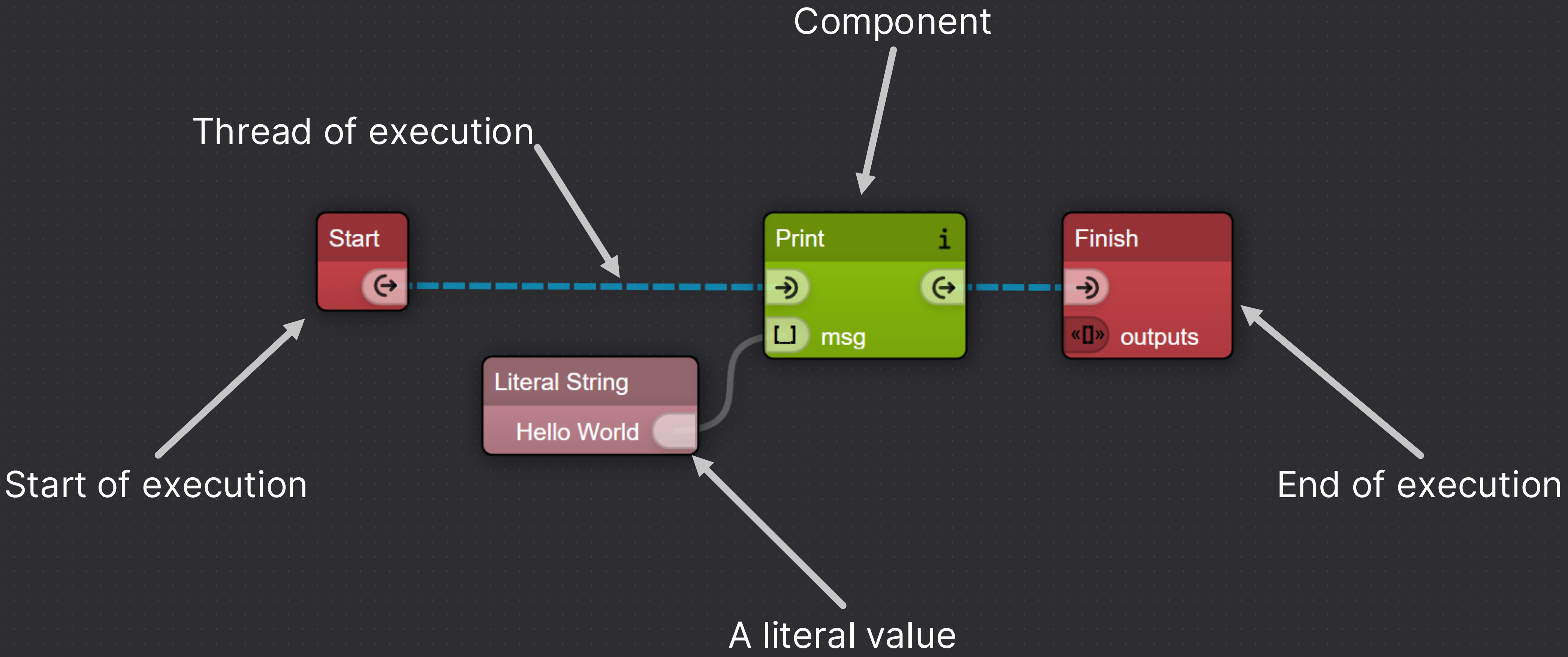
- Originally for ML workflows
  - Simplify creation of new workflows by creating reusable components
- Inspiration from other disciplines: Game Development, 3D Modeling, Video Editing, ...
- Code-optional, instead of No-Code
  - Components defined in python, allowing infinite extendibility





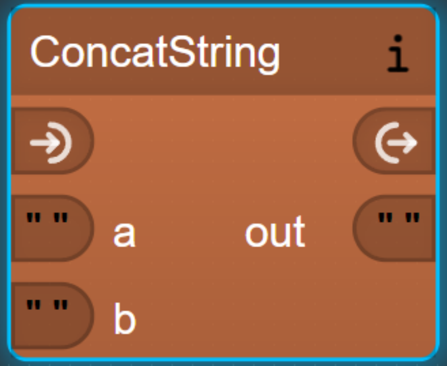


Hello World


Component
Thread of execution
Start
Print
Finish
msg
outputs
Literal String
Hello World
Start of execution
End of execution
A literal value

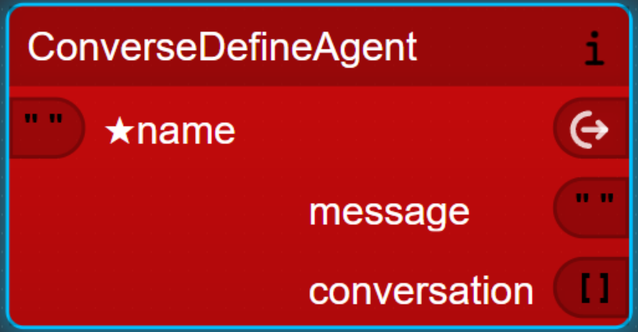
Custom Components

```
@xai_component
class ConcatString(Component):
    a: InArg[str]
    b: InArg[str]
    out: OutArg[str]

    def execute(self, ctx) -> None:
        self.out.value = "%s%s" % (self.a.value, self.b.value)
```

