

大模型重塑软件开发

Kongjie@tencent.com

个人简介



Konjie / 揭光发

Techparty 前组委主席。

2015年开始从事低无代码相关的研发与管理工作。

2020年度腾讯云TVP。

2021年以低代码技术专家身份加入腾讯，是低代码Oteam核心成员，参与信通院低代码标准、IEEE低代码标准的订制工作。

大模型AIGC与软件开发现状

AIGC的三种形态，从用途的角度区分

端到端内容生成（经典AIG Content）

AI生成最终产品，它们可以直接被最终消费者使用。例如文本、图片、音频和视频，它们都是可以直接供给最终用户的。

中间产物生成（AIG Code）

AI生成的中间产品，它们往往需要进一步的加工和使用。例如生成的代码片段和完整的复杂代码，它们通常被用作开发更复杂系统或应用的基础。

任务与动作生成（AIG aCtion）

AI生成的具体动作或任务，它们常常作为粘合层，连接不同系统和应用，以完成更复杂的任务。例如自然语言交互，复杂任务执行等。

AIG -- Code, 开启十倍程序员的时代

```
TS sentiments.ts  write_sql.go  parse_expenses.py  addresses.rb

1 import datetime
2
3 def parse_expenses(expenses_string):
4     """Parse the list of expenses and return the list of triples (date, value, currency).
5     Ignore lines starting with #.
6     Parse the date using datetime.
7     Example expenses_string:
8         2016-01-02 -34.01 USD
9         2016-01-03 2.59 DKK
10        2016-01-03 -2.72 EUR
11    """
12    expenses = []
13    for line in expenses_string.splitlines():
14        if line.startswith("#"):
15            continue
16        date, value, currency = line.split(" ")
17        expenses.append((datetime.datetime.strptime(date, "%Y-%m-%d"),
18                        float(value),
19                        currency))
20    return expenses
```

