CIDR
Conference on Innovative Data Systems Research
January 18-21
2026 Chaminade, USA

BridgeScope: A Universal Toolkit for Bridging Large Language Models and Databases

Lianggui Weng, Dandan Liu, Rong Zhu, Bolin Ding, Jingren Zhou

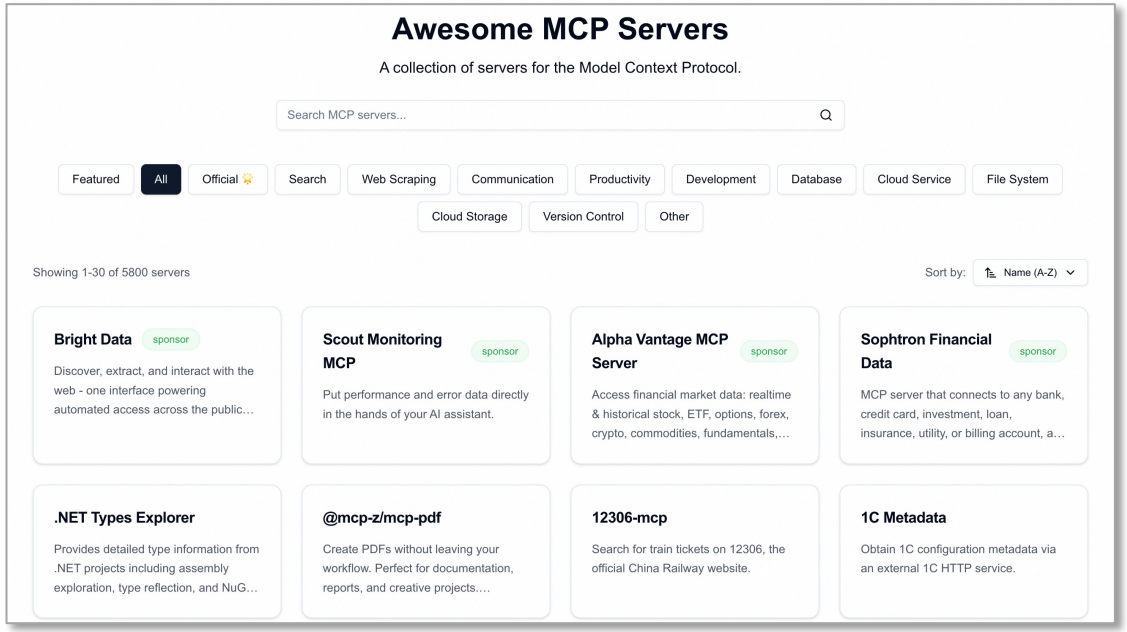
Alibaba Group

TONGYI

Background

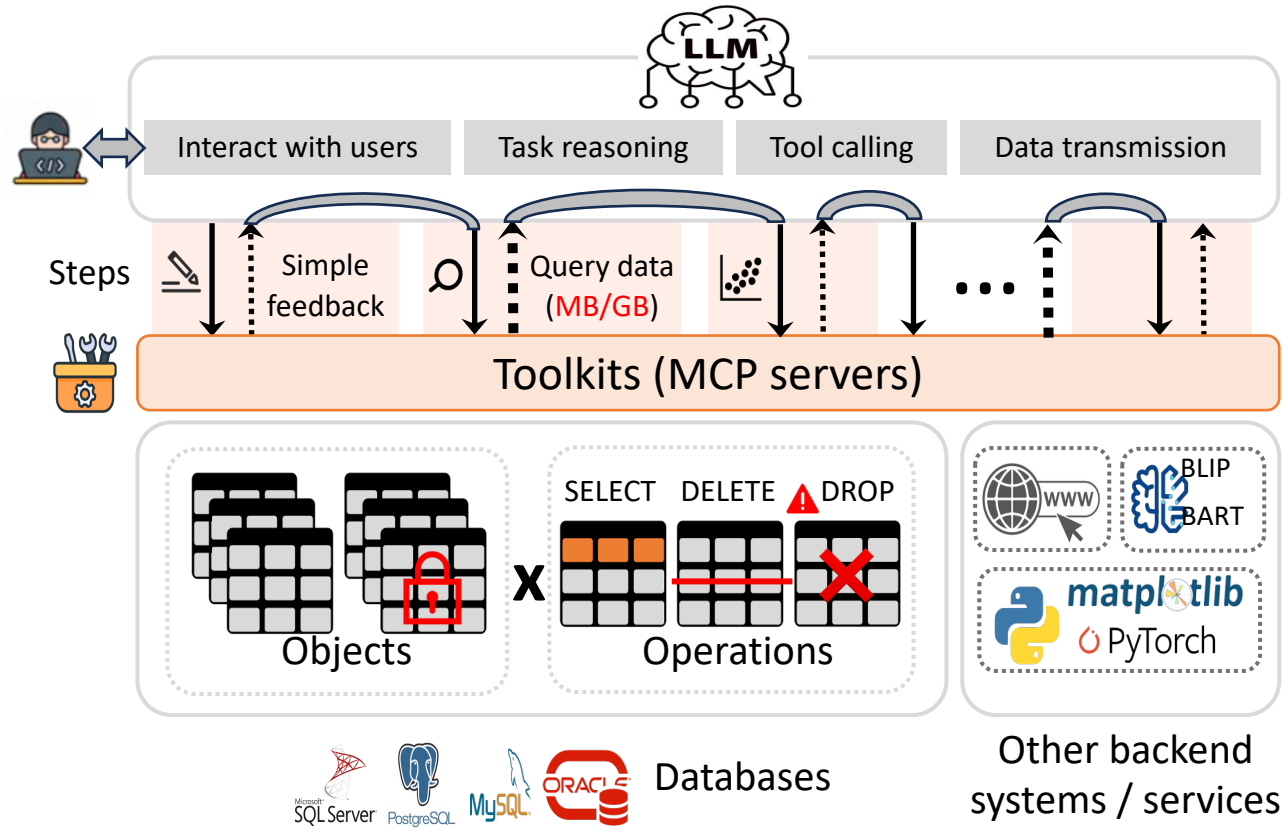
- The advancing of **LLMs' reasoning and orchestration capabilities** and **MCP ecosystems** is unlocking truly general-purpose agents.

