
Trovares Graph 2.0

Democratizing HPC Graph Analytics

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

September 5, 2024

David Haglin – david@trovares.com

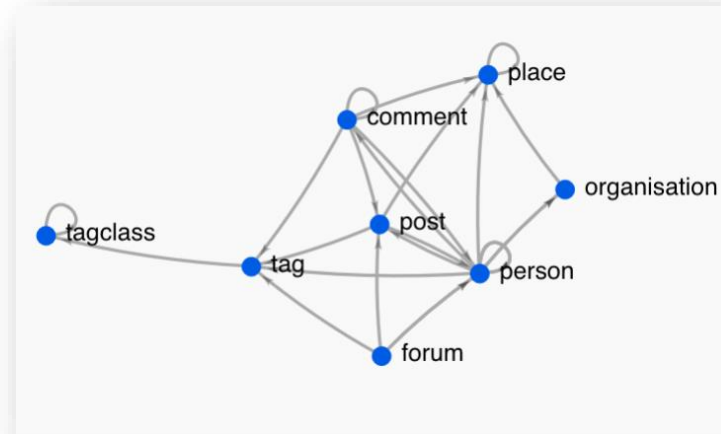
The Challenge of Graph Data Analysis

→ Complexity of graph model

- Graph schema is required
- Translating from a set of tables to a graph schema requires graph expertise

→ Complexity of graph queries

- Cypher/GQL queries can be elaborate
- Cypher/GQL is not SQL



```
Enter your query
MATCH (source:ibmaml__Account)-[t1:ibmaml__Transaction]->(intermediate1:ibmaml__Account),
      (source)-[t2:ibmaml__Transaction]->(intermediate2:ibmaml__Account),
      (intermediate1)-[t3:ibmaml__Transaction]->(target:ibmaml__Account),
      (intermediate2)-[t4:ibmaml__Transaction]->(target)
WHERE t1.paid_currency = t2.paid_currency
      AND t1.paid_currency <> t3.received_currency
      AND t1.paid_currency <> t4.received_currency
      AND t1.timestamp < t3.timestamp
      AND t2.timestamp < t4.timestamp
      AND t1.timestamp < t2.timestamp
      AND unique_vertices(source, intermediate1, intermediate2, target)
RETURN t1, t2, t3, t4
ORDER BY intermediate1.acct_id
```

