



Kern Chatbot Application

+ Agenda

Kern Chatbot Application

Topic	Page
1. Vision & Goals	4
2. Why NOT Public Chatbots?	7
3. Challenges Before The Chatbot Application	9
4. Kern Chatbot	11
5. What's next in the Chatbot Application	15
6. Summary	17



Vision & Goals



Goal : To Introduce an AI Into a Company

Vision & Goals



[1]

AI at the heart
of workflow



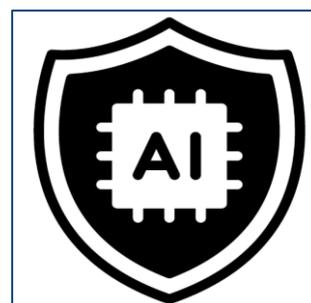
[1]

Faster answers,
higher
efficiency



[1]

Productivity
and growth
unlocked



[1]

Secure,
compliant,
enterprise-grade



[1]

Cost optimization
because of on-
premise server



[1]

Always-on,
round-the-clock
support



Goal : To Introduce an AI Into a Company

Vision & Goals

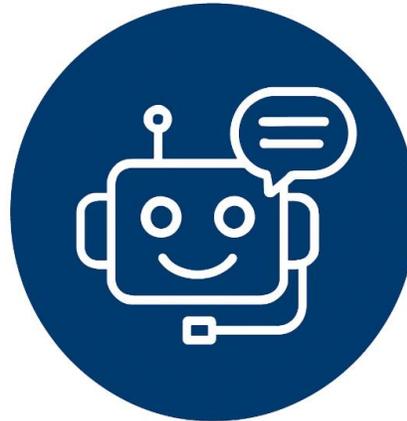
Goal: To create a Private Chatbot



To achieve a data privacy.



To get the help for code related tasks.



[2]



To get the information from internal documents of a company.



To connect the chatbot to external platforms like Gitlab, SAP, etc. for company purpose.

2

Why NOT Public Chatbots?



Why NOT Public Chatbots?

Chatbot application

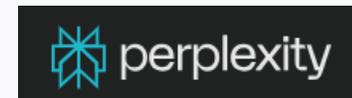
-  Sensitive-data privacy is at risk
-  No plug-ins for our core systems
-  Minimal options for customization
-  Answers come without citations
-  Premium plans are costly and rigid, High Total cost of Ownership (TCO)



[3]

The logo for DeepSeek, consisting of the word "deepseek" in a blue, lowercase, sans-serif font.

[3]



[3]



[3]

3

Challenges Before The Chatbot Application



Challenges Before The Chatbot Application

Chatbot application

-  Slow information retrieval delays work
-  Developers struggle with context & code clarity
-  Knowledge gaps cause inconsistent answers
-  No support outside working hours
-  No possible customization to public chatbots
-  Answers lack cited sources