+



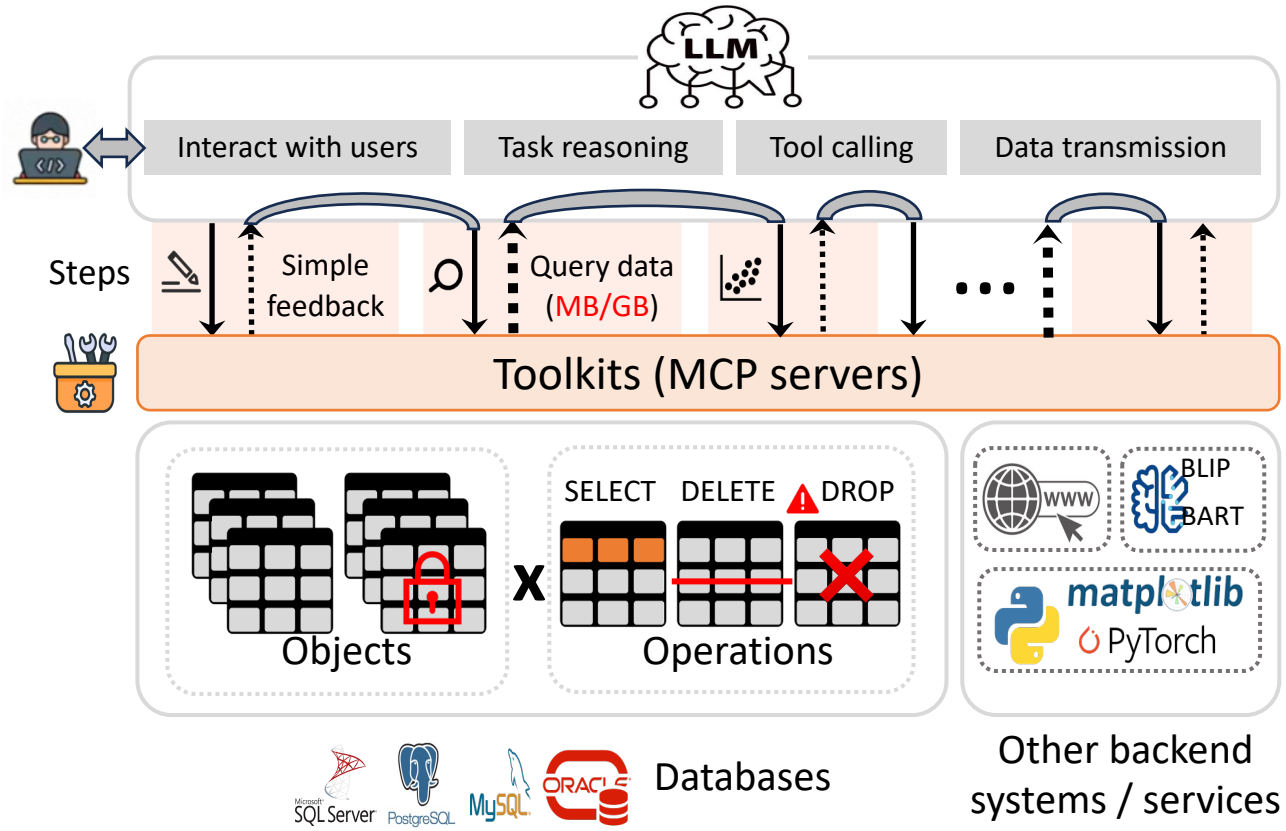
Background

- Agents handling **general-purpose**, **data-intensive** tasks can be highly problematic.

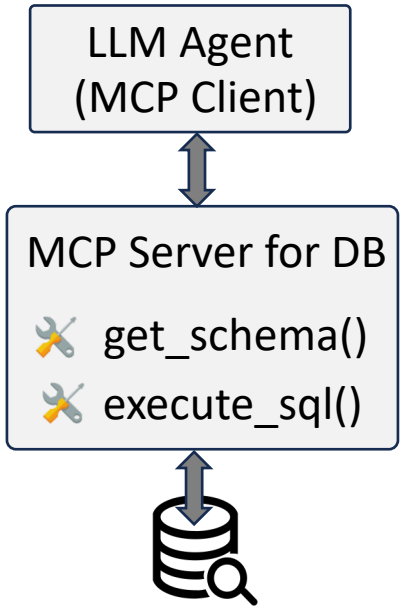


Background

- Agents handling **general-purpose**, **data-intensive** tasks can be highly problematic.