The Solution: Use GenAI/LLM to Assist User

→ Introduce Language Model as Assistant for User

- Insert into UI/UX as an option
- Supports beginners and advanced users

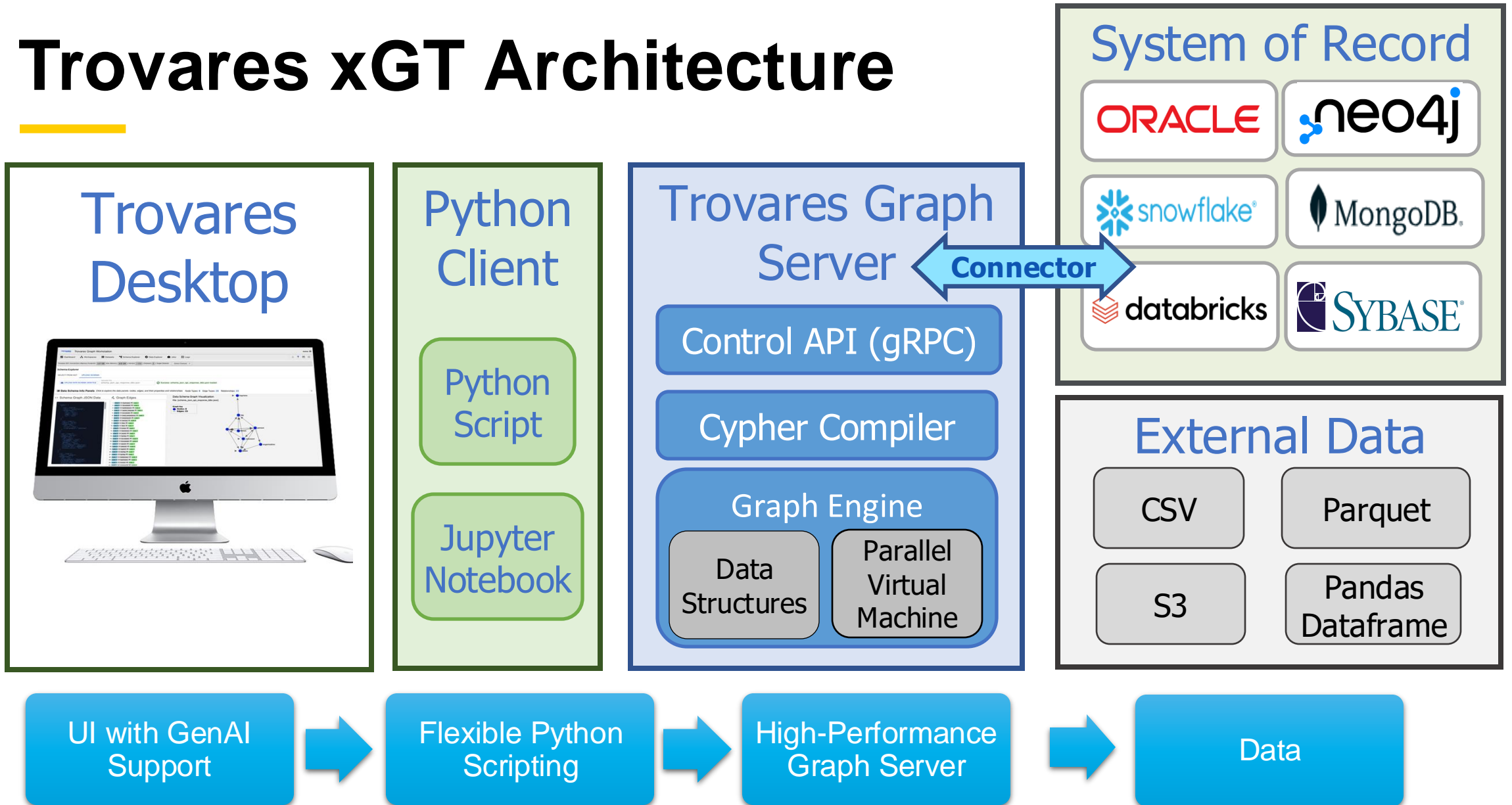
→ Provide Fallback

- Supports bypassing all use of GenAI/LLM

→ Embrace flexibility of LM integration

- Open internet API call
- On-Prem, air-gapped, custom API call

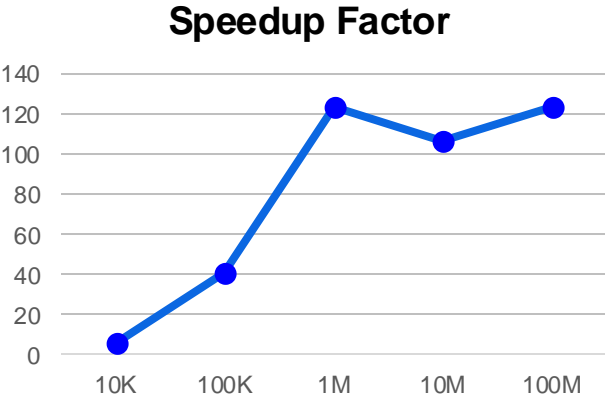
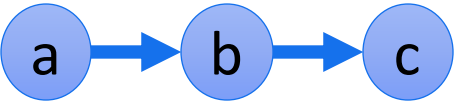
Trovares xGT Architecture



Comparison of Trovares versus Neo4j

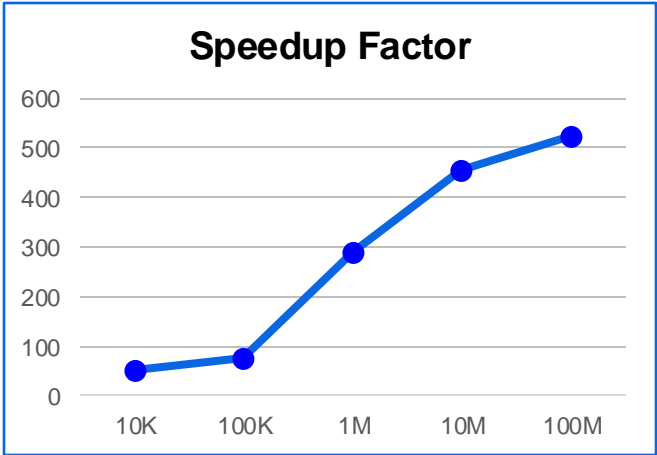
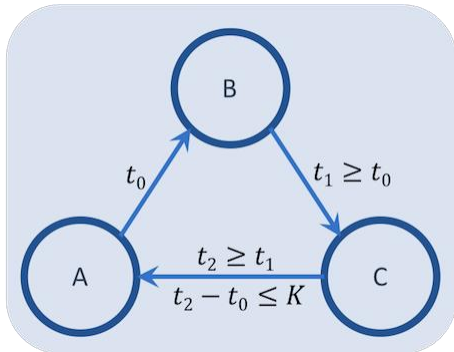
Two-Hop Motif

Product	10K	100K	1M	10M	100M
Neo4j	0.1	1.2	15	225	3,513
xGT	0.02	0.03	0.1	2	28
Speedup Factor	6x	41x	123x	106x	123x



Temporal Triangles Motif

Product	10K	100K	1M	10M	100M
Neo4j	0.5	2.3	29	456	7,344
xGT	0.01	0.03	0.1	1	14
Speedup Factor	54x	76x	363x	430x	516x



Use Case Setup: Financial Transaction Data

→ Data Source:

- IBM AML Anti-Money Laundering Data
- <https://ibm.ent.box.com/v/AML-Anti-Money-Laundering-Data/folder/132403326995>

→ Graph Representation:

- Nodes: Bank accounts
- Edges: Fund Transfer Transactions between accounts

Node: Account

Properties:

1. **acct_id**: text
2. **bank_number**: text
3. **account_number**: text

Edge: Transaction

Source: Account

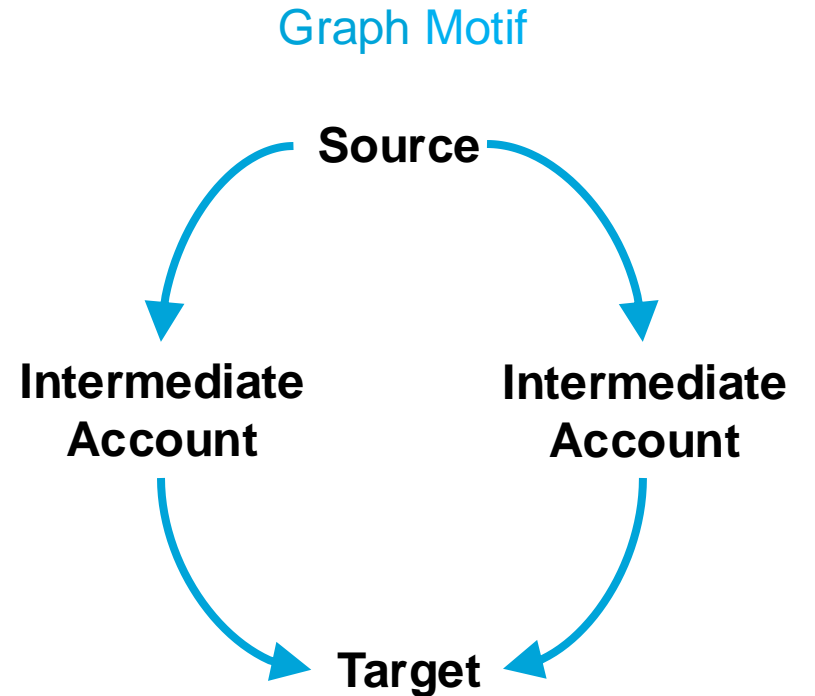
Target: Account

Properties:

