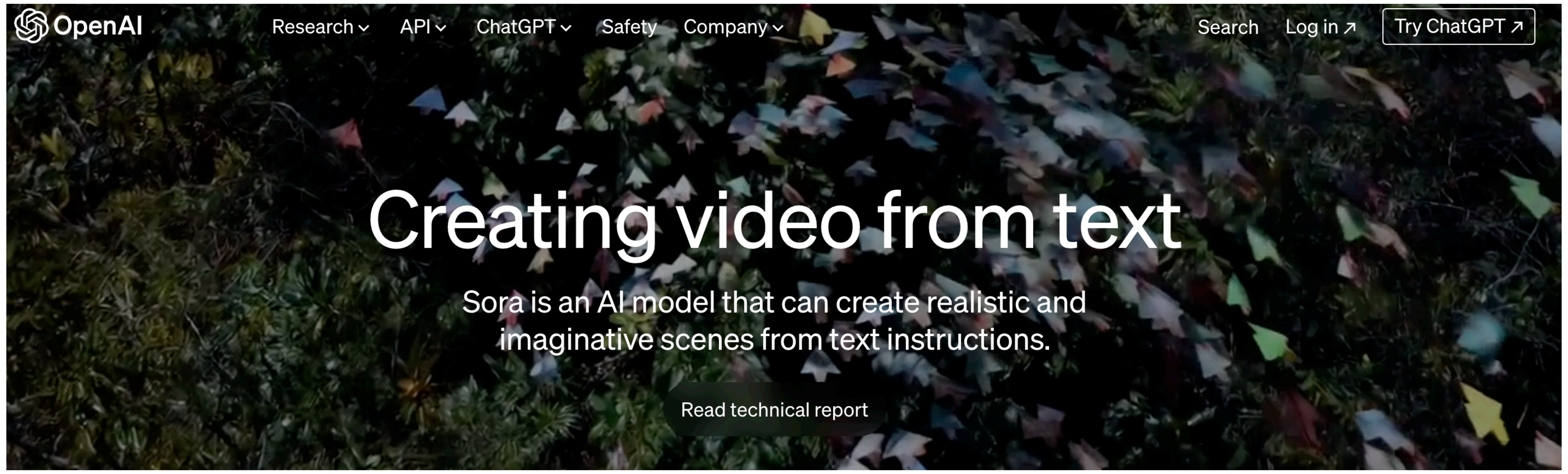
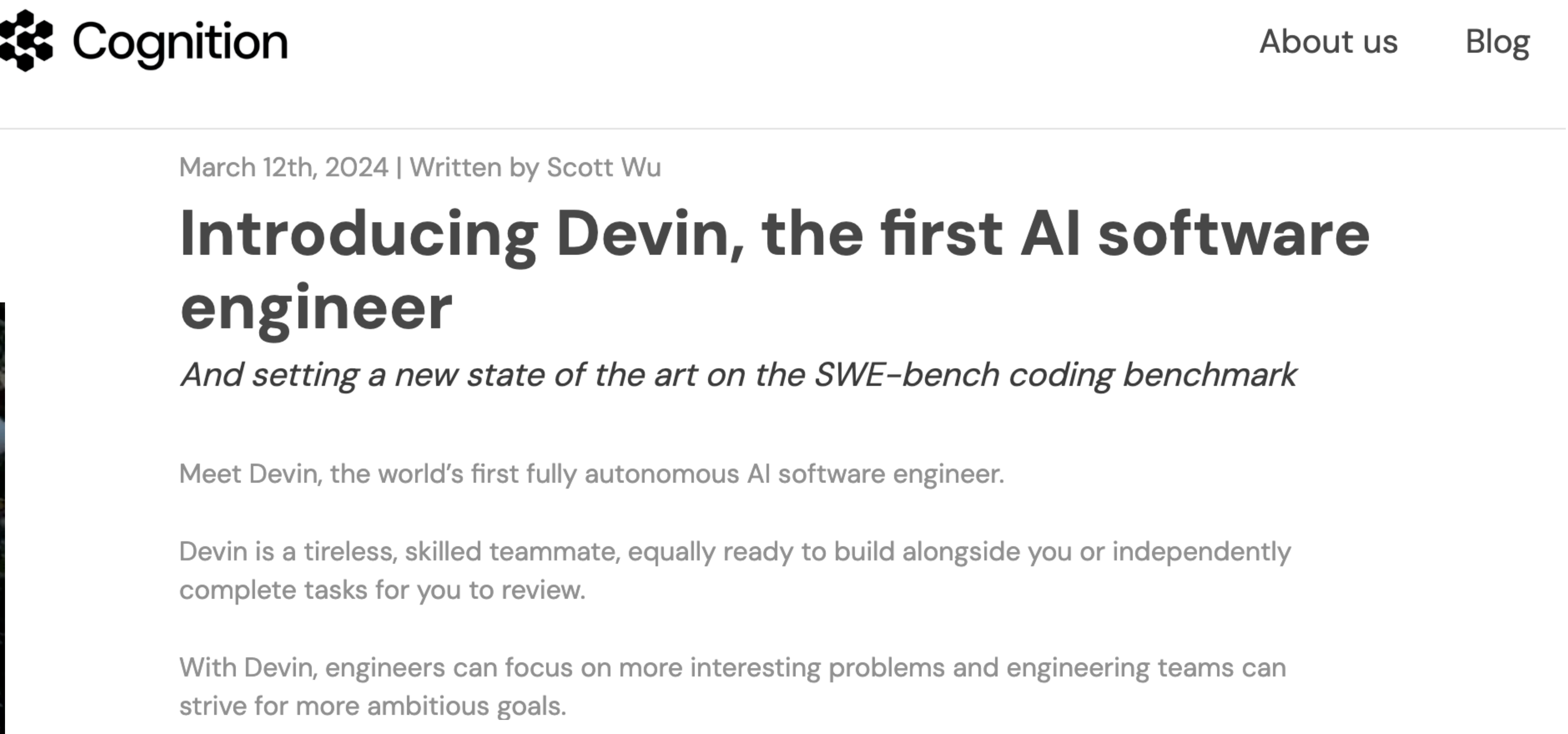
GenAI Everywhere!



Creating video from text

Sora is an AI model that can create realistic and imaginative scenes from text instructions.



March 12th, 2024 | Written by Scott Wu

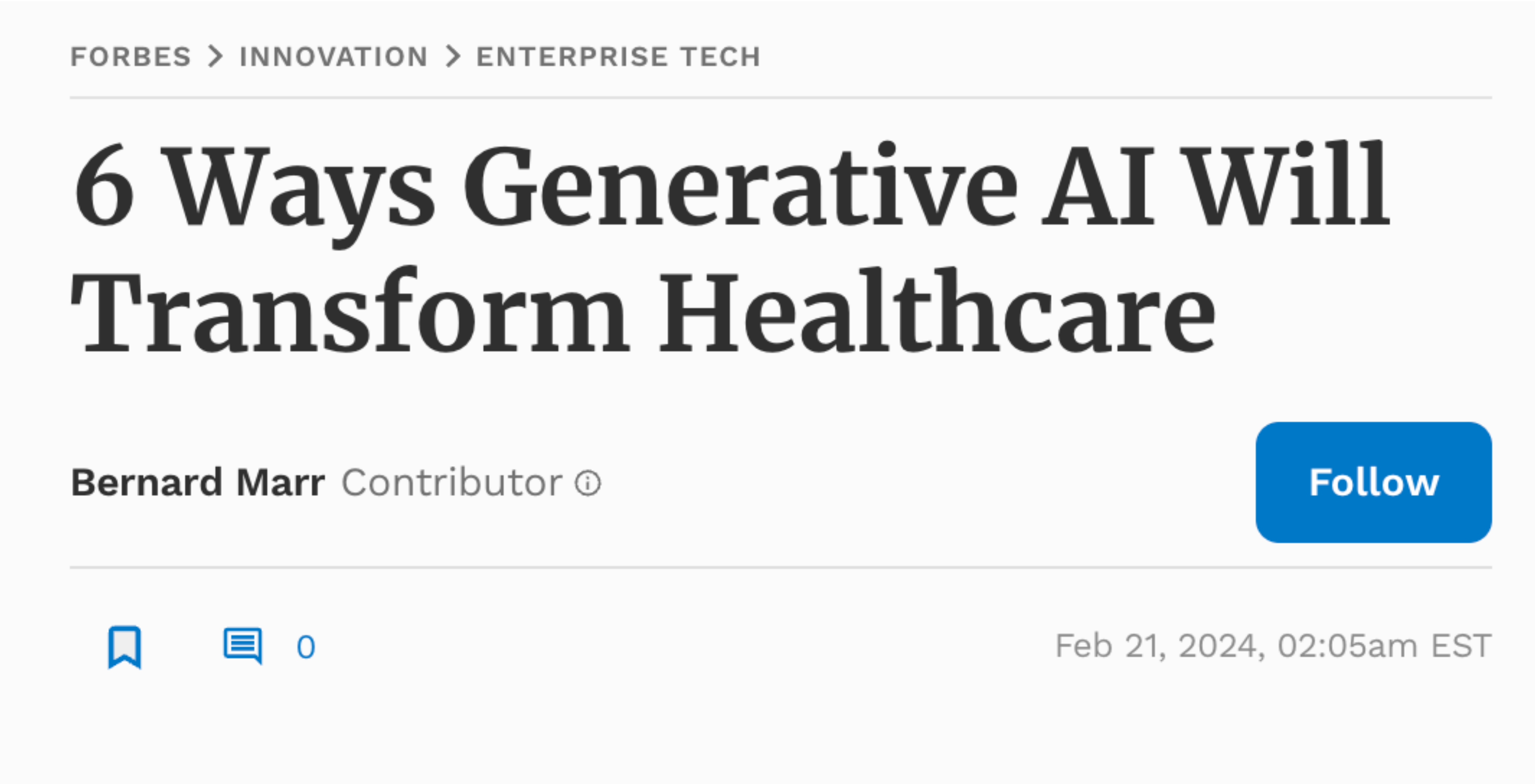
Introducing Devin, the first AI software engineer

And setting a new state of the art on the SWE-bench coding benchmark

Meet Devin, the world's first fully autonomous AI software engineer.

Devin is a tireless, skilled teammate, equally ready to build alongside you or independently complete tasks for you to review.

With Devin, engineers can focus on more interesting problems and engineering teams can strive for more ambitious goals.



6 Ways Generative AI Will Transform Healthcare

Bernard Marr Contributor

Feb 21, 2024, 02:05am EST

Even the car that dropped me ;)

Playing with Abstractions: Can LLMs be Architects New Best Friend?

Karthik Vaidhyanathan

1st International workshop on New Trends in Software Architecture@ICSE 2024

April 14, 2024

SERC
Software Engineering Research Centre

INTERNATIONAL INSTITUTE OF INFORMATION TECHNOLOGY
HYDERABAD

PORTUGAL
LISBON
ICSE 24 INTERNATIONAL CONFERENCE ON SOFTWARE ENGINEERING

ABOUT ME

Logic takes you from A to B, Immagination takes you elsewhere -- Albert Einstein

Karthik Vaidhyanathan
Assistant Professor
Software Engineering Research Center
IIIT Hyderabad, India

Research Interests

ML4SA

1. ML for continuous self-adaptation
2. Gen AI for Architectural Knowlege

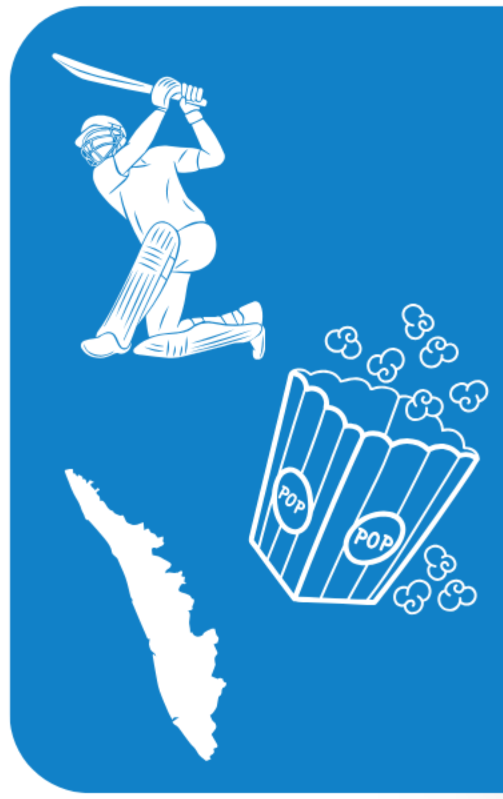
SA4ML

1. Sustainable ML-enabled systems
2. Autonomous ML-ops

Education

Double Master Degree - Software Architecture and Machine Learning
PhD from GSSI, Italy
Postdoc, University of L'Aquila, Italy

Fun Facts!



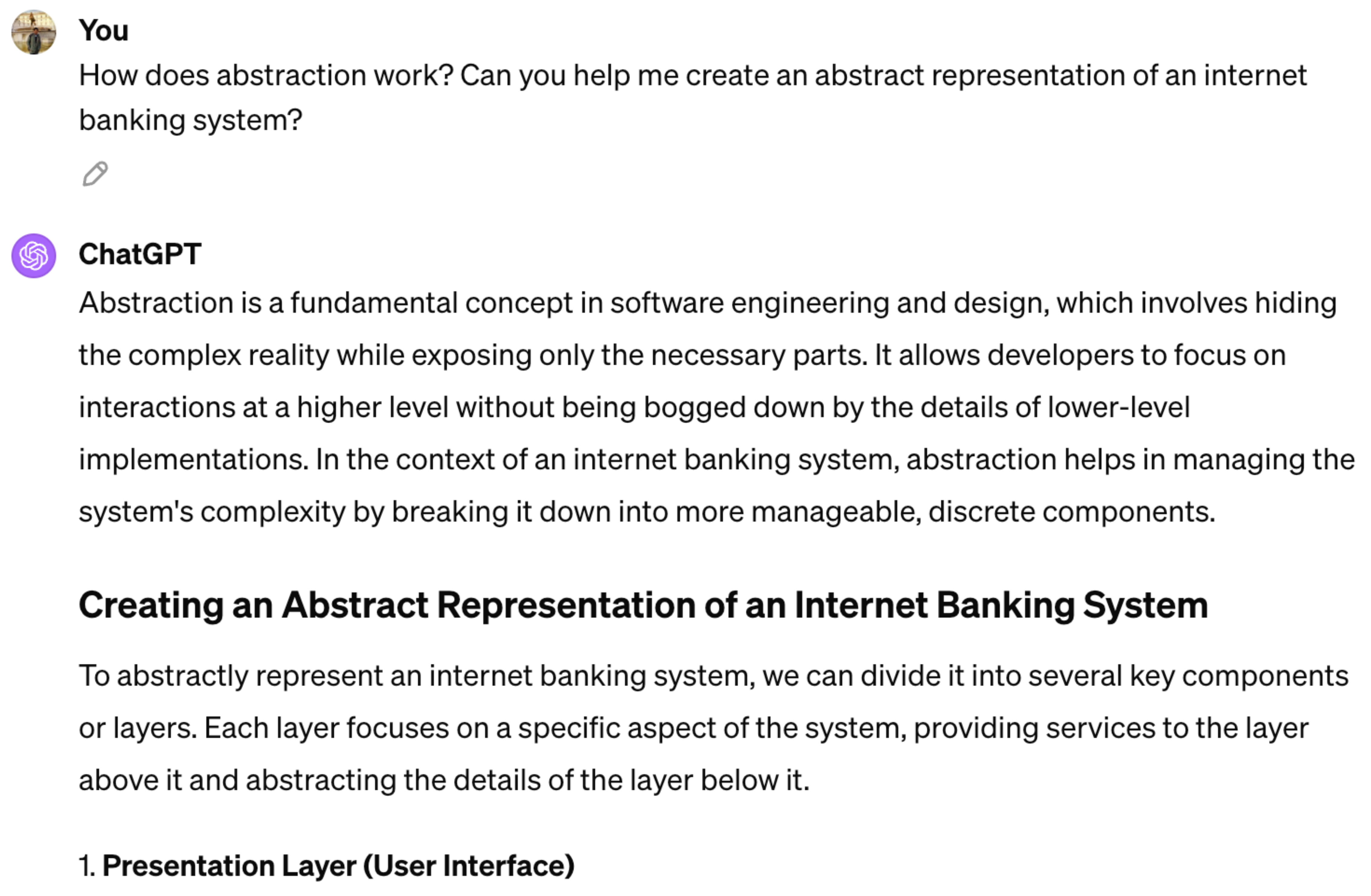
1. Cricket fanatic!
2. Movie buff!!
3. From God's own Country!!

SERC Software Engineering Research Centre

https://karthikvaidhyanathan.com
/in/karthikv1392/
karthi_ishere
karthik.vaidhyanathan@iiit.ac.in

The Power of Abstraction!

“Abstraction is the art of painting with thoughts, not brushes”



You

How does abstraction work? Can you help me create an abstract representation of an internet banking system?