➢ **Coarse-Grained Tooling**



> execute_sql("**DROP DB**")

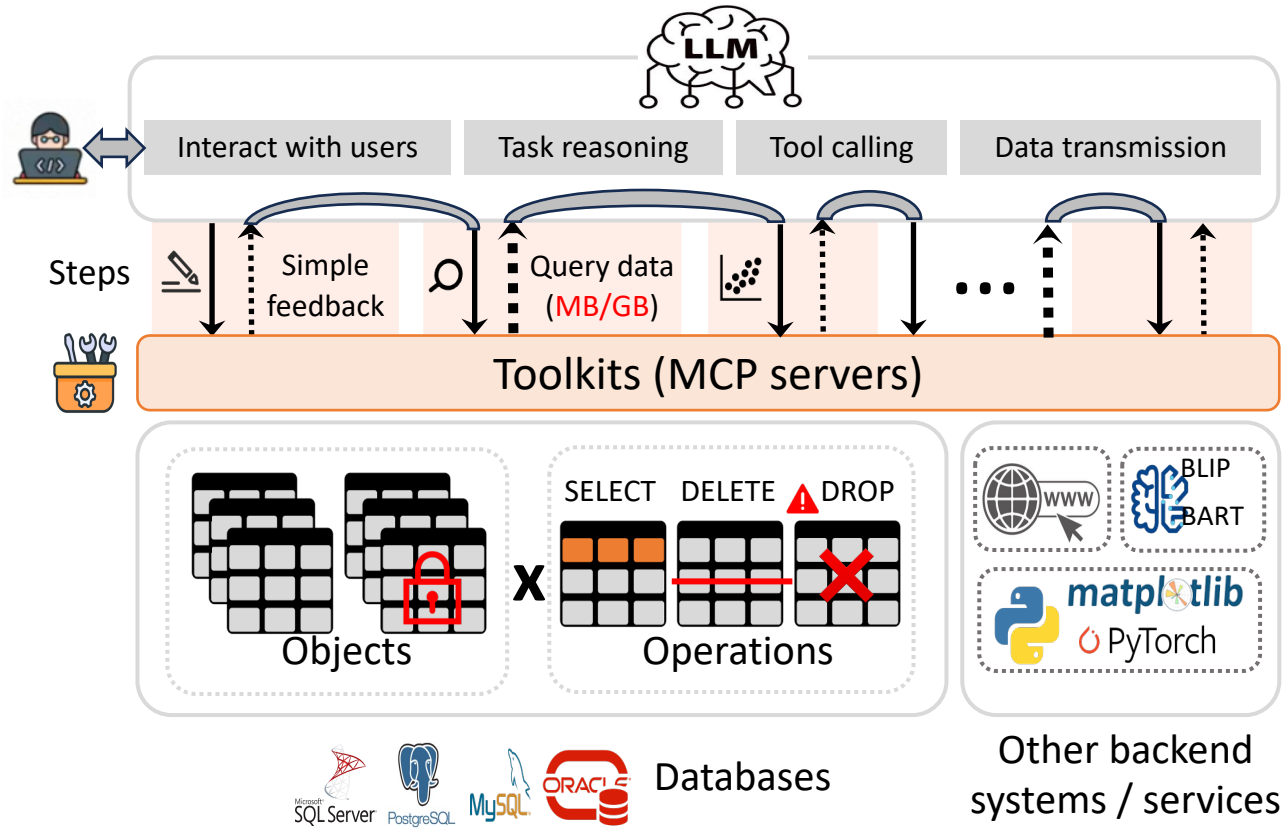
☹ Security Risk

> execute_sql("**SELECT**")
> execute_sql("**BEGIN**")
> execute_sql("**UPDATE**")
> execute_sql("**...**")

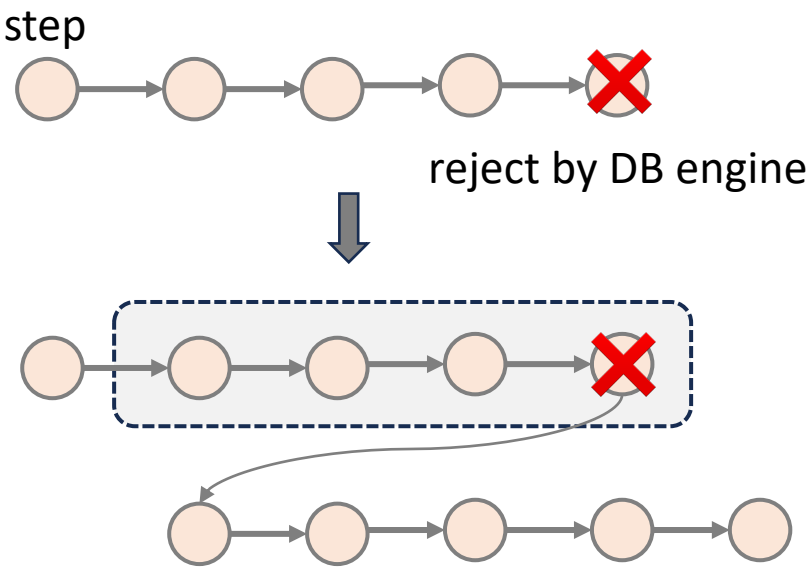
☹ Tool Selection Error

Background

- Agents handling **general-purpose**, **data-intensive** tasks can be highly problematic.