1. **from_account_id**: text
2. **to_account_id**: text
3. **timestamp**: datetime
4. **amount_paid**: float
5. **amount_received**: float
6. **paid_currency**: text
7. **received_currency**: text
8. **payment_type**: text

Analyst Suspicion About Money Laundering

- Transfer money from a source account to a target account.
- Route these funds through intermediate accounts.
- Each intermediate account converts and sends the funds in a different currency than the source account.



AML Exploration Workflow

- Describe pattern to look for in a Natural Language
- Ask a Language Model (LM) to generate a Cypher query for this pattern description and the Graph Schema
- Review the LM output, including explanation and insight
- Copy Cypher to query box
- Make desired changes (if any)
- Explore answer

Natural Language description

- Detect transfers from one source to two intermediates in the same currency, which then transfer to one target in different currencies than the one sent.
- Ensure transfers from the source to intermediates occur before those to the target, and all accounts are unique.
- Provide all four edges of this pattern.

Demo

Natural Language

LLM
OpenAI-gpt4o

xGT Query Interface

Query the Trovares xGT datasets by entering a question or cypher query below.

Query Info Panel: View Question and Query History

Enter your question
Detect transfers from one source to two intermediates in the same currency, which then transfer to one target in different currencies than the one sent. Ensure transfers from the source to intermediates occur before those to the target, and all accounts are unique. Provide all four edges of this pattern.

GET CYPHER

Suggested Cypher:

```
MATCH (source:aml__Account)-[t1:aml__Transaction]->(intermediate1:aml__Account), (source)-[t2:aml__Transaction]->(intermediate2:aml__Account), (intermediate1)-[t3:aml__Transaction]->(target:aml__Account), (intermediate2)-[t4:aml__Transaction]->(target) WHERE unique_vertices(source, intermediate1, intermediate2, target) AND t1.paid_currency = t2.paid_currency AND t3.received_currency <> t1.paid_currency AND t4.received_currency <> t1.paid_currency AND t1.timestamp < t3.timestamp AND t2.timestamp < t4.timestamp RETURN t1, t2, t3, t4
```

Cypher Explanation

This query matches a pattern where a source account transfers to two intermediate accounts in the same currency, and these intermediates then transfer to a target account in different currencies. It ensures all accounts are unique and the transfers from the source to intermediates occur before those to the target.

Cypher Insight

Running this query will help identify complex money transfer patterns that may indicate money laundering or other suspicious activities. By ensuring the transfers occur in different currencies and in a specific order, it highlights potential attempts to obscure the origin of funds.

Cypher Query

Explanations

RUN QUERY

xGT Connection | Host: xgt Port: 4367 Memory: 0.85 GiB / 16.00 GiB Version: 2.0.0 Datasets: 1

Select Dataset

aml

Suggested Cypher:

```
MATCH (source:aml__Account)-[t1:aml__Transaction]->(intermediate1:aml__Account), (source)-[t2:aml__Transaction]->(intermediate2:aml__Account), (intermediate1)-[t3:aml__Transaction]->(target:aml__Account), (intermediate2)-[t4:aml__Transaction]->(target) WHERE unique_vertices(source, intermediate1, intermediate2, target) AND t1.paid_currency = t2.paid_currency AND t3.received_currency <> t1.paid_currency AND t4.received_currency <> t1.paid_currency AND t1.timestamp < t3.timestamp AND t2.timestamp < t4.timestamp RETURN t1, t2, t3, t4
```

i Cypher Explanation

This query matches a pattern where a source account transfers to two intermediate accounts in the same currency, and these intermediates then transfer to a target account in different currencies. It ensures all accounts are unique and the transfers from the source to intermediates occur before those to the target.

📈 Cypher Insight

Running this query will help identify complex money transfer patterns that may indicate money laundering or other suspicious activities. By ensuring the transfers occur in different currencies and in a specific order, it highlights potential attempts to obscure the origin of funds.

Enter your query

```
MATCH (source:aml__Account)-[t1:aml__Transaction]->(intermediate1:aml__Account),
      (source)-[t2:aml__Transaction]->(intermediate2:aml__Account),
      (intermediate1)-[t3:aml__Transaction]->(target:aml__Account),
      (intermediate2)-[t4:aml__Transaction]->(target)
WHERE unique_vertices(source, intermediate1, intermediate2, target)
      AND t1.paid_currency = t2.paid_currency
      AND t3.received_currency <> t1.paid_currency
      AND t4.received_currency <> t1.paid_currency
      AND t1.timestamp < t3.timestamp
      AND t2.timestamp < t4.timestamp
RETURN t1, t2, t3, t4
```

RUN QUERY

Query Job: [21]

Job Answer Completed - 21 - Answer Complete with result.