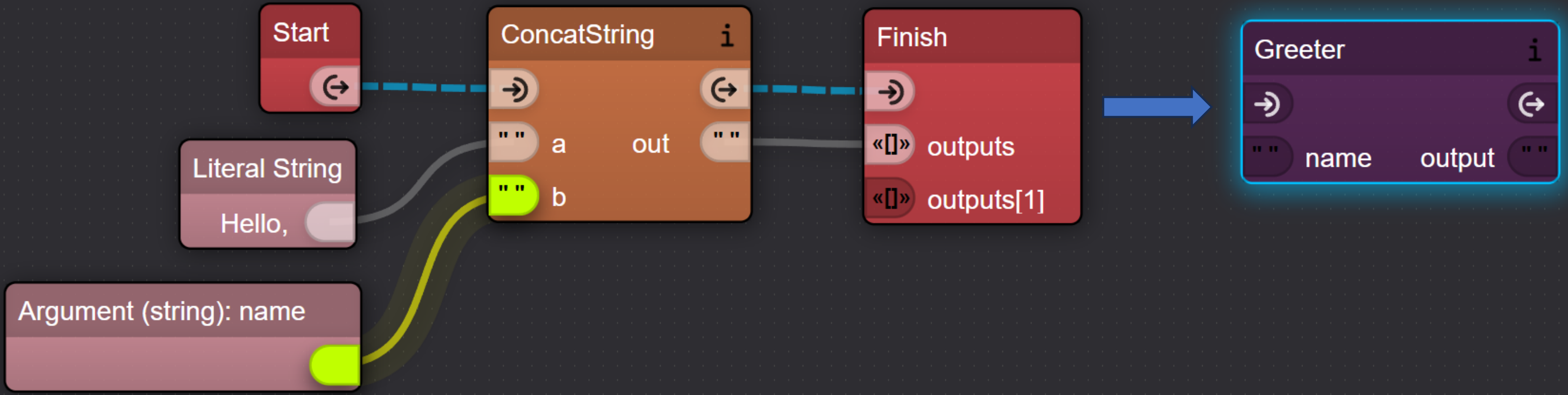


Advanced Components

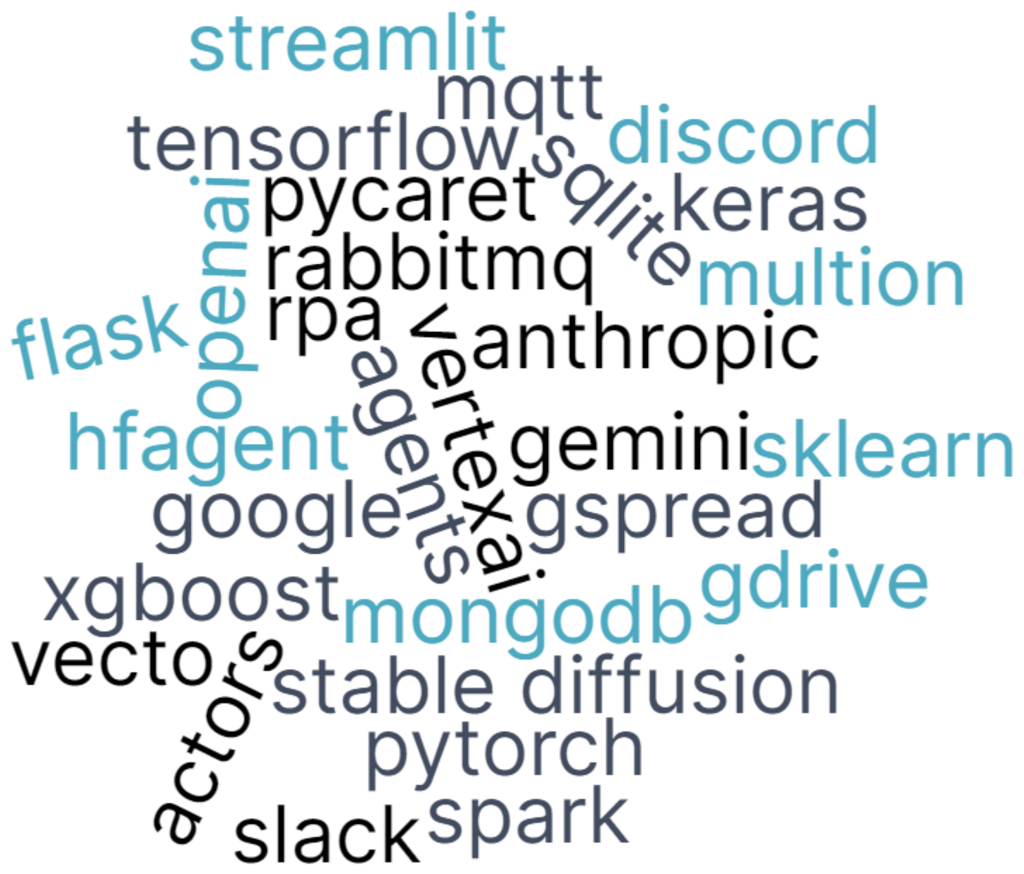
```python
@xai_component(type='Start', color='red')
class ConverseDefineAgent(Component):
    name: InCompArg[str]
    message: OutArg[str]
    conversation: OutArg[list]

    def init(self, ctx):
        ctx.setdefault(CONVERSE_AGENTS_KEY, {})[self.name.value] = self
```