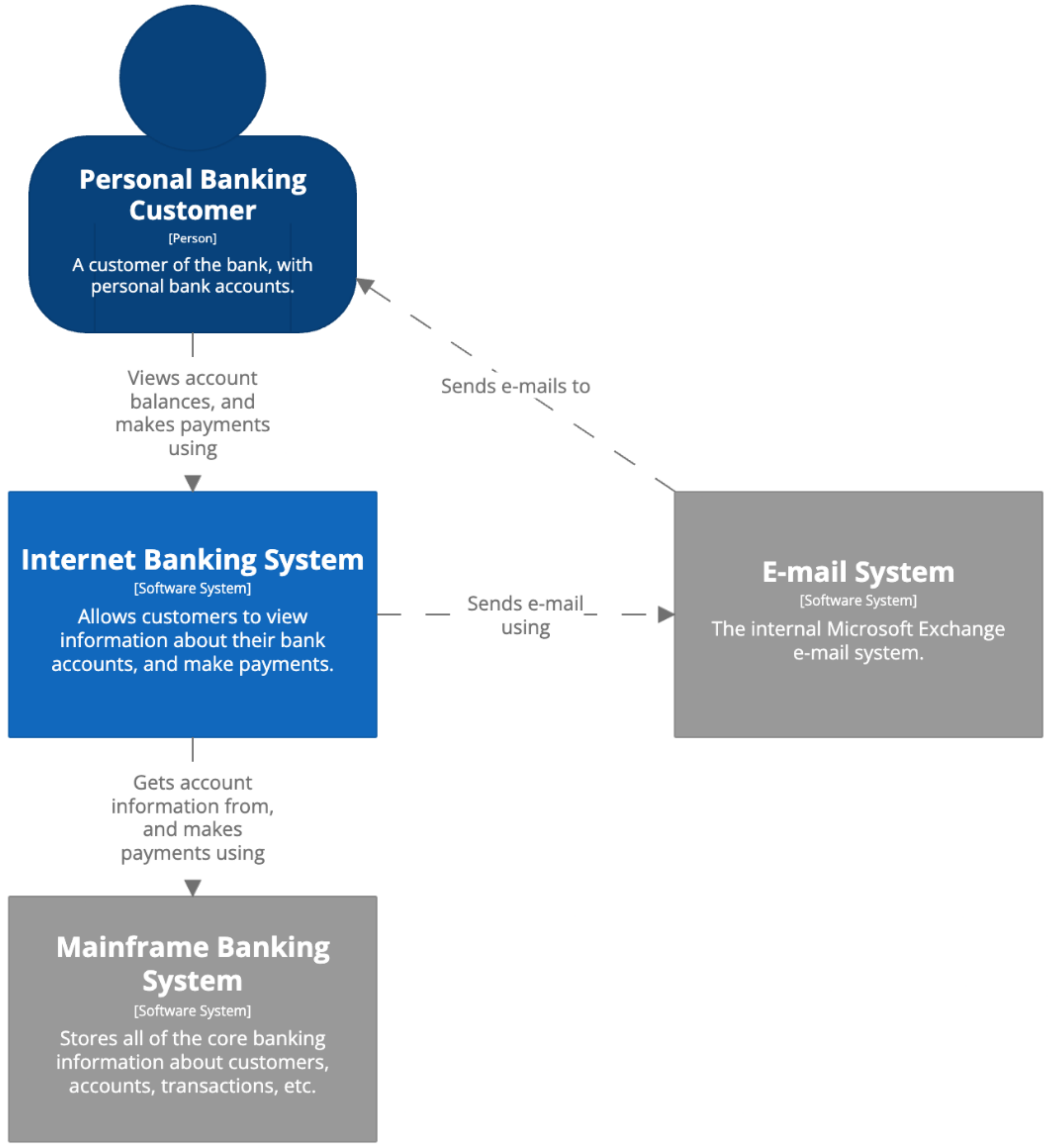
ChatGPT

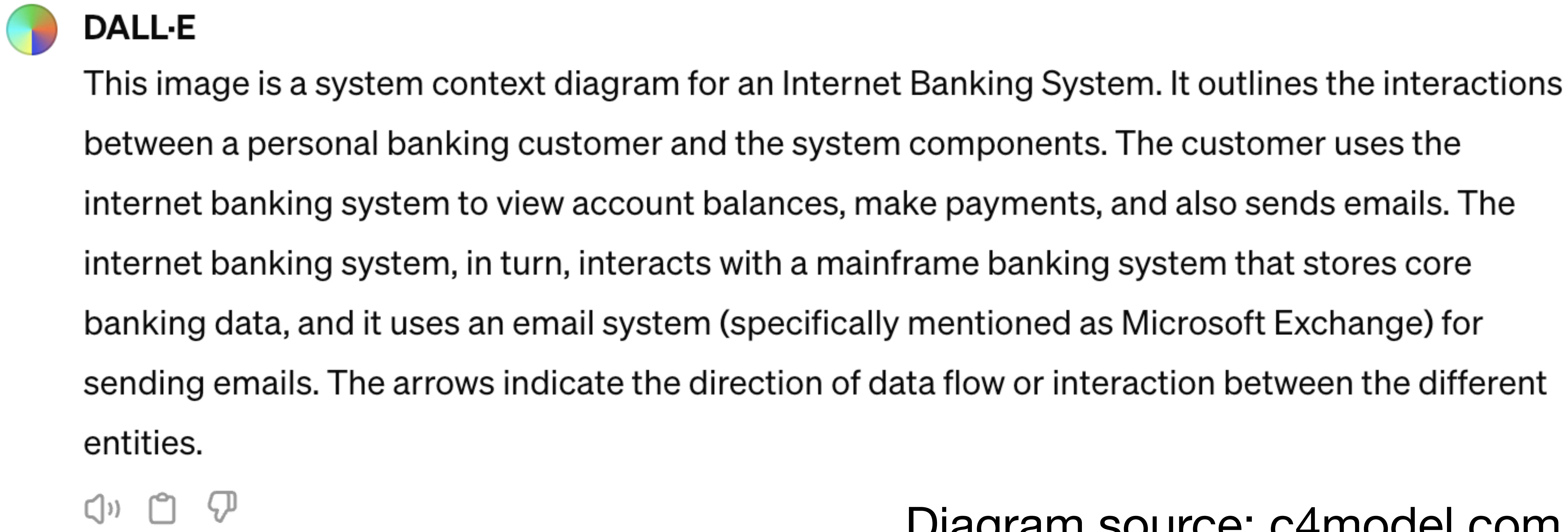
Abstraction is a fundamental concept in software engineering and design, which involves hiding the complex reality while exposing only the necessary parts. It allows developers to focus on interactions at a higher level without being bogged down by the details of lower-level implementations. In the context of an internet banking system, abstraction helps in managing the system's complexity by breaking it down into more manageable, discrete components.

Creating an Abstract Representation of an Internet Banking System

To abstractly represent an internet banking system, we can divide it into several key components or layers. Each layer focuses on a specific aspect of the system, providing services to the layer above it and abstracting the details of the layer below it.

1. **Presentation Layer (User Interface)**


Personal Banking Customer
[Person]
A customer of the bank, with personal bank accounts.
Views account balances, and makes payments using
Sends e-mails to
Internet Banking System
[Software System]
Allows customers to view information about their bank accounts, and make payments.
Sends e-mail using
E-mail System
[Software System]
The internal Microsoft Exchange e-mail system.
Gets account information from, and makes payments using
Mainframe Banking System
[Software System]
Stores all of the core banking information about customers, accounts, transactions, etc.



DALL·E

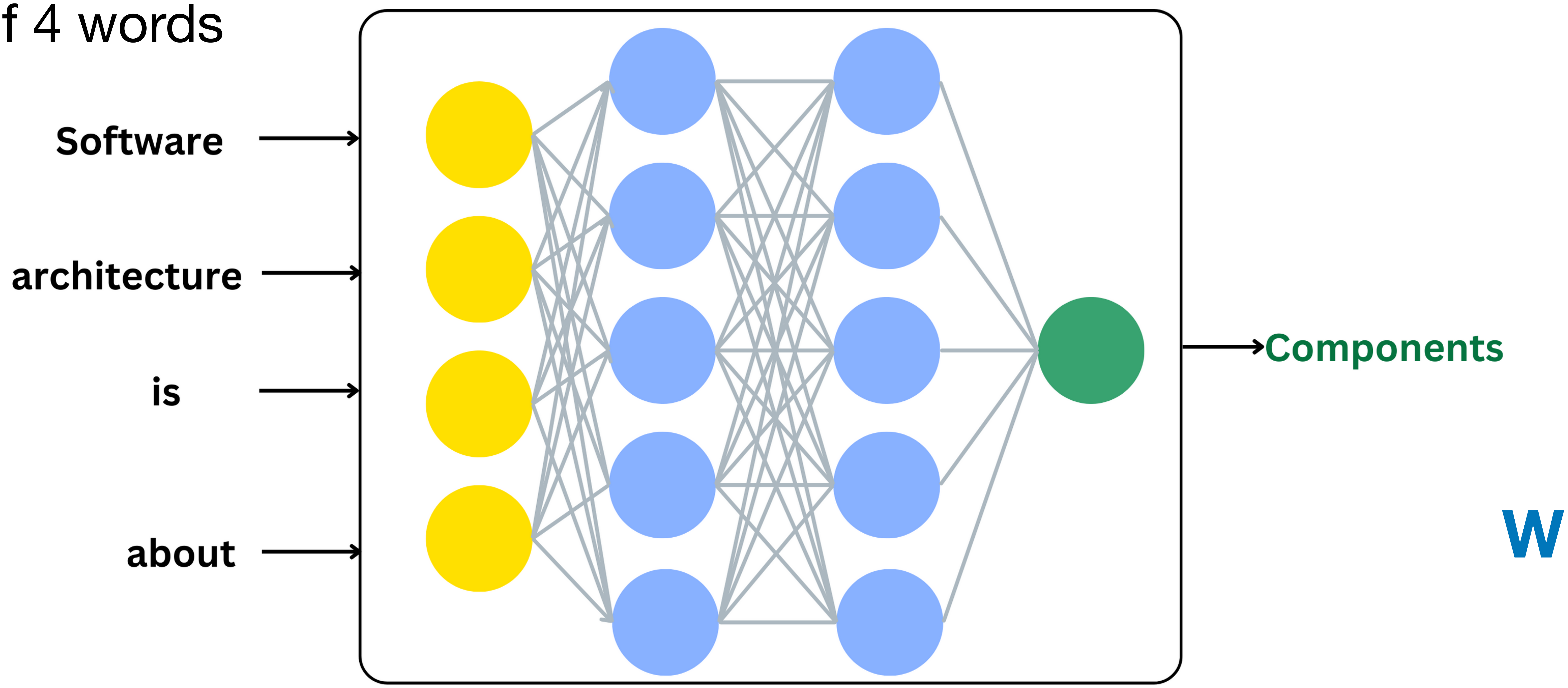
This image is a system context diagram for an Internet Banking System. It outlines the interactions between a personal banking customer and the system components. The customer uses the internet banking system to view account balances, make payments, and also sends emails. The internet banking system, in turn, interacts with a mainframe banking system that stores core banking data, and it uses an email system (specifically mentioned as Microsoft Exchange) for sending emails. The arrows indicate the direction of data flow or interaction between the different entities.

Diagram source: c4model.com

Its been a long journey MCP, ..backprop…RNN CNN,……..

Predict the next word in a sequence!

Eg: Context of 4 words



Software
architecture
is
about

Components

What does “it” refer to?

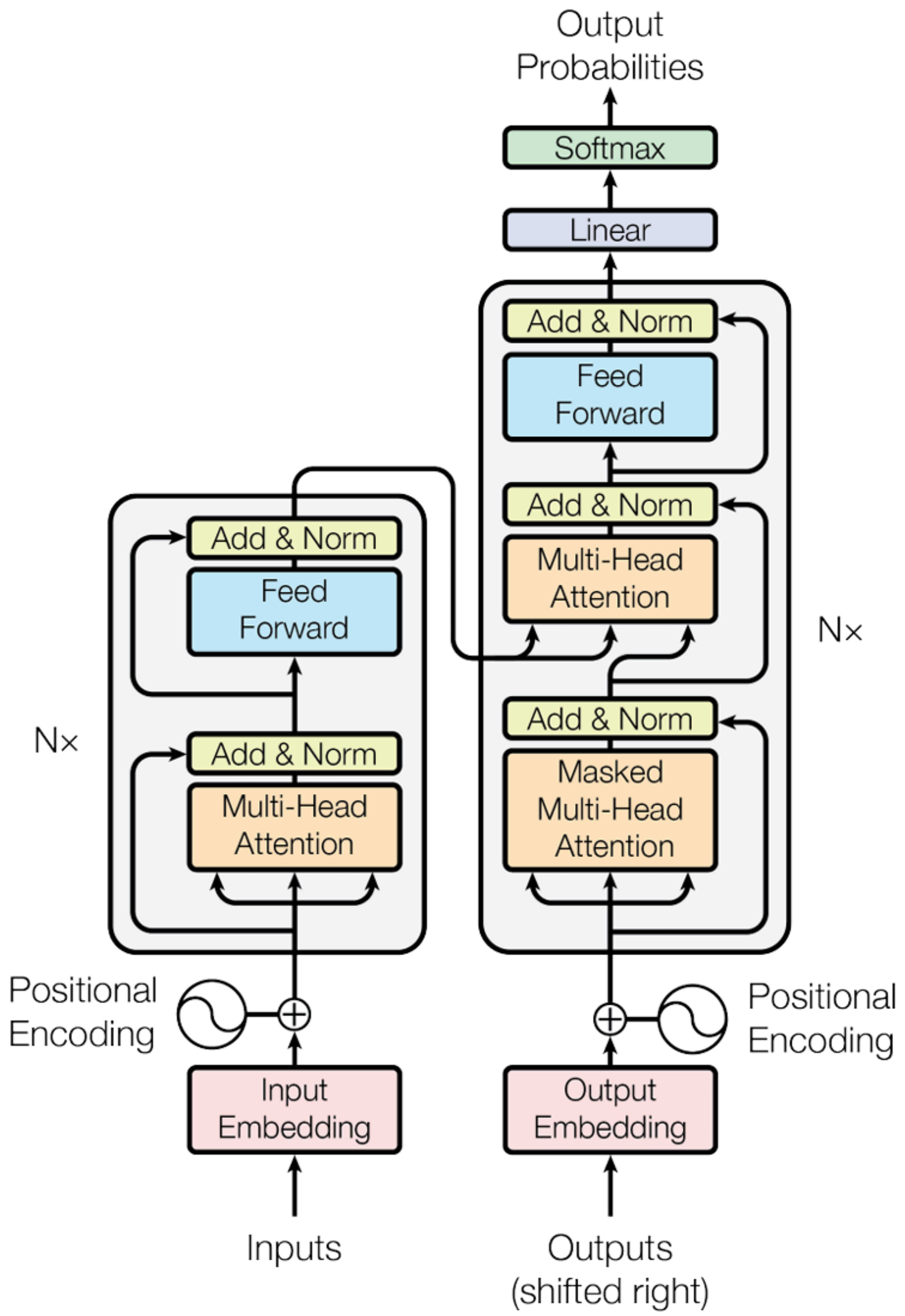
Software architecture is about components and

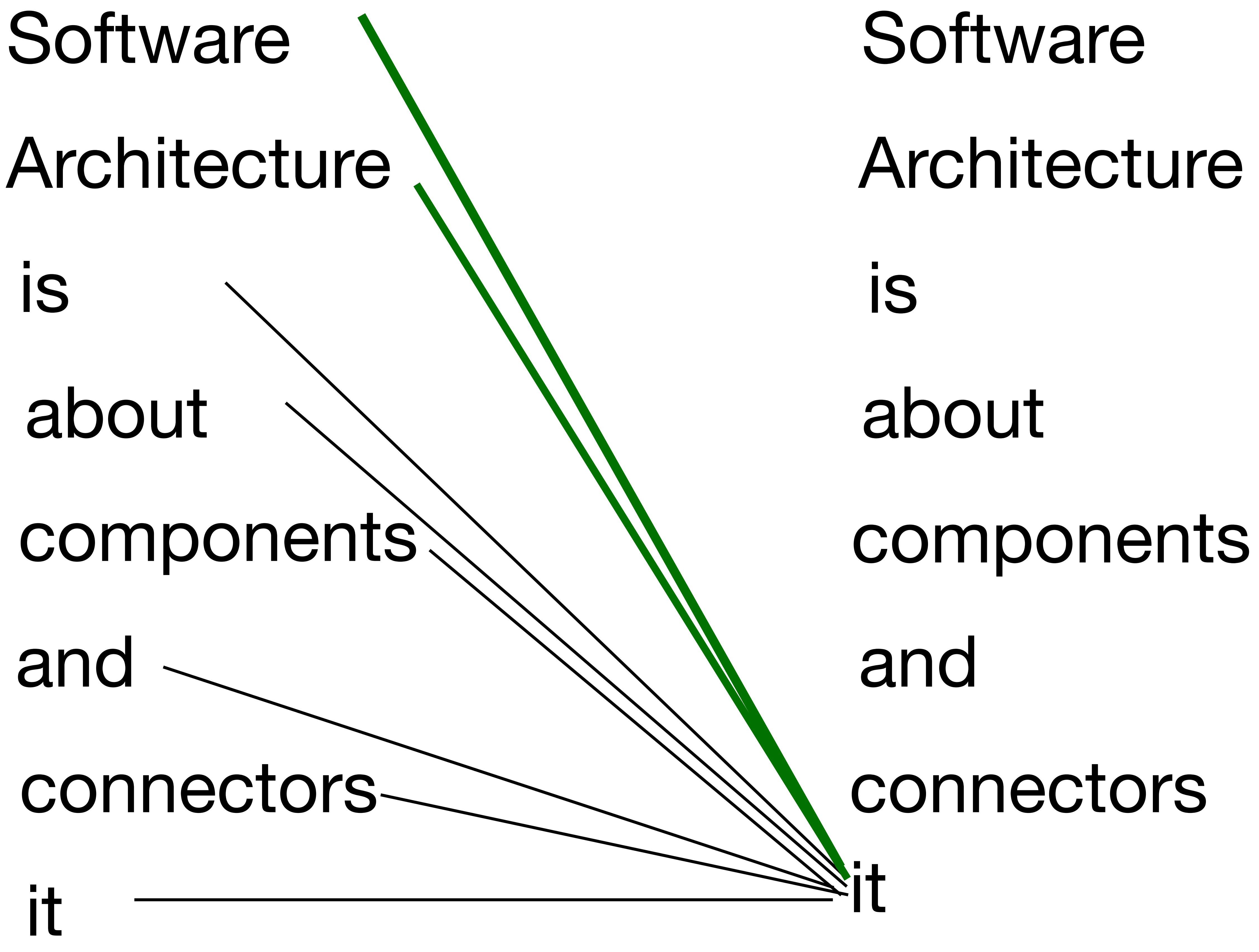
Software architecture is about components and connectors.

Software architecture is about components and connectors. It

Attention is all you need!

Age of Transformers





Software
Architecture
is
about
components
and
connectors
it

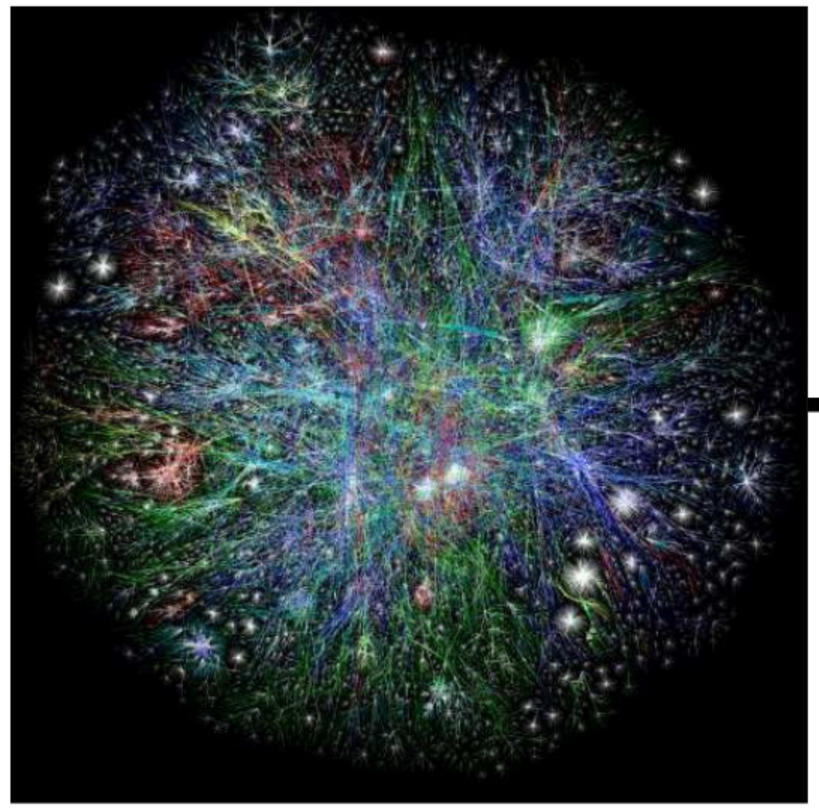
Software
Architecture
is
about
components
and
connectors
it

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł. and Polosukhin, I., 2017. **Attention is all you need.** *Advances in neural information processing systems*, *30*.

“Large” Language Models (LLM)

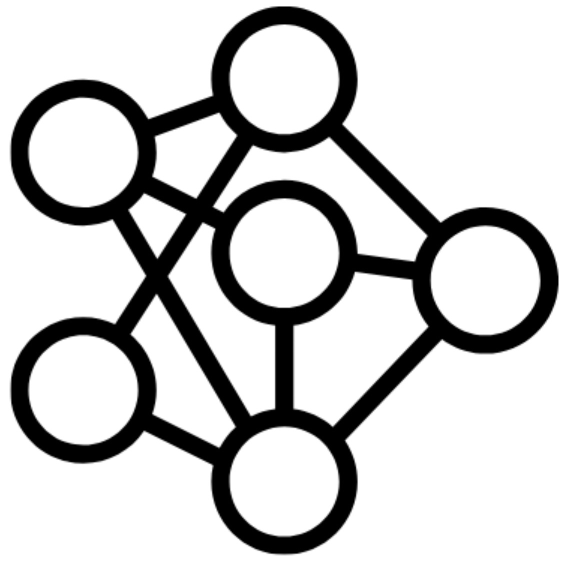
Do you have a ton of text and compute power?