Copilot

Replay

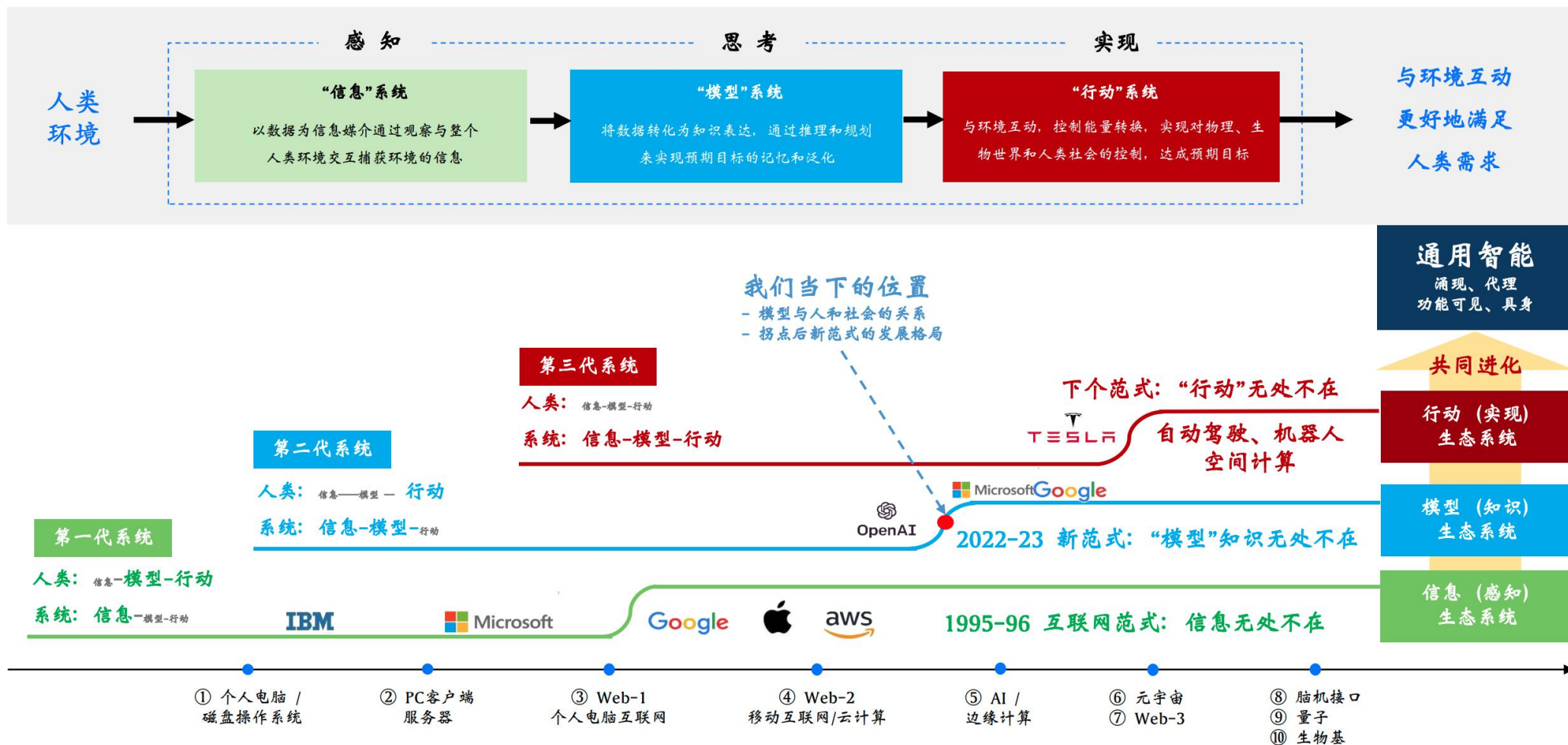


GitHub
Copilot

AIG - Action: 通往第三个拐点（行动无处不在）的道路

新范式的新拐点 @陆奇

“三位一体结构演化模式”：人、组织、社会，数字化



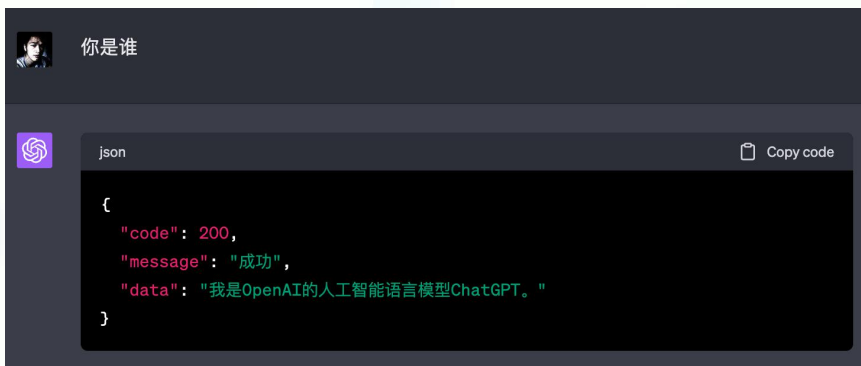
AIGX -- Action:低代码 + AI的最佳结合方式



AIGX -- Action: AI充当自然语言与既有系统的粘合层

AIGAction的本质是在AI生成的文本中嵌入特定格式的
代码或配置数据。

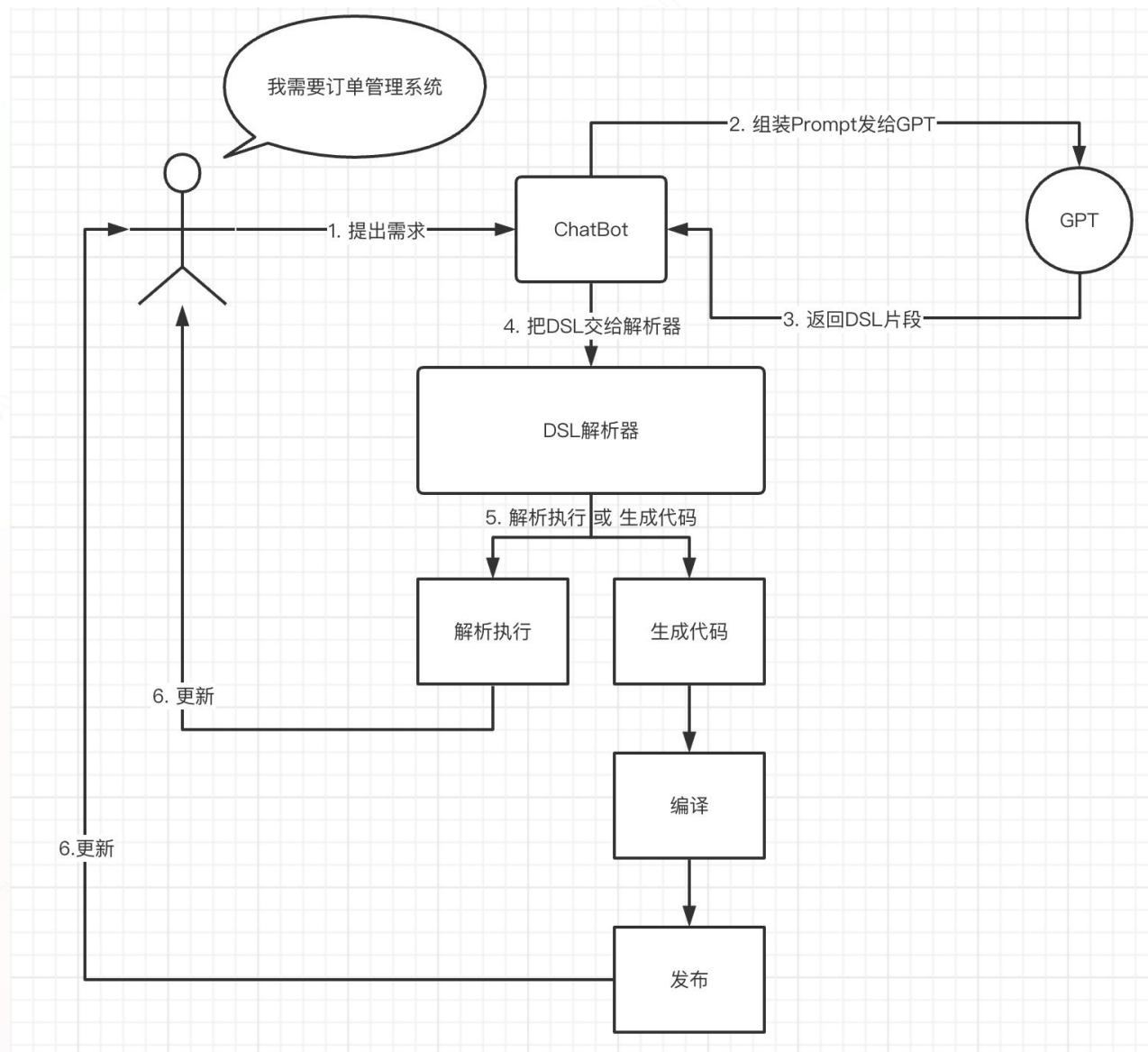
这些代码或数据可以被解析并用来指导特定的程序执行
相应的任务。



你是谁

```
json
{
  "code": 200,
  "message": "成功",
  "data": "我是OpenAI的人工智能语言模型ChatGPT。"
}
```

Copy code



AIGX -- Action: 当下大模型应用开发的主流形态

ChatGPT Plugin，插件提供具体的扩展能力，ChatGPT识别用户意图调用插件执行复杂任务。

Plugin store

New

Most popular

All plugins



Instacart

Install

Ask about recipes and discover the ingredients you need—then get it all delivered from local stores.



Zapier

Install

Interact with over 5,000+ apps like Google Sheets, Gmail, HubSpot, Salesforce, and thousands more.



KAYAK

Install

Search flights, stays & rental cars or get recommendations where you can go on your budget.



Expedia

Install

Bring your trip plans to life – get there, stay there, find things to see and do.



Wolfram

Uninstall

Access computation, math, curated knowledge & real-time data through Wolfram|Alpha and Wolfram Language.



Speak

Install

Learn how to say anything in another language with Speak, your AI-powered language tutor.

About plugins



腾讯最新的股票是多少



Finished browsing

腾讯的最新股票价格为44.33美元，数据截至2023年5月15日¹。



换算为港币和人民币



Finished browsing

腾讯的最新股票价格是44.33美元。换算为港币和人民币如下：

44.33美元 = 44.33 * 7.84225港币 = 347.564港币^{1,2}。

44.33美元 = 44.33 * 6.95965人民币 = 308.317人民币^{3,4}。

AIGX -- Action:当下大模型应用开发的主流形态

```
> Entering new AgentExecutor chain...
```

```
Question: 电视剧三体的作者是谁, 上映时间是什么时候?
```

```
Thought: We need to use the `get_teleplay_id` tool to get the teleplay id for "三体",  
and then use the `get_teleplay_info` tool to get the author and release date  
information.
```

```
Action:
```

```
...
```

```
{
```

```
  "action": "get_teleplay_id",
```

```
  "action_input": "三体"
```

```
}
```

```
...
```



LangChain

AIGX -- Action: ChatGPT最新模型支持Function Calling

Function calling example



What's the weather like in Boston right now?

Step 1 · OpenAI API

Call the model with functions and the user's input

Step 2 · Third party API

Use the model response to call your API

Step 3 · OpenAI API

Send the response back to the model to summarize



The weather in Boston is currently sunny with a temperature of 22 d

```
# Step 1, send model the user query and what functions it has access to
def run_conversation():
    response = openai.ChatCompletion.create(
        model="gpt-3.5-turbo-0613",
        messages=[{"role": "user", "content": "What's the weather like in Boston?"}],
        functions=[
            {
                "name": "get_current_weather",
                "description": "Get the current weather in a given location",
                "parameters": {
                    "type": "object",
                    "properties": {
                        "location": {
                            "type": "string",
                            "description": "The city and state, e.g. San Francisco, CA",
                        },
                        "unit": {"type": "string", "enum": ["celsius", "fahrenheit"]},
                    },
                    "required": ["location"],
                },
            },
        ],
        function_call="auto",
    )

    message = response["choices"][0]["message"]
```

大模型后的低代码开发业态 - 自然语言交互

Power apps Copilot

The screenshot displays the Power Apps Copilot interface. On the left, a table is visible with the following data:

Inspector type	Location
Daily	New York City, New York
Daily	Los Angeles, California
Weekly	Chicago, Illinois
Weekly	Houston, Texas
Weekly	Philadelphia, Pennsylvania
Daily	Phoenix, Arizona
Weekly	San Antonio, Texas
Weekly	San Diego, California
Weekly	Austin, Texas

On the right, the Copilot chat window is active, showing the following interactions:

- Copilot** (with a **Preview** button and a refresh icon)
- User Prompt:** "Add a column called Location and populate it with sample data" (timestamp: Just now)
- Copilot Response:** "Sure, here's the Location column with sample data." (timestamp: Just now, with thumbs up/down icons)
- User Prompt:** "Give me suggested columns for the table" (timestamp: Just now)