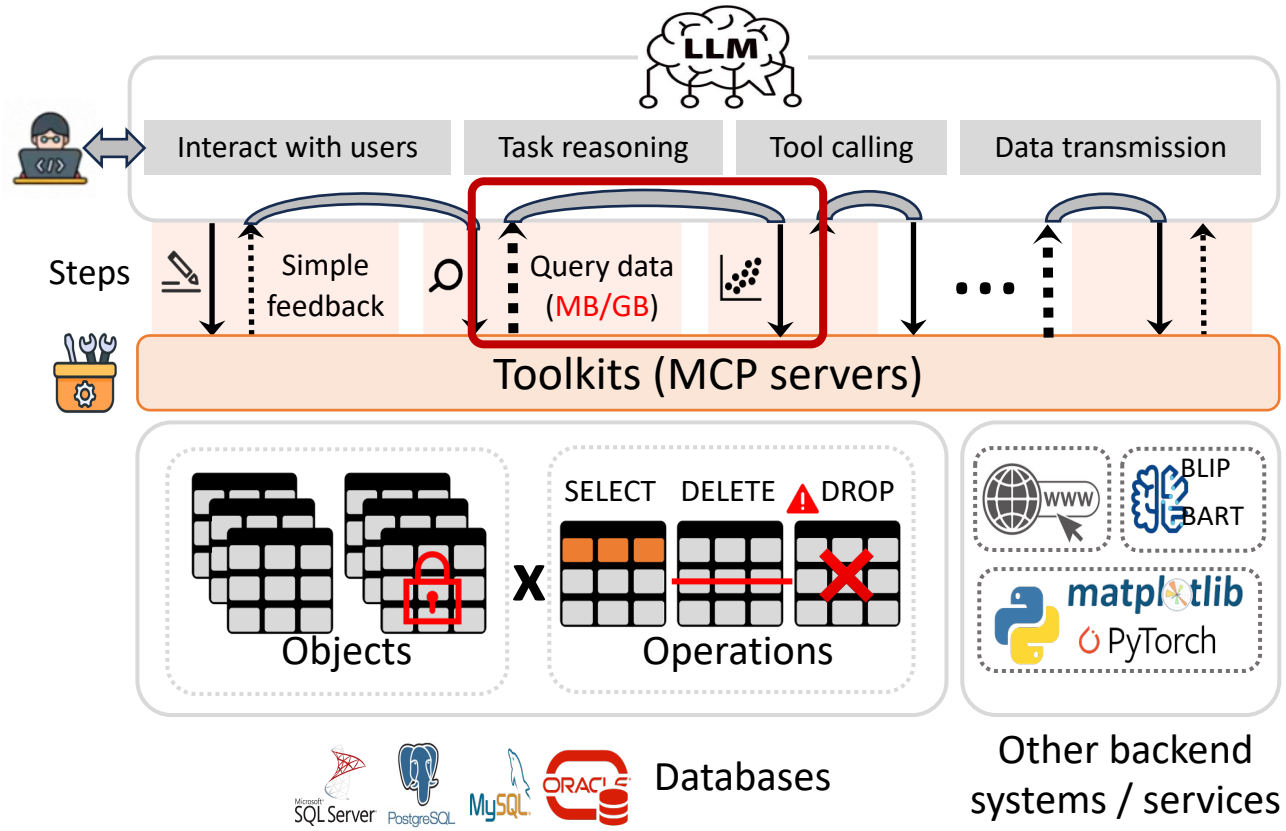
- **Coarse-Grained Tooling**
- **Privilege-Unaware Planning**



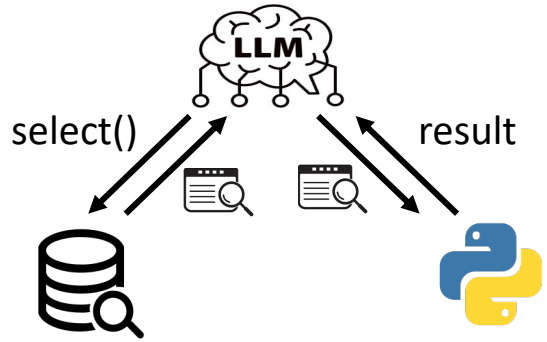
☹ Wasted planning/tool calling resource

Background

- Agents handling **general-purpose**, **data-intensive** tasks can be highly problematic.



