CLOUDERA

# GENERATIVE AI AND THE FUTURE OF SQL

Cloudera SQL AI Assistant

# TABLE OF CONTENTS

Björn

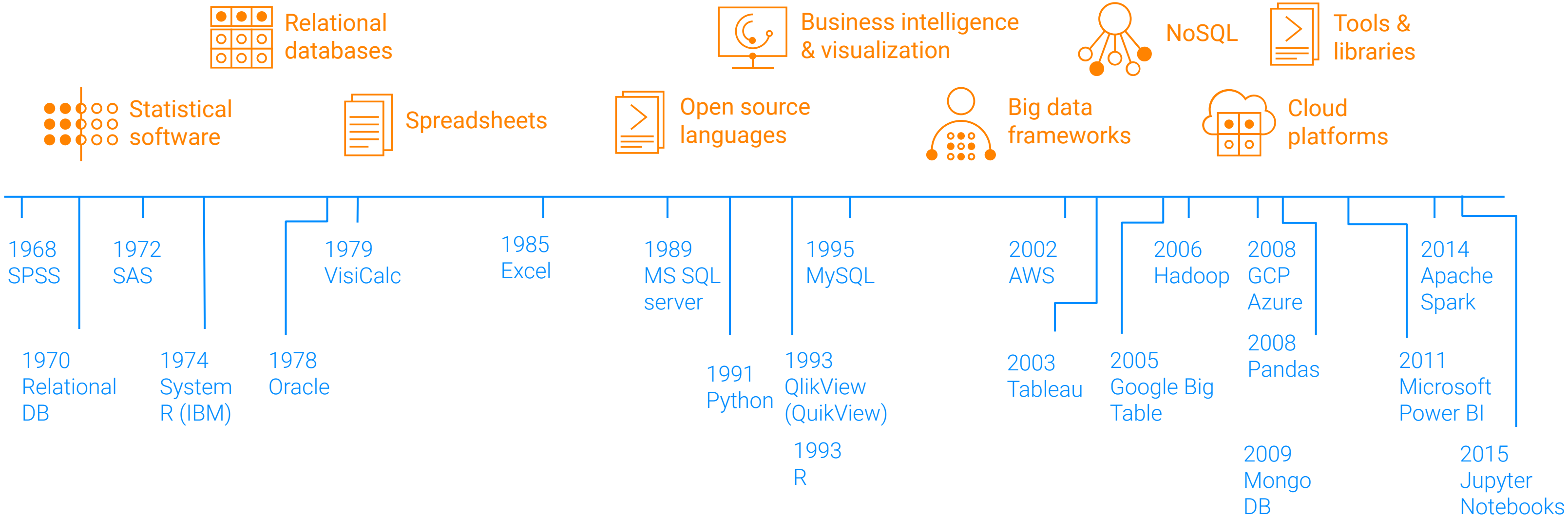
Alm Mangell

Senior Staff Software Developer

balm@cloudera.com

# THE TRADITIONAL TOOLBOX

## Early tools, DBMS, visualization, open source & big data



Statistical software

Relational databases

Spreadsheets

Open source languages

Business intelligence & visualization

Big data frameworks

NoSQL

Tools & libraries

Cloud platforms

1968 SPSS

1970 Relational DB

1972 SAS

1974 System R (IBM)

1978 Oracle

1979 VisiCalc

1985 Excel

1989 MS SQL server

1991 Python

1993 QlikView (QuikView)

1993 R

1995 MySQL

2002 AWS

2003 Tableau

2005 Google Big Table

2006 Hadoop

2008 GCP Azure

2008 Pandas

2009 Mongo DB

2011 Microsoft Power BI

2014 Apache Spark

2015 Jupyter Notebooks

# “SQL IS TO DATA WHAT THE BROWSER IS TO THE INTERNET”

ChatGPT 2024

# A 50 YEAR OLD COMPANION

## Structured Query Language

```
SELECT * FROM sales
WHERE year > 2022;
```

```
WITH RecursiveCTE AS (
   SELECT id, x_value, 1 AS Depth
   FROM tbl_Mystery
   WHERE x_value IS NOT NULL
   UNION ALL
   SELECT m.id, m.x_value, r.Depth + 1
   FROM tbl_Mystery m
   INNER JOIN RecursiveCTE r ON m.parent_id = r.id
   WHERE m.x_value > r.x_value
)
SELECT TOP 5 id, x_value, Depth
FROM RecursiveCTE
ORDER BY Depth DESC, x_value ASC;
```

# A 50 YEAR OLD COMPANION

## Hard numbers

48.66%

of SO Developers are using SQL

Which programming, scripting, and markup languages have you done extensive development work in over the past year, and which do you want to work in over the next year? *

3/3

most used DBMS run SQL

Which database environments have you done extensive development work in over the past year, and which do you want to work in over the next year? *

* ) https://survey.stackoverflow.co/2023/#most-popular-technologies-language

# A 50 YEAR OLD COMPANION

## Short about SQL

- First appeared in 1974 (SEQUEL)
- Single statement for multiple records
- Declarative with procedural elements
- Multiple type of sublanguages
- ANSI 1986, ISO 1987

DQL - Data query language
DDL - Data Definition language
DML - Data manipulation language
DCL - Data control language

# HUE - more than SQL

## A swiss army knife for the data analyst

- First released 2010
- Open source
- gethue.com
- github.com/cloudera/hue
- Distributions also available via
  - Cloudera Data Platform
  - Amazon AWS EMR
  - Google Cloud Dataproc
  - Azure HDInsight