Kern Chatbot



Kern Chatbot

Chatbot application

User Interface (UI) Overview and Key Features

Hosted on
'On-Premise' Server

<http://192.168.10.64:3000/>

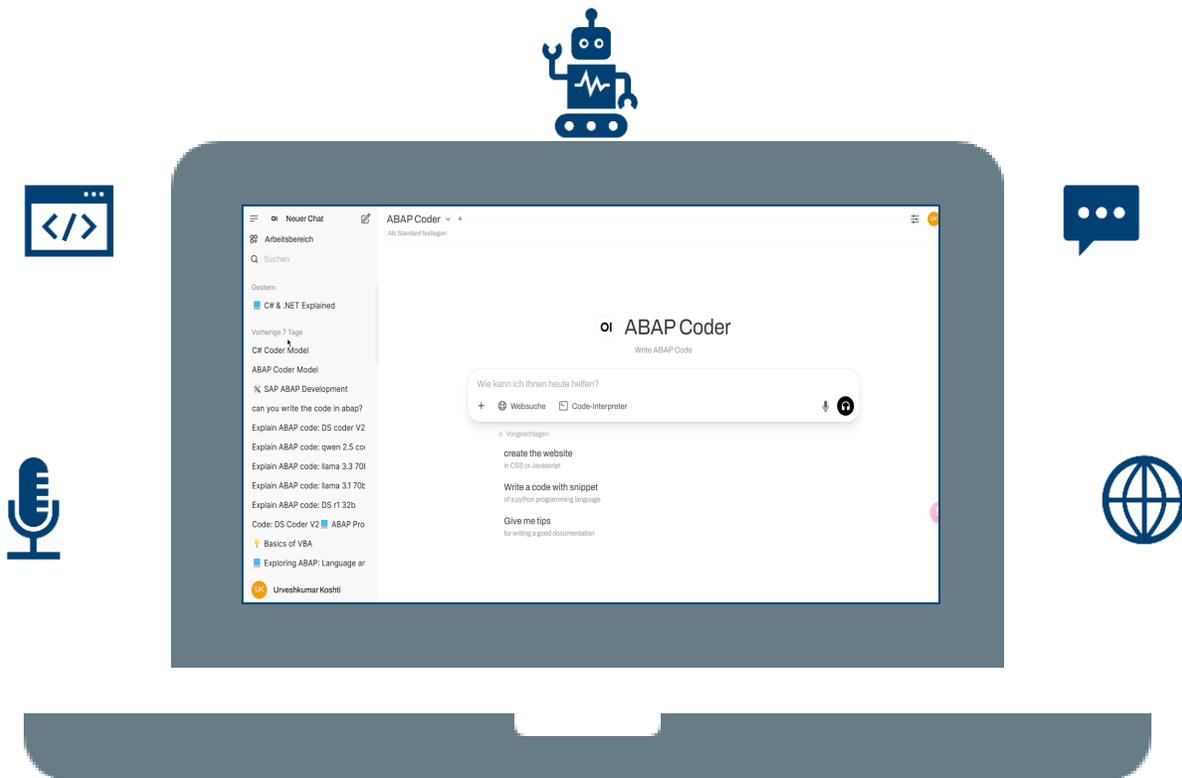
The screenshot displays the user interface of the Kern Chatbot. On the left is a sidebar with a menu containing 'Neuer Chat', 'Arbeitsbereich', and 'Suchen'. Below this is a list of chat history items, including 'C# & .NET Explained', 'C# Coder Model', 'ABAP Coder Model', 'SAP ABAP Development', and 'Basics of VBA'. The main chat area is titled 'ABAP Coder' and features a prompt 'Wie kann ich Ihnen heute helfen?' and a list of suggested topics: 'create the website', 'Write a code with snippet', and 'Give me tips'. A callout box labeled 'Code Assistant' points to the 'ABAP Coder' header, and another callout box labeled 'Question (Prompt)' points to the input field.



Kern Chatbot

Chatbot application

User Interface (UI) Overview and Key Features



Code assistants

- **Write**
- **Optimize**
- **Explain**



Web search: fresh info + citations.