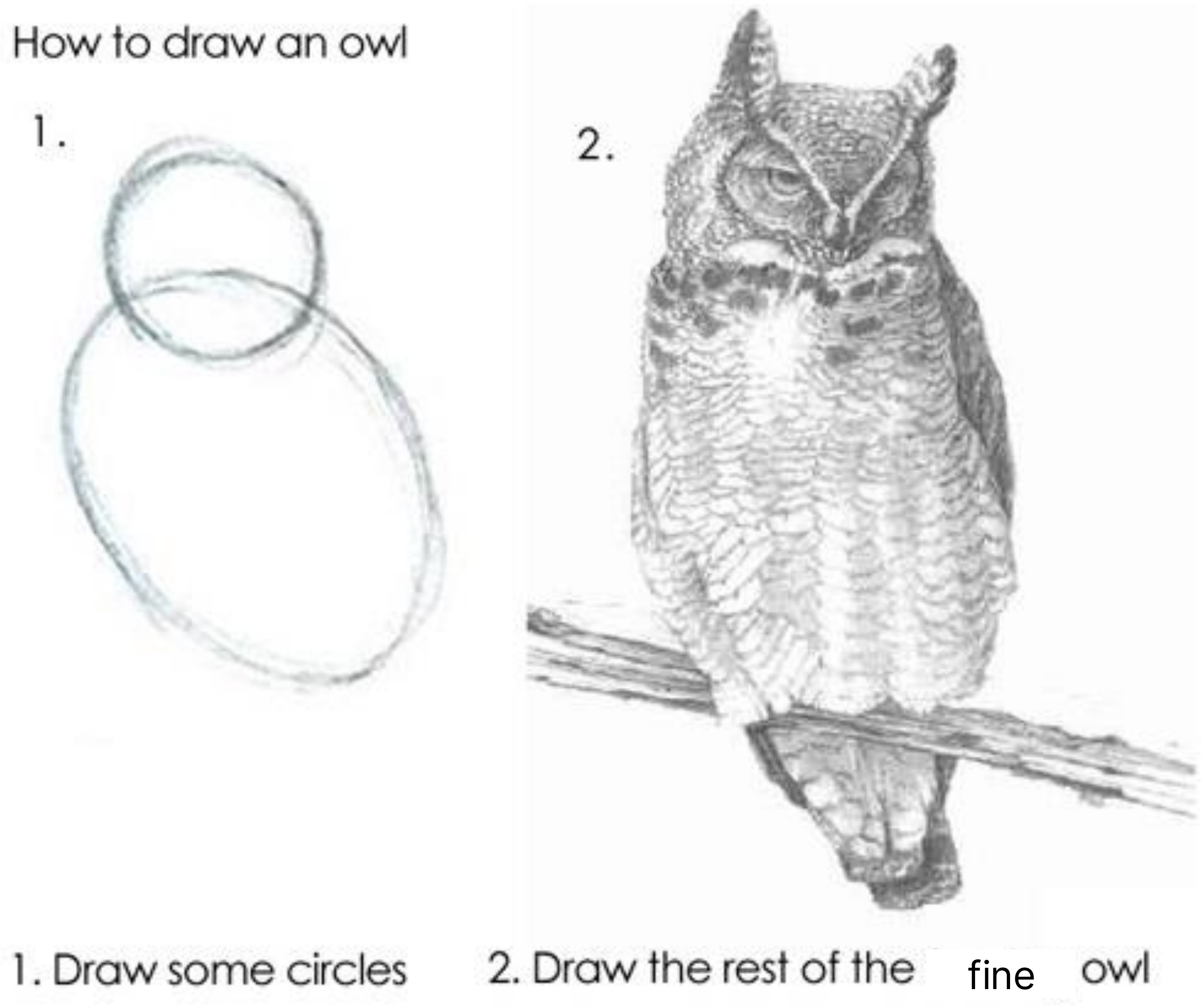


No-Code Components

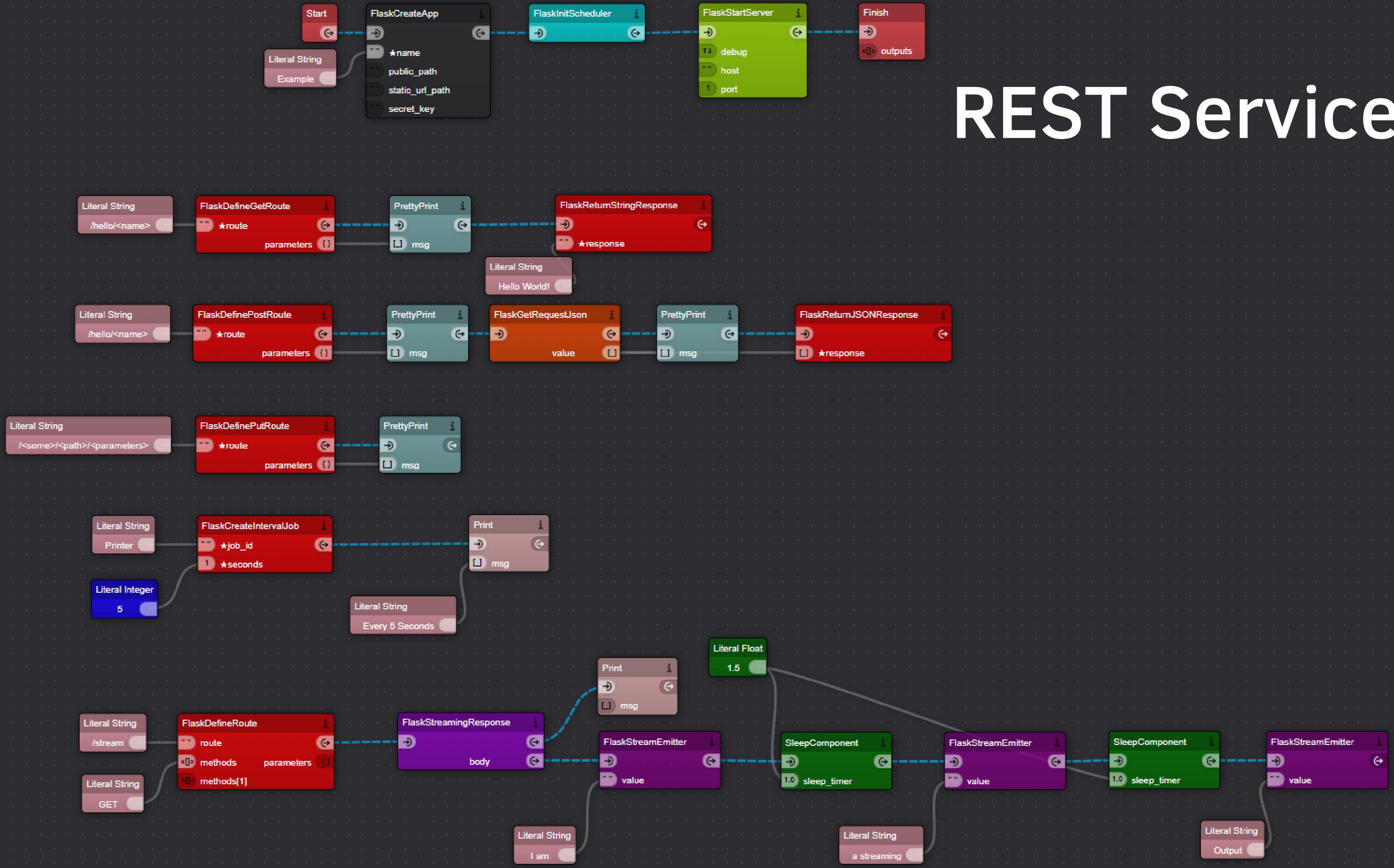


Batteries included




How to draw an owl
1.
2.
1. Draw some circles
2. Draw the rest of the fine owl

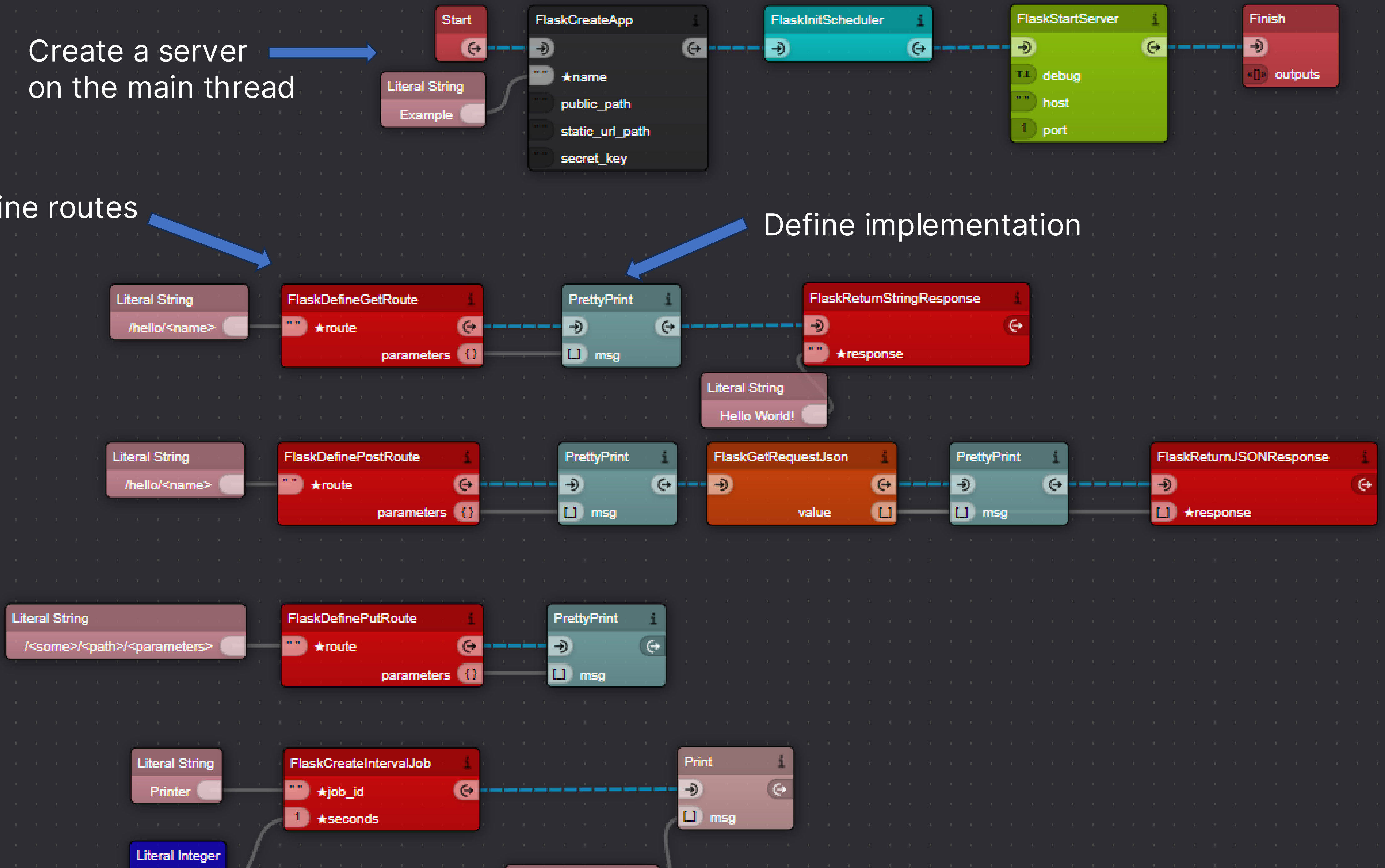