与逻辑编排的结合 – 支持CURD以外的个性化逻辑开发

Thor-chat 表格



暂无数据

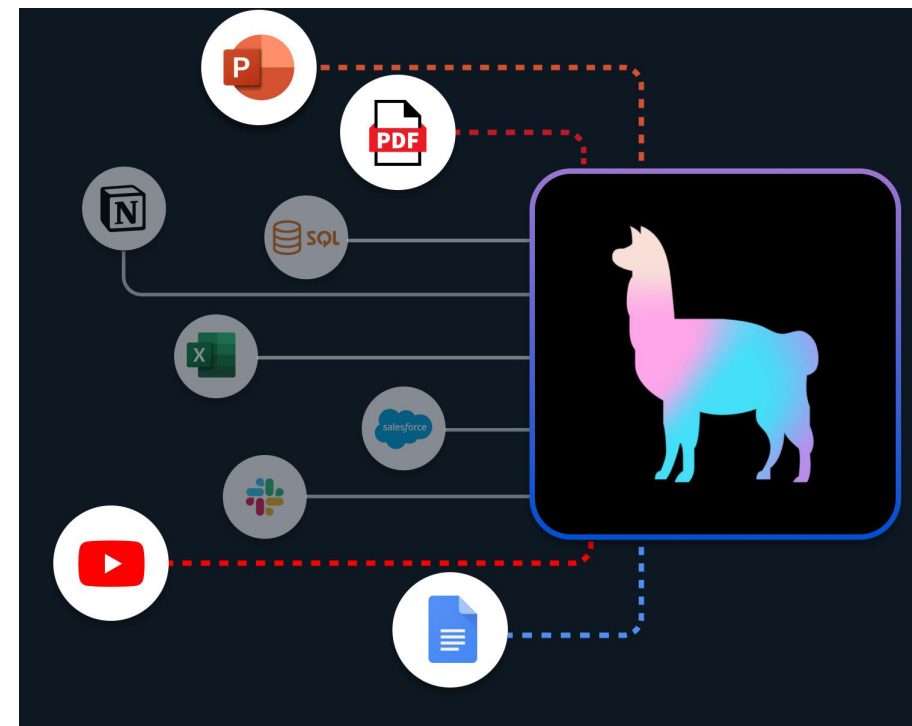
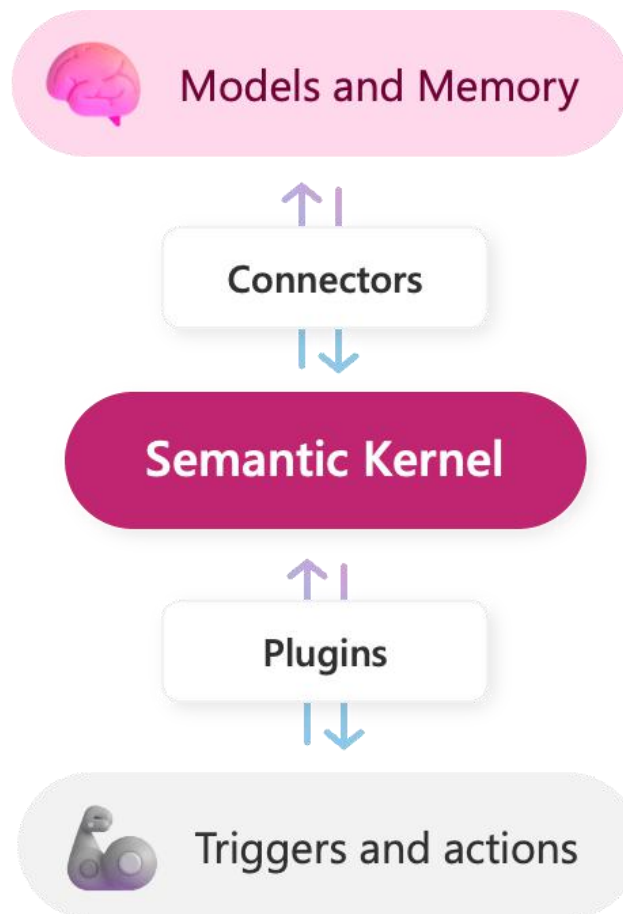