Free-form chat



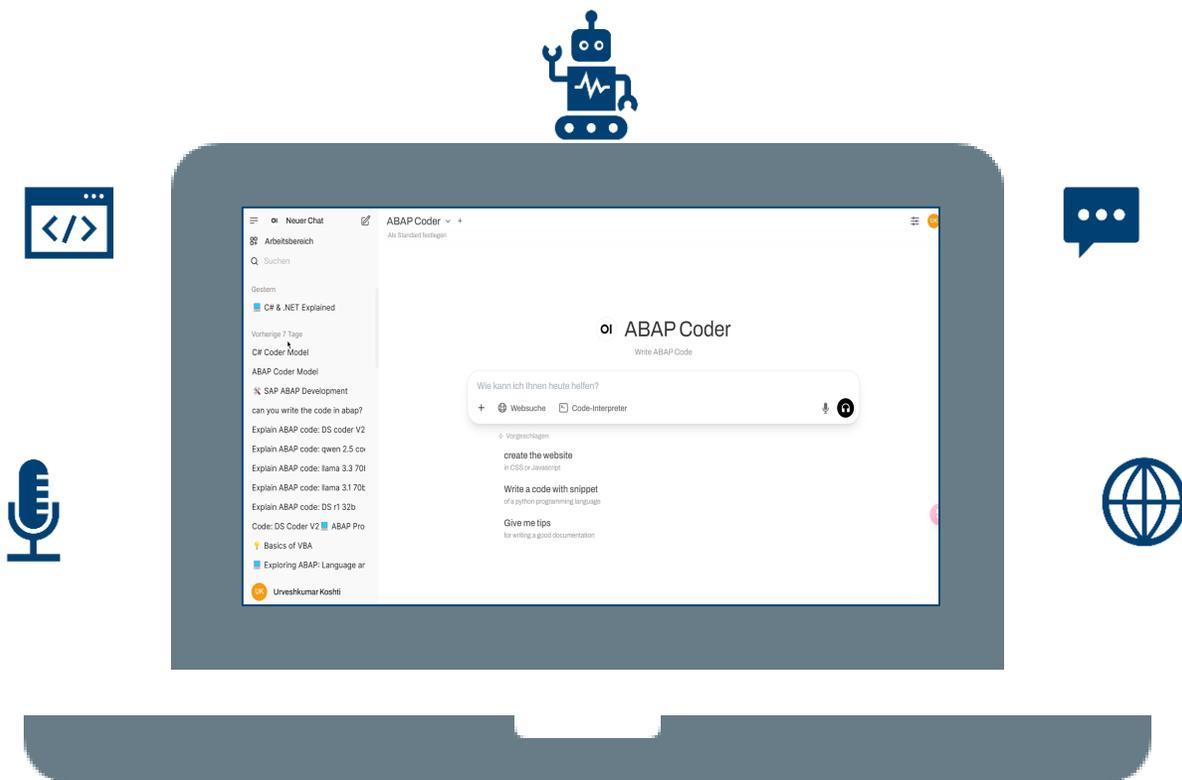
Speech to Text Functionality



Kern Chatbot

Chatbot application

User Interface (UI) Overview and Key Features



- ✓ Code-task support
- ✓ Strong data control & security
- ✓ Built-in web search
- ✓ Speech ↔ text
- ✓ Fast, reliable answers
- ✓ Multi-user collaboration

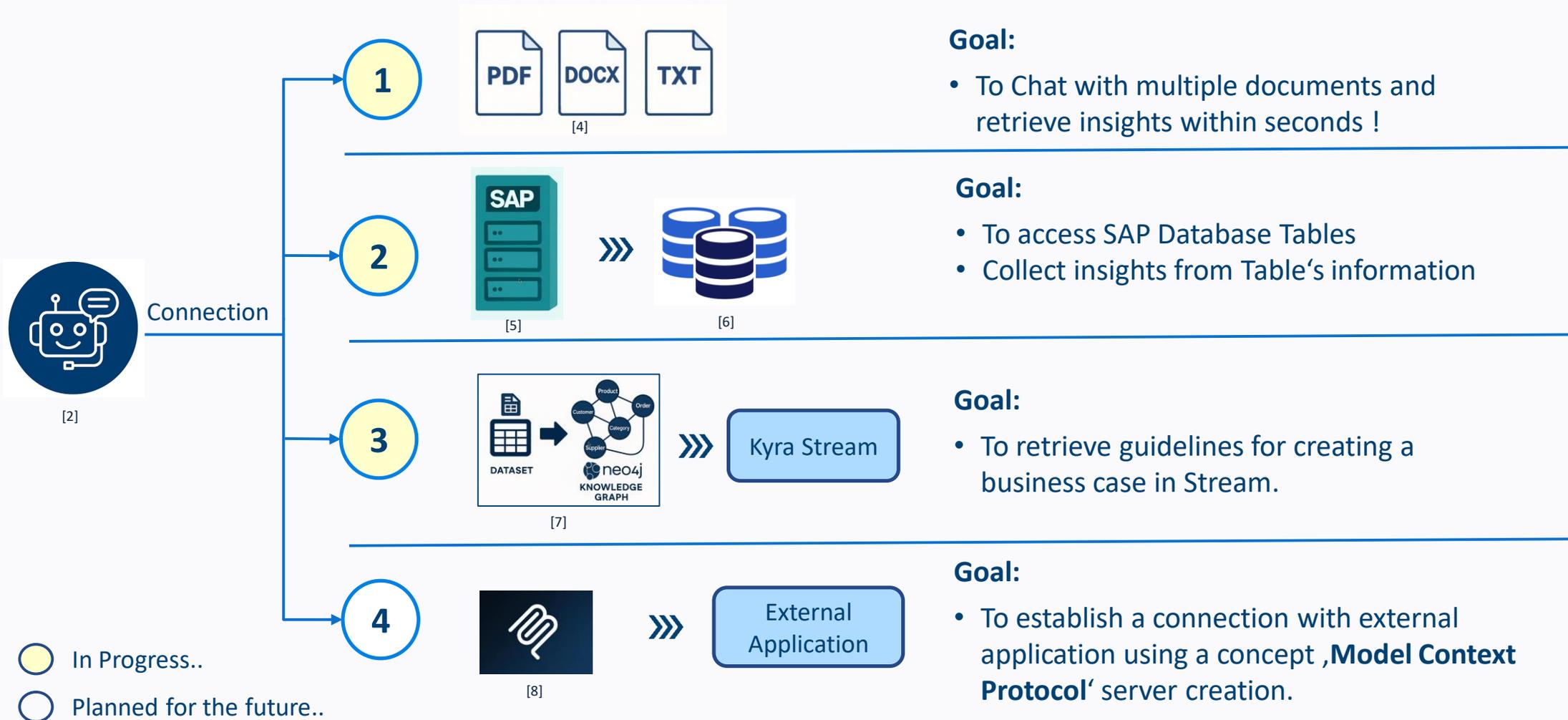
5

What's next in the Chatbot Application



What's next in the Chatbot Application

Chatbot application



In Progress..