Internet



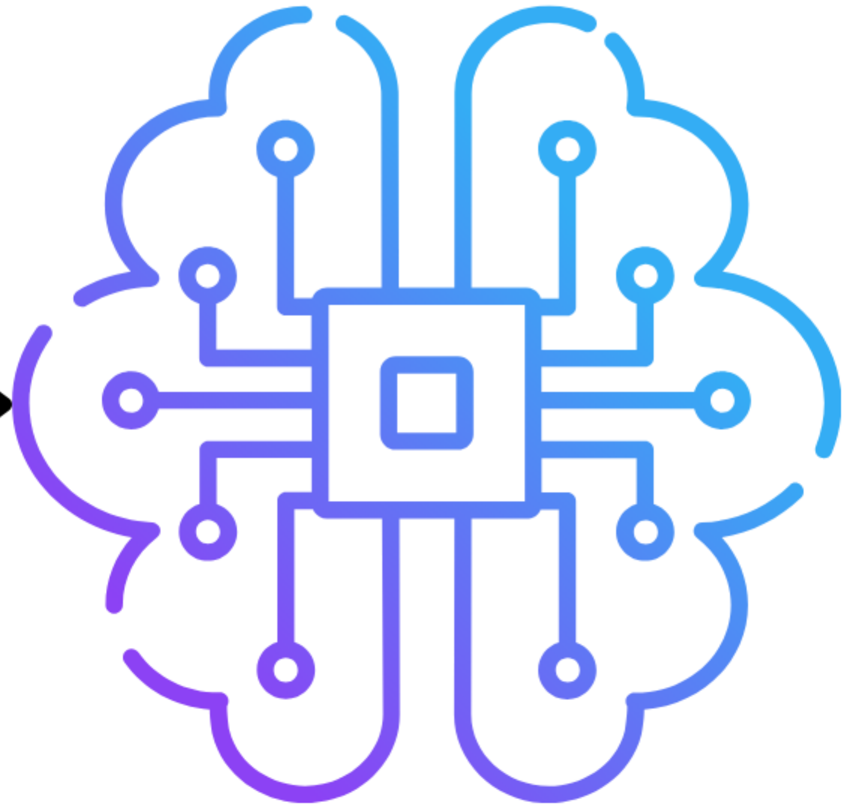
~10B of text from internet

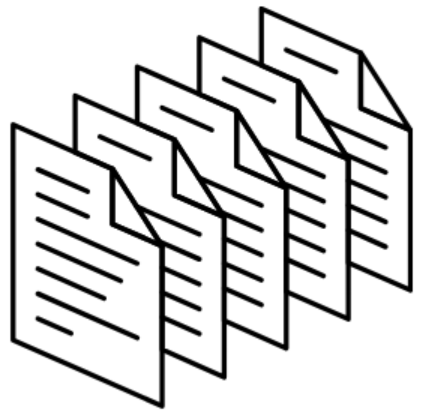
Compute



Transformer NN
trained on 1000s of GPUs
for days

**Foundation model/
Base model**



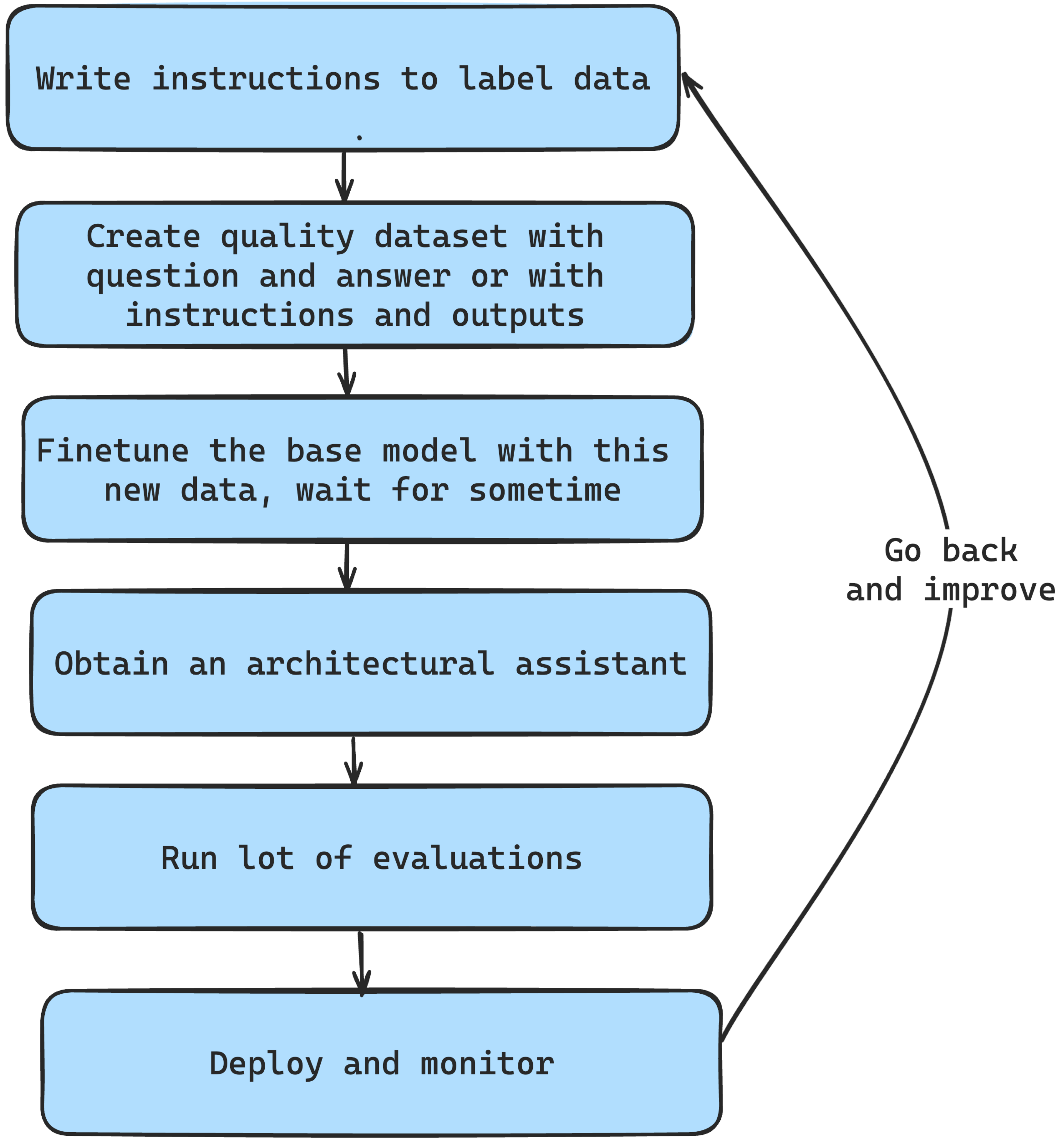


Model parameters
~some billions
100s of GB

Inspiration from slides of Andrej Karpathy, Introduction to large language models

I want something more specific to Architecture

May be you can fine-tune and create your model



Write instructions to label data

Create quality dataset with question and answer or with instructions and outputs

Finetune the base model with this new data, wait for sometime

Obtain an architectural assistant

Run lot of evaluations

Deploy and monitor

Go back and improve

<<USER>

What is Software Architecture?

<<ASSISTANT>>

Software architecture is a combination of elements, form and rationale

<<USER>

How to start thinking about architecture?

<<ASSISTANT>>

One can start with IEEE 42010 framework, understand stakeholders…

Some 1000s of examples

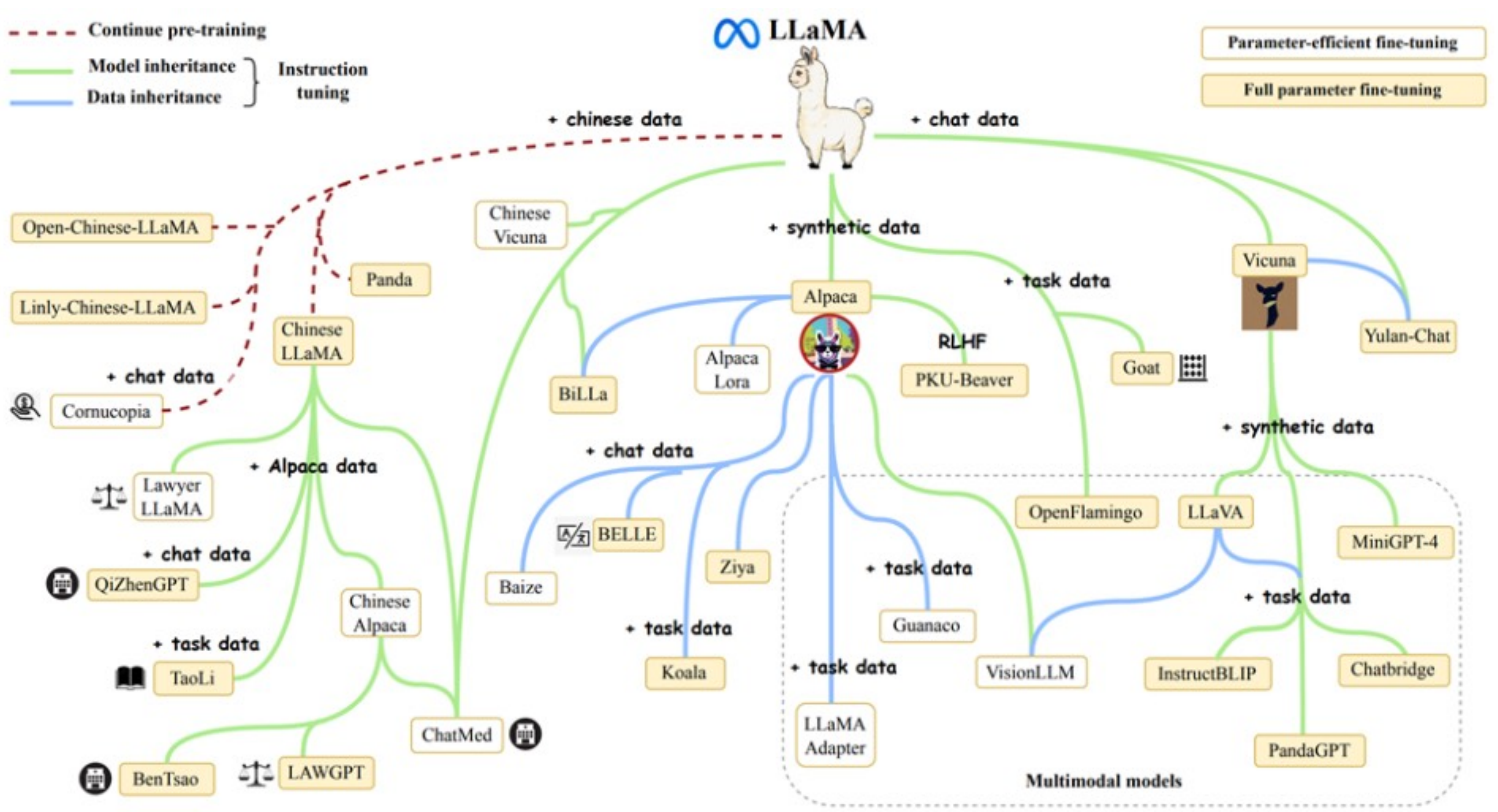
Eg: codexGPT, CodeBERT,…

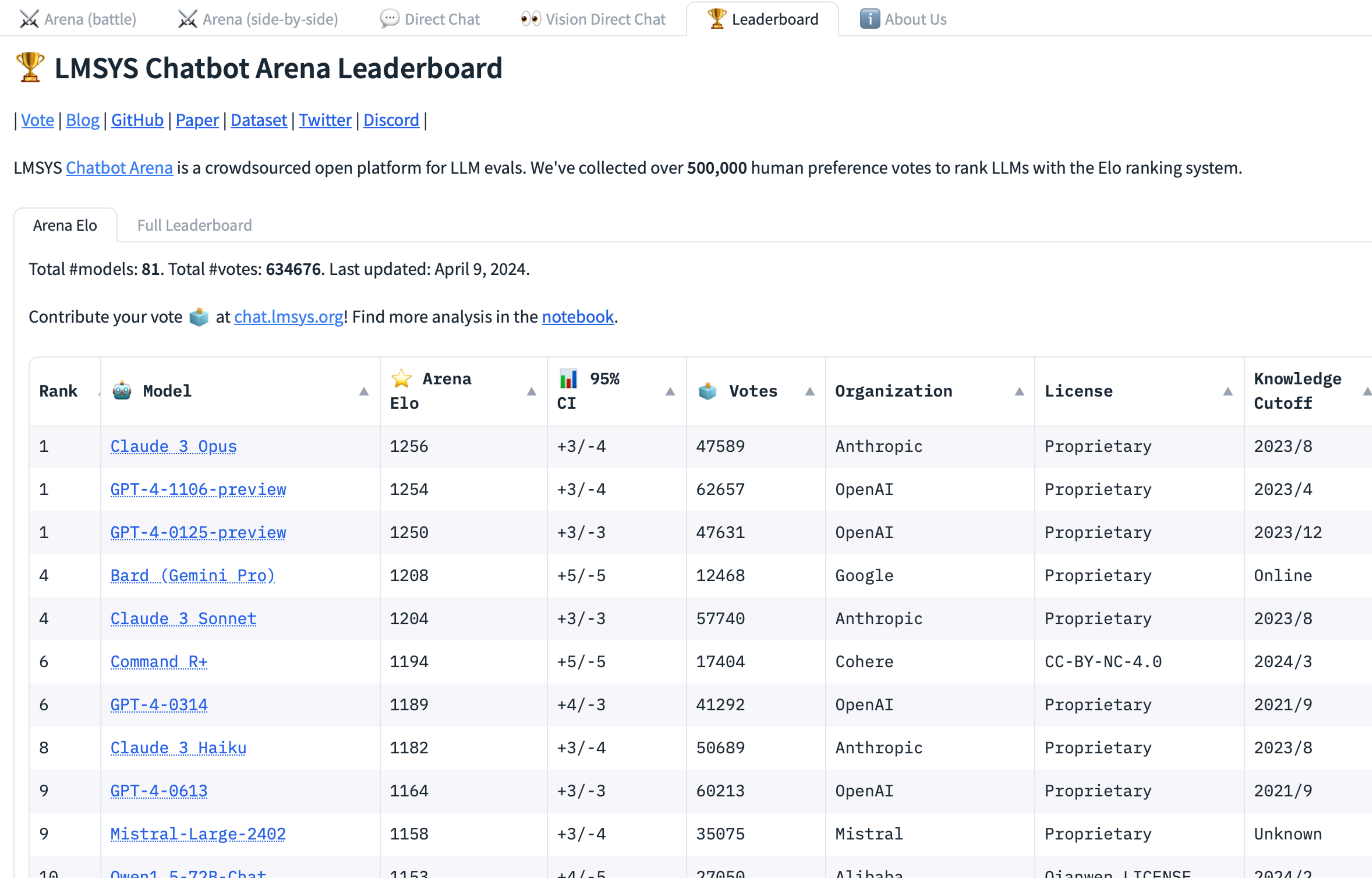
Today we have different varieties of LLMs

Proprietary

Bard

Open Source





🏆 LMSYS Chatbot Arena Leaderboard

| Vote | Blog | GitHub | Paper | Dataset | Twitter | Discord |

LMSYS Chatbot Arena is a crowdsourced open platform for LLM evals. We've collected over **500,000** human preference votes to rank LLMs with the Elo ranking system.

Total #models: **81**. Total #votes: **634676**. Last updated: April 9, 2024.

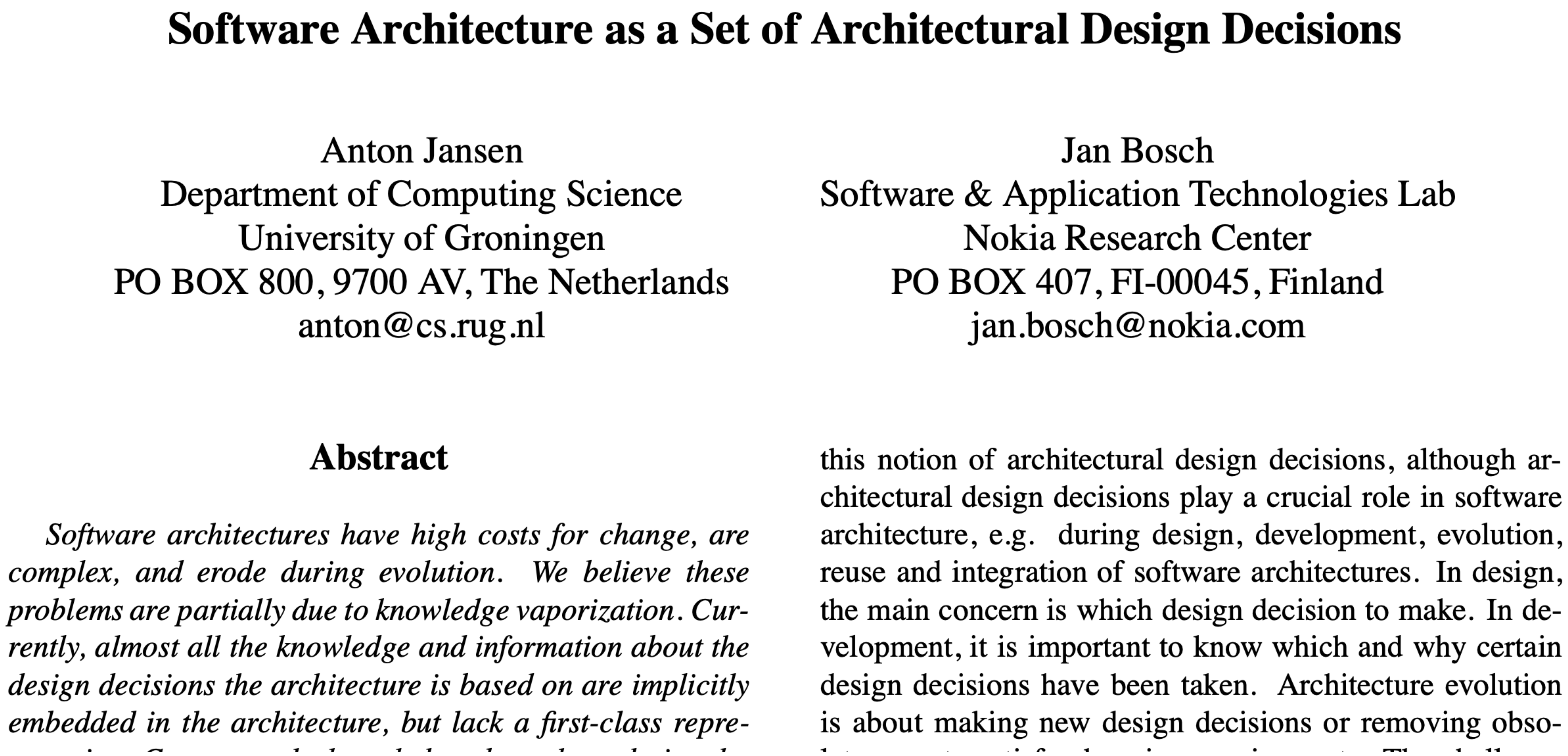
Contribute your vote at chat.lmsys.org! Find more analysis in the notebook.

Rank	Model	Arena Elo	95% CI	Votes	Organization	License	Knowledge Cutoff
1	Claude 3 Opus	1256	+3/-4	47589	Anthropic	Proprietary	2023/8
1	GPT-4-1106-preview	1254	+3/-4	62657	OpenAI	Proprietary	2023/4
1	GPT-4-0125-preview	1250	+3/-3	47631	OpenAI	Proprietary	2023/12
4	Bard (Gemini Pro)	1208	+5/-5	12468	Google	Proprietary	Online
4	Claude 3 Sonnet	1204	+3/-3	57740	Anthropic	Proprietary	2023/8
6	Command R+	1194	+5/-5	17404	Cohere	CC-BY-NC-4.0	2024/3
6	GPT-4-0314	1189	+4/-3	41292	OpenAI	Proprietary	2021/9
8	Claude 3 Haiku	1182	+3/-4	50689	Anthropic	Proprietary	2023/8
9	GPT-4-0613	1164	+3/-3	60213	OpenAI	Proprietary	2021/9
9	Mistral-Large-2402	1158	+3/-4	35075	Mistral	Proprietary	Unknown
10	Qwen1.5-72B-Chat	1153	+4/-5	27050	Alibaba	Qianwen LICENSE	2024/2

https://chat.lmsys.org/

How can all these help Software Architect?

Design Decisions is all you need!