Job Id: 21 Job Number: 21 Start Time: Fri, 09 Aug 2024 21:18:11 GMT End Time: Fri, 09 Aug 2024 21:18:11 GMT Status: completed Num Rows: 894 Columns Count: 32

- Enable Dynamic Visualization
- Enable Graph Visualization



AML Workflow Recap

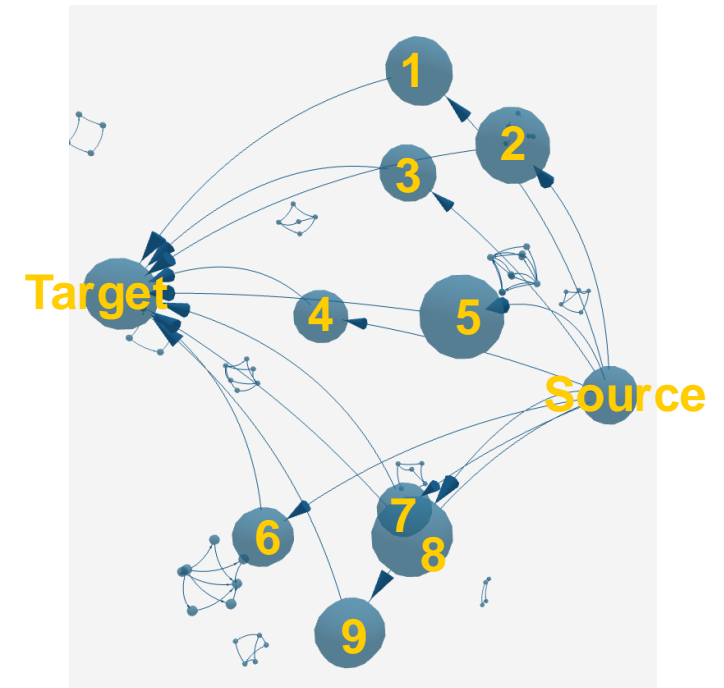
Natural Language to Graph Data Analytics

- Start from an English description of a pattern of interest
- Auto-translate description to Cypher using Gen AI
- Visualize all answers as a graph
- Human Perception drawn to clusters

Down-Select and Analyze

- Begin from way too much data to visualize
- Have some idea of behavior
- Use High-Performance Search to reduce data to subset of interest
- This process can be very iterative

Partial Image



AML Deeper Exploration

Edge: aml_Transaction

Source: 8019CFAC0I806AD2E50

Target: 80D75F170I80F5496C0

Properties:

1. **from_account_id**: 8019CFAC0I806AD2E50
2. **to_account_id**: 80D75F170I80F5496C0
3. **timestamp**: 2019-01-01 05:21:00
4. **amount_paid**: 5091.65
5. **amount_received**: 5091.65
6. **paid_currency**: Yuan
7. **received_currency**: Yuan
8. **payment_type**: ACH



Node: 80D75F170I80F5496C0

Edge: aml_Transaction

Source: 8017A5F70I8105243D0

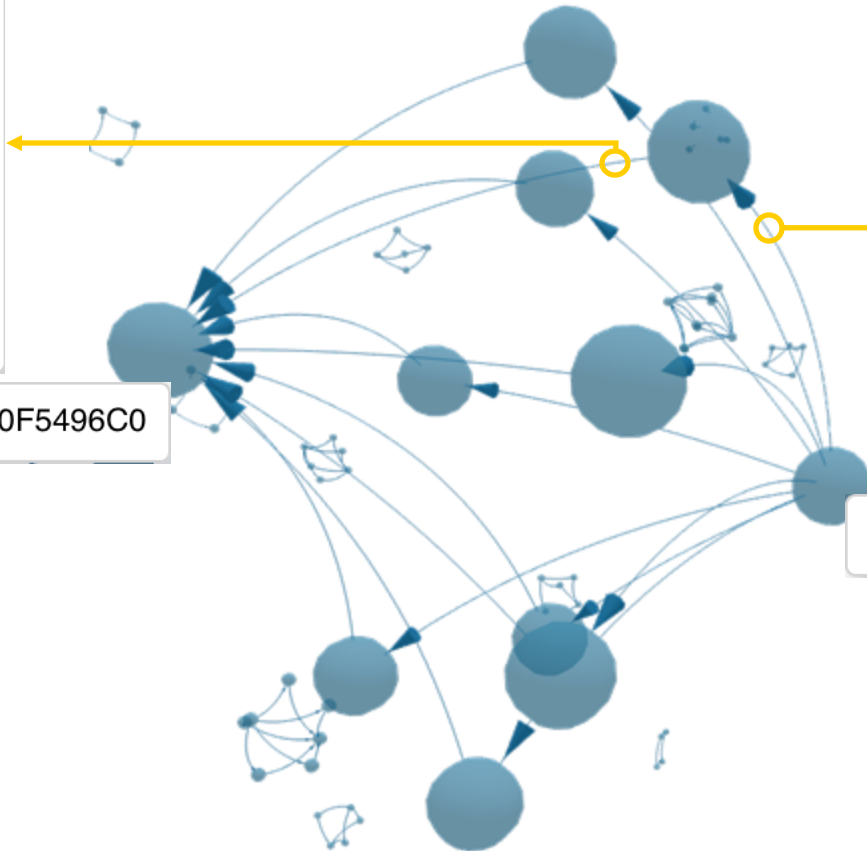
Target: 8019CFAC0I806AD2E50

Properties:

1. **from_account_id**: 8017A5F70I8105243D0
2. **to_account_id**: 8019CFAC0I806AD2E50
3. **timestamp**: 2019-01-01 03:41:00
4. **amount_paid**: 809.58
5. **amount_received**: 809.58
6. **paid_currency**: US Dollar
7. **received_currency**: US Dollar
8. **payment_type**: ACH