大模型正在加速重构软件开发

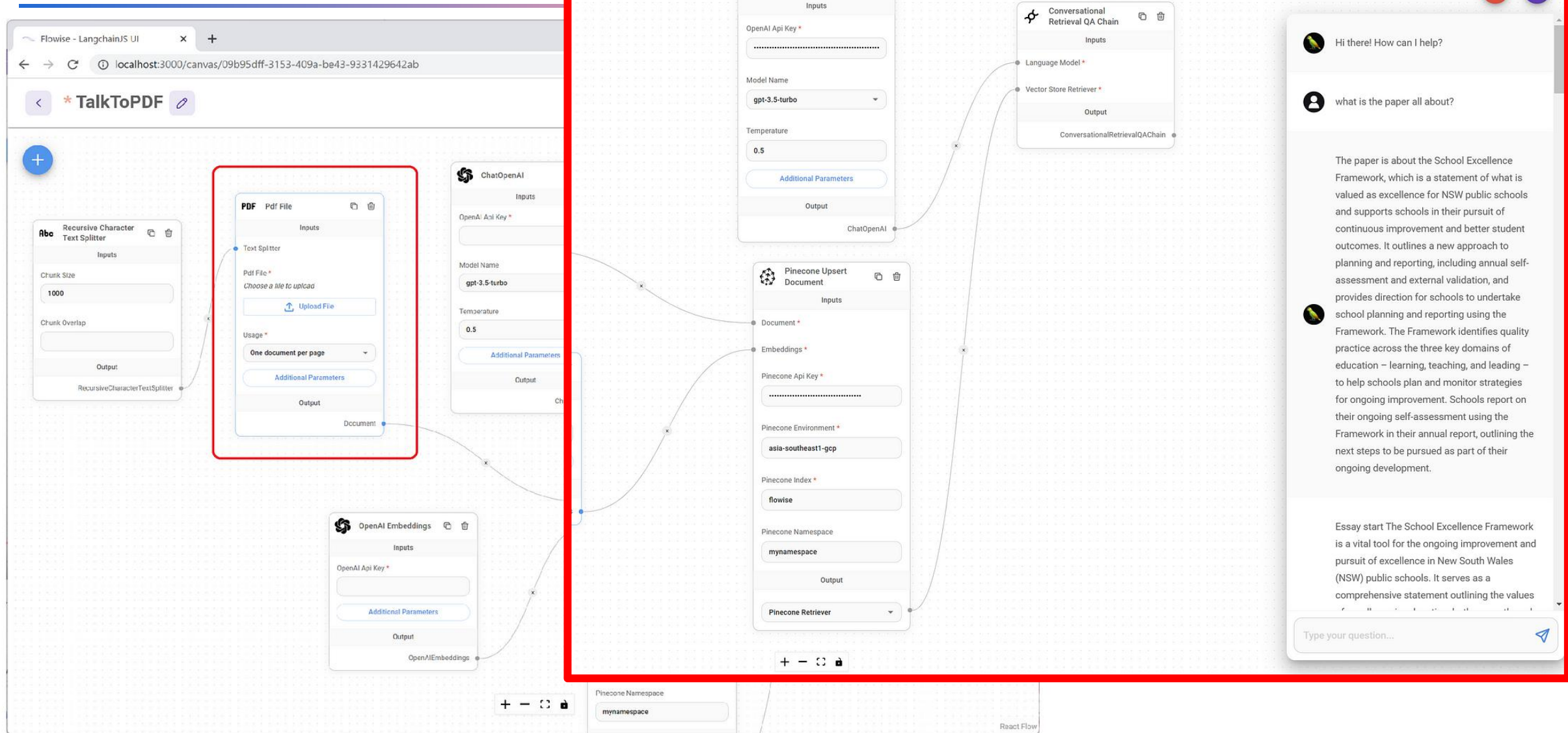
大模型应用开发框架飞速发展



LangChain



三分钟搞定Chat-PDF



自治Agent层出不穷，挑战更加复杂的任务

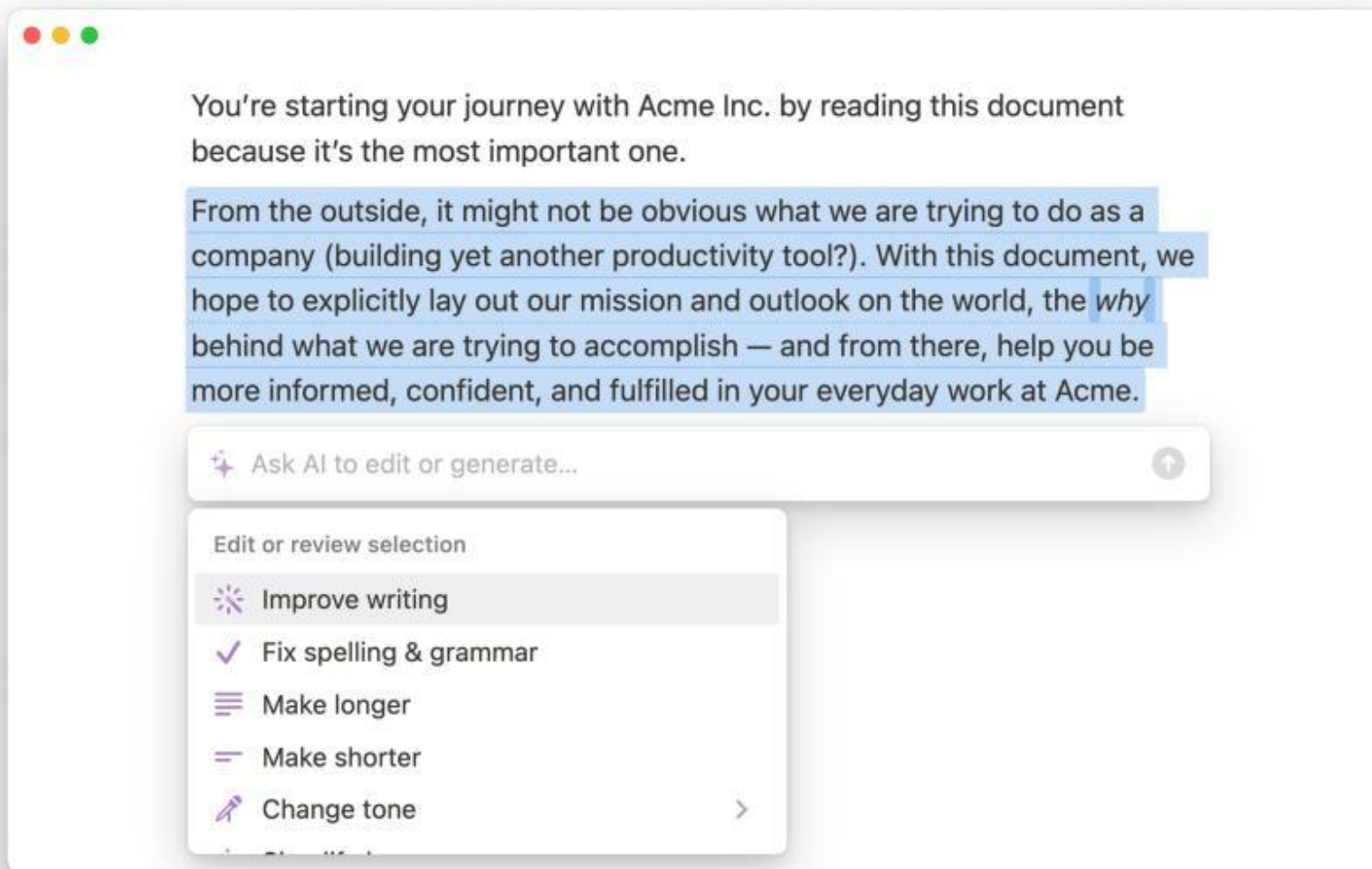


Baby AGI

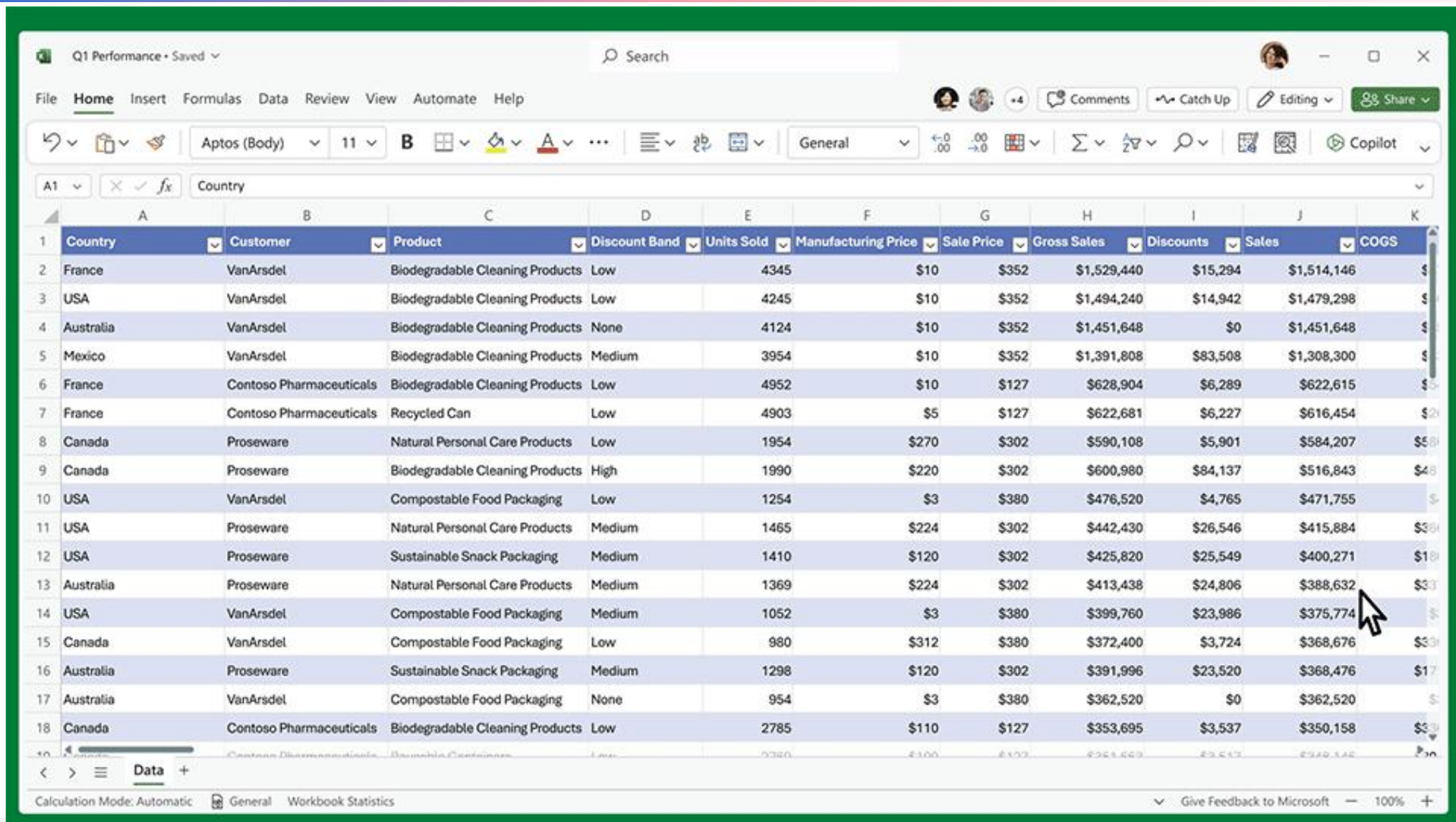
软件开发的新局面

1. 全新的软件交互体验
2. 软件开发自动驾驶

全新UX —— Chat every where, Copilot for Every thing



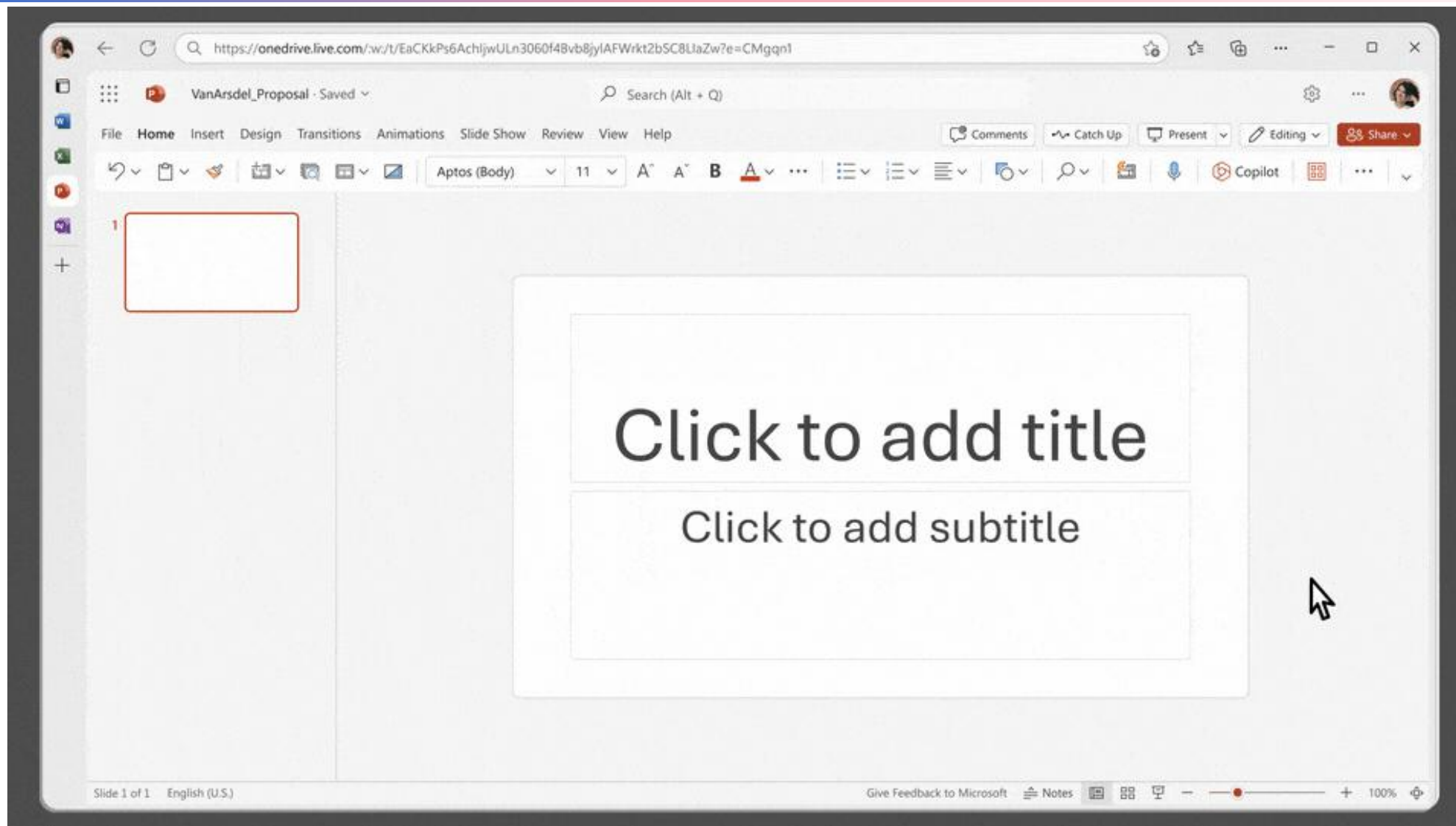
全新UX —— Chat every where, Copilot for Every thing



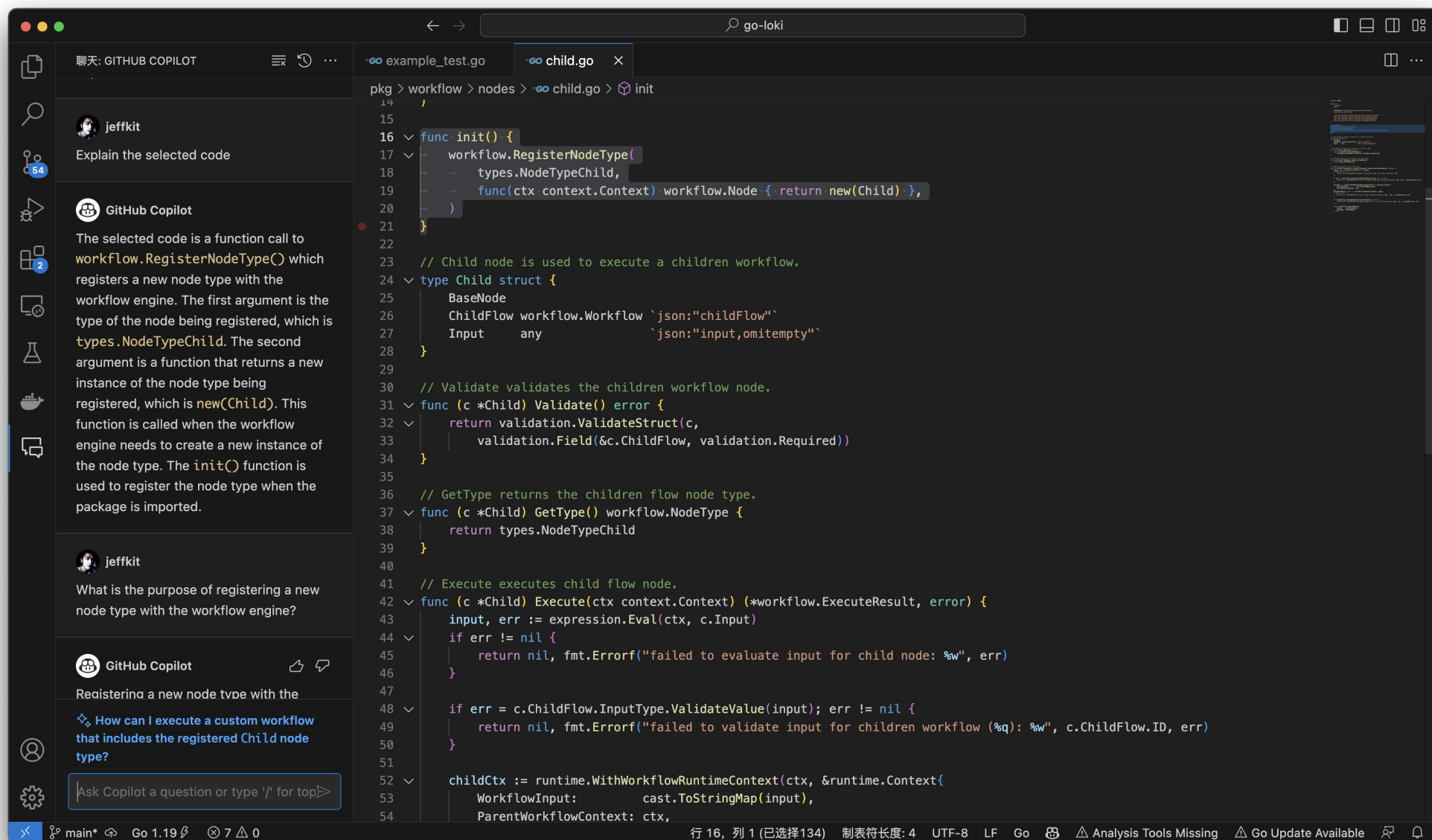
The screenshot displays the Microsoft Excel interface for a workbook titled "Q1 Performance - Saved". The ribbon includes tabs for File, Home, Insert, Formulas, Data, Review, View, Automate, and Help. The Home tab is active, showing the Font, Paragraph, and Styles groups. The Copilot icon is visible in the top right corner of the ribbon. The formula bar shows "Country" in cell A1. The main grid contains a table with 18 rows of data. A Copilot chat window is open on the right side of the grid, displaying a list of suggestions and a chat area with a text input field and a "Send" button. The data table has columns for Country, Customer, Product, Discount Band, Units Sold, Manufacturing Price, Sale Price, Gross Sales, Discounts, Sales, and COGS. The data is organized by country and product type, showing various metrics for each entry.