REST Service



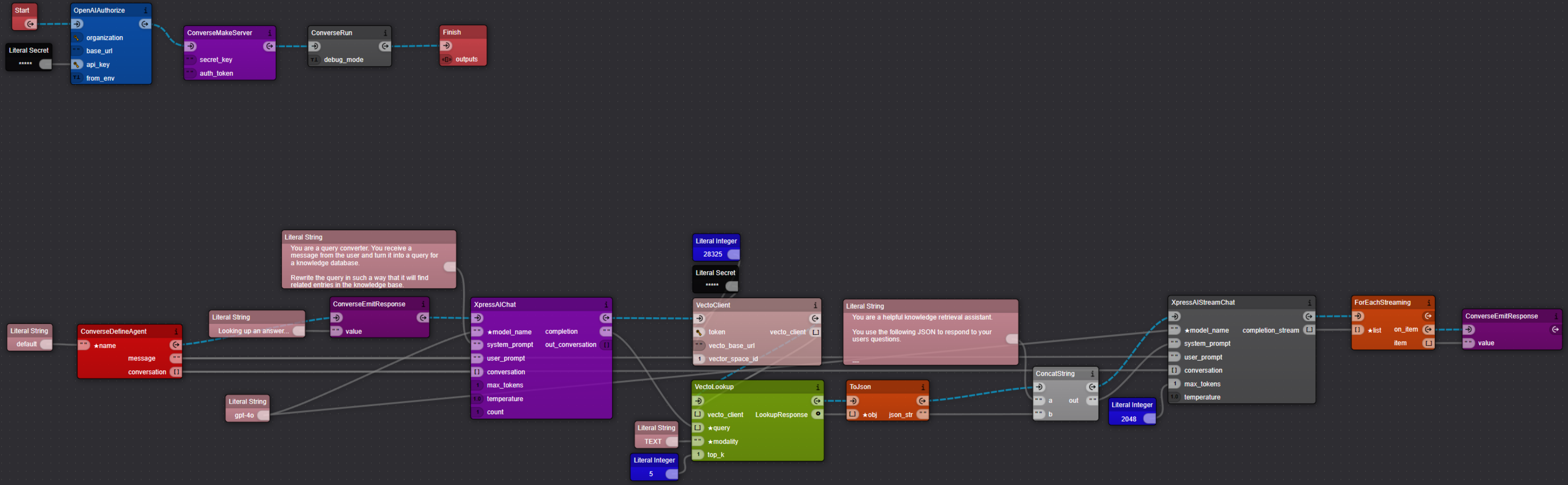
Create a server
on the main thread

Define routes

Define implementation



Retrieval Augmented Generation (RAG)



You are a query converter. You receive a message from the