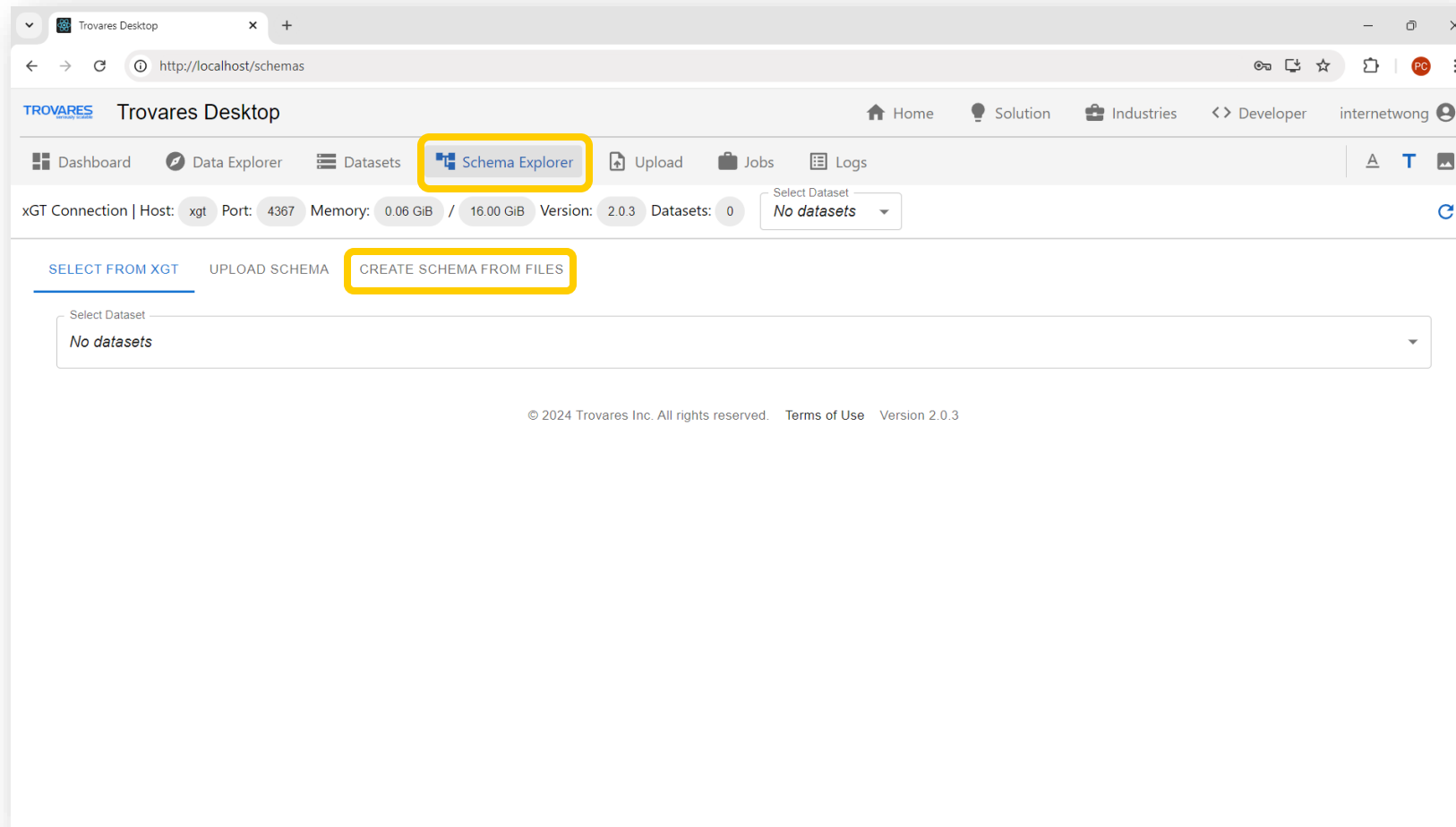


Node: 8017A5F70I8105243D0

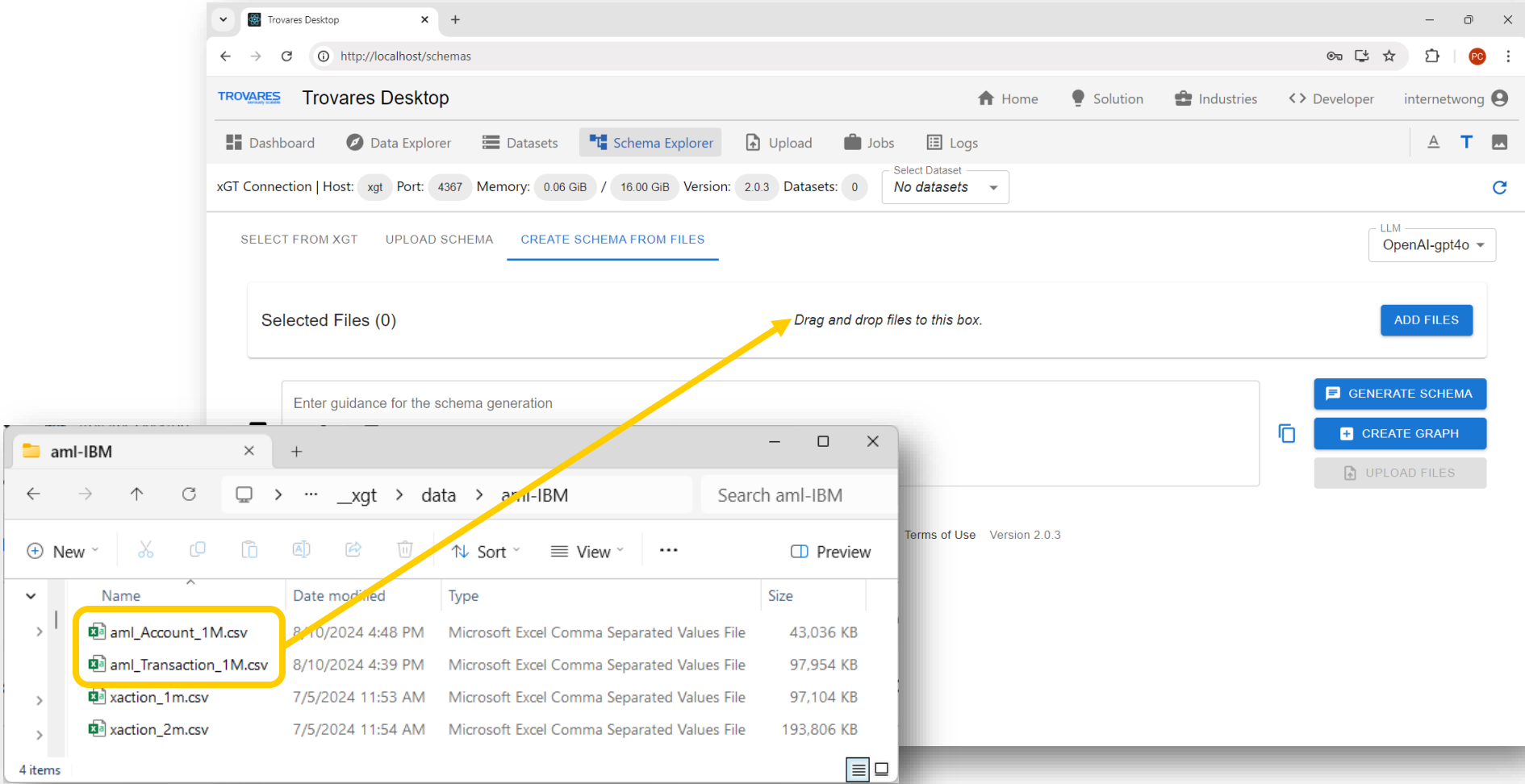