	A	B	C	D	E	F	G	H	I	J	K
1	Country	Customer	Product	Discount Band	Units Sold	Manufacturing Price	Sale Price	Gross Sales	Discounts	Sales	COGS
2	France	VanArsdel	Biodegradable Cleaning Products	Low	4345	\$10	\$352	\$1,529,440	\$15,294	\$1,514,146	\$
3	USA	VanArsdel	Biodegradable Cleaning Products	Low	4245	\$10	\$352	\$1,494,240	\$14,942	\$1,479,298	\$
4	Australia	VanArsdel	Biodegradable Cleaning Products	None	4124	\$10	\$352	\$1,451,648	\$0	\$1,451,648	\$
5	Mexico	VanArsdel	Biodegradable Cleaning Products	Medium	3954	\$10	\$352	\$1,391,808	\$83,508	\$1,308,300	\$
6	France	Contoso Pharmaceuticals	Biodegradable Cleaning Products	Low	4952	\$10	\$127	\$628,904	\$6,289	\$622,615	\$
7	France	Contoso Pharmaceuticals	Recycled Can	Low	4903	\$5	\$127	\$622,681	\$6,227	\$616,454	\$
8	Canada	Proseware	Natural Personal Care Products	Low	1954	\$270	\$302	\$590,108	\$5,901	\$584,207	\$
9	Canada	Proseware	Biodegradable Cleaning Products	High	1990	\$220	\$302	\$600,980	\$84,137	\$516,843	\$
10	USA	VanArsdel	Compostable Food Packaging	Low	1254	\$3	\$380	\$476,520	\$4,765	\$471,755	\$
11	USA	Proseware	Natural Personal Care Products	Medium	1465	\$224	\$302	\$442,430	\$26,546	\$415,884	\$
12	USA	Proseware	Sustainable Snack Packaging	Medium	1410	\$120	\$302	\$425,820	\$25,549	\$400,271	\$
13	Australia	Proseware	Natural Personal Care Products	Medium	1369	\$224	\$302	\$413,438	\$24,806	\$388,632	\$
14	USA	VanArsdel	Compostable Food Packaging	Medium	1052	\$3	\$380	\$399,760	\$23,986	\$375,774	\$
15	Canada	VanArsdel	Compostable Food Packaging	Low	980	\$312	\$380	\$372,400	\$3,724	\$368,676	\$
16	Australia	Proseware	Sustainable Snack Packaging	Medium	1298	\$120	\$302	\$391,996	\$23,520	\$368,476	\$
17	Australia	VanArsdel	Compostable Food Packaging	None	954	\$3	\$380	\$362,520	\$0	\$362,520	\$
18	Canada	Contoso Pharmaceuticals	Biodegradable Cleaning Products	Low	2785	\$110	\$127	\$353,695	\$3,537	\$350,158	\$

全新UX —— Chat every where, Copilot for Every thing



全新UX —— Chat every where, Copilot for Every thing



全新UX —— Chat every where, Copilot for Every thing

演讲嘉宾信息 (/speakers)

页面信息LessCode高级设置扩展插件上传素材新手指引搜索配置项(全局搜索)

主页面

模板通用组件数据源

输入字段名或ID

数据 (data)

数据源 (data)

auto_speakerSexCount (演讲嘉宾的男女数量)

auto_guestSexRate (嘉宾的男女比例) 单接口

页面变量 (state)

自定义页面变量

立刻添加

URL参数 (\$query)

你可以直接使用\$query.xxx读取url中的参数
在编辑页面时, 如果页面url没有对应参数, 会读取\$query示例值
在运行页面时, 如果页面url没有对应参数, 会读取\$query默认值

声明新的\$query

传给页面的参数 (props)

前端API (w)

全局实例 (\$app)

数据源依赖 (collectionRefs)

布局结构

页面

嘉宾男女比例

malefemale

演讲嘉宾男女数量

count

female

male

020406080

无极AI助手Beta

你好, 请描述你的BI需求, 目前我支持饼图、柱状图、折线图展示你的数据。(PS: 请放心授权数据表, 我只理解数据表的结构, 不读取数据。)
需求描述的例子:

- 基于这几个表提出3个统计需求并展示
- 展示客户的男女比例
- 展示指定客户名称 (比如"李四") 的订单时间分布情况

授权数据库

授权数据表 演讲嘉宾

展示嘉宾的男女比例

好的, 以下命令用于"展示嘉宾的男女比例"

commands

copy

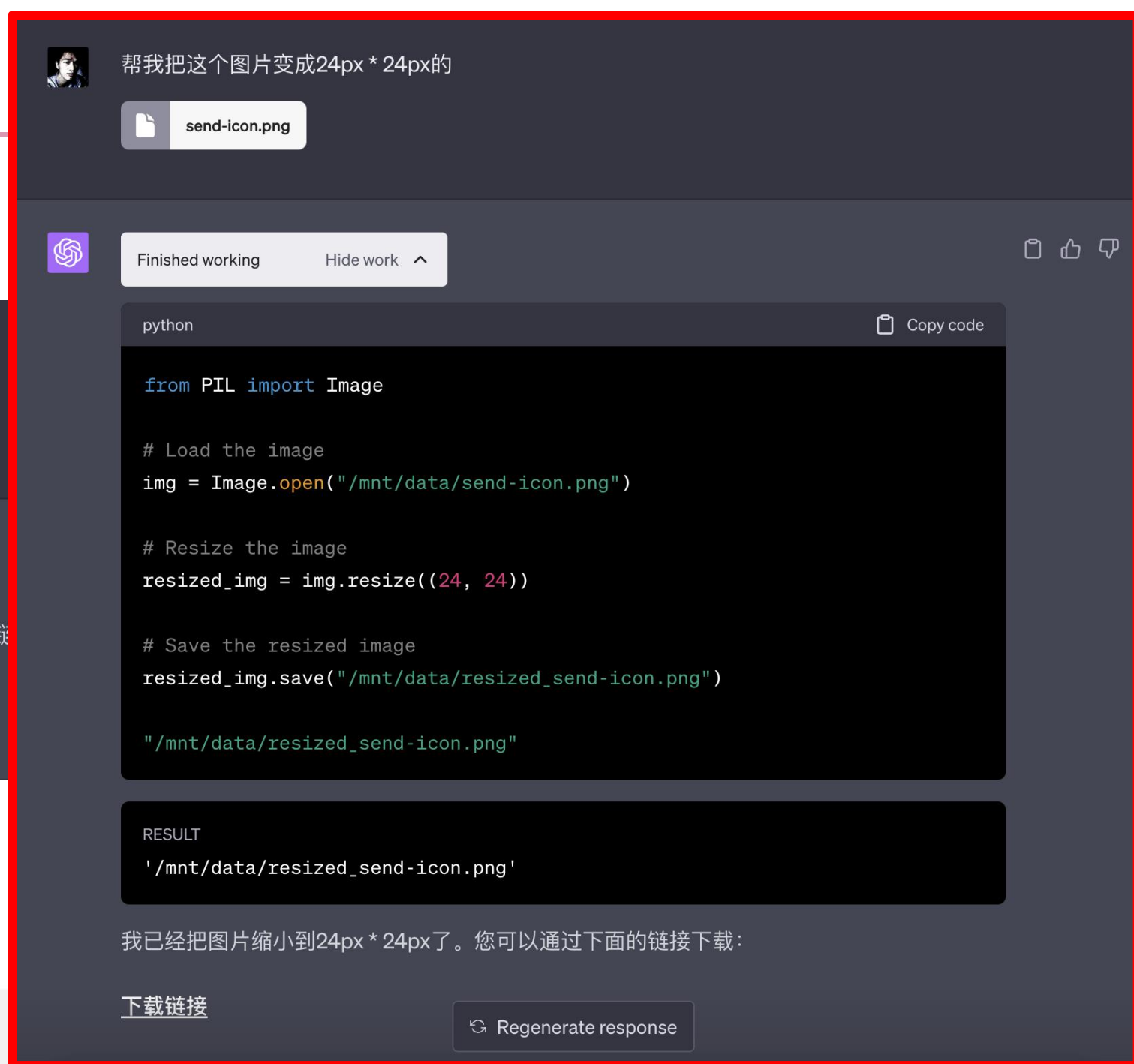
await addSQLSource("select sex, count(
compId: "public-w-chart-pie-base",
title: "嘉宾男女比例",
collectionId: "guestSexRate",
name: "嘉宾的男女比例"
});

