



Multiagentní systémy s ChatGPT a knihovnou LangGraph

Michal Vašínek

Odborný asistent, katedra informatiky FEI, VŠB-TUO

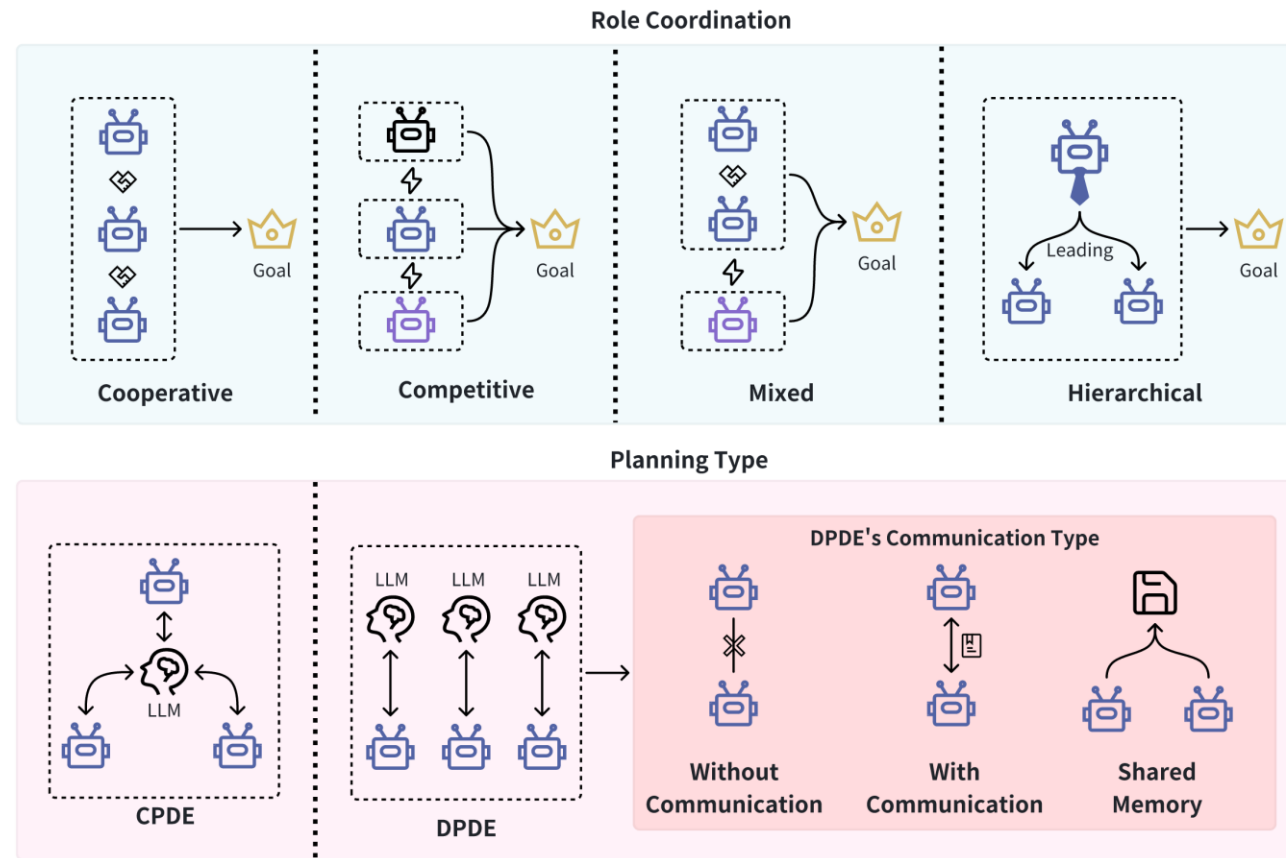


Multiagentní systemy

Multiagentní systém (MAS)

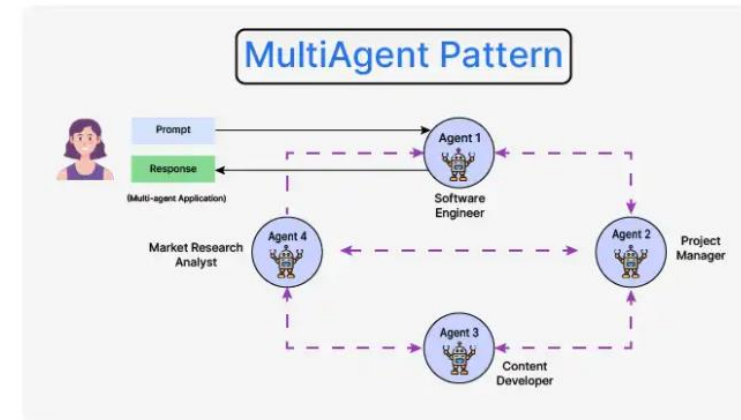
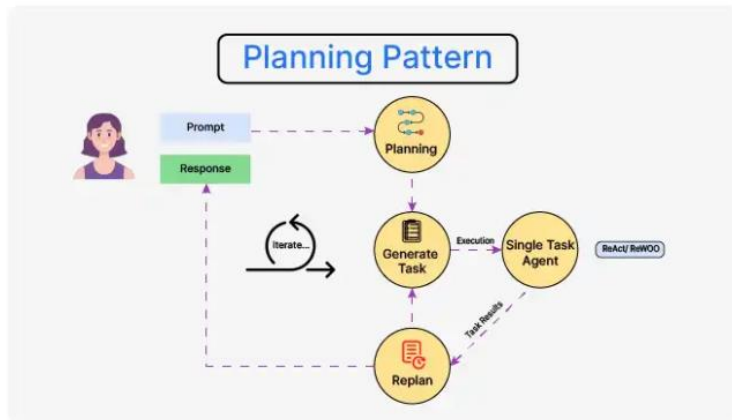
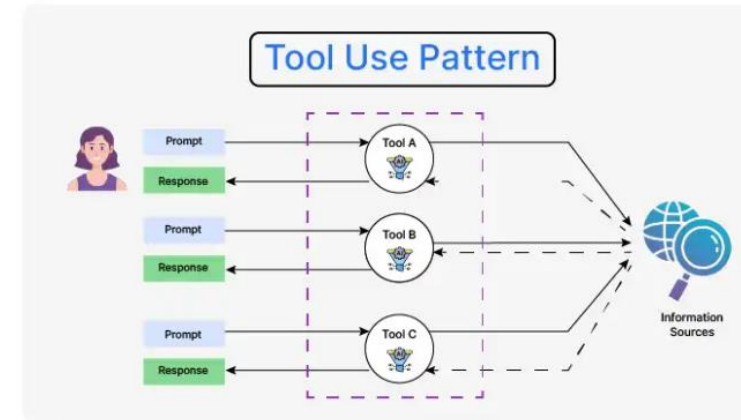
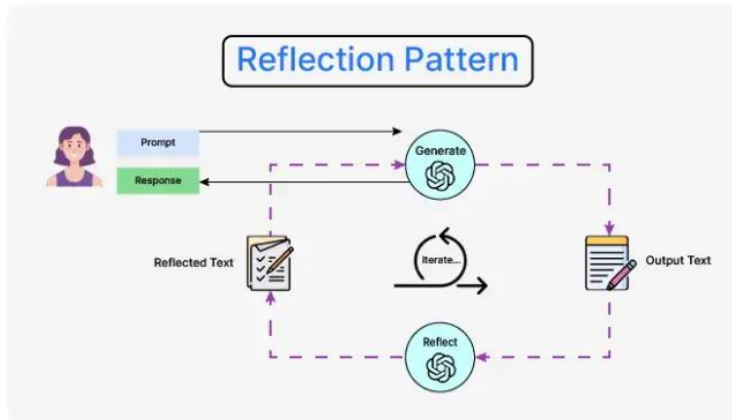
- MAS je kolekcí autonomních agentů, kteří spolupracují a komunikují.
- Vývoj od počátku 70 let, především v oblasti robotiky.
- Agenti jsou schopni přijímat zprávy z okolí, rozhodovat se a provádět akce.
- Agent nemusí být nutně počítačem řízený objekt.
- Různí agenti dostávají zodpovědnost za různé činnosti.