Planned for the future..



Summary



Summary

Chatbot application

Goal: Create a Chatbot

To achieve a data privacy.
Gather insights from internal data.
Code task assistance
Link chatbot to enterprise platforms

Challenges Before Chatbot Application

Developers struggle with context and code
Inconsistent answers lead to confusion
Support unavailable after hours
Lack of source citations reduces trust

Why Not Public Chatbots

Data Protection issue
Incapability of integrating company systems
Lack of source citations
High Total cost of Ownership (TCO)

Chatbot so far..

Achieved **coding tasks**

Achieved **data security**

Achieved **Web search**
functionality

Implemented **text-to-speech / speech - to - text** functionality

Implemented **collaboration with multiple users** functionality

What next..

Chat with Document functionality

Connection to **SAP**

Connection to **Stream**

Connection to **External Application**



References

Image references

Image References

- [1] <https://www.flaticon.com/free-icons/powerpoint>
- [2] <https://pngtree.com/so/chatbot-icon>
- [3] <https://www.wikipedia.org/>
- [4] <https://www.vecteezy.com/>
- [5] <https://www.shutterstock.com/>
- [6] <https://stock.adobe.com/>
- [7] <https://timbr.ai/>
- [8] <https://lobehub.com/>



Thank you 😊

Questions..







Backup Slide

Key Features of Chatbot

Key Features

Web Search

Web Search Tool ▾ + ... 🔍 UK

What are top 4 teams in Semi final of UEFA Champions league 2025?

OI Web Search Tool
Searched 12 sites ▾

The top 4 teams in the semi-finals of the UEFA Champions League 2025 are:

1. Arsenal
2. Paris Saint-Germain
3. Barcelona
4. Inter Milan

1 <https://www.si.com/soccer/2025-champions-league-semifi...> 2 <https://www.ndtvprofit.com/sports/champions-league-sem...>
3 <https://en.as.com/soccer/2025-champions-league-semi-fi...>



Backup Slide

Key Features of Chatbot

Key Features

Code Explanation

ABAP Coder v +

```

22 /orbit/cl_log=>log_sys( ).
23 REFRESH <ft_data>.
24 ENDIF.
25 endmethod.

```

Could you explain me this ABAP Code?

OI ABAP Coder 04/17/2025 at 6:01 PM

Sure, let's break down the provided ABAP code step-by-step to understand its functionality and components.

Explanation of the Code

The method `READ` is part of a class that seems to handle some operations related to COSS (Cost Object Structure) data based on specific fiscal years (`gjahr`). Here's a detailed explanation:

```

abap
1 DATA:
2   lo_params      TYPE REF TO ZZ_UKI_CL_PRM_PO_EXAMPLE,
3   lv_dummy1     TYPE string

```

ABAP Coder v +

```

25 endmethod.

```

Data Declarations and Initializations

- Data Declarations:**
 - `lo_params`: A reference to a parameter object of type `ZZ_UKI_CL_PRM_PO_EXAMPLE`.
 - `lv_dummy1`: A dummy string variable used for logging messages.
 - `lt_coss`: An internal table of COSS structures, which will hold the data retrieved from the database.
 - `<ft_data>`: A field symbol representing a dynamic table structure that can be assigned to any table type.
- Creating and Assigning Data:**
 - `CREATE DATA me->md_data TYPE STANDARD TABLE OF COSS;`: Creates an internal table of type COSS and assigns it to `me->md_data`.
 - `ASSIGN me->md_data->* TO <ft_data>;`: Uses a field symbol to assign the created table to `<ft_data>`.
 - `lo_params ?= me->mo_param;`: Casts `me->mo_param` to `lo_params`.

Data Retrieval

- Data Selection:**
 - `SELECT * FROM COSS INTO TABLE lt_coss WHERE gjahr IN lo_params->mr_gjahr;`: Selects all data from the table COSS where the fiscal year (`gjahr`) matches those in `lo_params->mr_gjahr`.