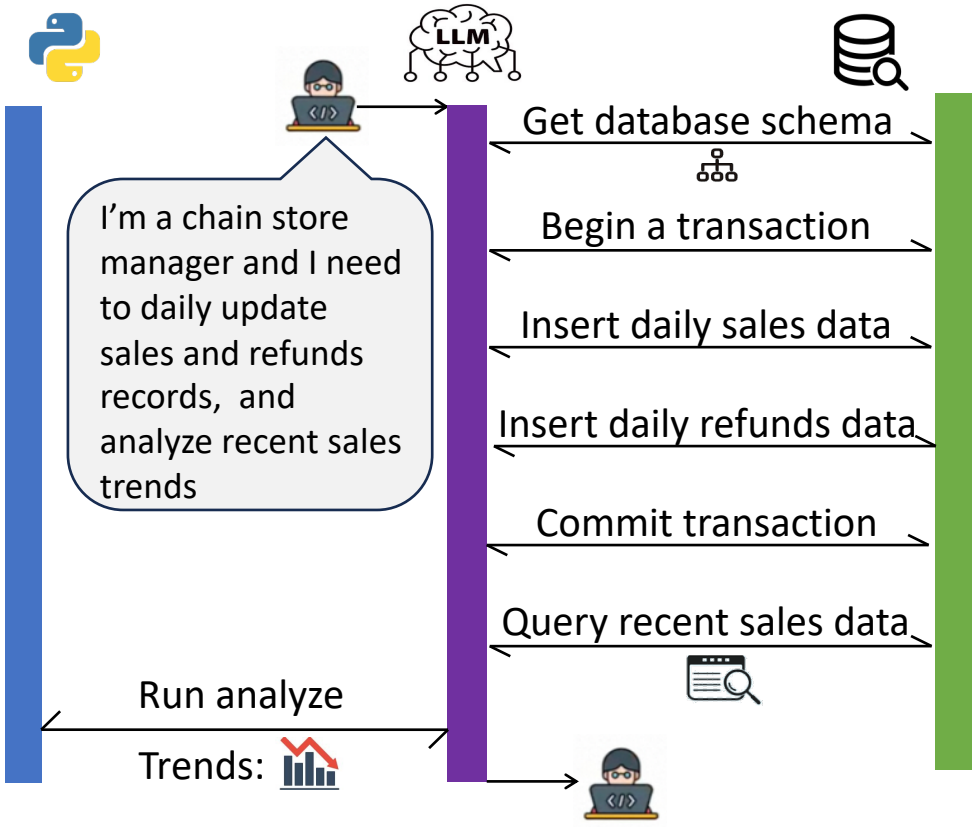
- **Coarse-Grained Tooling**
- **Privilege-Unaware Planning**
- **Data Transmission via LLM**