Vztahy mezi agenty



Zdroj: Cheng, et al., Exploring Large Language Model based Intelligent Agents: Definitions, Methods, and Prospects, arxiv.org

Návrhové vzory agentních systémů



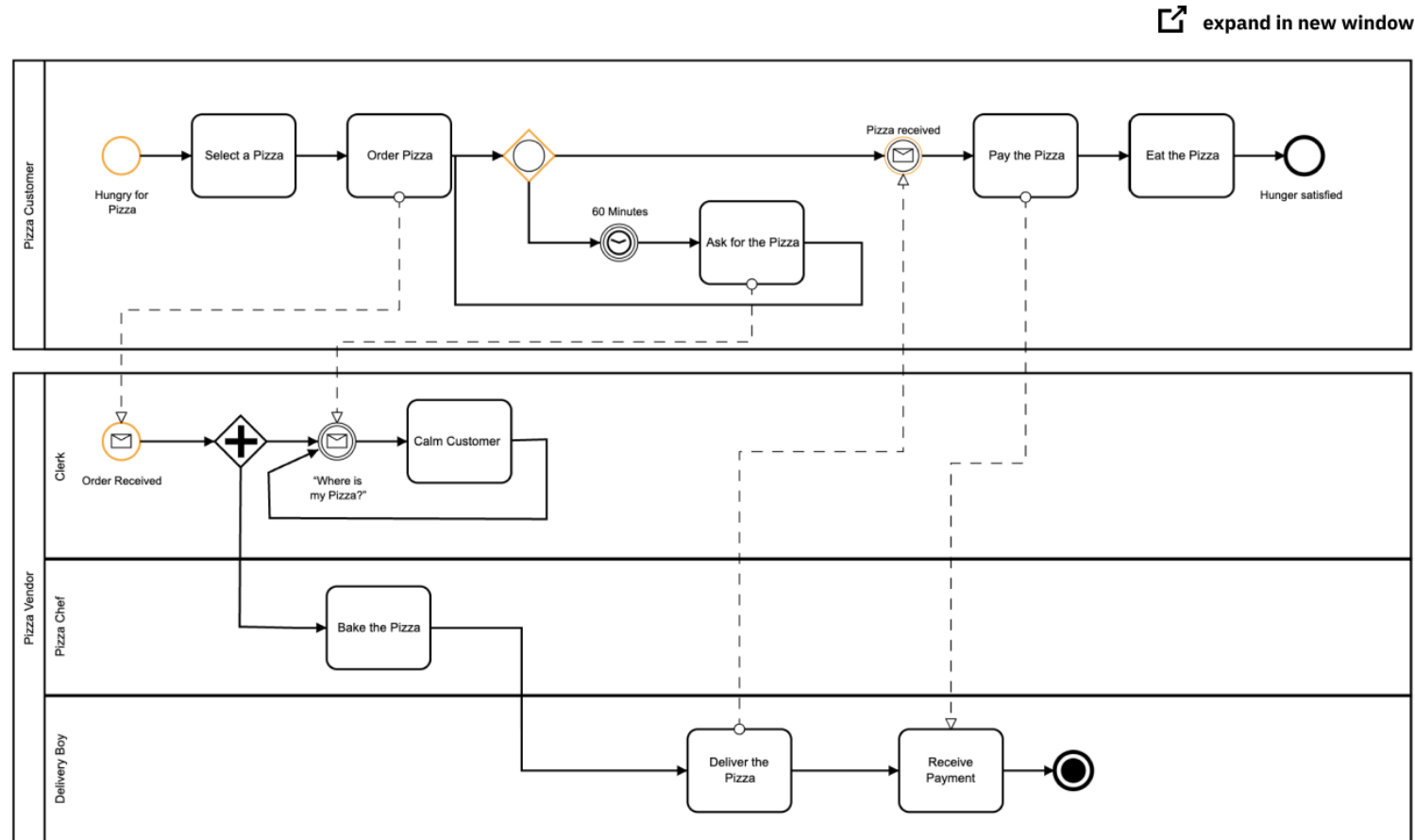
Zdroj: [Top 4 Agentic AI Design Patterns - Analytics Vidhya](#)

Business procesy

Business proces

- Posloupnost kroků a operací vedoucích k dosažení cíle.
- Cíle interní nebo vytvoření hodnoty pro zákazníka.
- Prvky procesu:
 - Vstupy – zdroje, data, např.
 - Aktivita – jednotlivé kroky a operace
 - Výstupy – konečná hodnota, produkt
 - Zdroje – lidé, technologie, nebo systémy
 - Kontrola – ověření a zajištění, že proces funguje správně, zpětná vazba

Business proces – objednání pizzy



Zdroj: [BPMN Tutorial: Learn Business Process Model and Notation | Camunda](#)

Analýza business procesu

- Chceme zlepšit efektivitu procesu
 - Identifikace opakujících se činností,
 - Automatizace,
 - Snížení nákladů => \$\$\$
- Chceme zjistit, kde nám může AI pomoci:
 - Zpracování faktur,
 - Analýza obrazu,
 - Řízení dodavatelský řetězců,
 - Zákaznická podpora
 - Marketing