Software Architecture as a Set of Architectural Design Decisions

Anton Jansen
Department of Computing Science
University of Groningen
PO BOX 800, 9700 AV, The Netherlands
anton@cs.rug.nl

Jan Bosch
Software & Application Technologies Lab
Nokia Research Center
PO BOX 407, FI-00045, Finland
jan.bosch@nokia.com

Abstract

Software architectures have high costs for change, are complex, and erode during evolution. We believe these problems are partially due to knowledge vaporization. Currently, almost all the knowledge and information about the design decisions the architecture is based on are implicitly embedded in the architecture, but lack a first-class repre-

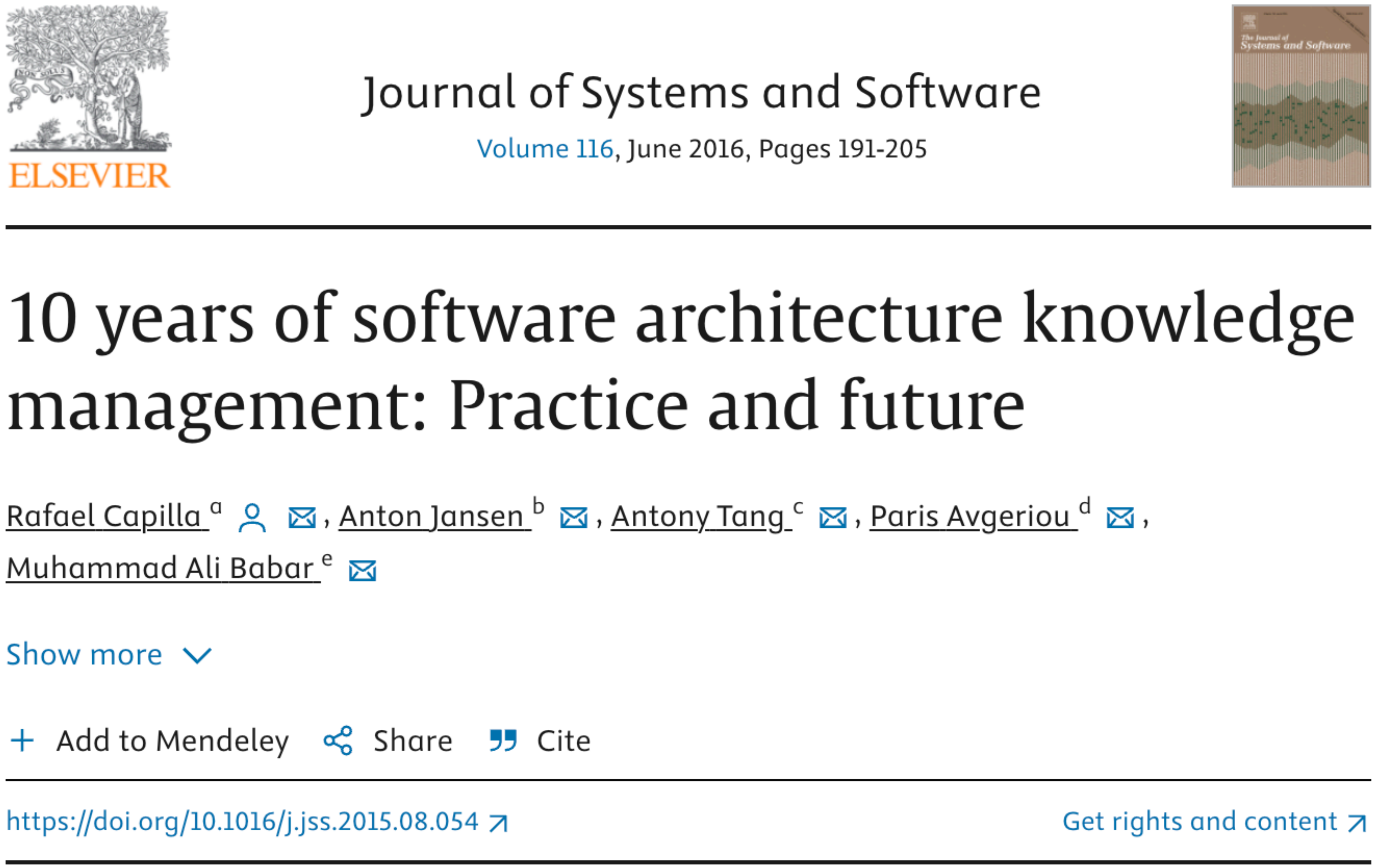
this notion of architectural design decisions, although architectural design decisions play a crucial role in software architecture, e.g. during design, development, evolution, reuse and integration of software architectures. In design, the main concern is which design decision to make. In development, it is important to know which and why certain design decisions have been taken. Architecture evolution is about making new design decisions or removing obso-

Software Architecture is a set of key design decisions

Jansen, A. and Bosch, J., 2005, November. Software architecture as a set of architectural design decisions. In *5th Working IEEE/IFIP Conference on Software Architecture (WICSA'05)*

Architecture Knowledge Management

*Architecture knowledge management (AKM) aims to **codify and maintain** the Architectural knowledge of a software system in a form that can be **easily accessed** by different stakeholders*



ELSEVIER

Journal of Systems and Software

Volume 116, June 2016, Pages 191-205

10 years of software architecture knowledge management: Practice and future

Rafael Capilla[a], Anton Jansen[b], Antony Tang[c], Paris Avgeriou[d], Muhammad Ali Babar[e]

Show more

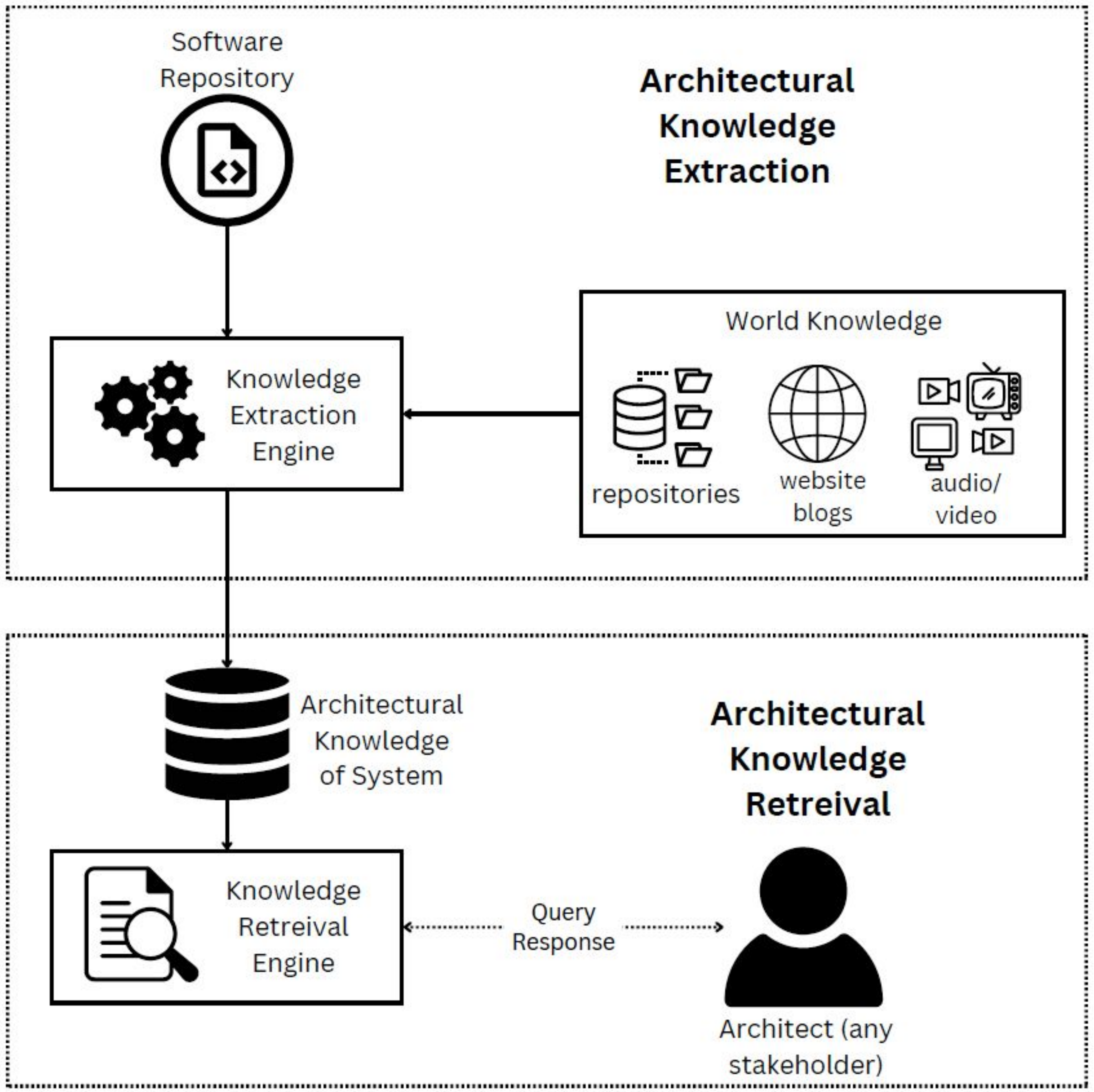
https://doi.org/10.1016/j.jss.2015.08.054

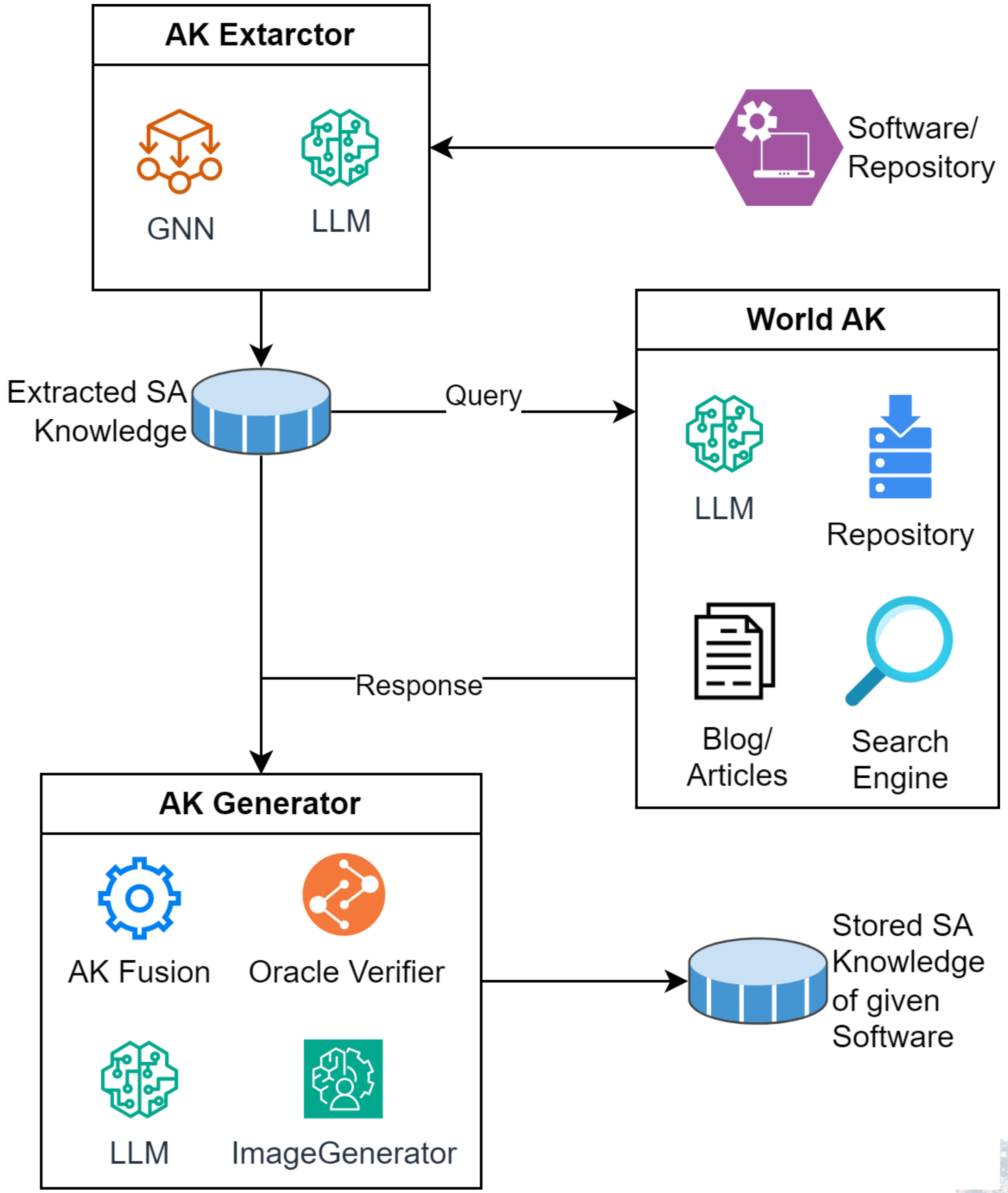
Get rights and content

Need for better tools => Automate using **ArchBots or a co-pilot**

Paris Avgeriou, Making Decisions - From Software Architecture Theory to Practice, Keynote, ICSA 2023

Generative AI for Architectural Knowledge Management


Software Repository
Architectural Knowledge Extraction
Knowledge Extraction Engine
World Knowledge
repositories
website blogs
audio/ video
Architectural Knowledge of System
Architectural Knowledge Retreival
Knowledge Retrieval Engine
Query
Response
Architect (any stakeholder)

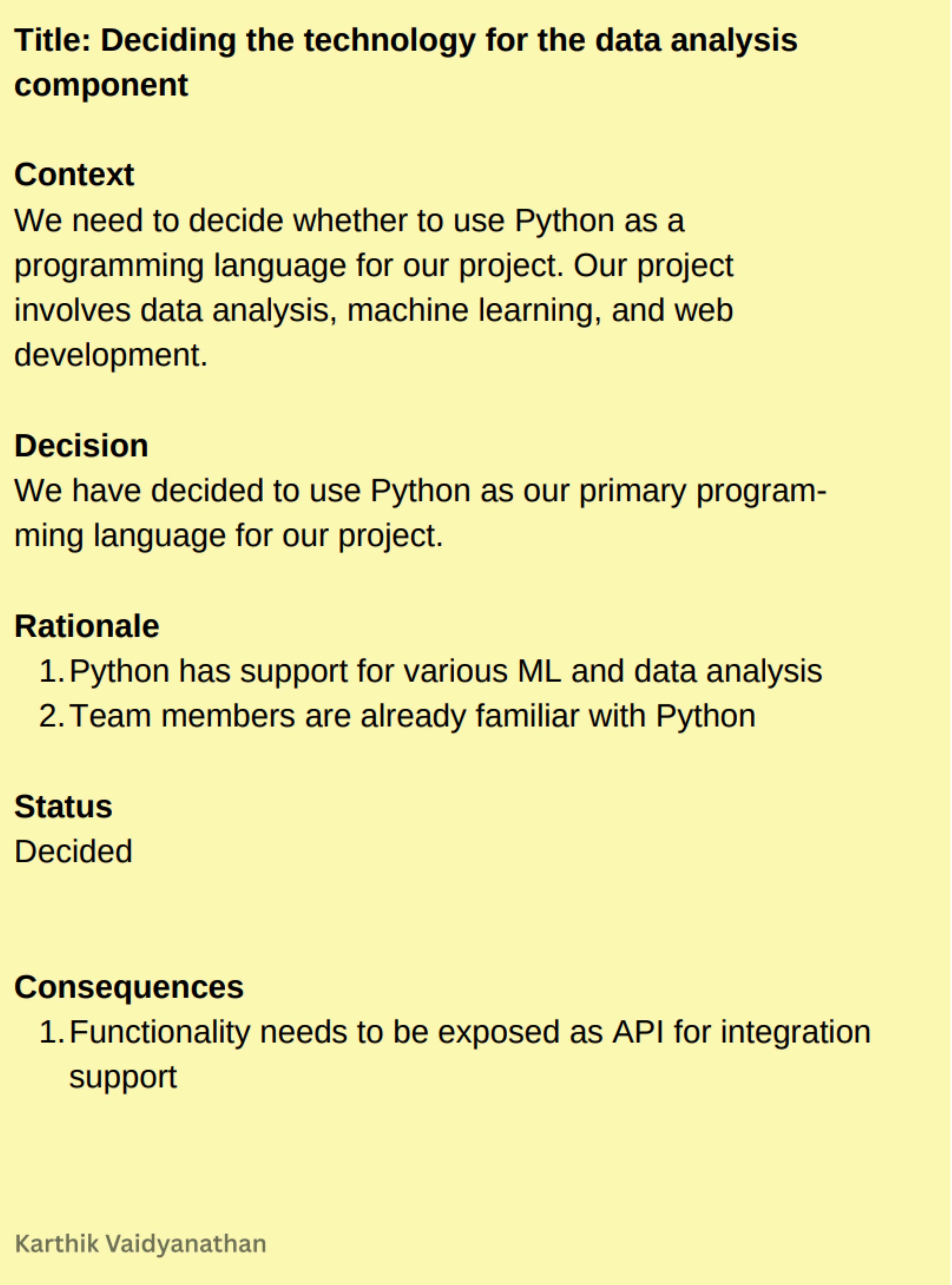


Rudra Dhar, Karthik Vaidhyanathan, Vasudeva Varma, **GenAI for Architecture Knowledge Management**, ICSA 2024, Poster

Starting with Design Decisions

- **Architecture Decision Records: ADR**
- Lightweight mechanism for documenting decisions
- Design decisions require careful considerations of various parameters
  - This requires broader understanding of domain as well as expertise
  - **Can we use LLMs to generate architecture design decisions?**
  - **Can LLMs be used to extract architectural information from design decisions?**

https://github.com/joelparkerhenderson/architecture-decision-record



Title: Deciding the technology for the data analysis component

Context

We need to decide whether to use Python as a programming language for our project. Our project involves data analysis, machine learning, and web development.

Decision

We have decided to use Python as our primary programming language for our project.