BI生成 2080

撤销操作

输入您的需求

全新UX —— 消失的软件？



全新UX —— 消失的软件 之 『0分钟搞定Chat-PDF』



总结一下这篇论文

 2210.03629.pdf



Finished working Show work ▾



这篇论文的题目是“ReAct：在语言模型中协同推理和行动”，作者包括 Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik Narasimhan, 以及 Yuan Cao。

该论文研究的是大型语言模型（LLMs）在推理（例如，链式思考提示）和行动（例如，行动计划生成）方面的能力，这两者以往常常被单独研究。在本文中，作者探索了如何利用大型语言模型（LLMs）以交叉方式生成推理迹象和任务特定的行动，使两者之间产生更大的协同效应：推理迹象可以帮助模型引导、跟踪和更新行动计划，以及处理异常，而行动可以使其接口与并从外部源（如知识库或环境）获取额外的信息。作者将这种方法称为“ReAct”。

作者将 ReAct 应用到了多种语言和决策任务中，并证明其在提高人类的可解释性和可信度方面的有效性，相比于最新的基线，其效果更好。具体来说，在问答（HotpotQA）和事实验证（Fever）中，ReAct 克服了链式思考推理中普遍存在的幻觉和错误传播问题，通过与一个简单的维基百科 API 进行交互，并生成与基线无法提供推理迹象的人类类似的任务解决路径。此外，在两个交互式决策制定基准（ALFWorld 和 WebShop）中，ReAct 的成功率分别比模仿和强化学习方法高出了 34% 和 10%，而且只需要一到两个上下文示例进行提示。

全新UX —— 趋势小结

软件回归解决需求本身，而非交互界面

复杂的交互界面更小众，ChatUI成主流

预先设计的、结构化、固化的软件功能会减少

大量个性化的需求可快速得到满足，传统软件开发模式难以照顾到

软件开发的AutoPilot

软件开发Auto Pilot之路

软件开发从Copilot 到 Auto Pilot

端到端内容生成（经典AIG Content）

AI生成最终产品，它们可以直接被最终消费者使用。例如文本、图片、音频和视频，它们都是可以直接供给最终用户的。

中间产物生成（AIG Code）

AI生成的中间产品，它们往往需要进一步的加工和使用。例如生成的代码片段和完整的复杂代码，它们通常被用作开发更复杂系统或应用的基础。

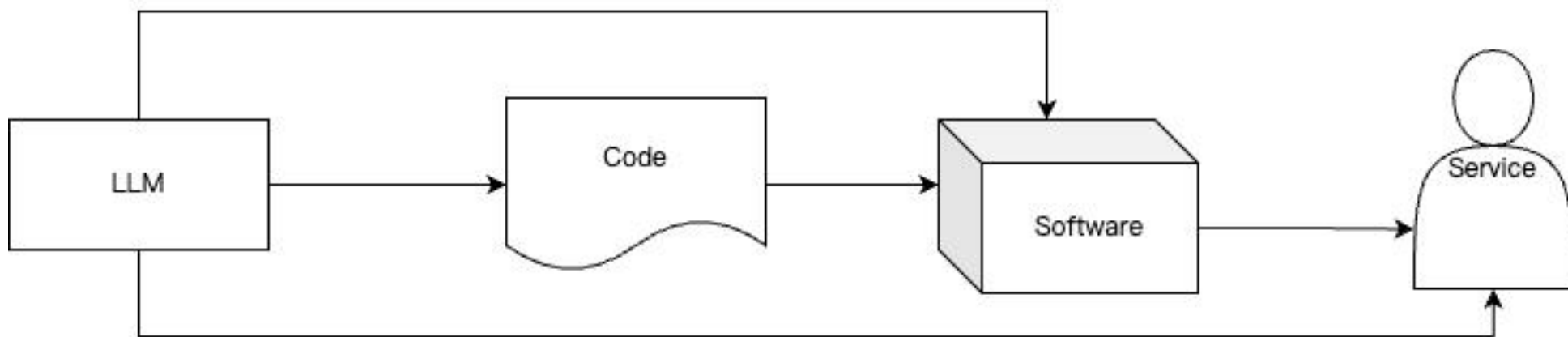
任务与动作生成（AIG aCtion）

AI生成的具体动作或任务，它们常常作为粘合层，连接不同系统和应用，以完成更复杂的任务。例如自然语言交互，复杂任务执行等。

软件开发的Auto Pilot —— 终局

端到端内容生成（经典AIG Content）

AI生成最终产品，它们可以直接被最终消费者使用。例如文本、图片、音频、视频、**代码、软件、服务**，它们都是可以直接供给最终用户的。



软件开发的Auto Pilot —— self refine agents

With programing tools



Auto GPT



Baby AGI

Plan and Execute Agent



LangChain

软件开发的Auto Pilot —— self refine + 人类反馈 = GPT Engineer

```
(base) → gpt_engineer git:(main) x bat README.md
(base) → gpt_engineer git:(main) x mkdir snake
(base) → gpt_engineer git:(main) x vim snake/main_prompt
(base) → gpt_engineer git:(main) x python main.py snake
```

Anything unclear?

Important areas that need clarification:

1. Game rules and mechanics: How exactly does the snake move, grow, and interact with other players? Are there any power-ups or special game elements?
2. Player connections: How many players can join a game? Is there a lobby system or matchmaking?
3. Game state updates: How frequently should the game state be updated and streamed to the players? What is the desired latency?
4. User interface: What should the game interface look like? Are there any specific design requirements or preferences?
5. Game controls: How do players control their snake? Are there any additional controls or settings?
6. Game end conditions: How does the game end? Is there a scoring system or leaderboard?
7. Code structure: Are there any specific requirements for the organization of the Python backend, HTML, and JavaScript code?

Clarifying question:

Can you please provide more details on the game rules and mechanics, such as how the snake moves, grows, and interacts with other players?

(answer in text, or "q" to move on)

No limit, keep it as simple to implement as possible

Anything unclear?

4 remaining questions.

Can you please provide more details on the user interface, such as the desired design and layout?