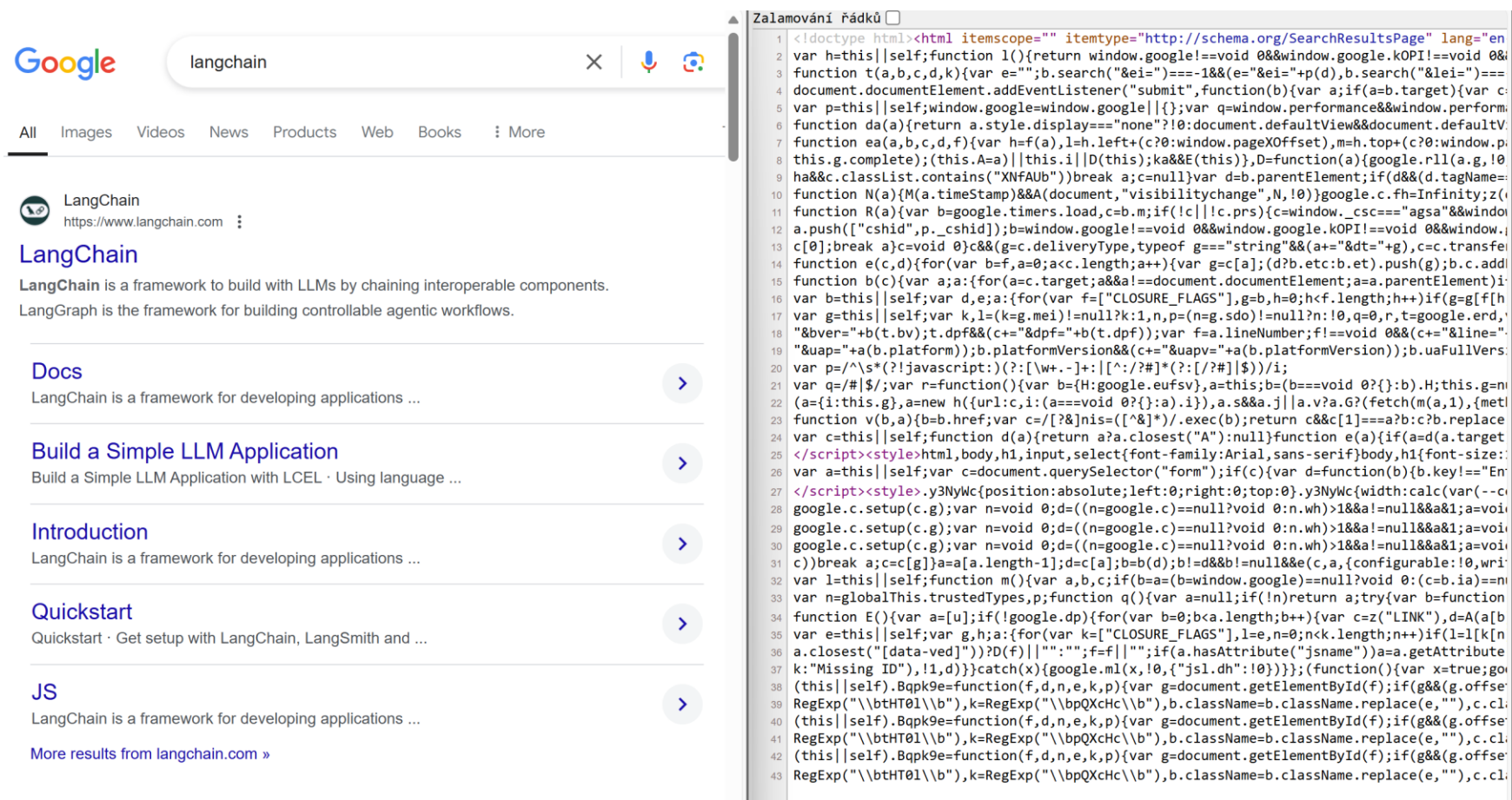
LangChain

LangChain - historie

- Framework pro práci s velkými jazykovými modely (LLM)
- Představen v roce 2022, původně pro chaining (řetězení) operací
- Abstrakce od konkrétního poskytovatele LLM (ChatGPT, Gemini, Llama, Groq,...)
- Postupně vyvinut pro interakce LLM s externími nástroji a vývoj multiagentních systémů



LangChain - ekosystém



The image shows a Google search for "langchain". The search results page displays the LangChain logo, the website URL "https://www.langchain.com", and a brief description: "LangChain is a framework to build with LLMs by chaining interoperable components. LangGraph is the framework for building controllable agentic workflows." Below this, there are links to "Docs", "Build a Simple LLM Application", "Introduction", "Quickstart", and "JS".

On the right side of the image, a snippet of JavaScript code is shown, which appears to be a search result snippet. The code is a complex function that handles search results and displays them in a specific format. It includes various DOM manipulation and event handling logic.

LangChain - ekosystém

https://serpapi.com

Google

Coffee

×

🔍

🔍 All

📍 Maps

🖼 Images

🛒 Shopping

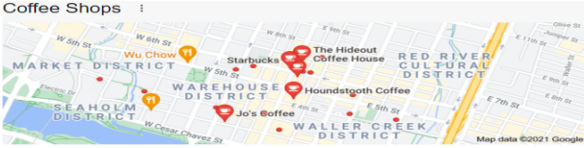
📰 News

⋮ More

Tools

About 2,520,000,000 results (1.13 seconds)

Coffee Shops



Rating

Price

Hours

Starbucks

4.1 ★★★★★ (506) · \$ · Coffee shop

600 Congress Ave

Dine-in · Takeout · Delivery

Houndstooth Coffee

4.6 ★★★★★ (740) · \$ · Coffee shop

401 Congress Ave #100C · In Frost Bank Tower

Dine-in · Takeout · No delivery

Lucky Lab Coffee

3.0 ★★★★★ (2) · Cafe

515 Congress Ave · In the Bank of America Financial Center

Dine-in · Takeout · No delivery

→

View all

https://en.wikipedia.org · wiki · Coffee

Coffee - Wikipedia

Coffee is a brewed drink prepared from roasted coffee beans, the seeds of berries from certain Coffea species. From the coffee fruit, the seeds are ...

Region of origin: Horn of Africa and South Ar... Color: Black, dark brown, light brown, beige

Introduced: 15th century

History · Coffee bean · Coffee preparation · Coffee production

People also ask



Is coffee good for your health?

What are the 4 types of coffee?

Does Austin have good coffee?

What does coffee do to your body?

Feedback



Coffee

Drink

Coffee is a brewed drink prepared from roasted coffee beans, the seeds of berries from certain Coffea species. From the coffee fruit, the seeds are separated to produce a stable, raw product: unroasted green coffee. Wikipedia

Patron saint: Saint Drogo of Sebourg

redrockroasters.com

Nutrition Facts

Coffee

Sources include: USDA

Amount Per 1 cup (8 fl oz) (237 g)

Calories 1

	% Daily Value*
Total Fat 0 g	0%
Saturated fat 0 g	0%
Trans fat regulation 0 g	
Cholesterol 0 mg	0%
Sodium 5 mg	0%
Potassium 116 mg	3%
Total Carbohydrate 0 g	0%
Dietary fiber 0 g	0%
Sugar 0 g	
Protein 0.3 g	0%
Caffeine 95 mg	
Vitamin C 0%	Calcium 0%
Iron 0%	Vitamin D 0%
Vitamin B6 0%	Cobalamin 0%
Magnesium 1%	

*Percent Daily Values are based on a 2,000 calorie diet. Your daily values may be higher or lower depending on your calorie needs.