user and turn it into a query for a knowledge database.

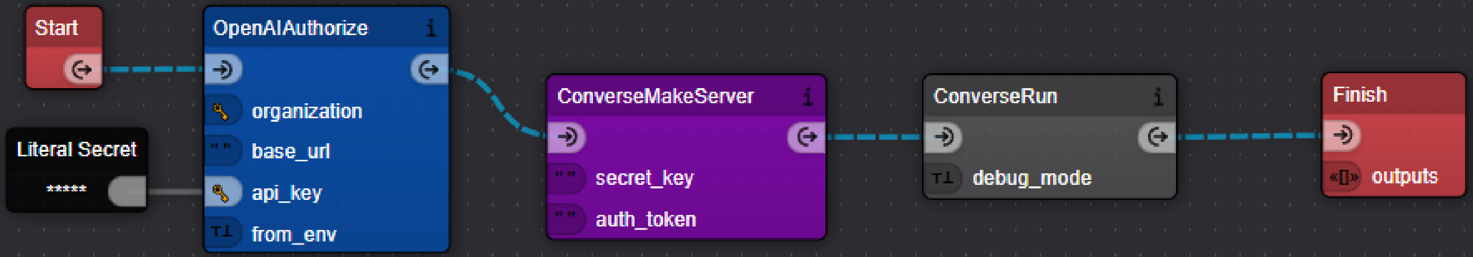
Rewrite the query in such a way that it will find related entries in the knowledge base.

You are a helpful knowledge retrieval assistant.