Conditional Handling

- Conditional Checking:**



Backup Slide

Key Features of Chatbot

Key Features

Question-Answer with Document

 Handbuch_StandardK...
File 1.4 MB

Summurize the document

OI llama3.3:70b

The provided text appears to be a user manual or guide for a software system called "Allevo-Koodai". The document covers various topics related to the system, including:

1. Introduction and overview of the system
2. Customizing columns and creating formulas
3. Using JavaScript functions and SVG templates
4. Managing views and calculations
5. Editing and navigating the system
6. Special topics such as matchcode tables, temporary PSP elements, and year-end closing

The document seems to be a comprehensive guide for users of the Allevo-Koodai system, providing detailed information on how to use and customize the system.

Some key points mentioned in the document include:

- CustomColumns can be used to create bespoke calculated columns
- Formulas can be created using JavaScript functions and SVG templates
- Views can be managed and calculations can be performed
- The system has a dashboard and allows for editing and navigation
- There are special topics such as matchcode tables and temporary PSP elements that need to be considered when using the system.

Overall, the document provides a detailed guide for users of the Allevo-Koodai system, covering various aspects of the system and its functionality.

1 Handbuch_StandardKoodai_12.pdf



Backup Slide

Key Features of Chatbot

Key Features

Code Writing

Could you please write the code (Report Program) in ABAP using Dynamic Programming containing Field Symbols, Internal table ?

ABAP Program should fetch the entire data from the Table 'ABCD' where 'ABCD' is the standard Table available in SAP. Do not fetch or filter the specific field from the database table. After fetching the entire data, it should paste the fetched data into an Internal Table.

OI **ABAP Coder 2** Today at 10:09 AM

Welcome to your ABAP assistant! I'll help you create a report program that fetches all data from a standard table named `ABCD` and stores it in an internal table using dynamic programming with field symbols and internal tables. Here's how you can do it:

Explanation of the Solution:

1. **Using Field Symbols:** These are used to dynamically access variables by name, which is useful when dealing with database tables where column names are not known in advance.
2. **Internal Tables:** This will store the fetched data from the database table `ABCD`.
3. **Dynamic SQL Queries:** By using dynamic programming, we can construct and execute SQL queries at runtime based on the structure of the standard table `ABCD`.
4. **Error Handling:** Using exception handling to manage potential errors during data fetching.

ABAP Code:

```
abap ⌵ Collapse Save Copy
1  REPORT zdynamic_fetch_abcd.
2
3  TABLES: abcd. " Standard table 'ABCD'
4
5  DATA: gt_abcd TYPE TABLE OF abcd, " Internal table to store fetched data
6         gs_abcd LIKE LINE OF gt_abcd. " Field symbol for a single record of ABCD
7
8  START-OF-SELECTION.
9    TRY.
10     " Open cursor on the standard table 'ABCD'
11     SELECT * FROM abcd INTO TABLE gt_abcd.
12     IF sy-subrc <> 0.
13       MESSAGE 'No data found in table ABCD' TYPE 'I'.
14     ENDIF.
15     CATCH cx_sy_open_sql_db INTO DATA(lo_sql_error).
16       " Handle SQL database open error
17       MESSAGE lo_sql_error->get_text RAISING sql_database_operation_failed.
18     ENDTRY.
19
20 " Display the fetched data (Optional, for testing purposes)
21 LOOP AT gt_abcd INTO gs_abcd.
22   WRITE: / gs_abcd-field1, " Replace 'field1' with actual field names from ABCD
23           gs_abcd-field2,
24           gs_abcd-field3. " Add more fields as needed
25 ENDLOOP. ⌵
```