Tools

```
{
  "search_metadata": {
    "id": "6165916694c6c7025deef5ab",
    "status": "Success",
    "json_endpoint": "https://serpapi.com/searches/87fa874d05a7fcc5/6165916694c6c7025deef5ab.json",
    "created_at": "2021-10-12 13:45:10 UTC",
    "processed_at": "2021-10-12 13:45:11 UTC",
    "google_url": "https://www.google.com/search?q=Coffee&oeq=Coffee&uule=w+CAIQICIAQXVzdGluLFR1eGFZl",
    "raw_html_file": "https://serpapi.com/searches/87fa874d05a7fcc5/6165916694c6c7025deef5ab.html",
    "total_time_taken": 1.85
  },
  "search_parameters": {
    "engine": "google",
    "q": "Coffee",
    "location_requested": "Austin, Texas, United States",
    "location_used": "Austin,Texas,United States",
    "google_domain": "google.com",
    "hl": "en",
    "gl": "us",
    "device": "desktop"
  },
  "search_information": {
    "organic_results_state": "Results for exact spelling",
    "query_displayed": "Coffee",
    "total_results": 2520000000,
    "time_taken_displayed": 1.13
  },
  "recipes_results": [
    {
      "title": "20 Great Coffee Drinks",
      "link": "https://www.acouplecooks.com/coffee-drinks/",
      "source": "A Couple Cooks",
      "total_time": "5 min",
      "ingredients": [
        "Moka pot",
        "dairy free"
      ],
      "thumbnail": "https://encrypted-tbn0.gstatic.com/images?q=tbn:ANd9GcTSfn3D91nInvziiA2XG6VNbLO9K"
    },
    {
      "title": "How to Make Iced Coffee",

```

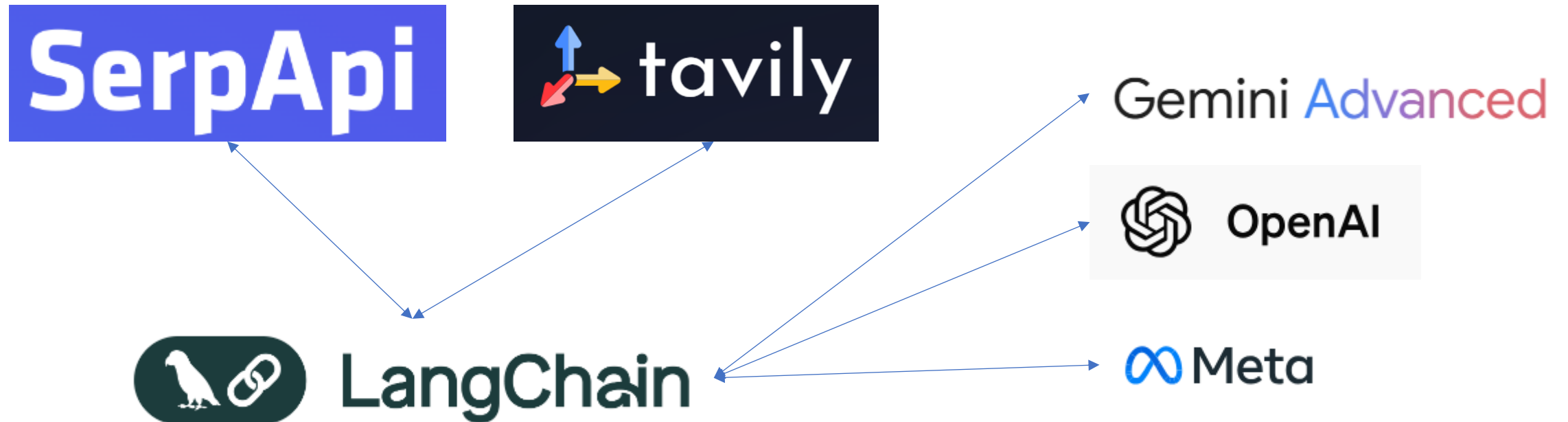
13

LangChain - ekosystém

```
{
  "title": "How to Make Iced Coffee",
  "link": "https://www.recipegirl.com/how-to-make-iced-coffee/",
  "source": "Recipe Girl",
  "total_time": "5 min",
  "ingredients": [
    "Simple syrup",
    "cream",
    "coffee frozen"
  ],
  "thumbnail": "https://encrypted-tbn0.gstatic.com/images?q=tbn:A",
},
```

LangChain - ekosystém

- Interakce agentů s externími nástroji a zdroji informací



LangChain - ekosystém

- Celá řada nástrojů a služeb, které zjednodušují jednotlivé úkony
- [Tools | !\[\]\(cd3e54d951a9fb854f48e4697cf550f9_img.jpg\) !\[\]\(cc729e263f29c0a76fbdc4cfe67fceb0_img.jpg\) LangChain](#)

Tool/Toolkit	Free/Paid	Return Data
Bing Search	Paid	URL, Snippet, Title
Brave Search	Free	URL, Snippet, Title
DuckDuckgoSearch	Free	URL, Snippet, Title
Exa Search	1000 free searches/month	URL, Author, Title, Published Date
Google Search	Paid	URL, Snippet, Title
Google Serper	Free	URL, Snippet, Title, Search Rank, Site Links

Cena interakce s ChatGPT

- Účetní jednotkou interakce s LLM modely je token
- Služby LLM jsou zpoplatněny na základě spotřebovaných tokenů

Model	Pricing
gpt-4o	\$2.50 / 1M input tokens
	\$1.25 / 1M cached** input tokens
	\$10.00 / 1M output tokens
gpt-4o-2024-08-06	\$2.50 / 1M input tokens
	\$1.25 / 1M cached** input tokens
	\$10.00 / 1M output tokens

Token – co a za kolik

GPT-4o & GPT-4o mini

GPT-3.5 & GPT-4

GPT-3 (Legacy)

Ahoj všem, jak se máte?

Clear

Show example

Tokens

10

Characters

23

Ahoj všem, jak se máte?