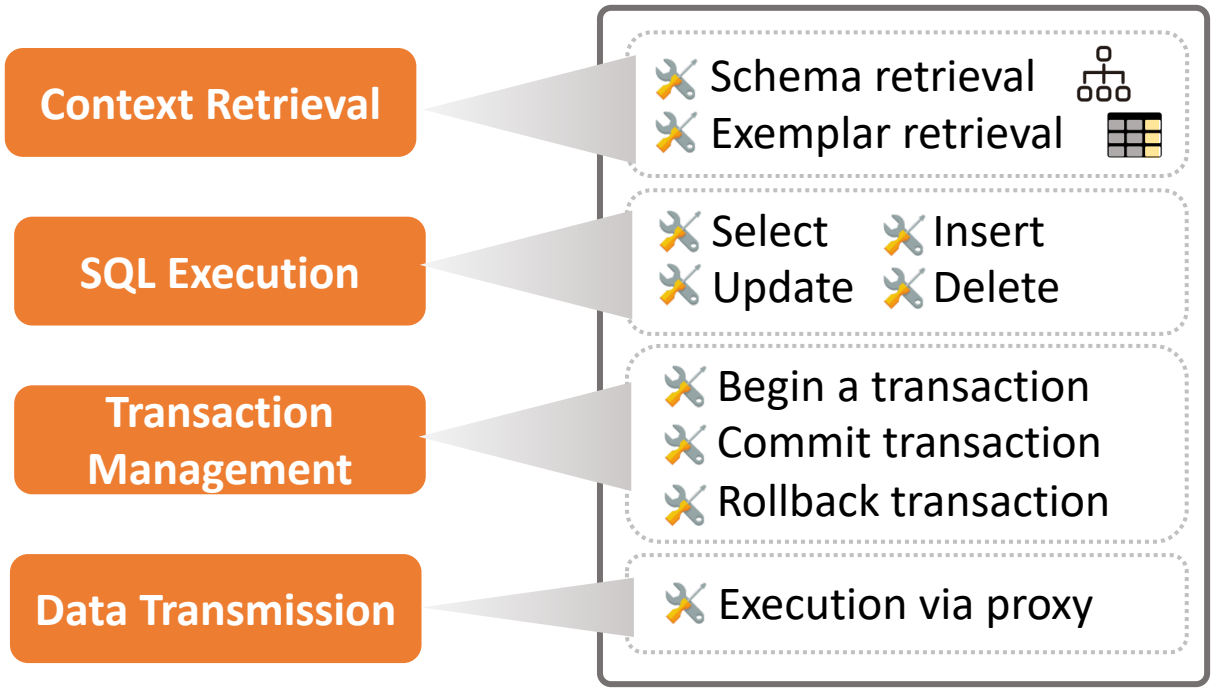
☹ Hallucinations in data transmission

☹ Task failure upon LLM context exhausted

BridgeScope Overview



A chain store use case



Functionalities

BridgeScope

Context Retrieval

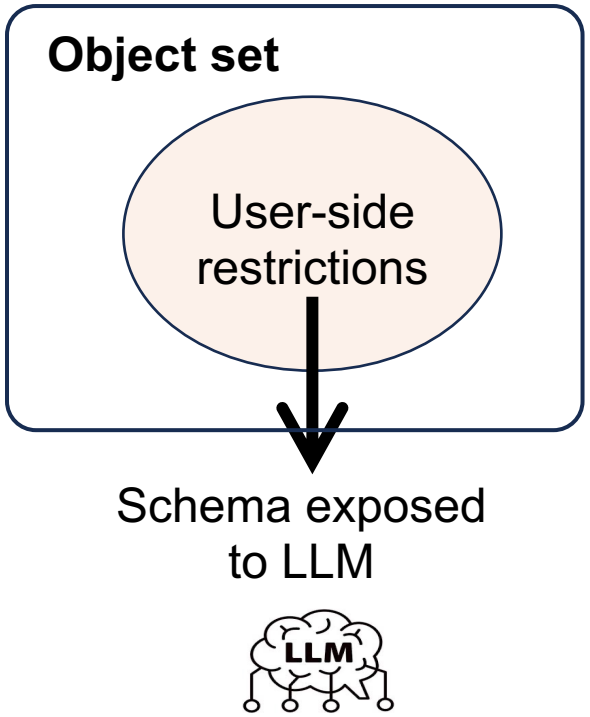
- DB-side privileges and user-side restrictions are made explicit to the LLM via annotations

```
-- Access:True, Permissions:SELECT
CREATE TABLE warehouse (
product_id TEXT
product_type TEXT
manager TEXT
last_audit_date DATE
description TEXT
... ...
```

```
-- Access:True, Permissions:ALL
CREATE TABLE fruit_order (
order_id TEXT PRIMARY KEY
item_id TEXT
purchase_count INT
... ...
FOREIGN KEY (item_id) REFERENCES
↪  fruit_inventory(id));
```

```
-- Access:True, Permissions:ALL
CREATE TABLE brand_A_sales (
id TEXT PRIMARY KEY
date DATE
item_id INT
... ...
FOREIGN KEY (item_id) REFERENCES
↪  brand_A_items(id));
```

```
-- Access:False
CREATE TABLE brand_B_sales (
id TEXT PRIMARY KEY
date DATE
item_id INT
... ...
FOREIGN KEY (item_id) REFERENCES
↪  brand_B_items(id));
```



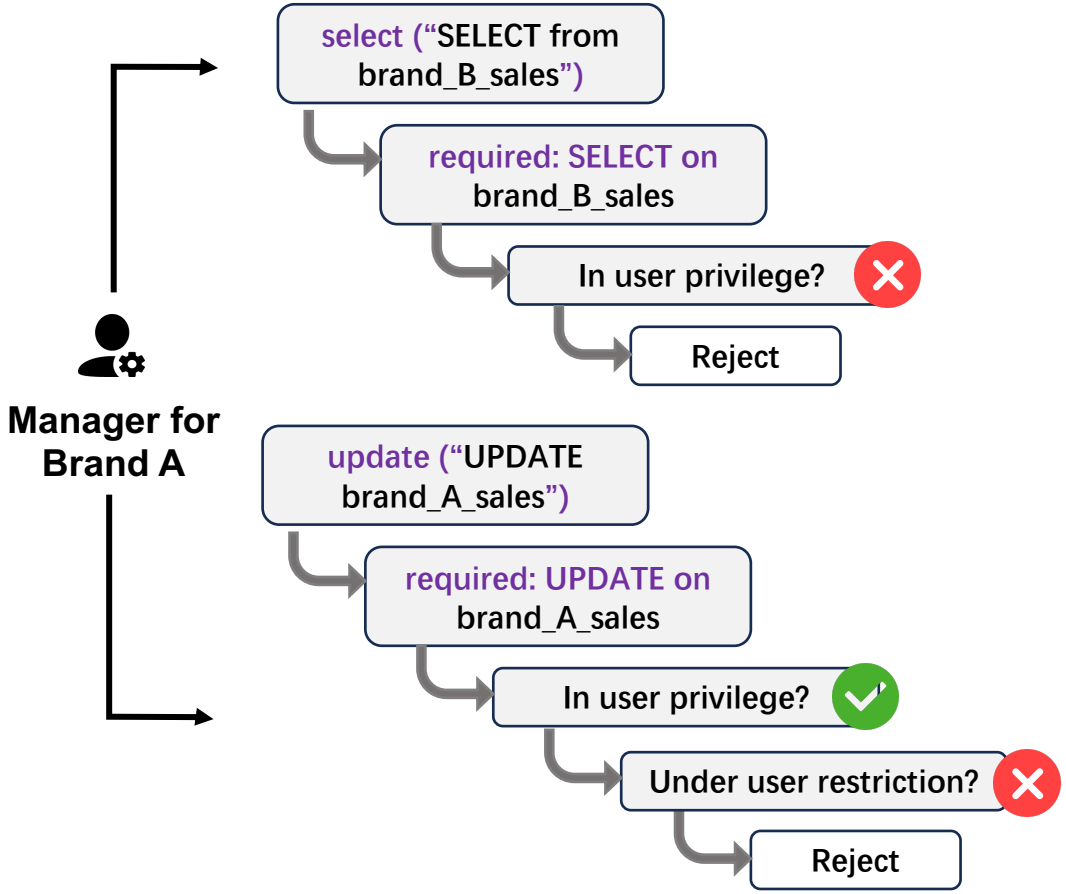
SQL Execution

- Action-level tool modularization

```
def execute(sql): # execute an arbitrary SQL
```