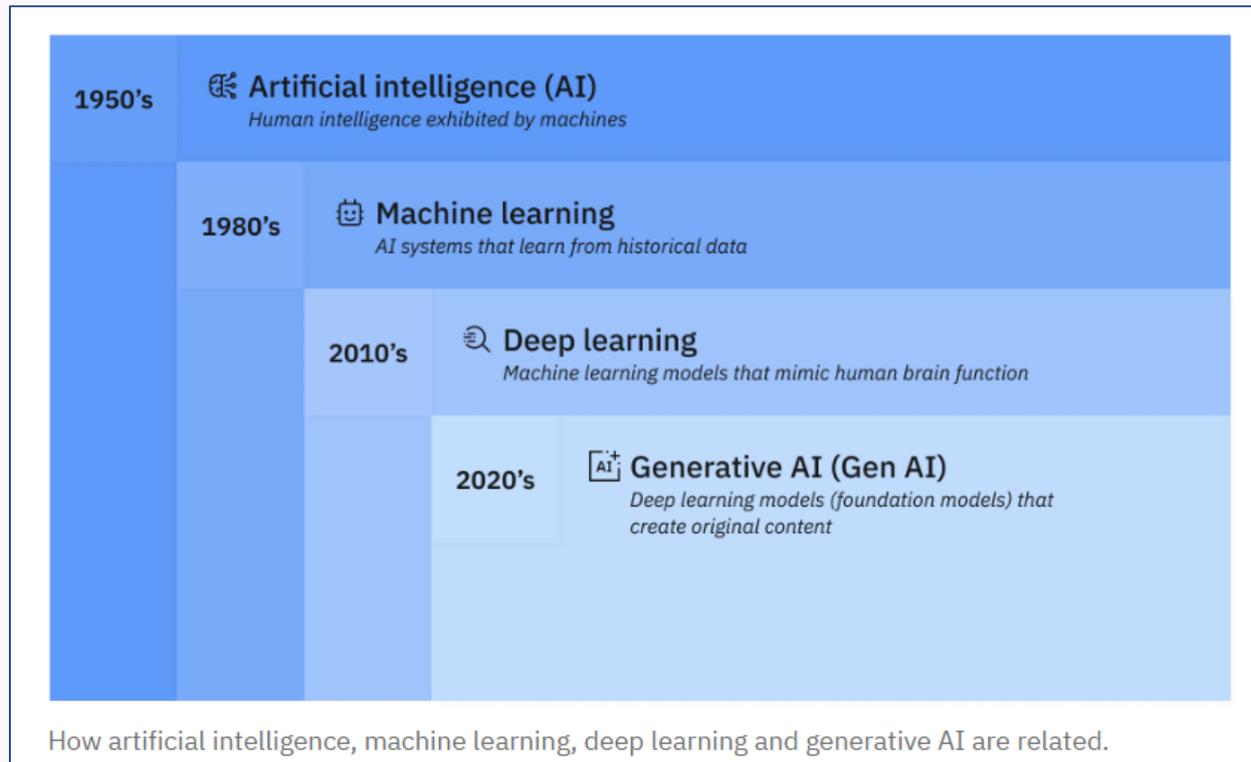


Backup Slide

Artificial Intelligence

What is AI

Artificial intelligence (AI) is technology that enables computers and machines to simulate human learning, comprehension, problem solving, decision making, understanding language, recognizing patterns, learning from experience, and making decisions.