<https://platform.openai.com/tokenizer>

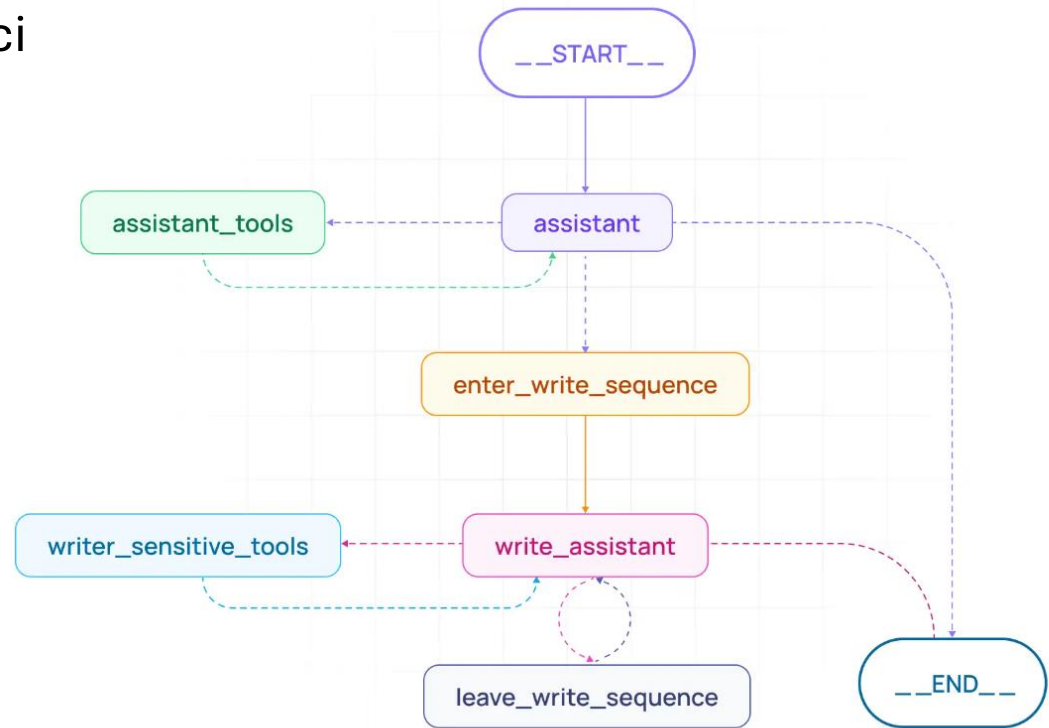
Cena interakce se SerpApi

Plan	Price / month	Searches / month	Legal US Shield
DEVELOPER	\$75	5,000	No
PRODUCTION	\$150	15,000	Yes
BIG DATA	\$275	30,000	Yes
FREE	\$0	100	No

LangGraph

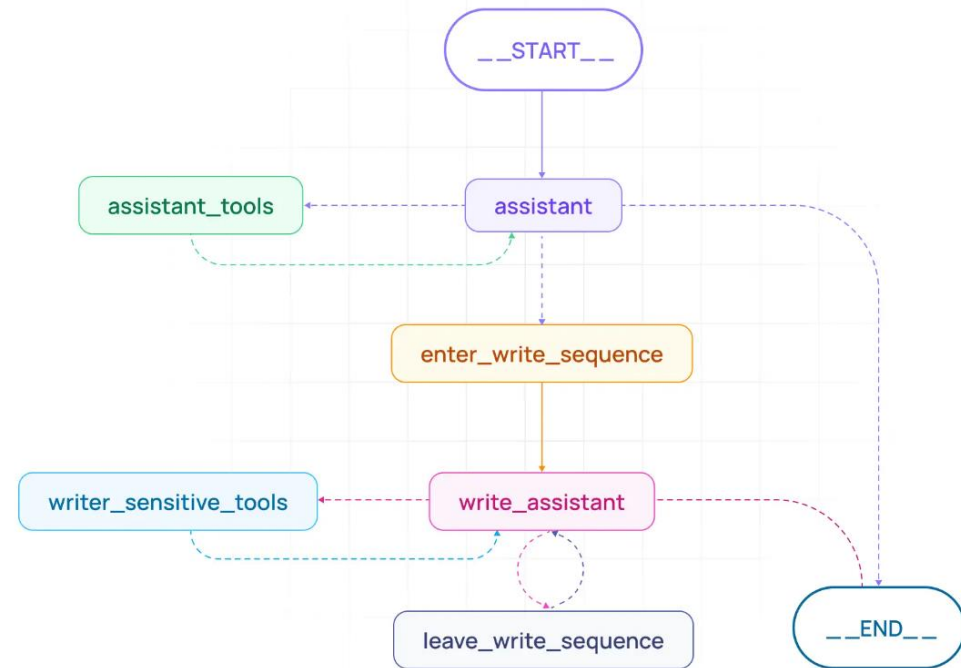
LangGraph

- Framework umožňující vytváření a interakci agentů založených (nejen) na LLM
- Integrovan s LangChainem
- Umožňuje cykly a větvení



LangGraph

- Agenti organizování v orientovaném grafu
- Uzly reprezentují agenty
- Hrany reprezentují tok informací mezi agenty, jejich komunikaci
- Počáteční a koncové stavy



LangGraph – stav MAS

- Stav systému popisujeme proměnnými
- Jednotliví agenti MAS modifikují stav systému
- Agenti reagují na základě stavu systému, tj. hodnot proměnných

```
class AgentState(TypedDict):  
    variable1: str  
    variable2: int  
    listVariable: List[int]
```

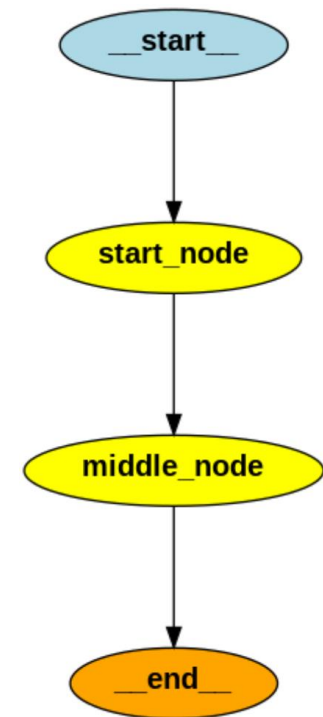
LangGraph – sestavení grafu

- Voláním metod **add_node** a **add_edge** sestavíme graf multiagentního systému

```
builder = StateGraph(AgentState)
builder.add_node("start_node", start_agent)
builder.add_node("middle_node", middle_agent)

builder.add_edge("start_node", "middle_node")
builder.add_edge("middle_node", END)

builder.set_entry_point("start_node")
graph = builder.compile()
```

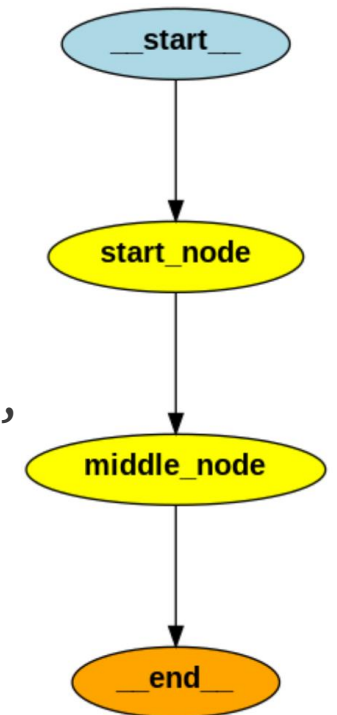


LangGraph – agent

- Agent je reprezentován funkcí, která obdrží a modifikuje stav multiagentního systému

```
builder.add_node("start_node", start_agent)
```

```
def start_agent(state: AgentState):  
    messages = [  
        SystemMessage(content="Jsi milý, komunikační asistent."),  
        HumanMessage(content="Ahoj."),  
    ]  
    response = model.invoke(messages)  
    return {"response": response.content}
```