# DEMO

Live demo of Hue & SQL AI Assistant

# BIG DATA CHALLENGES

## Providing the right context

- Foundation models know SQL
- Will gladly hallucinate
- Foundation models do not know your
  - Specific SQL task
  - SQL dialect
  - Metadata (DDL)
  - Sample data

**PROMPT:**
*Generate SQL that lists all customers who bought something last year*

**OUTPUT:**

```
SELECT DISTINCT c.customer_id, c.name
FROM customers c
JOIN orders o ON c.customer_id = o.customer_id
WHERE o.order_date >= '2023-01-01'
AND o.order_date <= '2023-12-31';
```

# BIG DATA CHALLENGES

## AI bandwidth limitations

- Metadata for thousands of tables
- Limited bandwidth for LLM in-context learning
- Cost of maximizing context window

**> 4.8MB**

Metadata + sample data for 2000 tables

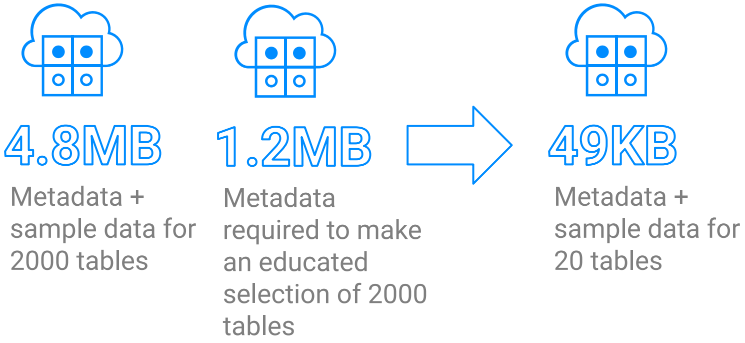
**128KB**

Maximum current context window of GPT-4 Turbo

# BIG DATA CHALLENGES

## Overcoming bandwidth limitations using RAG

*Only pass in metadata about relevant DBs and tables*



4.8MB

Metadata + sample data for 2000 tables

1.2MB

Metadata required to make an educated selection of 2000 tables

49KB

Metadata + sample data for 20 tables

- Narrow the search space upfront by
  - VectorDB
  - Semantic search
  - Use the LLM
- Augment the prompt by providing the LLM with the data needed

# BIG DATA CHALLENGES

## Overcoming bandwidth limitations using RAG

1

Create an embedding from the user input

2

Search for matching embeddings describing a unique table

3

Retrieve the metadata and sample data for the best matching tables

4

Append the relevant data to the prompt and make an LLM request

# BIG DATA CHALLENGES

## Challenges introduced by RAG

- Potential loss of information
- Additional dependencies
- Caching and syncing
- Latency
- Potential for information leakage

# BIG DATA CHALLENGES

## Other challenges

- Security
  - Access control
  - Data leakage
  - Dangerous content
- Quality
  - Reduce hallucinations
  - Improve SQL quality
  - How to verify quality
- LLMs execute poorly on multiple goals

# “THE REPORTS OF MY DEATH ARE GREATLY EXAGGERATED”

Mark Twain

# THE END OF SQL

## Is it different this time?

### AYES

- Object–relational mismatch
- Large (many keywords)
- Technical (complex SQL)
- AI

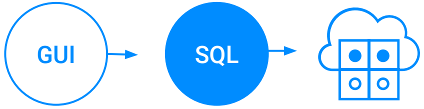
### NAYS

- Standardized
- Wide Ecosystem and Tooling
- Declarative
- SQL translators
- NoSQL SQL
- Language adaptation
- Better data formats & engines

# THE END OF SQL

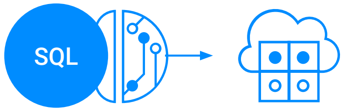
## Future interaction with data

### TRADITIONAL UI



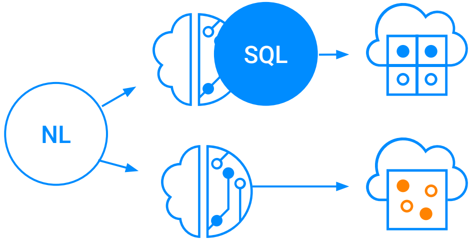
Click, select, drag & drop — Same as always, although with “smarter” functions with the help of AI

### AI ASSISTED CODING



SQL and code editors — will remain but become much smarter and more user friendly with AI

### NATURAL LANGUAGE



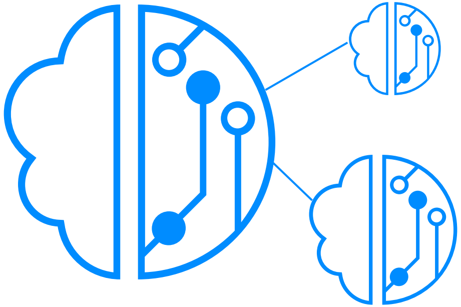
Chats — will increase in adaptation and fundamentally change how we interact with data

CLOUDERA

# THE END OF SQL

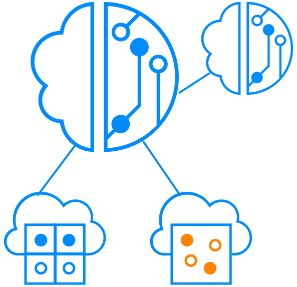
## Accessing data via AI

KNOWS EVERYTHING



VS

CAN ACCESS EVERYTHING



# THE END OF SQL

## Adapting to a changing scenery

```
SELECT SUMMARIZE(feedback_col)  as summary
FROM sales
WHERE year_col > 2022;

SELECT filepath
FROM images
WHERE image_col DEPICTS “cat”;

SELECT *
FROM video_files
WHERE PROMPT(video_col, “door is being opened”);
```

# THANK YOU

CLOUDERA