<https://www.ibm.com/>

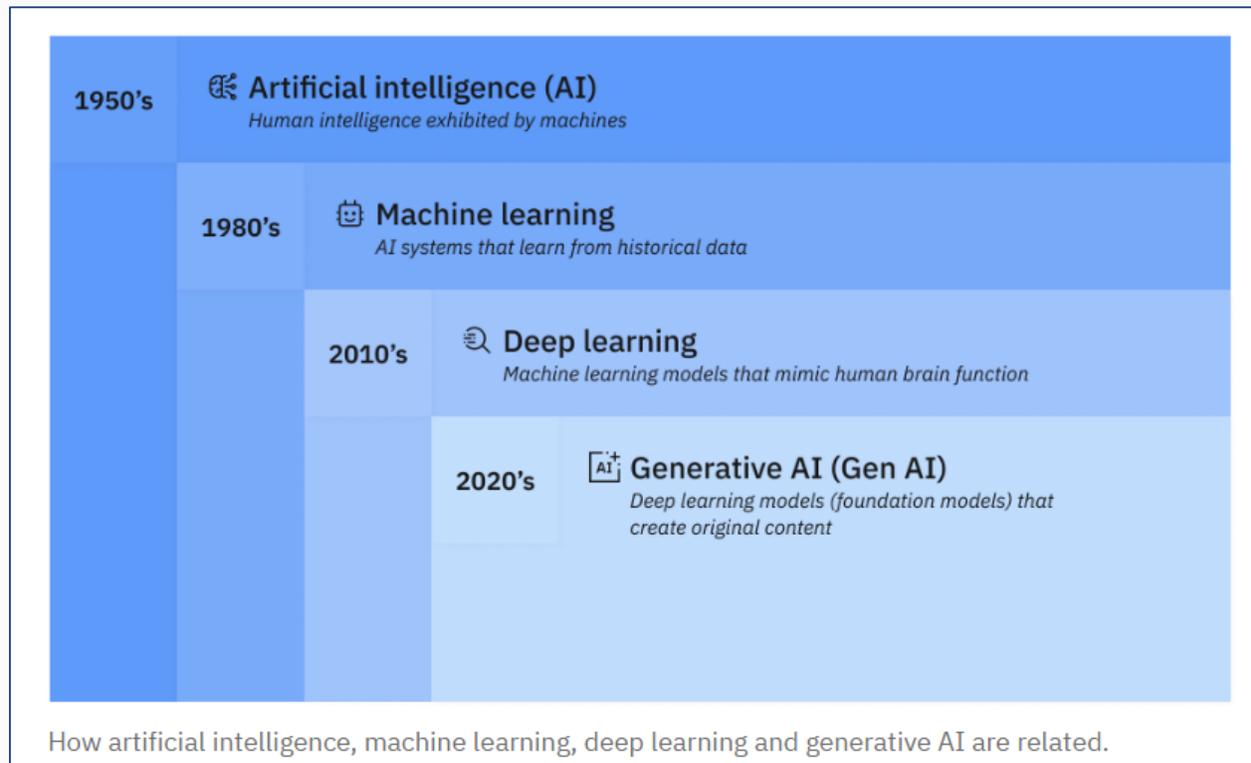


Backup Slide

Artificial Intelligence

What is Generative AI

Generative AI is a branch of artificial intelligence focused on creating new content—such as text, images, audio, video, code, or 3-D models.



What Gen AI does:

1. Write an Article,
2. Generate Images
3. Compose music
4. Design graphics
etc...

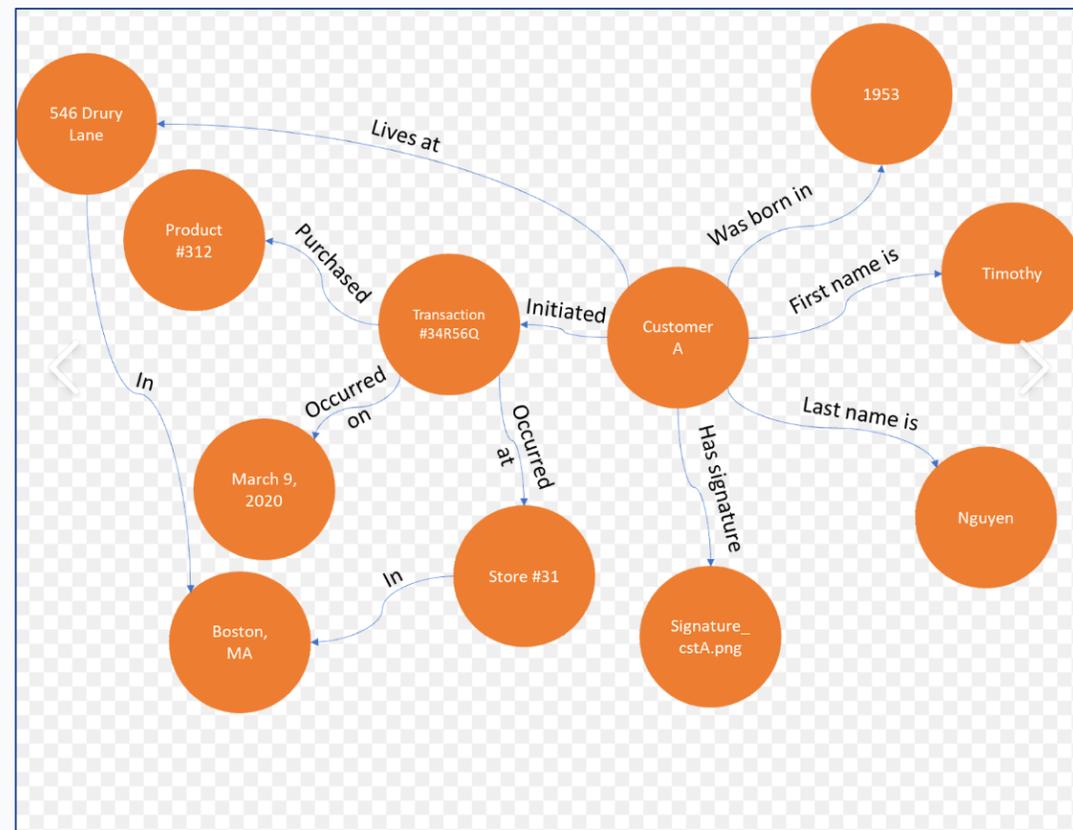


Backup Slide

Artificial Intelligence

What is Knowledge Graph

A knowledge graph, also known as a semantic network, represents a network of real-world entities—such as objects, events, situations or concepts—and illustrates the relationship between them. This information is usually stored in a graph database and visualized as a graph structure, prompting the term knowledge “graph.” [A]