(answer in text, or "q" to move on)

软件开发的Auto Pilot —— 我们是怎样开发软件的？



换个角度，如何软件开发的Auto Pilot是个管理问题呢？

顶级LLM(gpt-4)的单兵能力已经超过大多数行业的大多数人

LLM + 工具可完成单一功能代码编写、修复、功能验证，最大化减小幻觉

记忆不再是问题，大容量Token、摘要、Embedding多手段解决

引入结构化项目管理

软件开发的Auto Pilot —— 结构化协同的AI Agent团队

职业Agent：项目经理、需求分析师、架构师、程序员、测试，针对职业深度订制Prompt和工具

如给AI程序员提供工具：

- 本地代码文件读写能力
- 代码执行的验证的环境
- 命令行工具

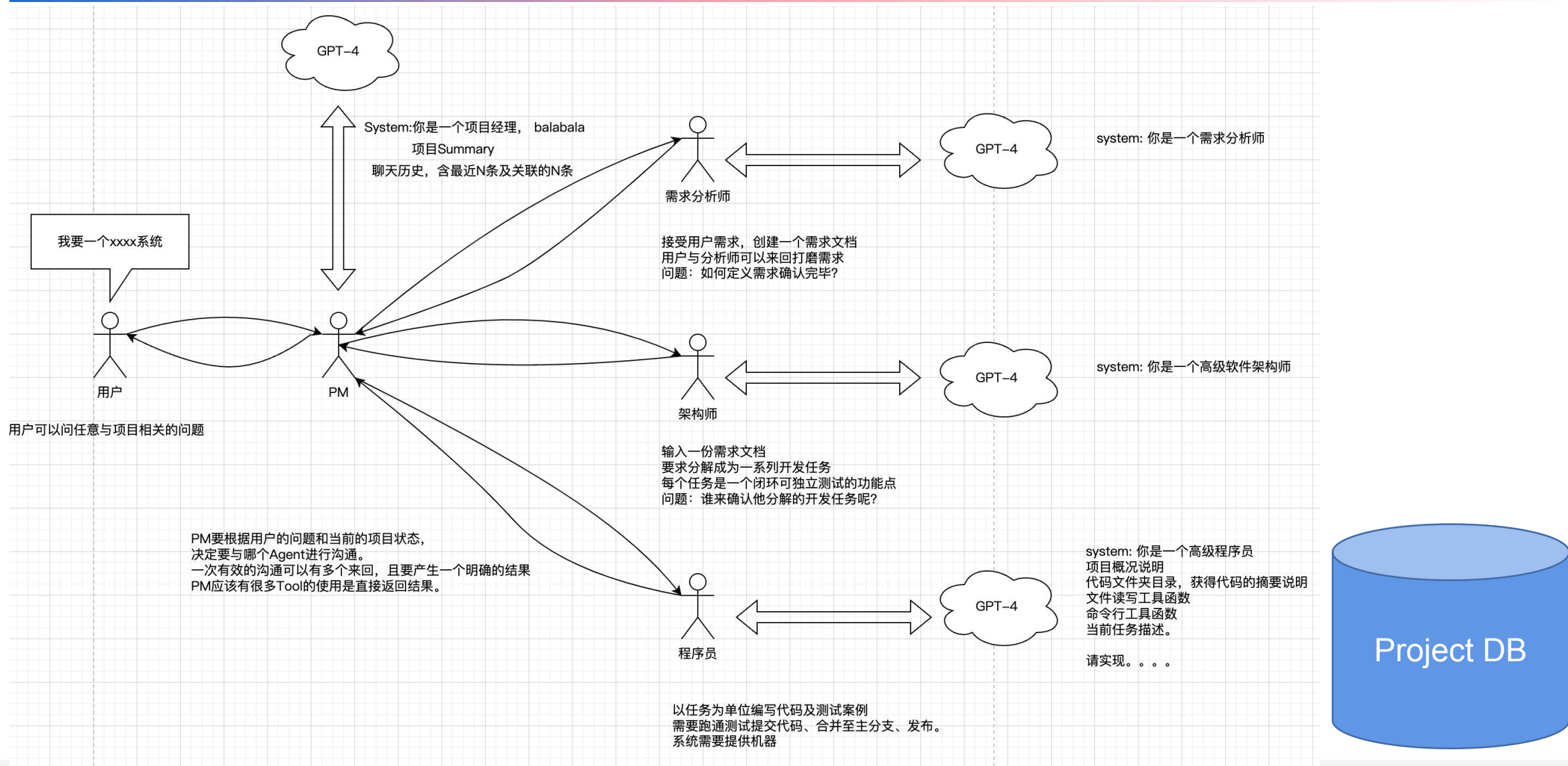
项目研发管理流程工具化：人与Agent、Agent与Agent关于项目、需求、代码、任务协同工具

大项目拆解成需求，需求拆解成任务

职业Agent们围绕任务进行协同，可随时停止和断点续写

人作为copilot参与需求确认、方案评审等

软件开发的Auto Pilot —— 结构化协同的AI Agent团队



软件开发的Auto Pilot —— MetaGPT，职业Agent的范例

```
from metagpt.software_company import SoftwareCompany
from metagpt.roles import ProjectManager, ProductManager, Architect, Engineer

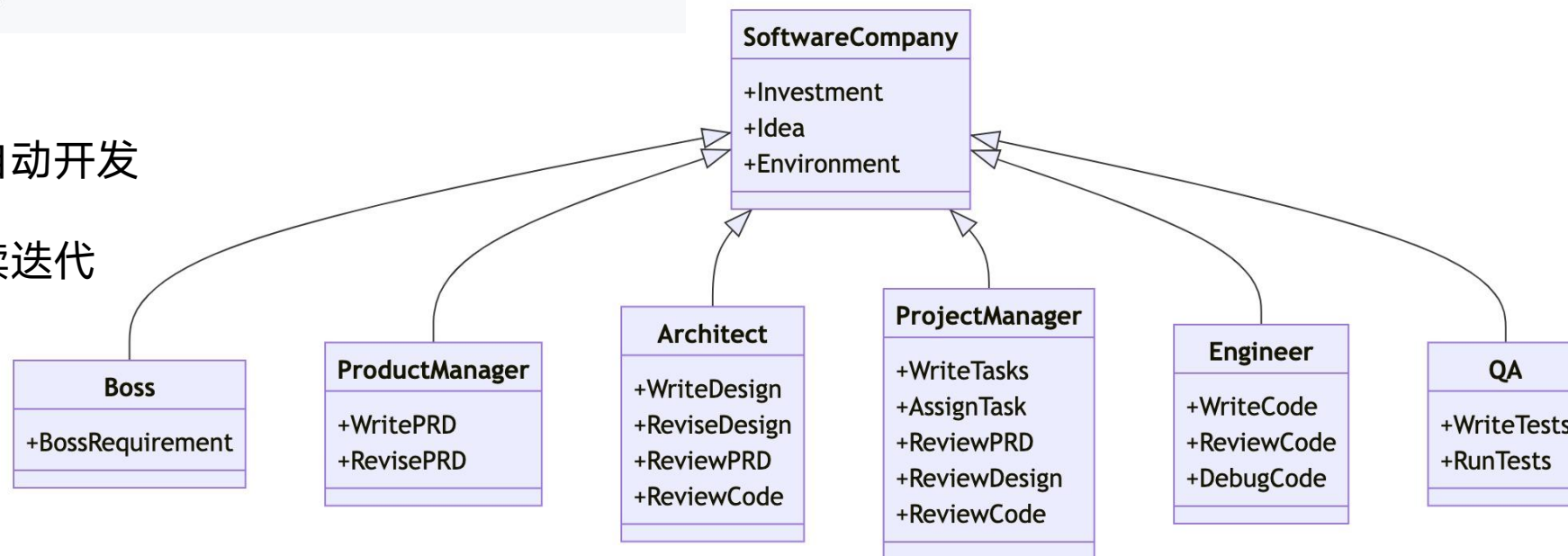
async def startup(idea: str, investment: float = 3.0, n_round: int = 5):
    """Run a startup. Be a boss."""
    company = SoftwareCompany()
    company.hire([ProductManager(), Architect(), ProjectManager(), Engineer()])
    company.invest(investment)
    company.start_project(idea)
    await company.run(n_round=n_round)
```



目标更泛化，不只专注于软件自动开发

一波流，无状态，产物不可持续迭代

缺乏人类参与反馈机制



未来，开发还存在吗

我们又将如何自处？



THANKS