No-Code Schema Creation and Ingest

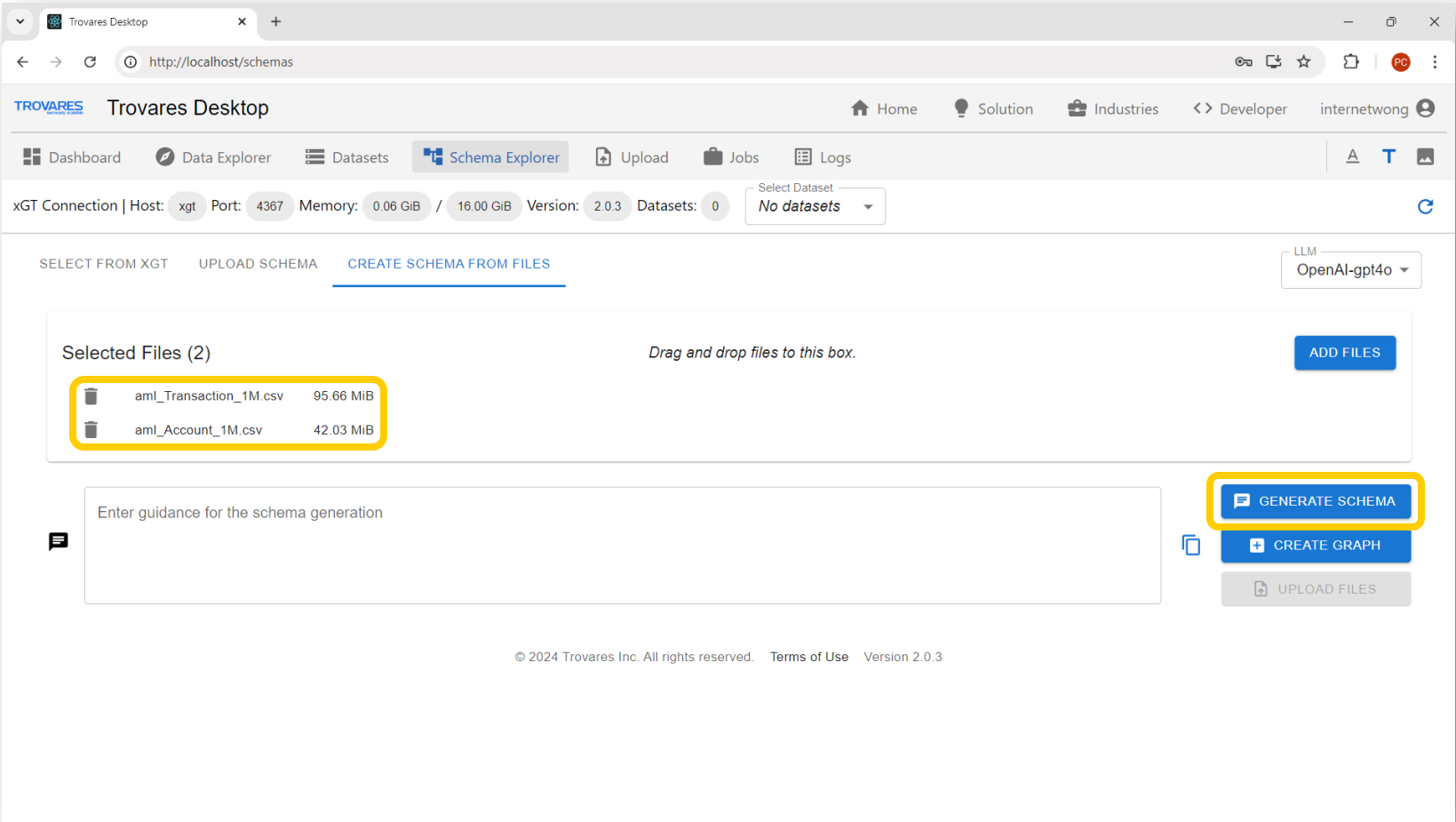
Schema Explorer: Create Schema from Files