[B]

[A] <https://www.ibm.com/think/topics/knowledge-graph>

[B] <https://www.eckerson.com/>

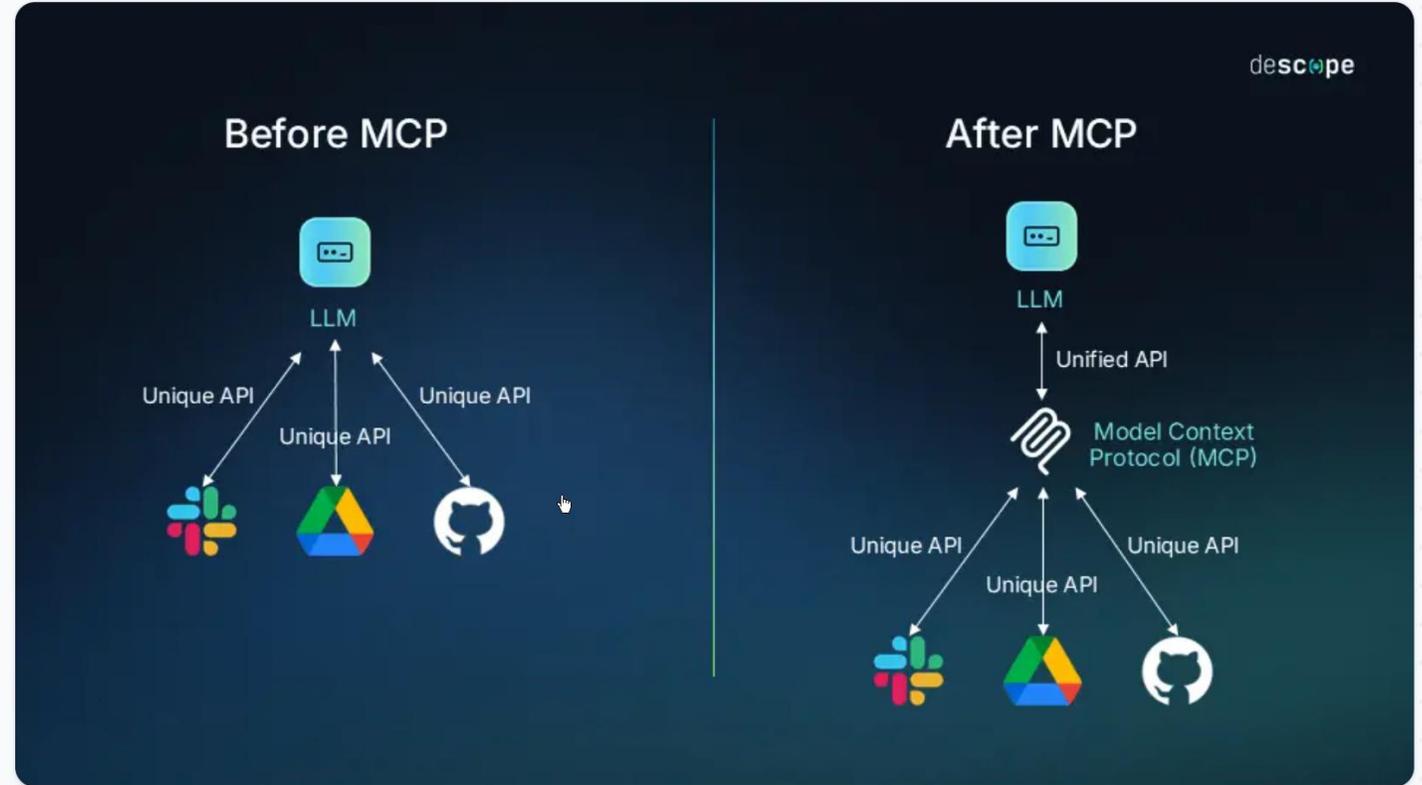


Backup Slide

Artificial Intelligence

What is Model Context Protocol (MCP)

MCP is a two-way communication bridge between AI assistants and external tools, providing access to information.^[A]



[B]

^[A] <https://zapier.com/blog/mcp/>

^[B] <https://www.descope.com/learn/post/mcp/>