You use the following JSON to respond to your users questions.

—

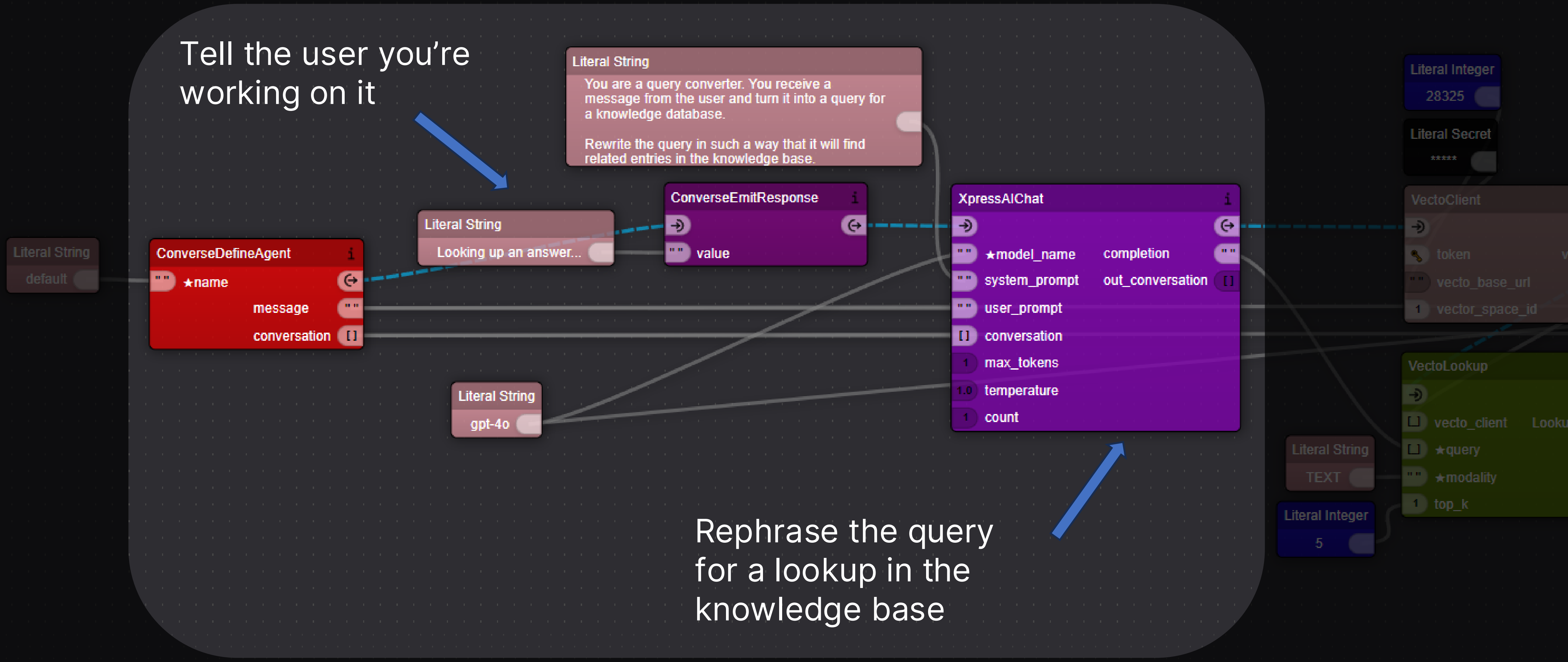
Create a server
on the main thread



Authorize to OpenAI

Tell the user you're working on it

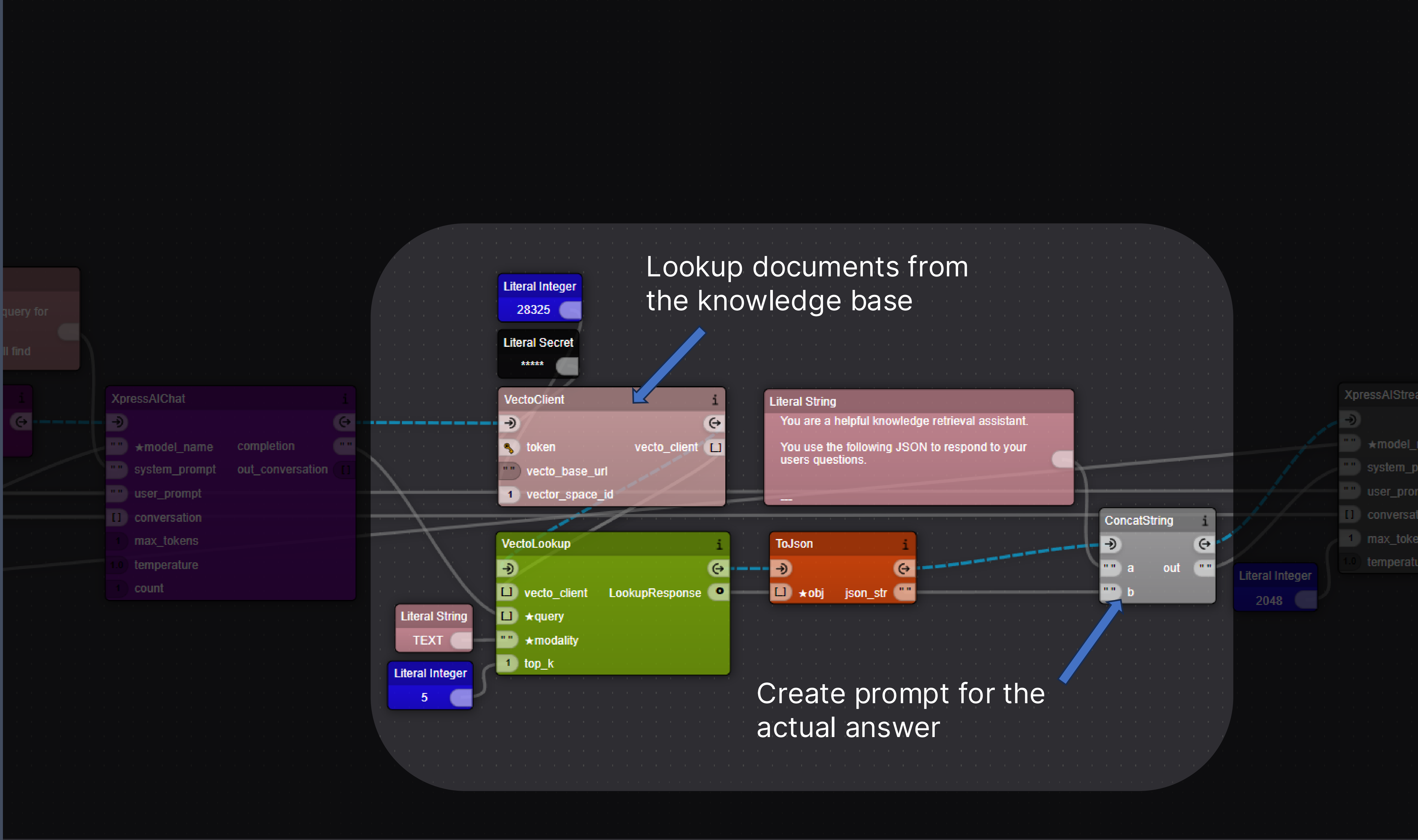
Rephrase the query for a lookup in the knowledge base



Literal String