LangGraph – ChatGPT

- Připojení vlastním OpenAI API klíčem.
- Řešíme přes platform.openai.com

```
import os
from langchain_openai import ChatOpenAI
model = ChatOpenAI(model="gpt-4o",
openai_api_key=os.environ["OPENAI_PRIVATE_API_KEY"])

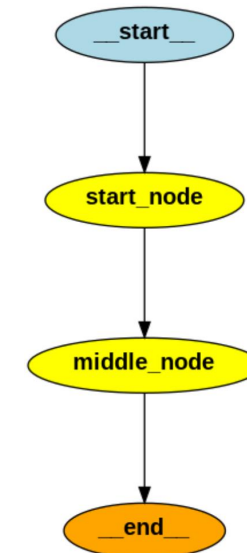
...

model.invoke(messages)
model.with_structured_output(Structured).invoke(messages)
```

LangGraph – spuštění

- Postupný průchod grafem pomocí `graph.stream` metody
- Každý průchod grafem může mít přiřazeno unikátní `id`, abychom mohli vést více konverzací zároveň

```
thread = {"configurable": {"thread_id": "1"}}
state = None
for s in graph.stream({
    'msg': "Ahoj světe!"
}, thread):
    state = s
```



Praktická ukázka

Emaily

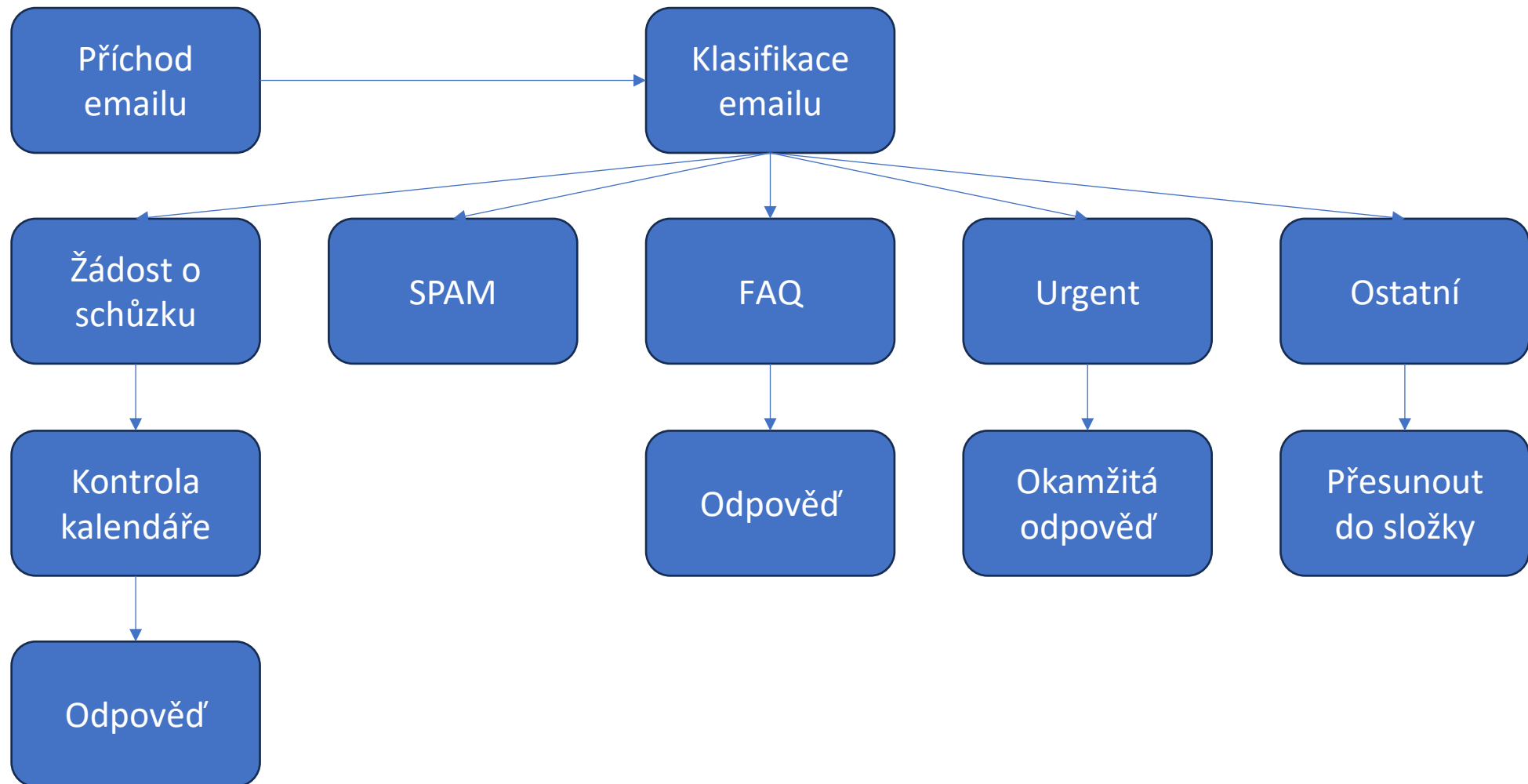


DALL-E: Vytvoř obrázek pro moji prezentaci, na obrázku by mělo být vystihnuto, že se uživatel může zbláznit z množství emailů, které mu denně dojdou.