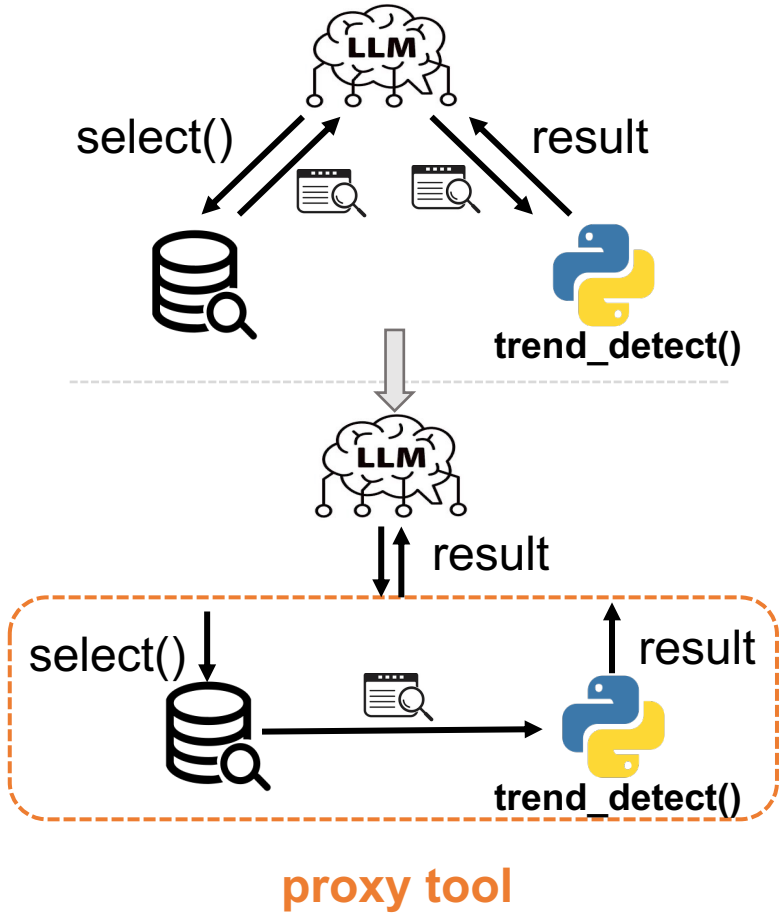
```
def select(sql): # execute a SELECT SQL
def update(sql): # execute an UPDATE SQL
def insert(sql): # execute an INSERT SQL
def delete(sql): # execute a DELECT SQL
...
```

- Object-level verification



Execution via Proxy

- Organize data flows instead of LLMs




```
lambda data:
    [[row[0] for row in data],
     [row[1] for row in data]]],
     [row[2] for row in data],
     [row[3] for row in data]]
```

transform

producer(s) → consumer

date	product	Prices/ weight	weight	supplier
06-01	apple	3.14	1	A
06-01	pear	2.79	2	A
06-02	grape	7.02	1	A

```
def trend_detect (
        date,
        product,
        price_per_weight,
        weight)
```

Experimental Settings

- ReAct agents built on
  - GPT-4
  - Claude-4

- Benchmark
  - BIRD-Ext: Extended from BIRD, including SELECT, INSERT, DELETE, UPDATE NL2SQL tasks, 50 each.

write SQL = write semantic + write condition

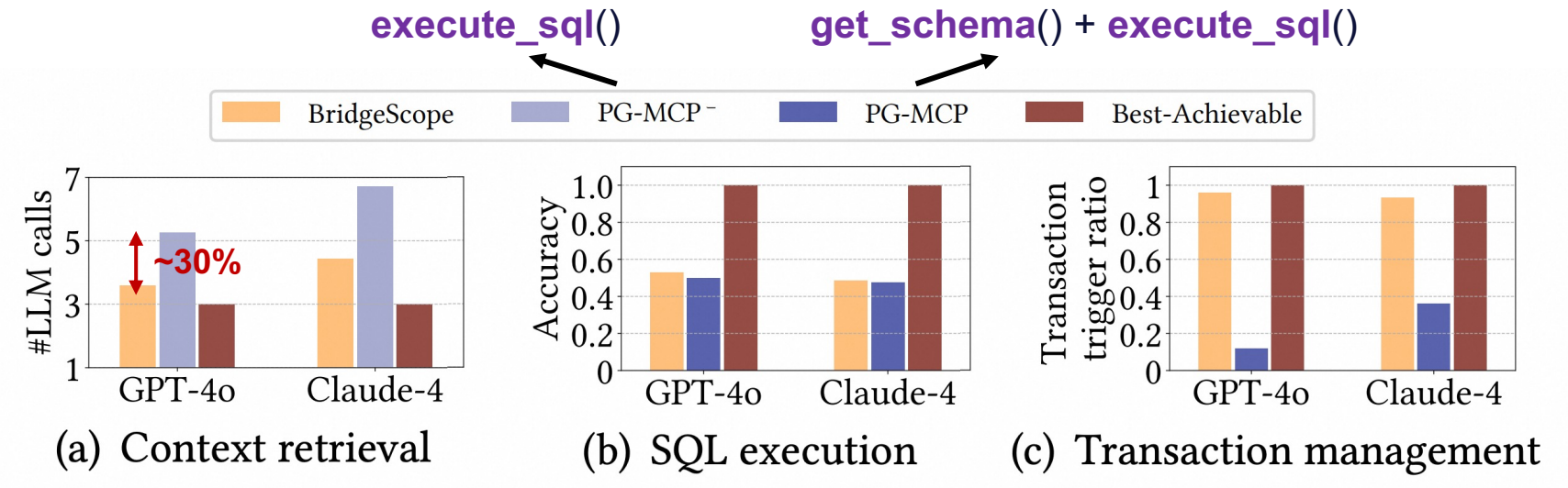
Rule-based Generation **Origin BIRD SELECT task**