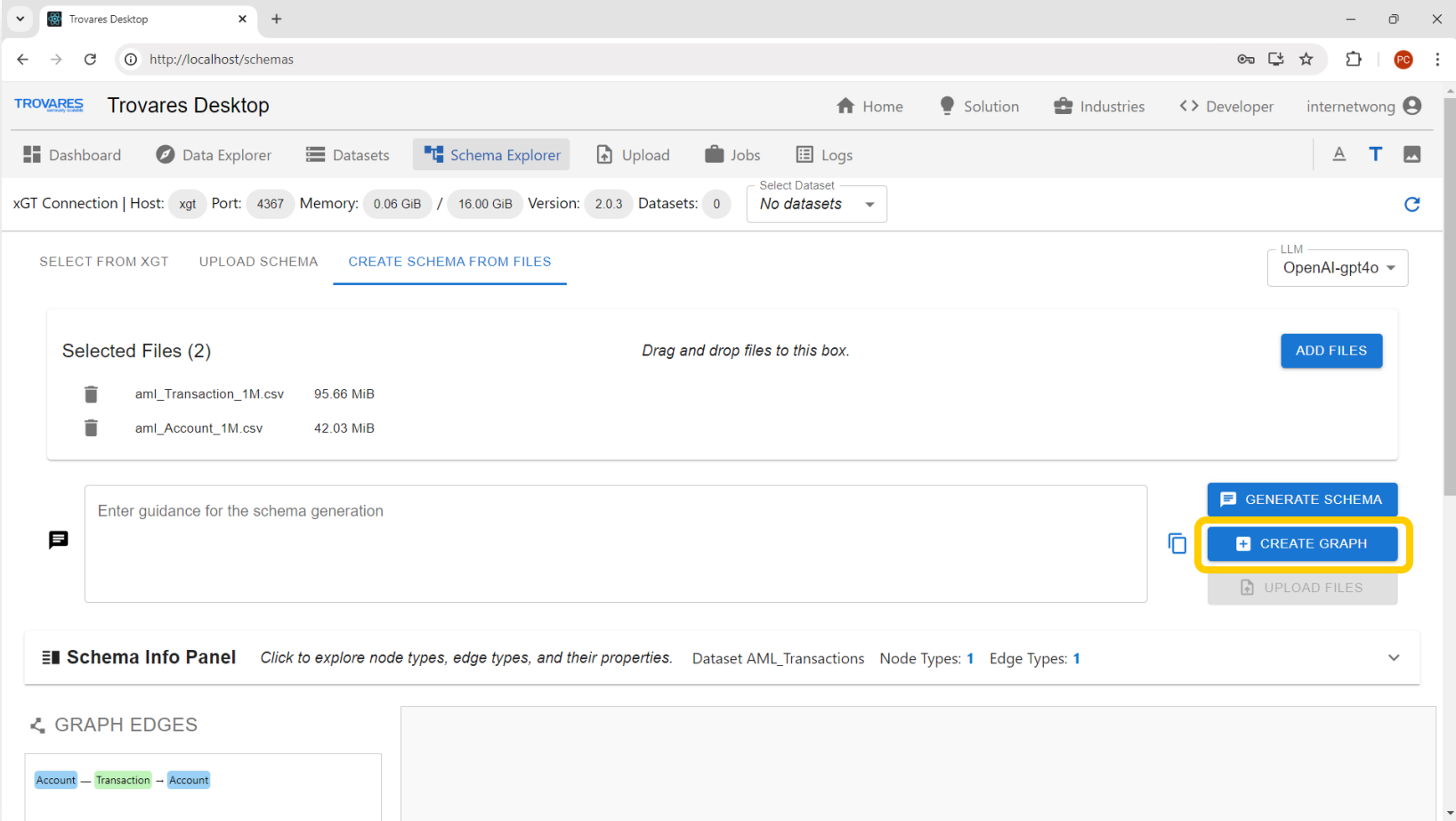
Drag and Drop Data Files



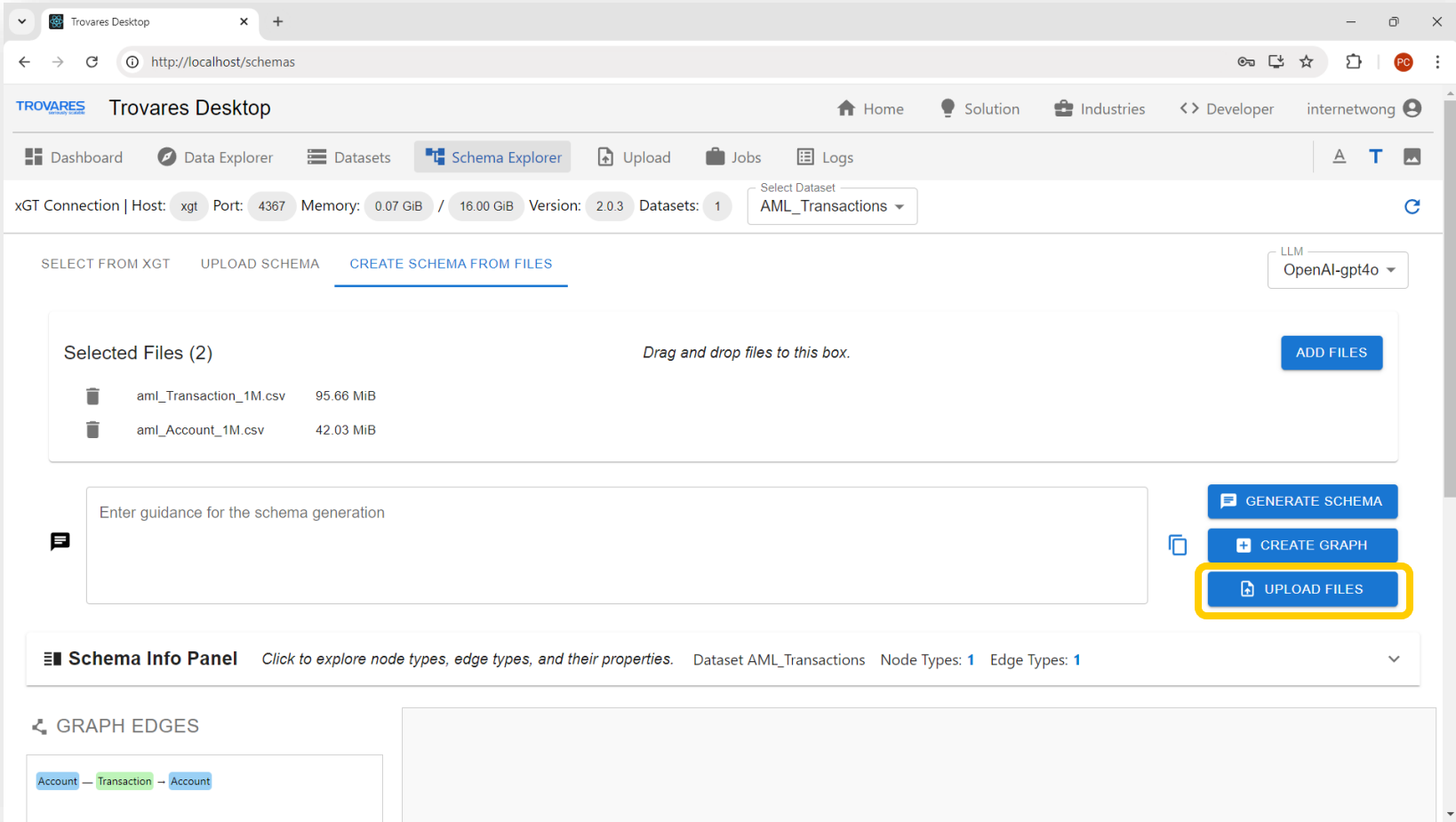
Generate Schema



Create Graph



Upload Files



Scroll Down to Header Mode

The screenshot shows the Trovares Desktop web interface. The browser address bar shows `http://localhost/upload`. The navigation menu includes Dashboard, Data Explorer, Datasets, Schema Explorer, Upload, Jobs, and Logs. The status bar shows xGT Connection | Host: xgt | Port: 4367 | Memory: 0.07 GiB / 16.00 GiB | Version: 2.0.3 | Datasets: 1 | Select Dataset: AML_Transactions.

The main content area is titled "Upload Files" and contains the following sections:

- Upload Files**: Choose which files to upload. (Select properties below to adjust frame location and other settings.)
- Files (2 total, 137.68 MiB) *Drag and drop files to this box.* [REMOVE] [ADD FILES] [ADD FOLDER]
- | <input type="checkbox"/> | Name | Graph | Frame | Size | Suppress Errors | Delimiter | Header Mode | Duplicate Keys | Block Size |
|--------------------------|------------------------|------------------|-------------|-----------|-----------------|-----------|-------------|----------------|------------|
| <input type="checkbox"/> | aml_Account_1M.csv | AML_Transactions | Account | 42.03 MiB | Yes | , | Normal | Skip | 100.00 MiB |
| <input type="checkbox"/> | aml_Transaction_1M.csv | AML_Transactions | Transaction | 95.66 MiB | Yes | , | Normal | Skip | 100.00 MiB |
- File Upload Properties**: Adjust the location and method of file uploading. (Select files above to change multiple at once.)

The "File Upload Properties" section includes the following settings:

- Destination Frame: Transaction
- Block Size: 100, Units: MiB
- Header Mode: None
- Delimiter: ,
- Suppress Errors: Suppress Errors
- Duplicate Keys: Error



Determine if the First Data Row Is a Header