Rationale

1. Python has support for various ML and data analysis
2. Team members are already familiar with Python

Status

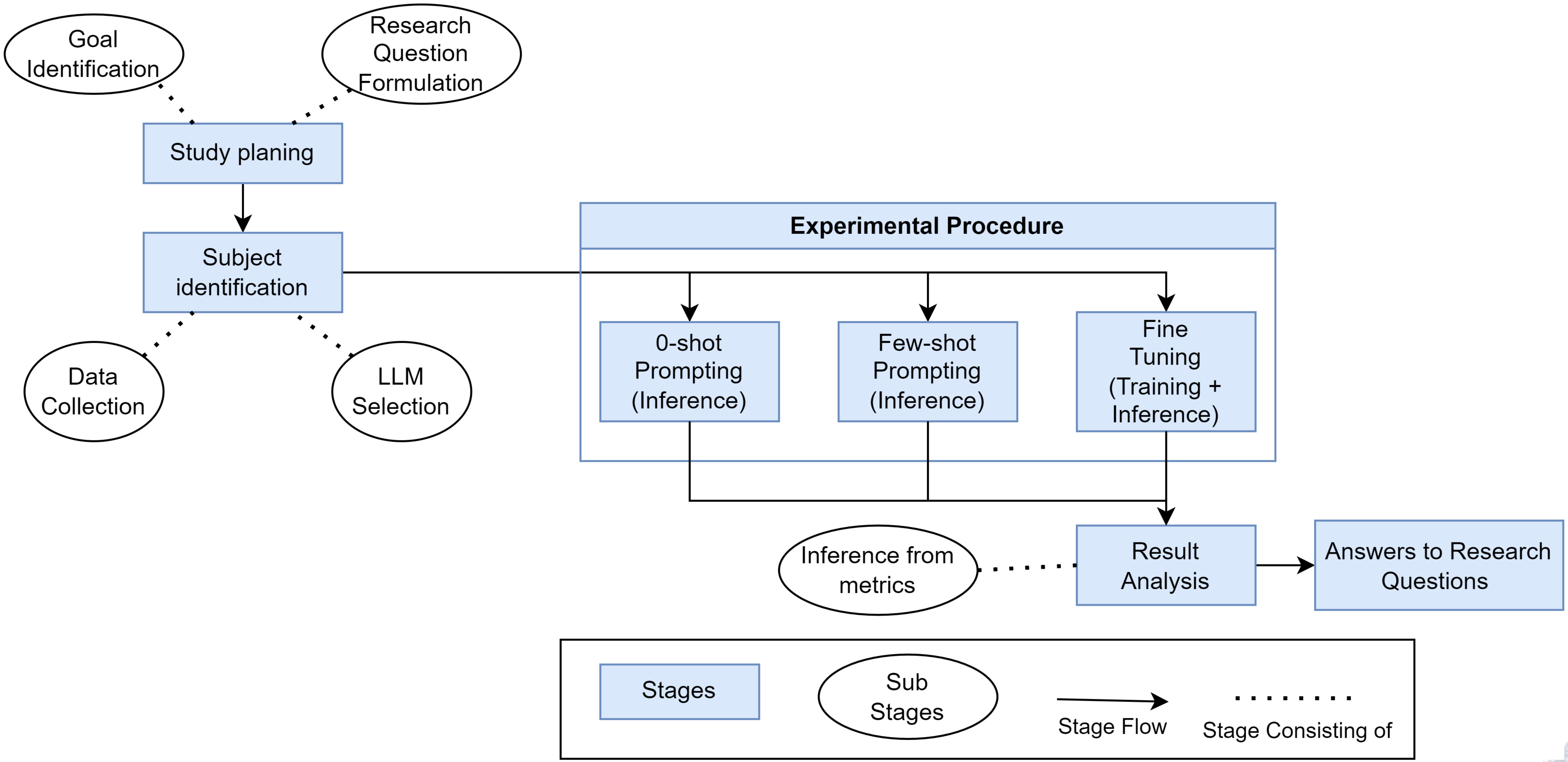
Decided

Consequences

1. Functionality needs to be exposed as API for integration support

Karthik Vaidyanathan

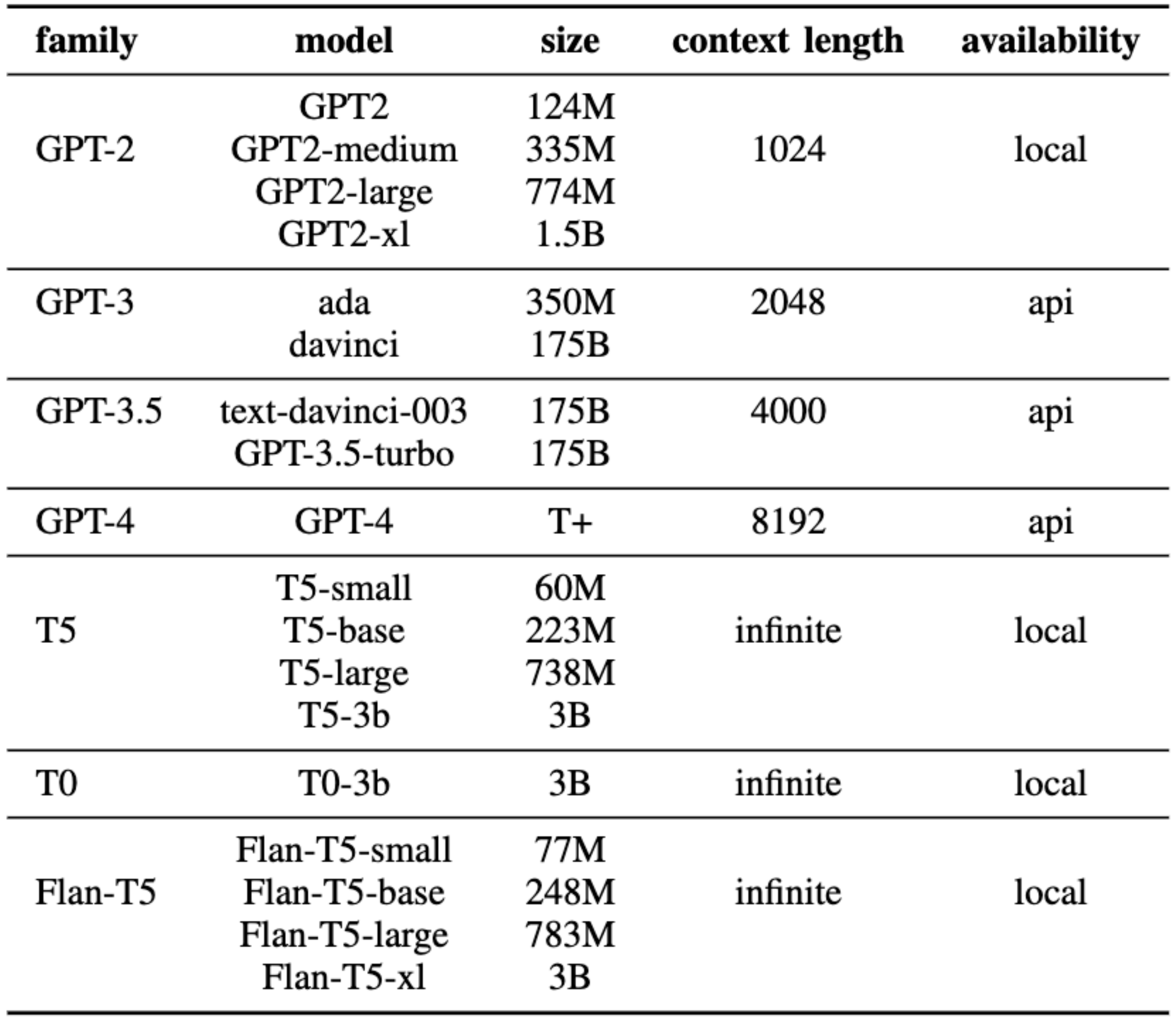
Performing an Exploratory Study



Dhar, R., Vaidhyanathan, K. and Varma, V. **Can LLMs Generate Architectural Design Decisions? - An Exploratory Empirical study,** ICSA 2024

Selected LLMs

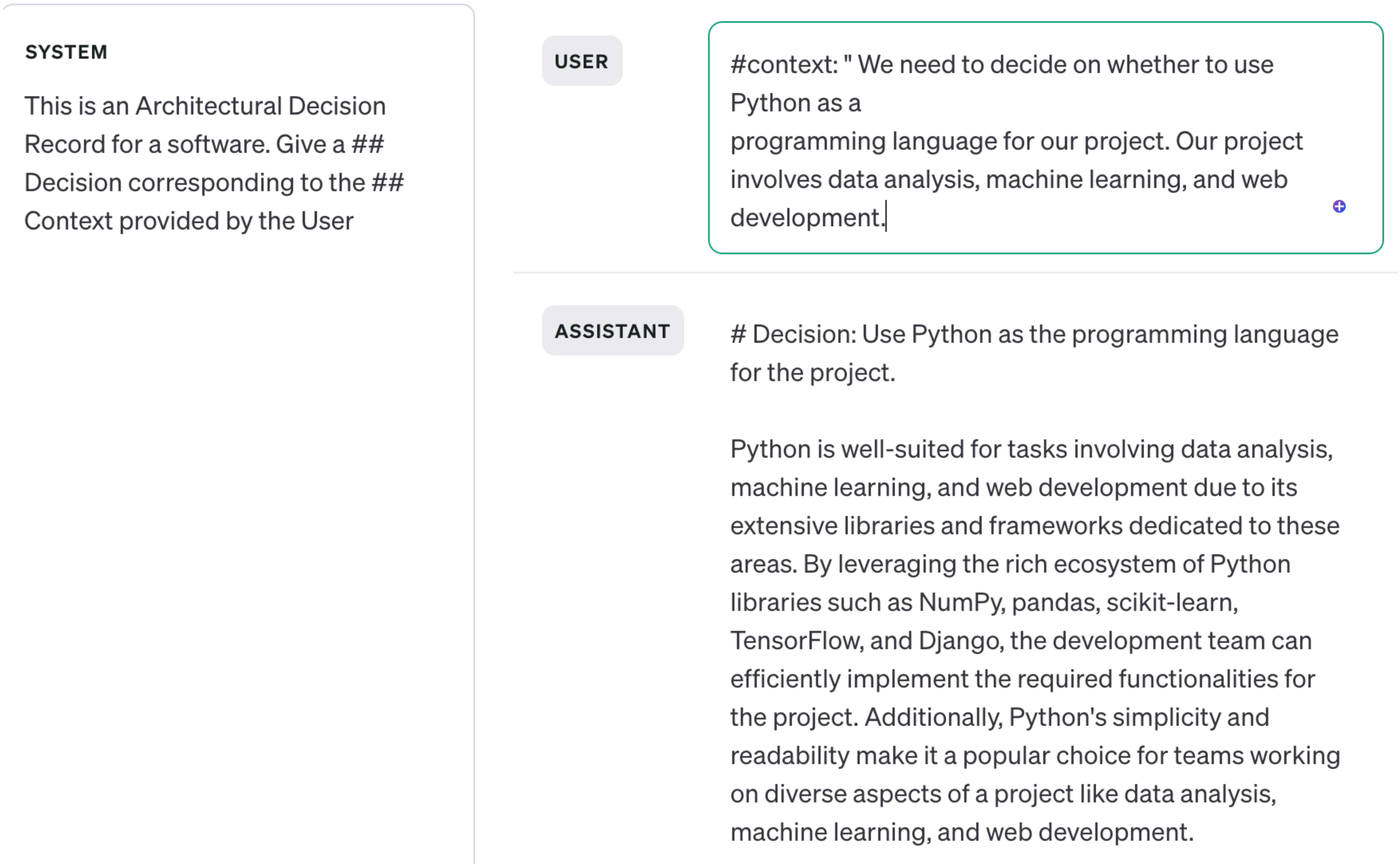
- **No of models:** 18
- **Model Series**
  - Encoder-decoder (T5 Series)
  - Decoder only (GPT series)
- **Model types**
  - Chat models
  - Text completion models
- From **small to large** models



family	model	size	context length	availability
GPT-2	GPT2 GPT2-medium GPT2-large GPT2-xl	124M 335M 774M 1.5B	1024	local
GPT-3	ada davinci	350M 175B	2048	api
GPT-3.5	text-davinci-003 GPT-3.5-turbo	175B 175B	4000	api
GPT-4	GPT-4	T+	8192	api
T5	T5-small T5-base T5-large T5-3b	60M 223M 738M 3B	infinite	local
T0	T0-3b	3B	infinite	local
Flan-T5	Flan-T5-small Flan-T5-base Flan-T5-large Flan-T5-xl	77M 248M 783M 3B	infinite	local

Generation Approaches

Zero Shot Prompting - Just ask the model and expect the answer



This is an Architectural Decision Record for a software. Give a ## Decision corresponding to the ## Context provided by the User

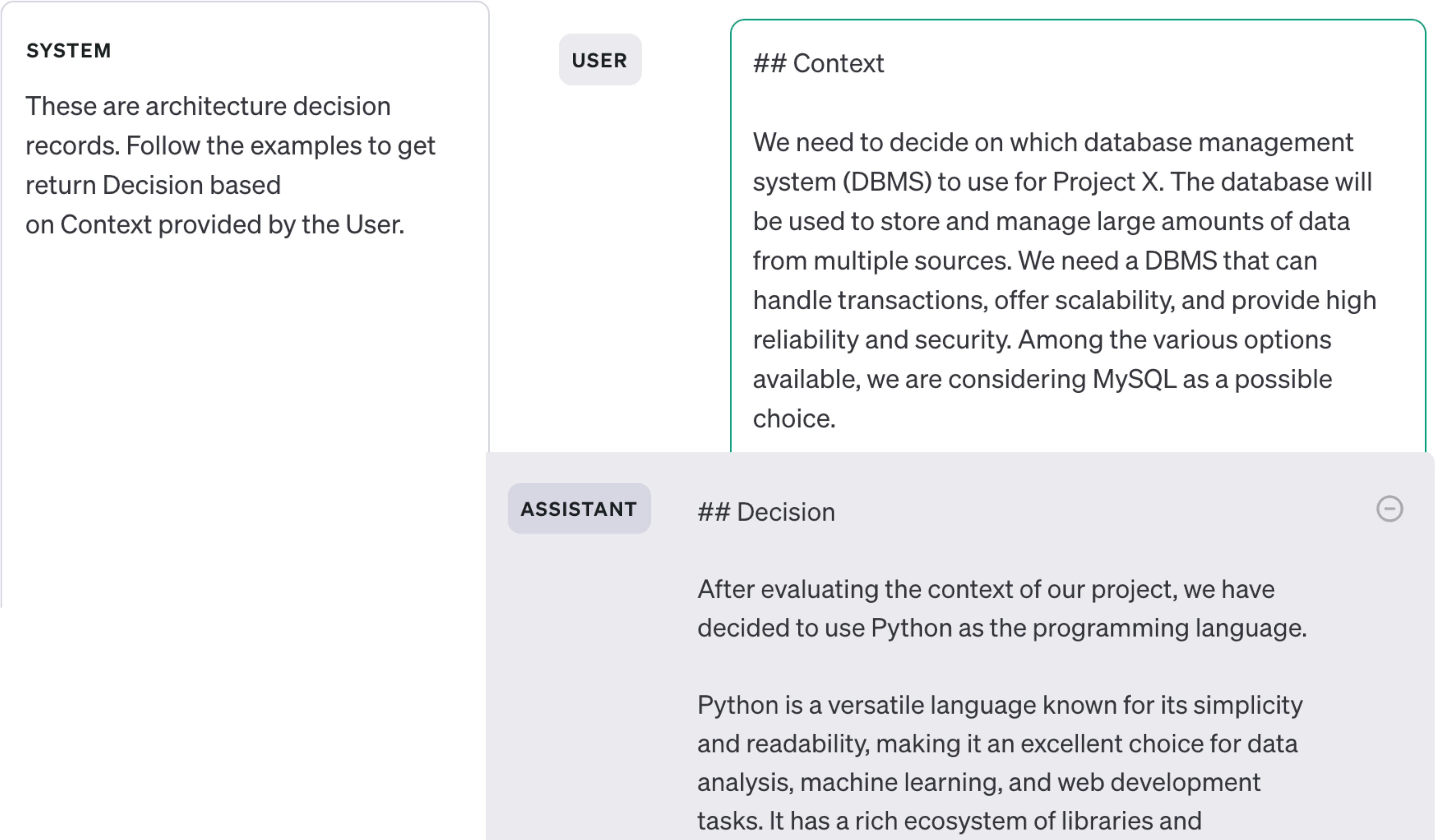
#context: " We need to decide on whether to use Python as a
programming language for our project. Our project involves data analysis, machine learning, and web development.

Decision: Use Python as the programming language for the project.

Python is well-suited for tasks involving data analysis, machine learning, and web development due to its extensive libraries and frameworks dedicated to these areas. By leveraging the rich ecosystem of Python libraries such as NumPy, pandas, scikit-learn, TensorFlow, and Django, the development team can efficiently implement the required functionalities for the project. Additionally, Python's simplicity and readability make it a popular choice for teams working on diverse aspects of a project like data analysis, machine learning, and web development.

Generation Approaches

Few Shot Prompting - Give some examples in the context



SYSTEM

These are architecture decision records. Follow the examples to get return Decision based
on Context provided by the User.

USER

Context

We need to decide on which database management system (DBMS) to use for Project X. The database will be used to store and manage large amounts of data from multiple sources. We need a DBMS that can handle transactions, offer scalability, and provide high reliability and security. Among the various options available, we are considering MySQL as a possible choice.

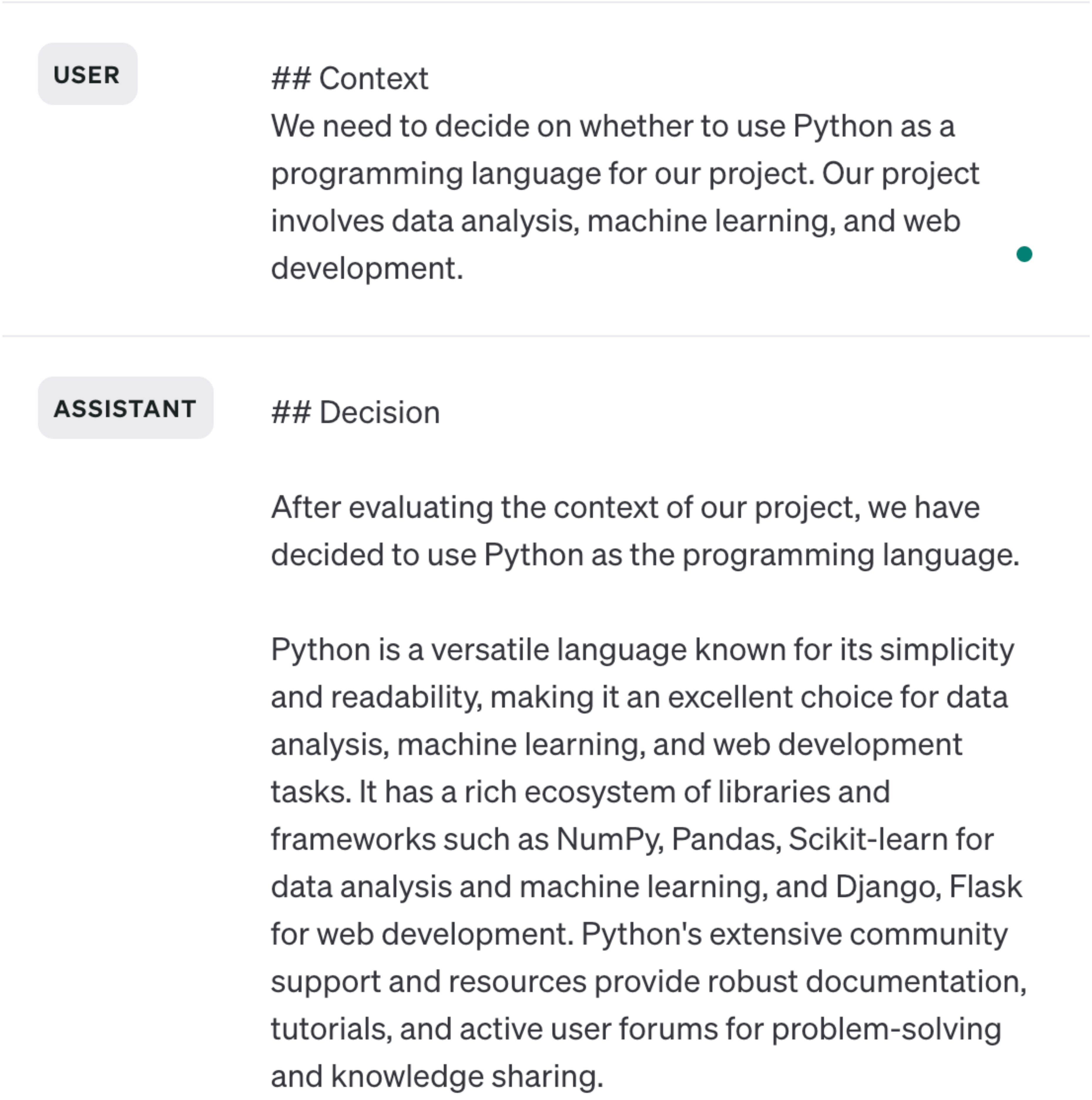
ASSISTANT

Decision

After evaluating the context of our project, we have decided to use Python as the programming language.

Python is a versatile language known for its simplicity and readability, making it an excellent choice for data analysis, machine learning, and web development tasks. It has a rich ecosystem of libraries and

Give some examples



USER

Context

We need to decide on whether to use Python as a programming language for our project. Our project involves data analysis, machine learning, and web development.

ASSISTANT

Decision

After evaluating the context of our project, we have decided to use Python as the programming language.

Python is a versatile language known for its simplicity and readability, making it an excellent choice for data analysis, machine learning, and web development tasks. It has a rich ecosystem of libraries and frameworks such as NumPy, Pandas, Scikit-learn for data analysis and machine learning, and Django, Flask for web development. Python's extensive community support and resources provide robust documentation, tutorials, and active user forums for problem-solving and knowledge sharing.