You are a query converter. You receive a message from the user and turn it into a query for a knowledge database.

Rewrite the query in such a way that it will find related entries in the knowledge base.

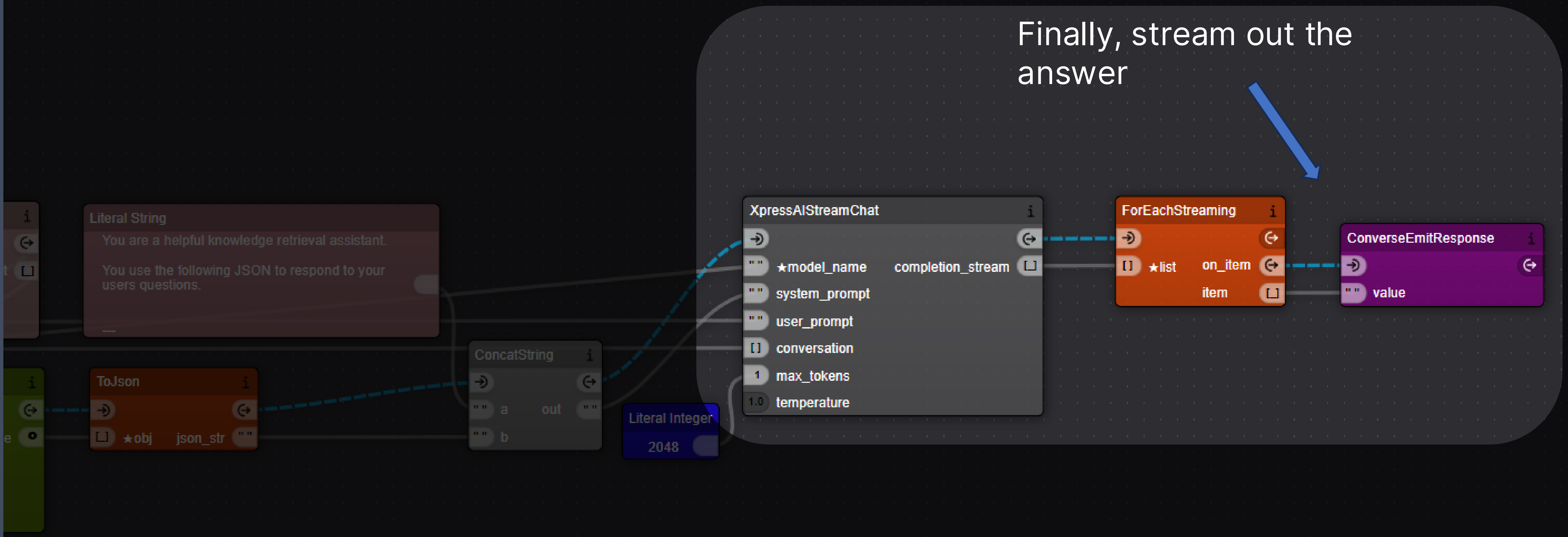


Literal String
You are a helpful knowledge retrieval assistant.

You use the following JSON to respond to your users questions.