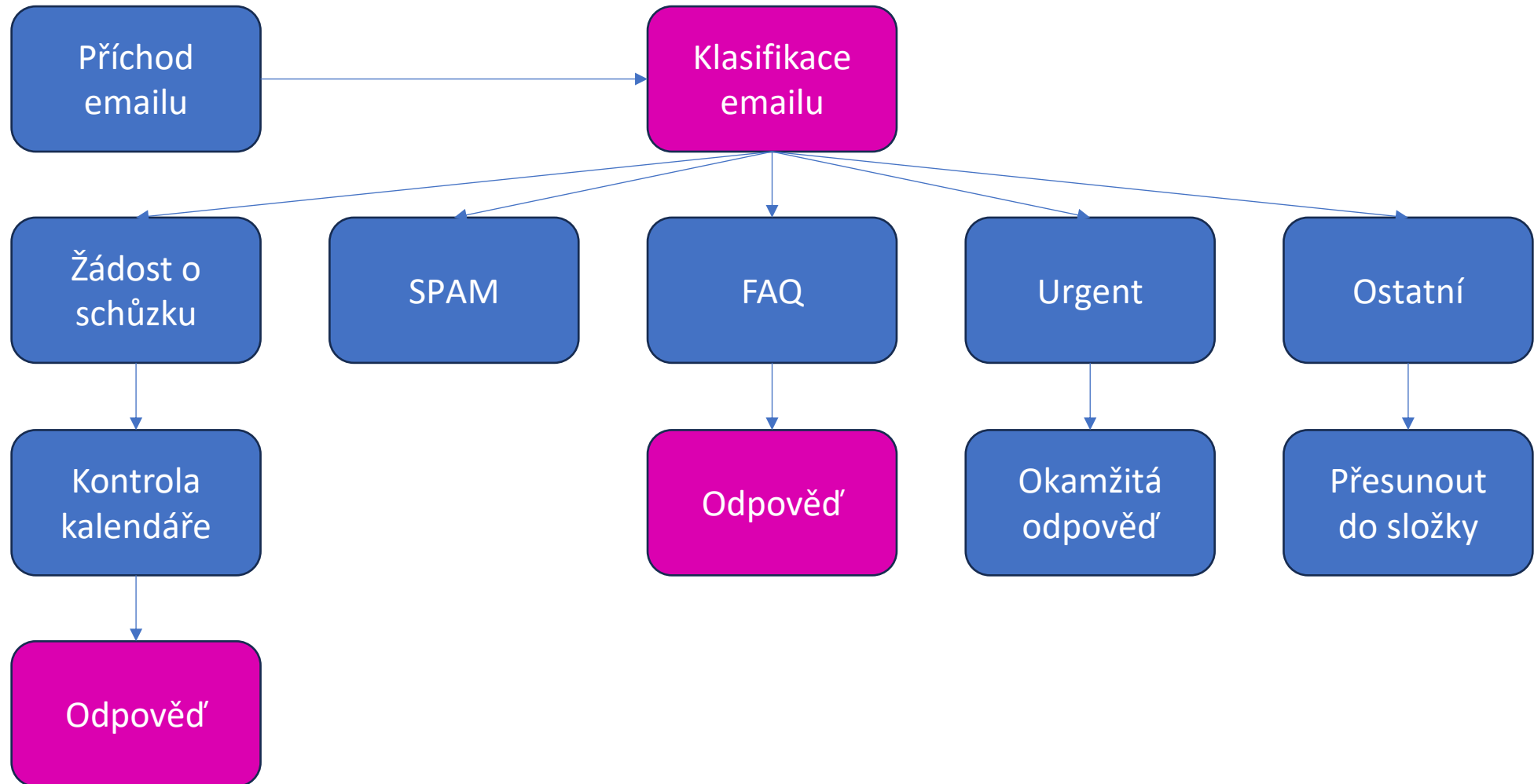
Ředitelé mají asisten-ty/tky

- Zpracování emailu - aktivita, kterou řešíme všichni každý den
- Jak a kdy reagujeme na emaily?
 - Mají nějaké emaily prioritu, čím jsou charakteristické
- Existuje skupina emailů, které nepotřebují mou interakci?
 - Dají se zodpovědět beze mě => **automatizace**
- Jsou emaily, k jejichž zodpovězení stačí databáze znalostí?
 - Retrieval augmented generation (RAG)
- Je email nositelem zajímavé informace?
 - Nezahazovat, ale zobrazit/přesunout do složky „zajímavé“