Get a response

Generation Approaches

Fine tuning - Fine tune base model with dataset

GPT2

Generate: {context} {decision}

Generate: {context} {decision}

Generate: {context} {decision}

T5

Read: {context}
Generate: {decision}

Read: {context}
Generate: {decision}

Read: {context}
Generate: {decision}

Results

Approach	**Model**	**Rouge-1**	**BLEU**	**Meteor**	**Bert-Score**			**Support**
					Precision	Recall	F1	
0-shot	GPT3.5-text-davinci-003	0.242	**0.031**	0.198	0.846	0.849	0.847	91
	GPT-4	**0.259**	0.028	**0.219**	0.847	**0.851**	**0.849**	95
	T0-3b	0.187	0.005	0.122	**0.856**	0.823	0.839	91
Few-shot	GPT2-xl	0.122	0.018	0.18	0.794	**0.857**	0.823	13
	GPT3.5-text-davinci-003	**0.245**	0.028	0.207	**0.849**	0.851	**0.849**	91
	GPT-3.5-turbo	0.226	0.027	**0.219**	0.832	0.85	0.84	92
	GPT-4	0.23	**0.03**	0.21	0.836	0.851	0.843	93
Fine-Tuning	Flan-T5-base	**0.231**	**0.028**	**0.171**	**0.842**	**0.841**	**0.841**	18

Smaller models performing well has lot of importance in software architecture domain!

Generated Decisions

GPT4 zero shot

We will use Python as our programming language for this project. Python has emerged as a reliable, versatile language with a rich ecosystem of libraries and frameworks that cater to our specific needs including data analysis (Pandas, NumPy), machine learning (scikit-learn, TensorFlow), and web

GPT3.5 few shot

After considering the various criteria discussed, we have decided to go ahead and use Python for our project.

Python is a versatile, powerful language with a wide range of features, .. community of developers who can provide support.. Python is also open-source, free to use

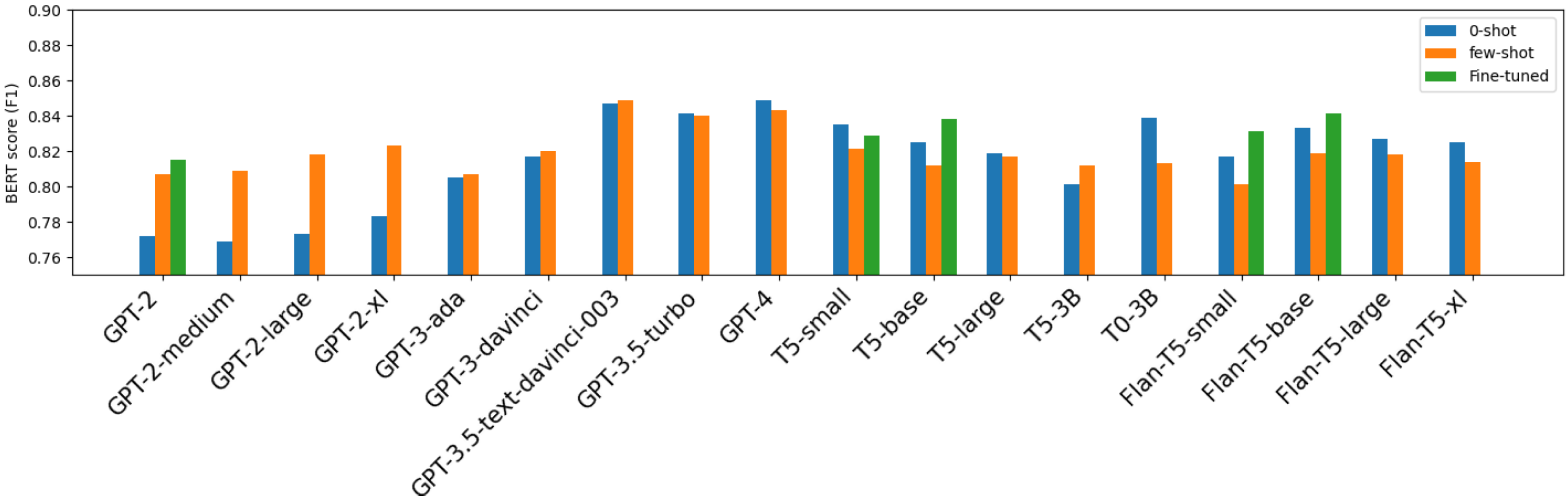
Python is also a great choice in terms of **performance and scalability….**, Python is an interpreted language..easier to debug and **maintain** than compiled languages.

Overall, Python is a great choice for our

Flan-T5 fine-tuned

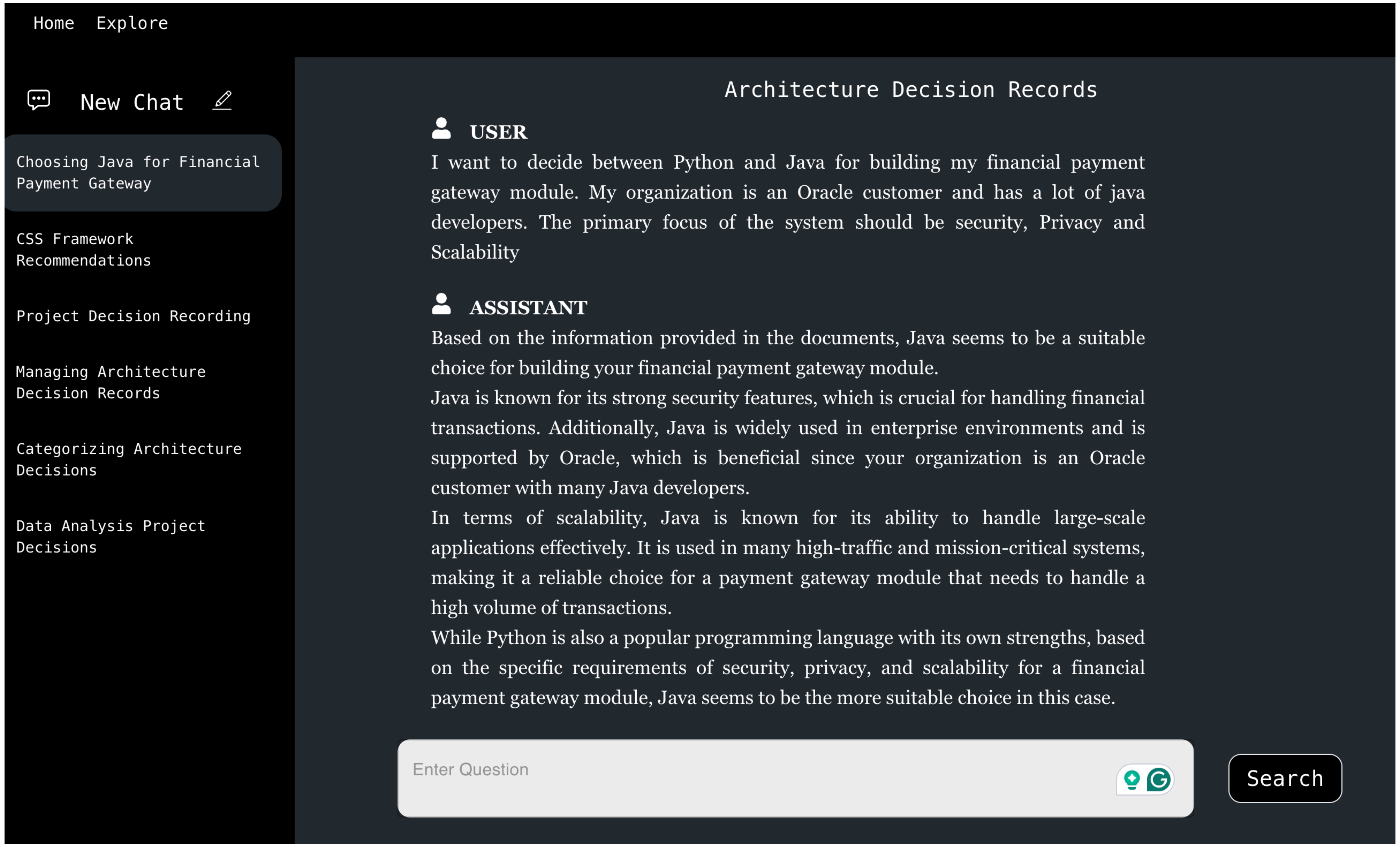
It will have a large dataset, which will display a lot of data. This can help us in preparing the application. We need to choose Python as a programming language, because all of our projects will require Python to use it.

The Overall Results



There is a need for more data and definitely more context!

Introducing the ArchCopilot


Home
Explore
New Chat
Choosing Java for Financial Payment Gateway
CSS Framework Recommendations
Project Decision Recording
Managing Architecture Decision Records
Categorizing Architecture Decisions
Data Analysis Project Decisions
Enter Question
Search

Architecture Decision Records

USER

I want to decide between Python and Java for building my financial payment gateway module. My organization is an Oracle customer and has a lot of java developers. The primary focus of the system should be security, Privacy and Scalability

ASSISTANT

Based on the information provided in the documents, Java seems to be a suitable choice for building your financial payment gateway module.

Java is known for its strong security features, which is crucial for handling financial transactions. Additionally, Java is widely used in enterprise environments and is supported by Oracle, which is beneficial since your organization is an Oracle customer with many Java developers.

In terms of scalability, Java is known for its ability to handle large-scale applications effectively. It is used in many high-traffic and mission-critical systems, making it a reliable choice for a payment gateway module that needs to handle a high volume of transactions.

While Python is also a popular programming language with its own strengths, based on the specific requirements of security, privacy, and scalability for a financial payment gateway module, Java seems to be the more suitable choice in this case.

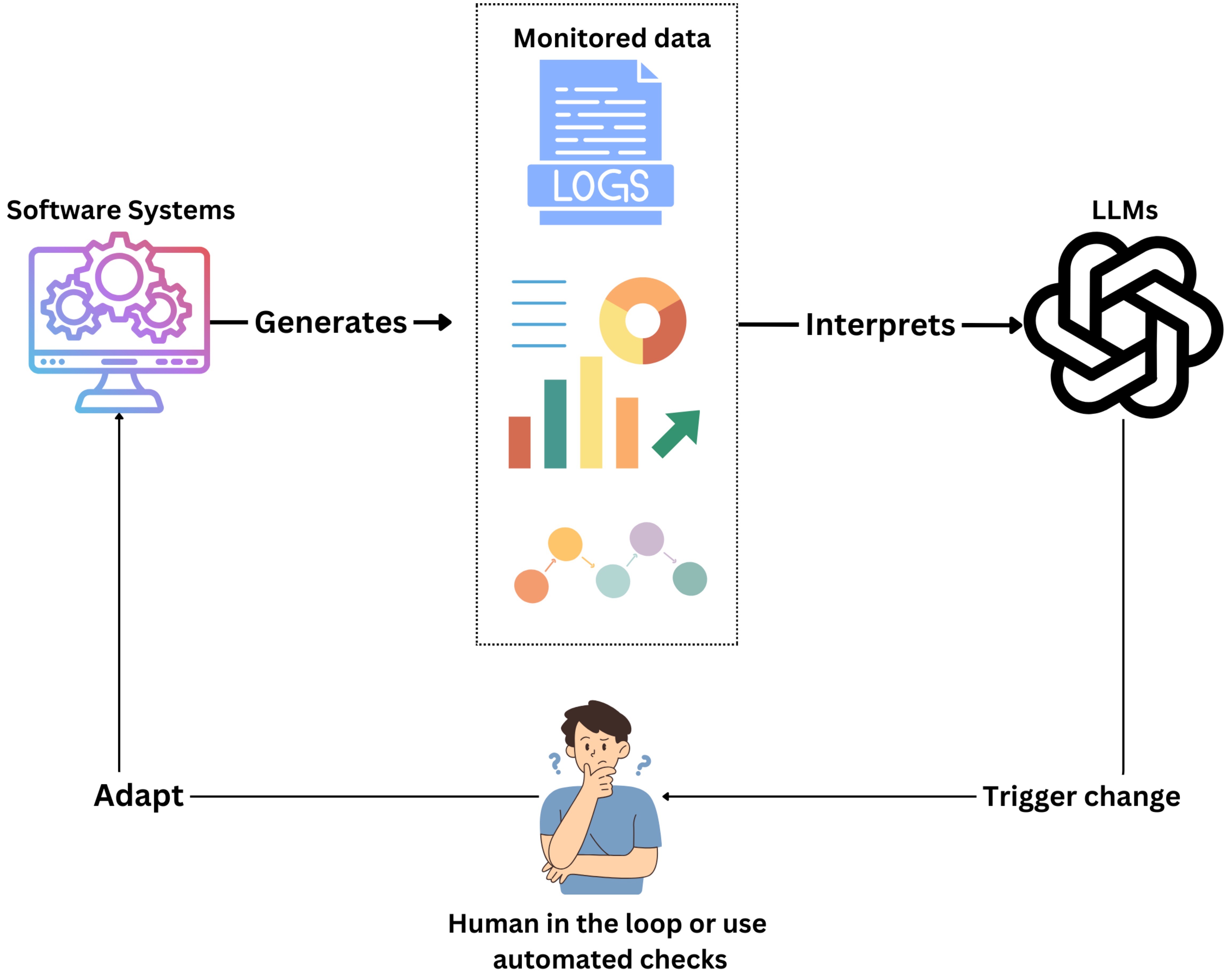
1. Add knowledge - decisions..
2. Search for similar decisions
3. Retrieve knowledge
4. Generate decision records for a given context

A companion for architect!

Thanks to Nikunj Garg, Pabba Ananya, Vansh Pravin, Yatharth Gupta, Ananya Sudi, Rupasree Yeduru and Rudra Dhar

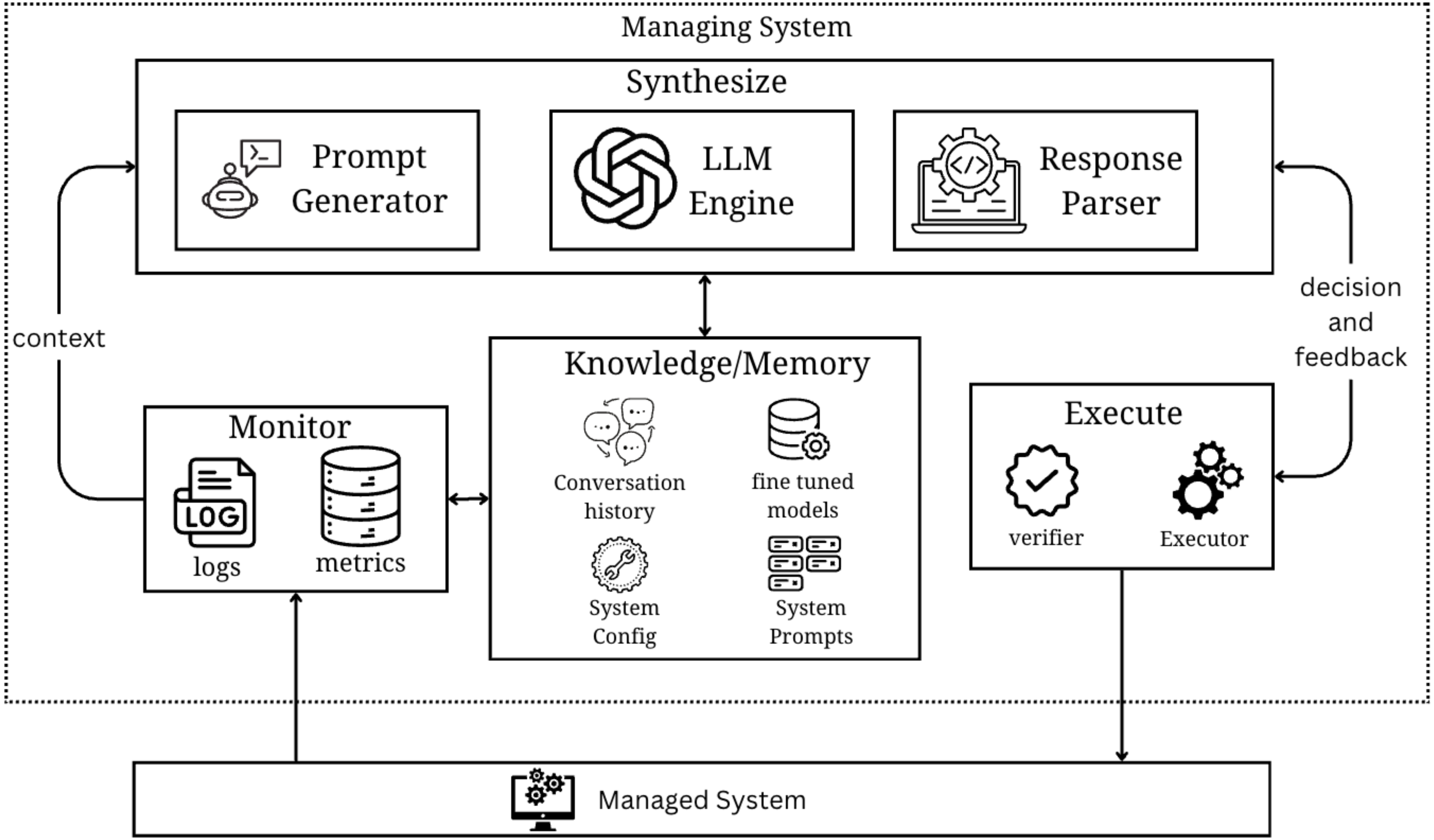
Extending beyond

Design time to run-time adaptation - Can LLMs help?