—

Finally, stream out the answer

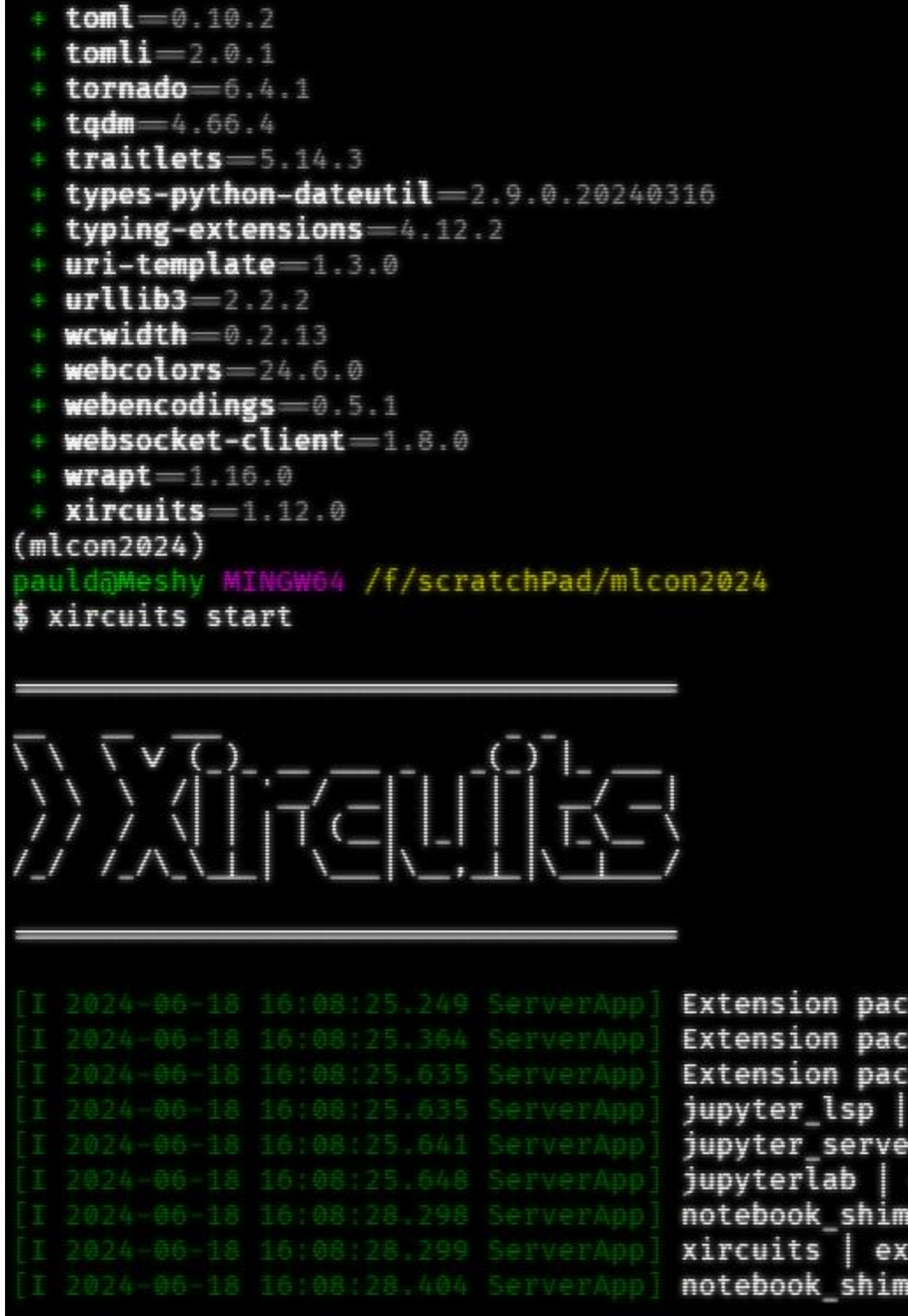


Demo: Agent in Action

Demo: Agent Implementation

Try it yourself

```
$ pip install xircuits
$ xircuits start
```



```
 + toml==0.10.2
 + tomli==2.0.1
 + tornado==6.4.1
 + tqdm==4.66.4
 + traitlets==5.14.3
 + types-python-dateutil==2.9.0.20240316
 + typing-extensions==4.12.2
 + uri-template==1.3.0
 + urllib3==2.2.2
 + wcwidth==0.2.13
 + webcolors==24.6.0
 + webencodings==0.5.1
 + websocket-client==1.8.0
 + wrapt==1.16.0
 + xircuits==1.12.0
(mlcon2024)
pauld@Meshy MINGW64 /f/scratchPad/mlcon2024
$ xircuits start

[I 2024-06-18 16:08:25.249 ServerApp] Extension pac
[I 2024-06-18 16:08:25.364 ServerApp] Extension pac
[I 2024-06-18 16:08:25.635 ServerApp] Extension pac
[I 2024-06-18 16:08:25.635 ServerApp] jupyter_lsp |
[I 2024-06-18 16:08:25.641 ServerApp] jupyter_serve
[I 2024-06-18 16:08:25.648 ServerApp] jupyterlab |
[I 2024-06-18 16:08:28.298 ServerApp] notebook_shim
[I 2024-06-18 16:08:28.299 ServerApp] xircuits | ex
[I 2024-06-18 16:08:28.404 ServerApp] notebook_shim
```

Questions?

Slides & Links to more materials:

https://www.dubs.tech/mlcon2024

SCAN ME