Business proces – zpracování emailu



Kde můžeme použít AI



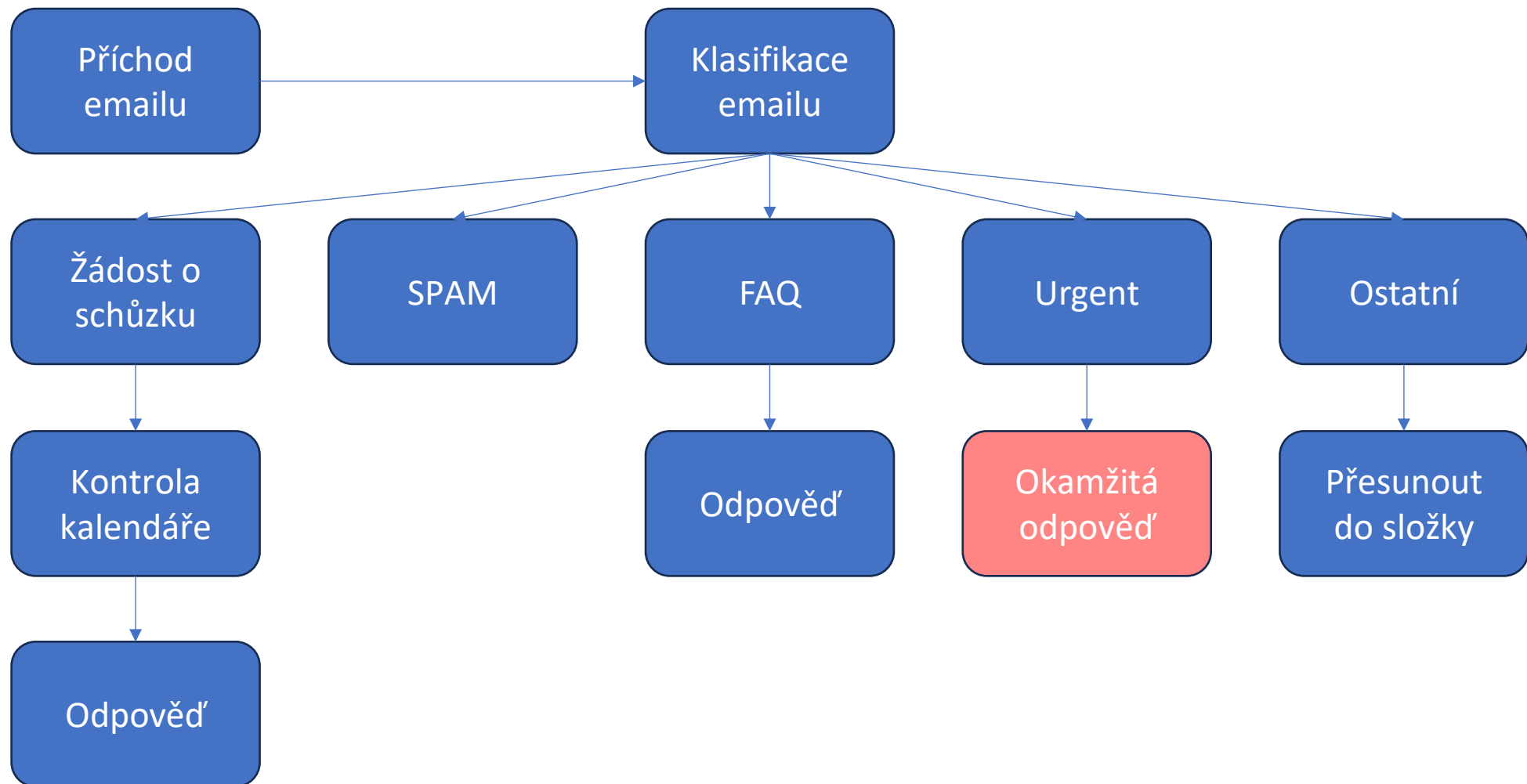
Agenti

Klasifikátor
emailů

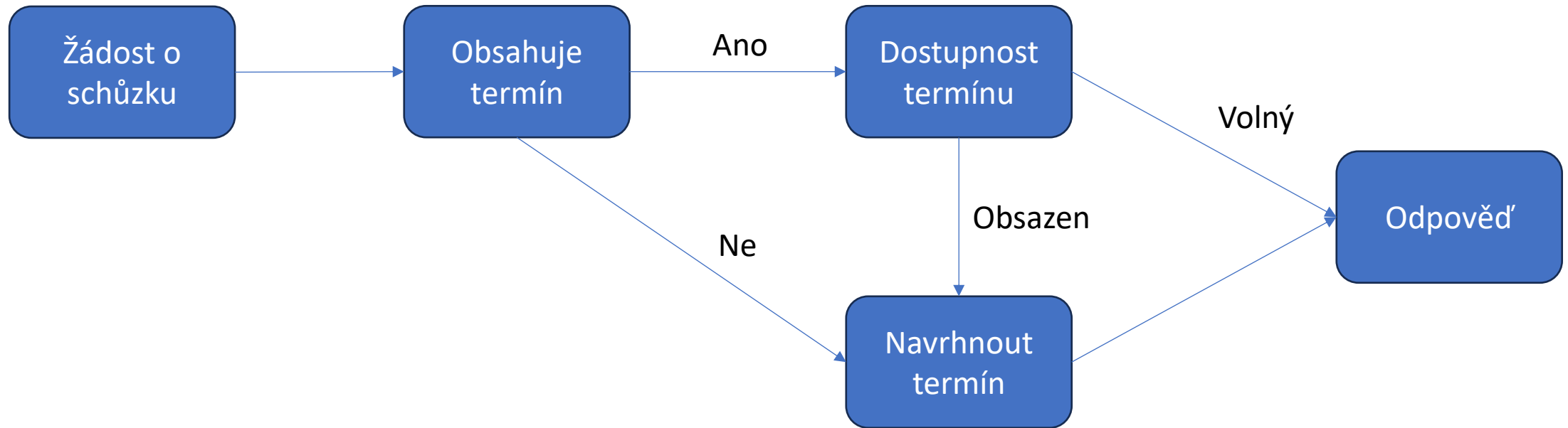
Asistent
schůzek

Asistent
FAQ

Business proces – human agent



Business proces – zpracování emailu



Klasifikátor

```
MAIL_CLASSIFY_PROMPT = """
```

```
Jsi asistent, který je expertem na klasifikaci e-mailů podle obsahu. \  
Klasifikuj e-mail do jedné z následujících kategorií dle jejich popisků. \  
Zároveň se podívej, zda e-mail obsahuje nějakou otázku, která by mohla být v FAQ. \  

```

```
Email: \  
{email} \  

```

```
Kategorie: \  
{categories} \  

```

```
FAQ: \  
{faq} \  
"""
```

Asistent - schůzky

```
RESPONSE_MEETING_PROMPT = """
```

```
Jsi asistent, který odpovídá na e-maily na domluvení schůzky. \  
Napiš odpověď na následující email: \  

```

```
Email: \  

```

```
{email} \  

```

```
Pokud je schůzka možná dle volných termínů, napiš, že je možné se sejít. \  
Pokud schůzka neodpovídá volným termínům, navrhní vhodné volné termíny. Dle níže uvedených  
volných termínů. \  

```

```
Volné termíny: \  

```

```
{free_times}
```

```
"""
```

Few shot prompt

- Dáme klasifikátoru příklady nebo popisky pro předvolené kategorie

```
MAIL_CATEGORY = {  
    ("SPAM", "Nevyžádaná pošta, nevyžádané nabídky, phishingové  
emaily."),  
    ("FAQ", "Často kladené otázky."),  
    ("MEETING", "Email s žádostí o schůzku."),  
    ("URGENT", "Email vyžadující okamžitou odpověď."),  
    ("OTHER", "Ostatní emaily."),  
}
```

Retrieval Augmented Generation

- Doplníme chatbota o externí zdroje informací
- Databáze, dokumenty, webové stránky

```
FAQ = [  
    ("Lze dopsat test, pokud jsem byl nemocný?", "Ano, lze na dalším  
    cvičení."),  
    ("Jaké máte termíny konzultací?", "Konzultace probíhají vždy ve  
    středu od 14:00 do 16:00."),  
    ("Jaké je heslo k Wi-Fi?", "Heslo k Wi-Fi je '12345678'."),  
]
```

Napojení externích služeb

- Můžeme připojit k externím službám – např. kalendář
- Interakce s kalendářem v rukou programátora, bez AI

```
FREE_TIMES = [  
    ("pondělí", "12:30"),  
    ("úterý", "14:00"),  
    ("středa", "16:00"),  
    ("čtvrtek", "10:00"),  
    ("pátek", "8:00"),  
]
```


Reflexe – zpětná vazba

- Agent, který má za úkol verifikovat odpověď systému

```
MAIL_REVIEW_PROMPT = """
```

```
Jsi expertem na psaní textů. Zkontroluj odpověď na email, zda je  
gramaticky správně, \  
zda odpovídá na otázku, zda je slušná a formální a zda je vhodná  
pro odeslání. Pokud \  
není v pořádku, navrhní připomínky. \  
"""
```

```
Původní Email:  
{email} \  
"""
```

```
Odpověď:  
{response} \  
"""
```

Excesy

- Detailní zpracování procesu nám může pomoci eliminovat chyby
- Pracuji v Ostravě, co když chatbot tuto informaci nemá:

""

Dobrý den,

tento týden jsem v Praze,
můžeme se sejít na konzultaci
na Staroměstském náměstí
v pondělí ve 12:30?

S pozdravem,

Jan Novák

""

""

Dobrý den pane Nováku,

děkuji za Váš e-mail. Schůzka na
Staroměstském náměstí v pondělí
ve 12:30 je možná. Těším se na
setkání s Vámi.

S pozdravem,

[Vaše jméno]

""

Graf systému

```
builder = StateGraph(AgentState)
builder.add_node("classifier", classifier_node)
builder.add_node("responder", response_node)

builder.add_edge("classifier", "responder")
builder.add_edge("responder", END)

builder.set_entry_point("classifier")
graph = builder.compile(checkpointer=memory)
```

Stav systému

- Stav systému zachycen ve třech proměnných
- Reprezentace vstupního emailu, detekovaná kategorie a vygenerovaná odpověď

```
class AgentState(TypedDict):  
    email: str  
    category: str  
    response: str
```

Klasifikační agent

```
def classifier_node(state: AgentState):  
    categories = [name + "-" + description for name, description in MAIL_CATEGORY]  
    messages = [  
        SystemMessage(content=MAIL_CLASSIFY_PROMPT.format(email=state["email"],  
categories=", ".join(categories), faq=", ".join([q for q, a in FAQ]))),  
        HumanMessage(content="Přiřaď e-mail do kategorie."),  
    ]  
    category = model.with_structured_output(Category).invoke(messages)  
  
    return {"category": category.name}
```

Odpovídající agent

```
def response_node(state: AgentState):  
    if state["category"] == "MEETING":  
        messages = [  
            SystemMessage(content=RESPONSE_MEETING_PROMPT.format(email=state["email"],  
free_times=", ".join([f"{day} v {time}" for day, time in FREE_TIMES]))),  
            HumanMessage(content="Napiš odpověď na e-mail:"),  
        ]  
        response = model.invoke(messages)  
    else:  
        response = "Děkuji za e-mail. Odpovím co nejdříve to bude možné."  
  
    return {"response": response}
```



Díky za pozornost.
Otázky či diskuze !?

github.com/mvasinek



[GitHub](#)



[\(Michal Vašinek |](#)