R.Donakanti, P.Jain, S.Kulkarni, K.Vaidhyanathan, **Reimagining Self-adaptation in the age of Large Language Models,** ICSA 2024, Poster

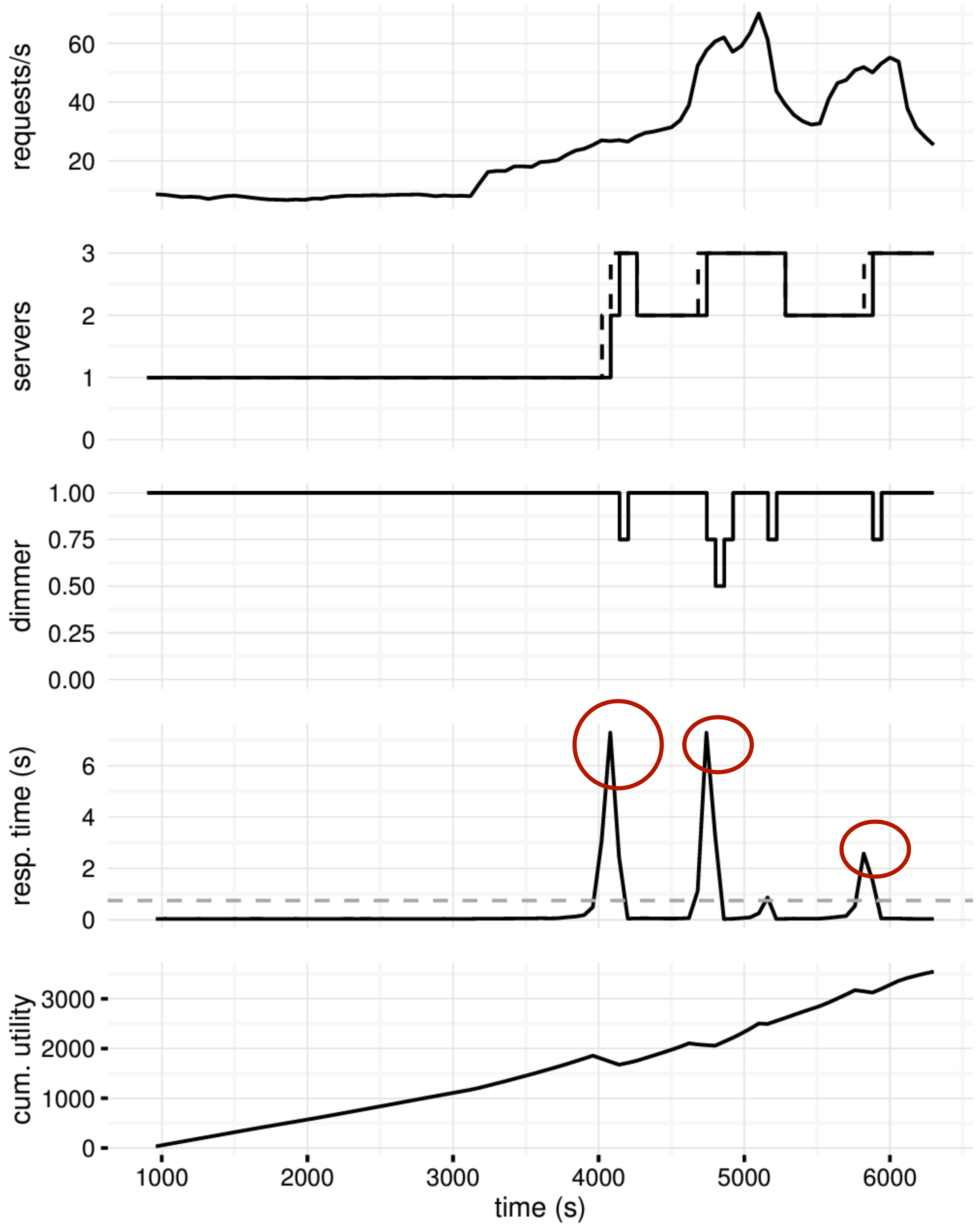
Reimagining Self-adaptation loop



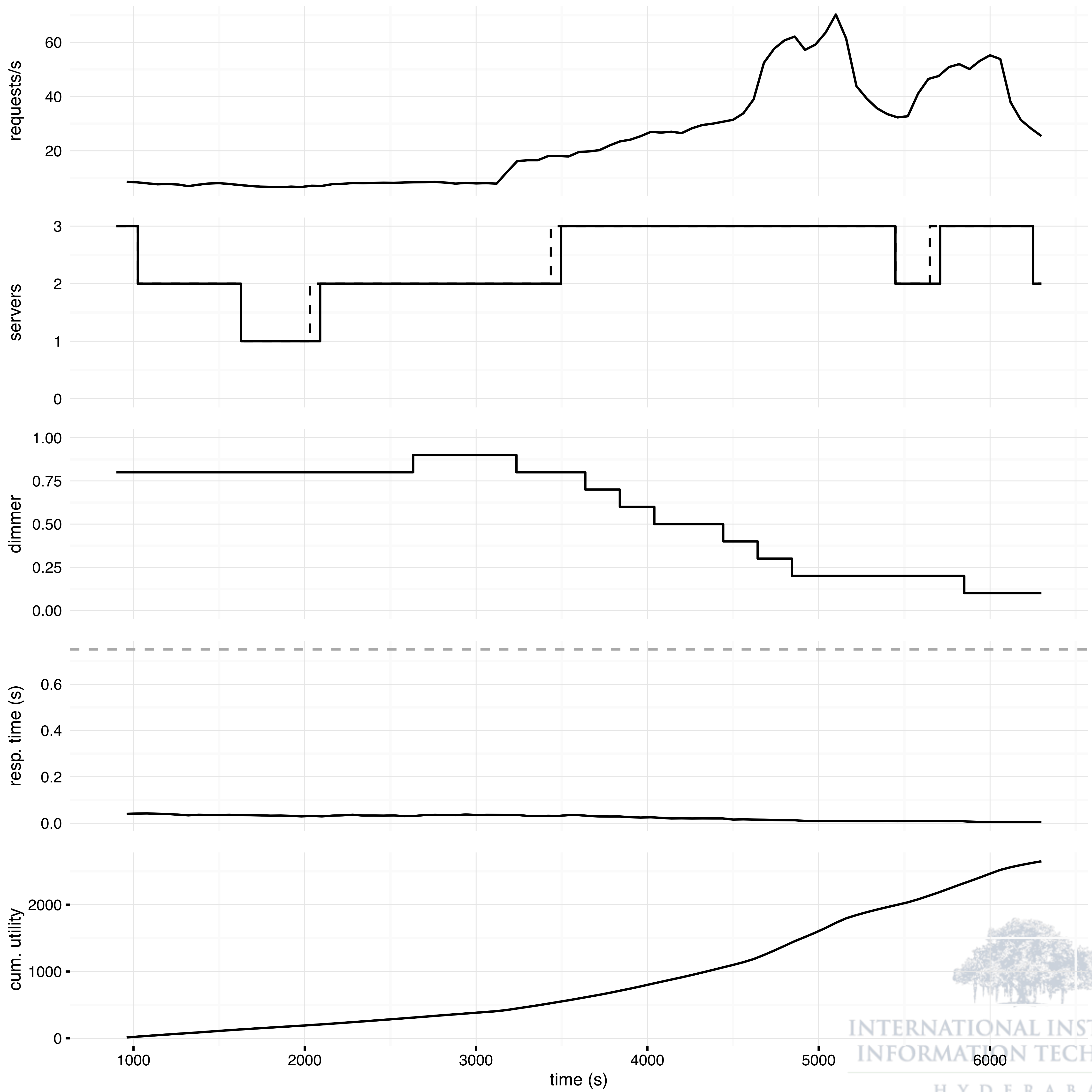
R.Donakanti, P.Jain, S.Kulkarni, K.Vaidhyanathan, **Reimagining Self-adaptation in the age of Large Language Models,** ICSA 2024, Poster

Some Initial Results

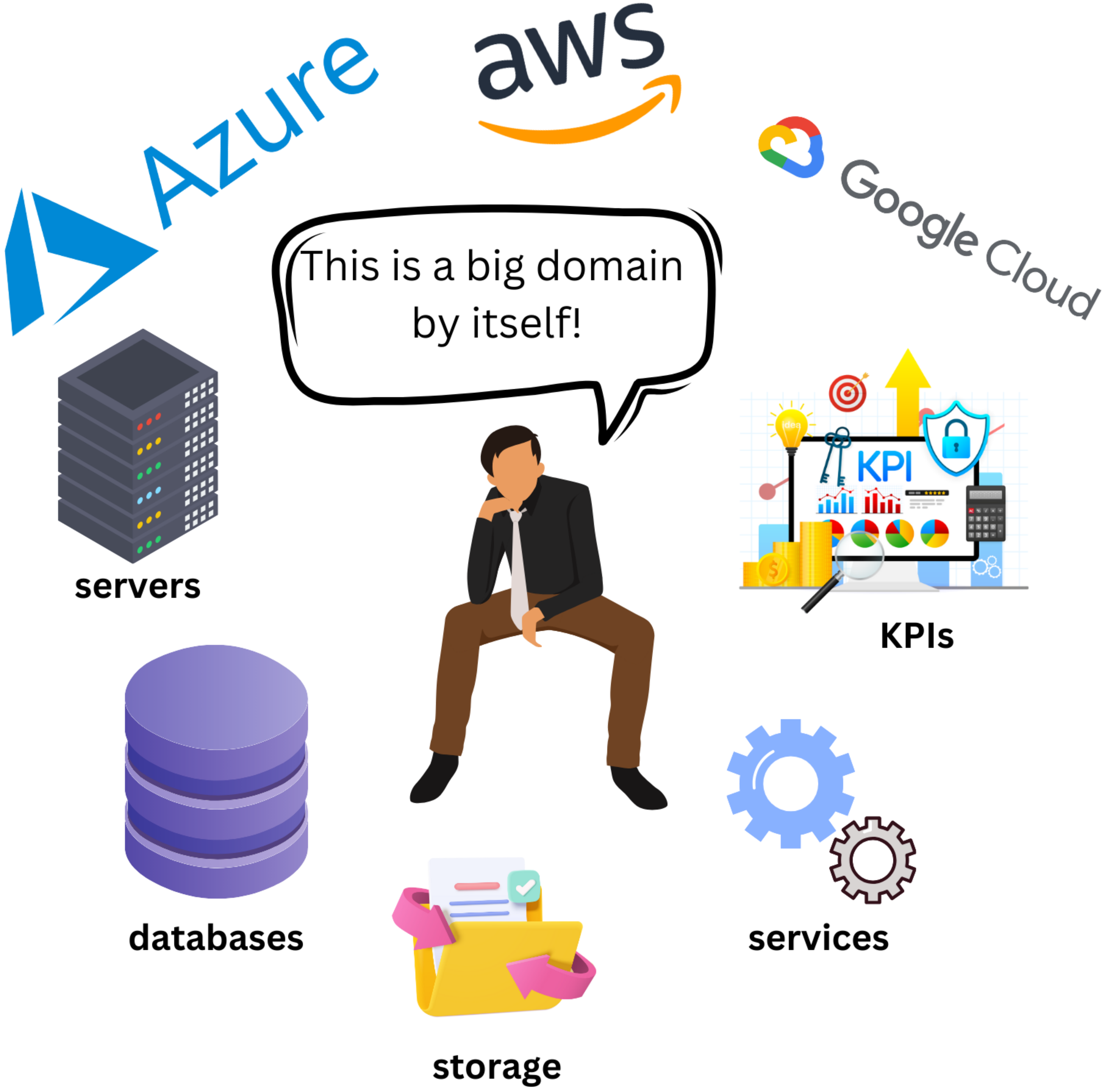
Using SWIM reactive adaptation



Using GPT-4, Promising but..



Making it more Concrete - CloudOps domain



AWS Well Architected Framework

Helps cloud architects build resilient, secure and high performing infrastructure

- **Build around six pillars**
  - Operational Efficiency
  - Security
  - Reliability
  - Performance Efficiency
  - Sustainability
  - Cost

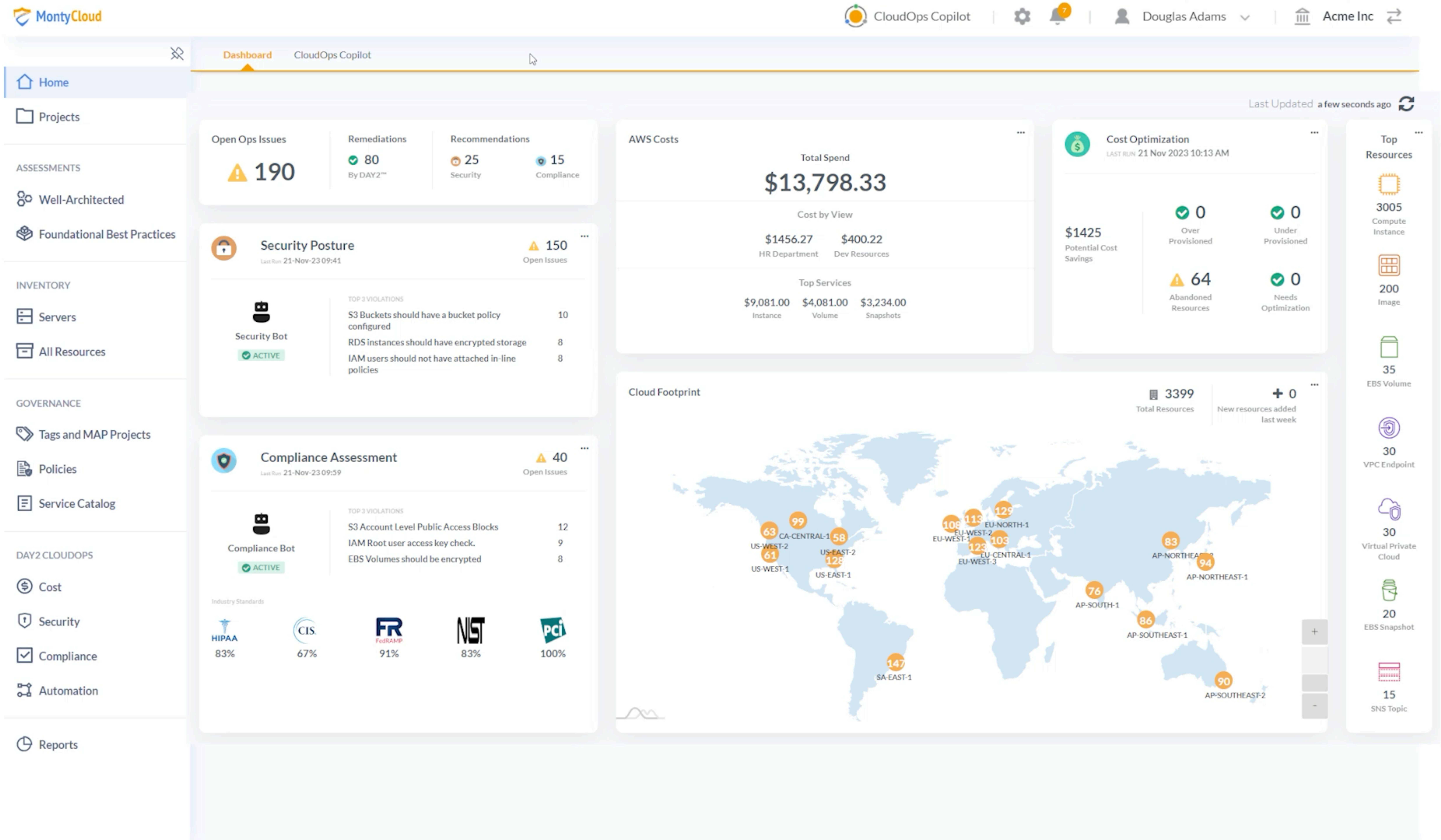
CloudOps Copilot

- Conversational support for cloud SMEs and architects
- Autonomously manage cloud footprint
  - Provision or decommission resources (human-in the loop)
  - Generate tickets or raise alerts
  - Provide insights on the large data captured
  - Perform regular mundane tasks

MontyCloud

No-Code Cloudops Company founded in 2018

https://montycloud.com/cloudops-copilot



Last Updated a few seconds ago

Open Ops Issues
190

Remediations
80
By DAY2™

Recommendations
25 Security
15 Compliance

Security Posture
Last Run 21-Nov-23 09:41
150 Open Issues

Security Bot
ACTIVE

TOP 3 VIOLATIONS

S3 Buckets should have a bucket policy configured	10
RDS instances should have encrypted storage	8
IAM users should not have attached in-line policies	8

Compliance Assessment
Last Run 21-Nov-23 09:59
40 Open Issues

Compliance Bot
ACTIVE

TOP 3 VIOLATIONS

S3 Account Level Public Access Blocks	12
IAM Root user access key check.	9
EBS Volumes should be encrypted	8

Industry Standards

HIPAA	CIS	FedRAMP	NIST	PCI
83%	67%	91%	83%	100%

AWS Costs

Total Spend
$13,798.33

Cost by View
$1456.27 HR Department
$400.22 Dev Resources

Top Services
$9,081.00 Instance
$4,081.00 Volume
$3,234.00 Snapshots

Cost Optimization
LAST RUN 21 Nov 2023 10:13 AM

$1425 Potential Cost Savings
0 Over Provisioned
0 Under Provisioned
64 Abandoned Resources
0 Needs Optimization

Top Resources
3005 Compute Instance
200 Image
35 EBS Volume
30 VPC Endpoint
30 Virtual Private Cloud
20 EBS Snapshot
15 SNS Topic

Cloud Footprint
3399 Total Resources
0 New resources added last week

MontyCloud
CloudOps Copilot
7
Douglas Adams
Acme Inc
Dashboard
CloudOps Copilot
Home
Projects
ASSESSMENTS
Well-Architected
Foundational Best Practices
INVENTORY
Servers
All Resources
GOVERNANCE
Tags and MAP Projects
Policies
Service Catalog
DAY2 CLOUDOPS
Cost
Security
Compliance
Automation
Reports

CloudOps Copilot

AI powered interactive Agent for Cloud Operations Preview

What would you like me to help with?

About my cloud footprint	Well-Architected Assessments	About CloudOps Copilot
Stay on top of your CloudOps What is the latest about my cloud operations?	**Optimize, Improve and Automate** How are my workloads doing against the six Well-Architected pillars?	**We are the easy button for Amazon AWS** How does MontyCloud help my business?
Know the latest about infrastructure deployments How has my infrastructure changed since the last time I asked you?	**Perform c[illegible]sive assessment of your workloads** Summarize insights for "Demo for Amado" workload	**You can innovate more, operate less** What is autonomous CloudOps?
Optimize your cloud cost How can I optimize my cloud spend?	**Secure your workload with encrypted resources** Do I have any unencrypted EBS volumes and EBS snapshots in "Demo for Amado" workload?	**Let's connect** Who can I contact at MontyCloud?

Your personal and company data are protected in this chat

CloudOps Copilot is powered by AI, so surprises and mistakes are possible. Please share feedback so we can improve!

The Future is here

- **Context is needed** - Capturing organizational aspects needs work, code can help!
- **LLMs will hallucinate** - No stopping that but we can reduce it - better engineering!
- **Multiple agents collaborating** together to help architects
- **Large action models (LAMs)** for self-adaptation, task generation
- Lot of potential for support in **architecture migration**

Key Takeaways

LLMs can be a best friend to the architect if used wisely!

- LLM presents a great opportunity for effective AKM!
- We need to have an effort to have qualitative data on architecture!
- Domain specific LLMs which are smaller shall be the way forward - SLMs!
- Need for better ways to architect/engineer systems around LLMs
- LLMs are not here to replace but to support!