  - NL2ML: ML tasks on a table of 20w rows of 3 complexity levels

L1	Data Query	Model Training		
L2	Data Query	Data Processing	Model Training	
L3	Data Query	Data Processing	Model Training	Predicting

Experimental Evaluation

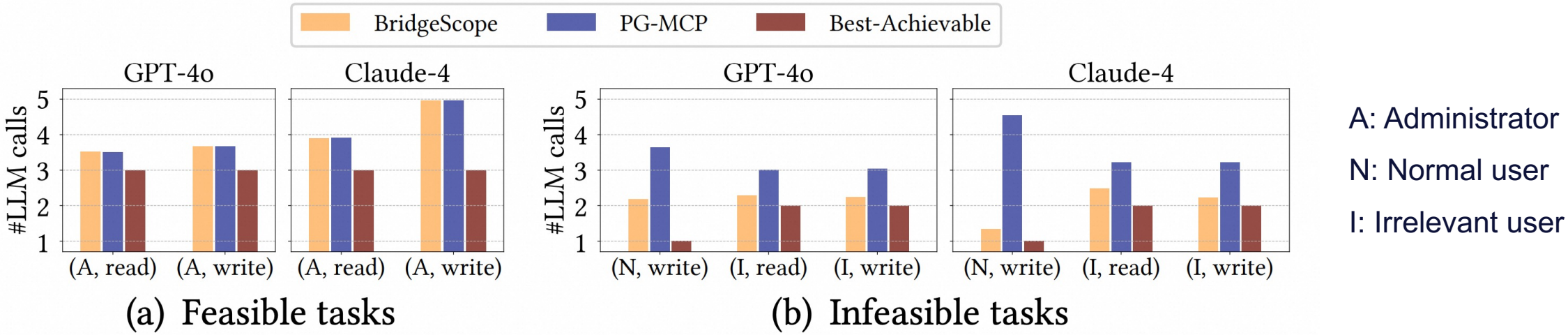
- Coarse-Grained vs. Fine-Grained Tooling



1. Explicit context retrieval tools **reduce schema hallucinations** and reduce LLM calls by 30%
2. Fine-grained tools **maintain accuracy** while enabling enhanced security controls
3. Explicit transaction tools **enable correct transaction initiation**, which PG-MCP fails to achieve

Experimental Evaluation

- Effectiveness of Privilege-Aware Tooling



A: Administrator

N: Normal user

I: Irrelevant user

1. Privilege-aware design adds no overhead when users have sufficient privileges
2. BridgeScope reduces LLM calls by **23–71%** and token costs by **30–82%** for infeasible tasks
3. Restricting visible tools based on privileges helps LLMs quickly recognize task infeasibility

Experimental Evaluation

- Effectiveness of Proxy

Metric	Agent	BridgeScope	PG-MCP	PG-MCP-S
Task Completion Rate	GPT-4o	1.0	0.0	1.0
	Claude-4	1.0	0.0	1.0
Token Usage (On Average)	GPT-4o	13,449.7	-	21,047.6
	Claude-4	15,622.3	-	22,353.1
#LLM Calls (On Average)	GPT-4o	3.37	-	5.07
	Claude-4	3.40	-	5.07

1. Modern LLMs are capable of abstracting and using the proxy tool to organize data flows
2. Direct data routing **enables 100% task completion** vs. 0% for context-based routing
3. BridgeScope **maintains high efficiency in terms of both token usage and LLM calls** on full datasets while context-based approaches fail or degrade even on small data scales

Conclusions

- Existing database toolkits offer coarse-grained tools that are insecure, privilege-unaware, and fail on data-intensive tasks.
- BridgeScope enables **secure, efficient, and scalable database operations** by three key innovations: fine-grained tool modularization, privilege-aware tool implementation, and a proxy mechanism for efficient data transmission.
- Being **database-agnostic** and **working transparently**, BridgeScope integrates seamlessly with any agentic framework and MCP server to support diverse data-intensive tasks.

Thanks for Listening!

BridgeScope Repo

https://github.com/duoyw/bridgescope/

Contact Us

lianggui.wlg@alibaba-inc.com