Thanks to my team - SA4S@SERC

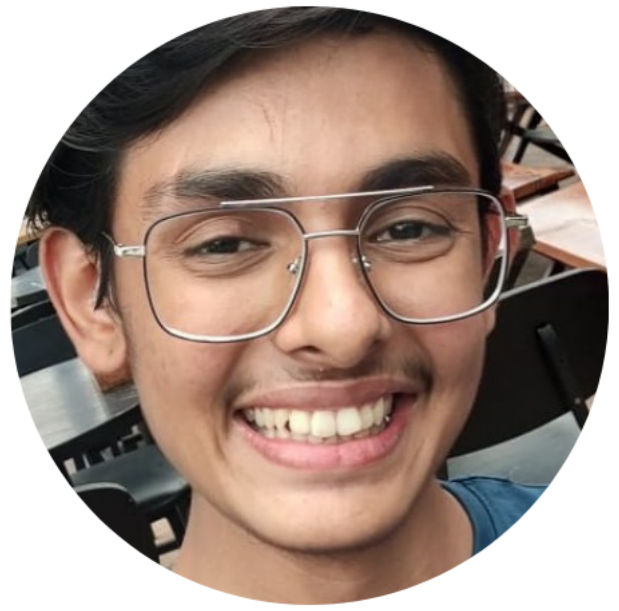
Rudra Dhar

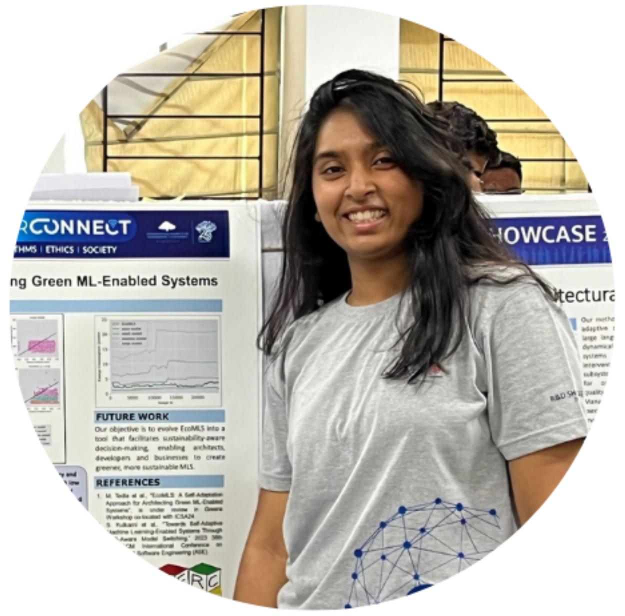
Akhila Matathammal

G V V Satyanarayana Raju

Shubham Kulkarni

Adyansh Kakran


Arya Pravin Marda


Meghana Tedla

Prakhar Jain

Shrikara A

Hiya Bhatt

Chandrasekar S

Team ArchCopilot

Nikunj Garg

Pabba Ananya

Vansh Pravin

Yatharth Gupta

SERC
Software Engineering Research Centre
HYDERABAD

ICSA 2024

SCAN ME

QUATIC 2024

Thank you

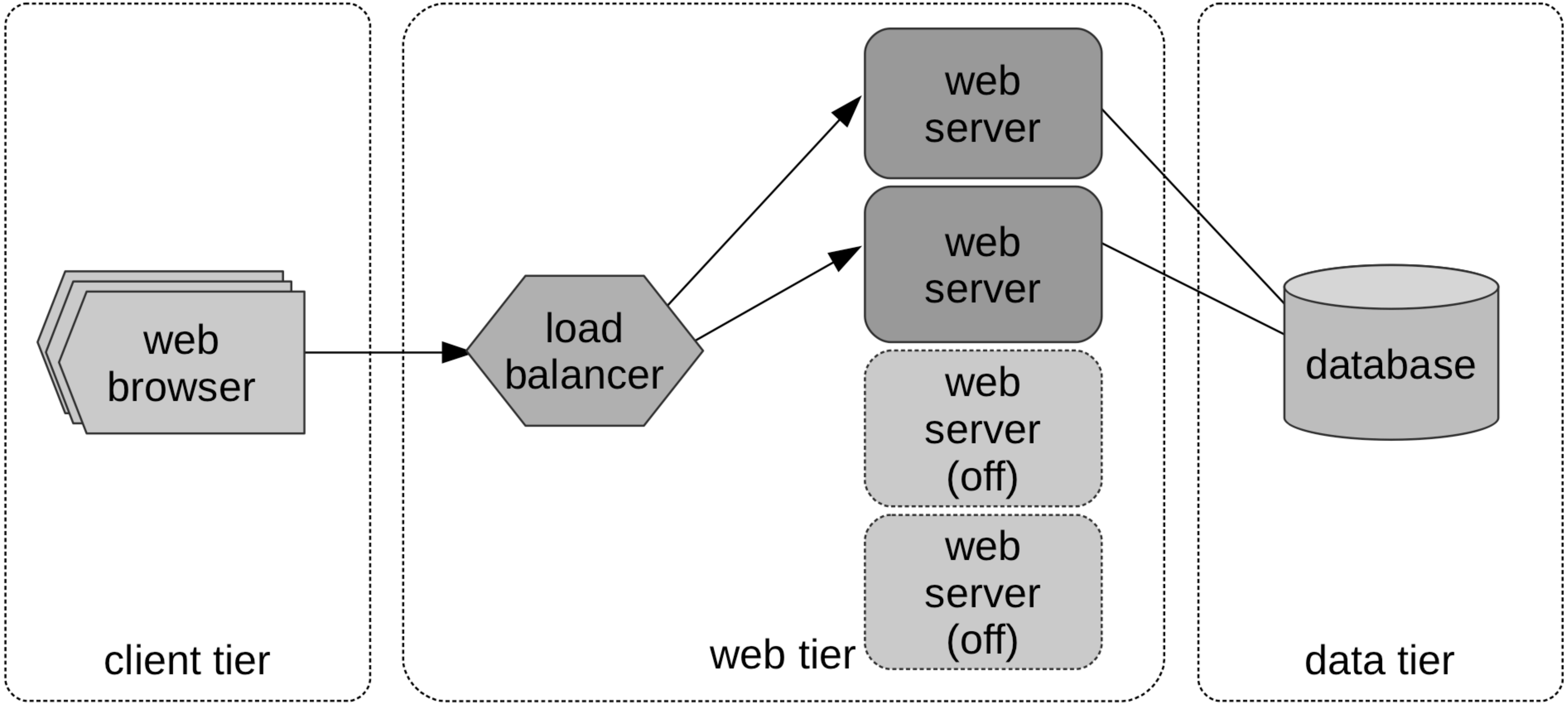
Web: karthikvaidhyanathan.com
Email: karthik.vaidhyanathan@iiit.ac.in
Twitter: @karthi_ishere

SERC
Software Engineering Research Centre

INTERNATIONAL INSTITUTE OF
INFORMATION TECHNOLOGY
HYDERABAD

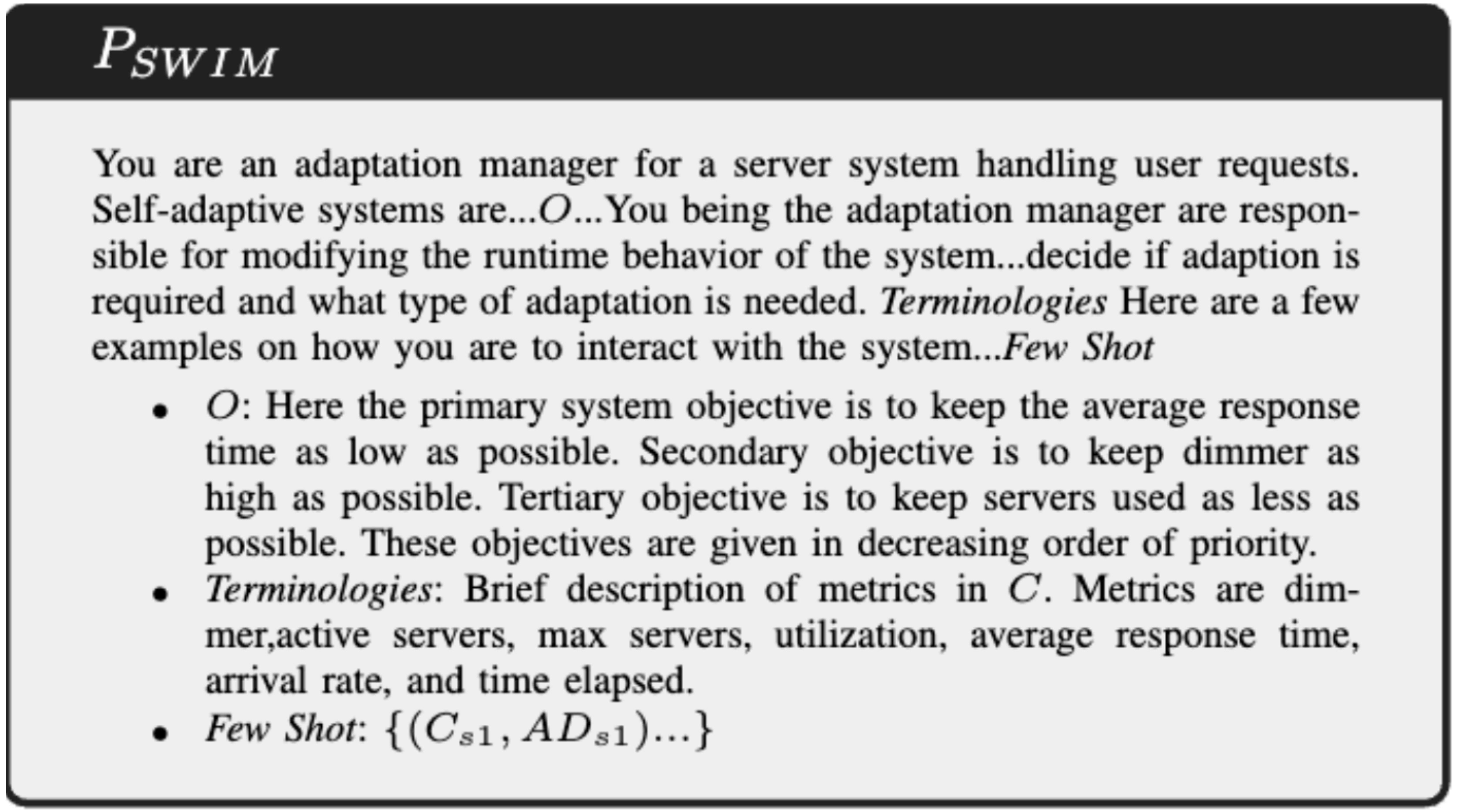
SWIM case study

Web Infrastructure Simulator



Moreno, Gabriel A., Bradley Schmerl, and David Garlan. **Swim: an exemplar for evaluation and comparison of self-adaptation approaches for web applications.** In 2018 IEEE/ACM 13th International Symposium on Software Engineering for Adaptive and Self-Managing Systems (SEAMS), pp. 137-143. IEEE,

Prompts used for SWIM



P_{SWIM}

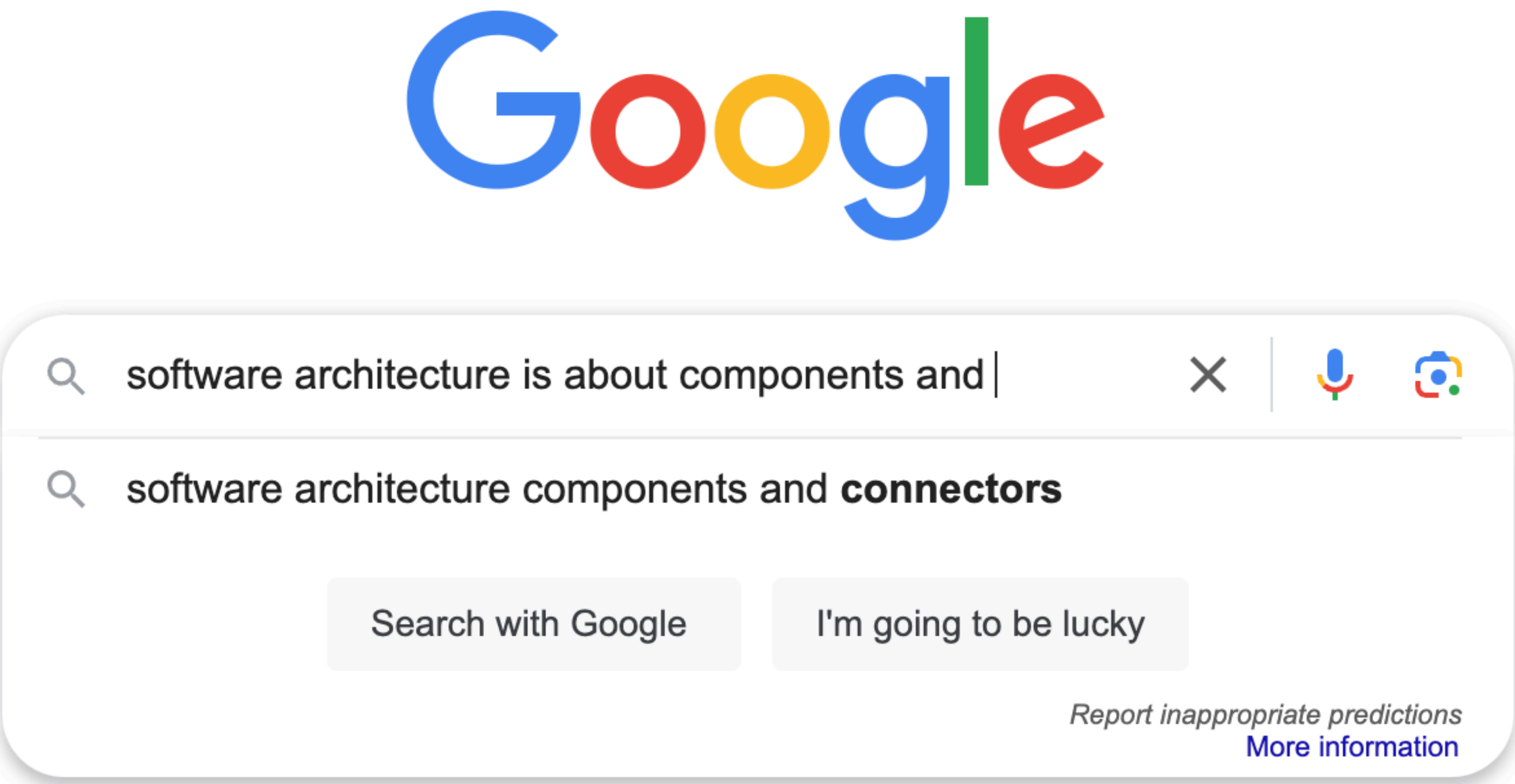
You are an adaptation manager for a server system handling user requests. Self-adaptive systems are...O...You being the adaptation manager are responsible for modifying the runtime behavior of the system...decide if adaption is required and what type of adaptation is needed. *Terminologies* Here are a few examples on how you are to interact with the system...*Few Shot*

- O: Here the primary system objective is to keep the average response time as low as possible. Secondary objective is to keep dimmer as high as possible. Tertiary objective is to keep servers used as less as possible. These objectives are given in decreasing order of priority.
- *Terminologies*: Brief description of metrics in C. Metrics are dimmer,active servers, max servers, utilization, average response time, arrival rate, and time elapsed.
- *Few Shot*: $\{(C_{s1}, AD_{s1})...\}$

RAG Working

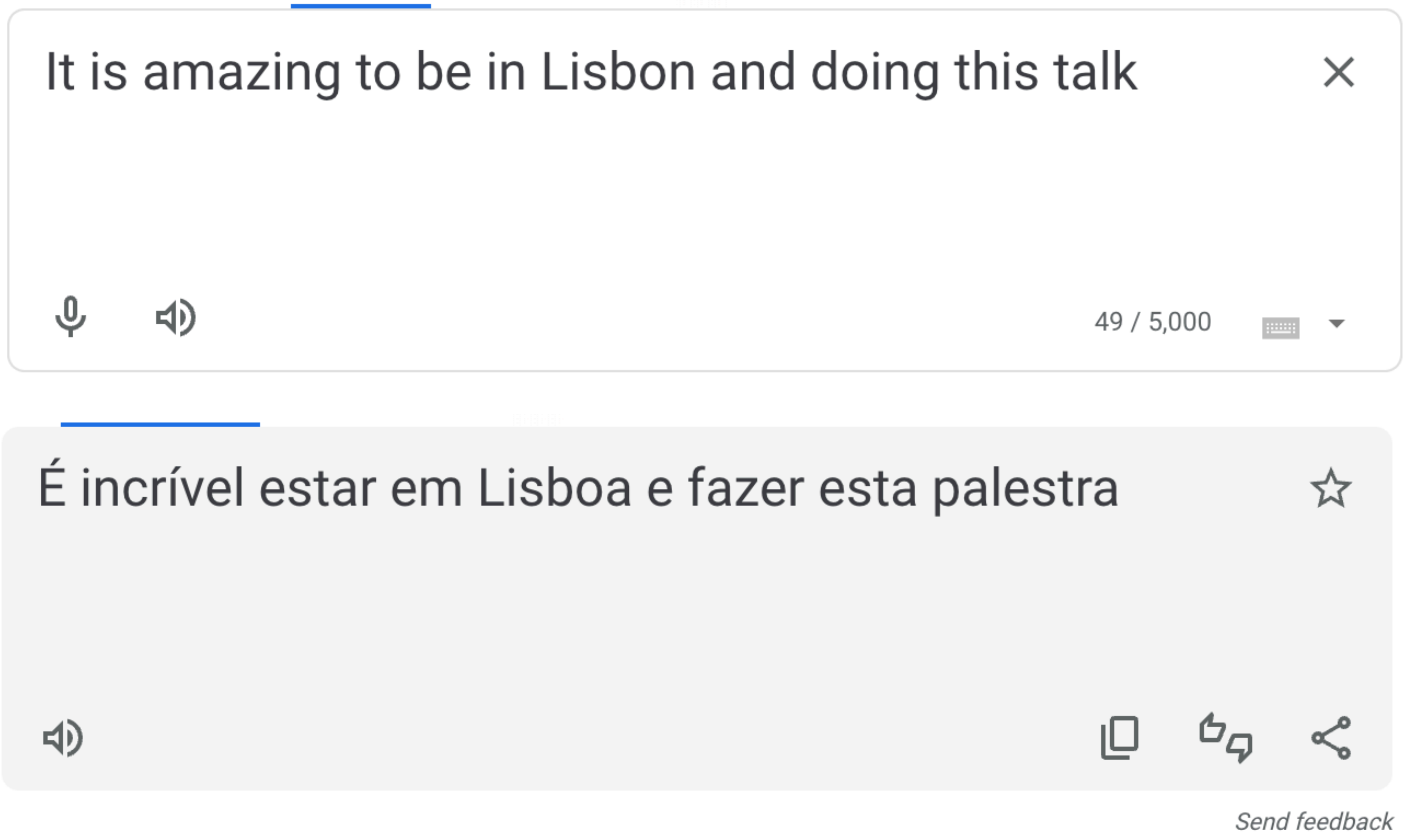
- Software architecture is set of components and connectors.
- Software architecture is about forms, elements and rationale
  - [software, architecture, is, set, of, components, and, connectors]
  - Vector embedding: [1.5, 0.8, 0.6, 0.9. 0.4. 0.2, 0.5] - v1
  - [1.8, 0.2, 0.5, 0.9. 0.4. 0.6, 0.9] - v2
- Store vector embeddings in a VDB
- Retriever
  - “What is software architecture”?: Matching documents V1 + V2
- Generation: Query + retrieved -> LLM -> “software architecture is about components and connectors. It is also about forms, elements and rationale”

Amazing Power of Language Models



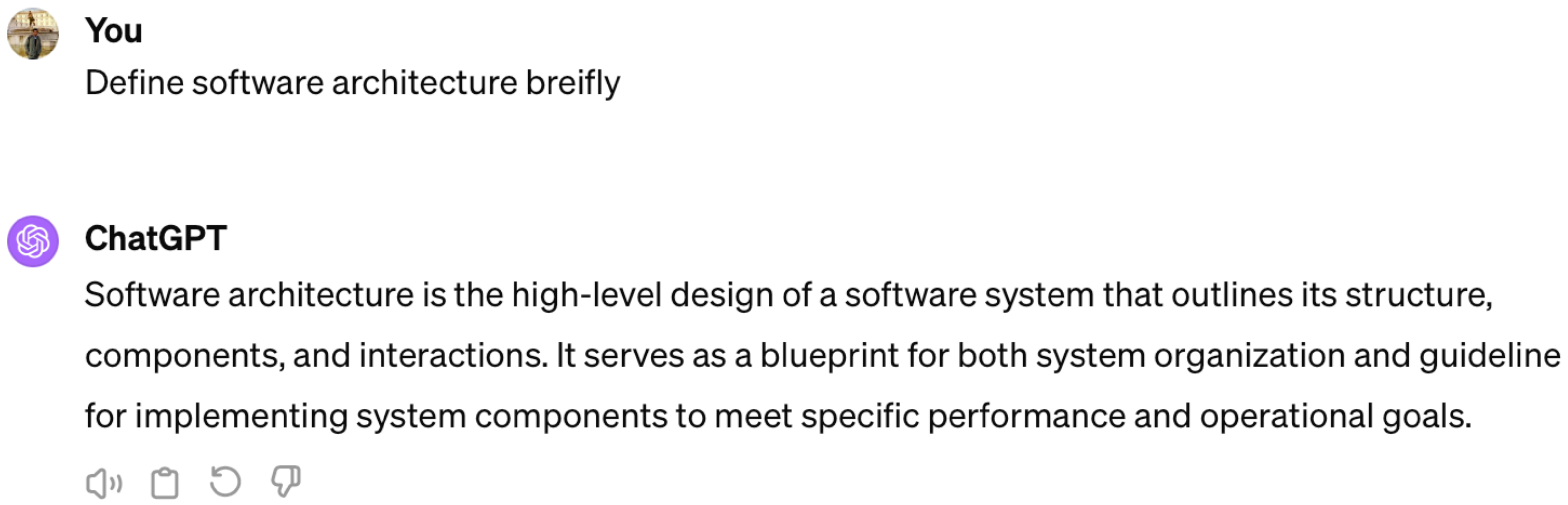
software architecture is about components and

software architecture components and **connectors**



It is amazing to be in Lisbon and doing this talk

É incrível estar em Lisboa e fazer esta palestra



You

Define software architecture breifly

ChatGPT

Software architecture is the high-level design of a software system that outlines its structure, components, and interactions. It serves as a blueprint for both system organization and guideline for implementing system components to meet specific performance and operational goals.