The screenshot shows the Trovares Desktop web interface for uploading files. At the top, there's a navigation bar with 'Home', 'Solution', 'Industries', 'Developer', and a user profile 'internetwong'. Below that, a secondary navigation bar includes 'Dashboard', 'Data Explorer', 'Datasets', 'Schema Explorer', 'Upload', 'Jobs', and 'Logs'. The main content area is titled 'Trovares Desktop' and shows 'xGT Connection | Host: xgt | Port: 4367 | Memory: 0.07 GiB / 16.00 GiB | Version: 2.0.3 | Datasets: 1'. A 'Select Dataset' dropdown is set to 'AML_Transactions'. The main instruction is 'Choose which files to upload. (Select properties below to adjust frame location and other settings.)'. There are two buttons: 'REMOVE' and 'ADD FILES'. Below this is a table of files to be uploaded:

<input type="checkbox"/>	Name	Graph	Frame	Size	Suppress Errors	Delimiter	Header Mode	Duplicate Keys	Block Size
<input type="checkbox"/>	aml_Account_1M.csv	AML_Transactions	Account	42.03 MiB	Yes	,	Normal	Skip	100.00 MiB
<input type="checkbox"/>	aml_Transaction_1M.csv	AML_Transactions	Transaction	95.66 MiB	Yes	,	Normal	Skip	100.00 MiB

Below the table is the 'File Upload Properties' section. It includes several settings:

- Destination Frame: Transaction
- Block Size: 100 MiB
- Header Mode: None (highlighted with a yellow box)
- Suppress Errors:
- Delimiter: ,
- Duplicate Keys: Error

Additional text explains: 'Frame destination where to store the file.', 'Size of each upload request.', 'How the first line of the CSV should be handled.', 'Character used to separate values in a CSV file.', and 'Handling of duplicate keys in Vertex frame uploads.'

Select Normal Mode and Scroll Down

The screenshot shows the Trovares Desktop web interface. The browser address bar shows `http://localhost/upload`. The page title is "Trovares Desktop". The navigation menu includes "Dashboard", "Data Explorer", "Datasets", "Schema Explorer", "Upload", "Jobs", and "Logs". The "Upload" tab is active. The page displays the following information:

- XGT Connection | Host: xgt | Port: 4367 | Memory: 0.07 GiB / 16.00 GiB | Version: 2.0.3 | Datasets: 1
- Select Dataset: AML_Transactions
- Choose which files to upload. (Select properties below to adjust frame location and other settings.)
- Files (2 total, 137.68 MiB) *Drag and drop files to this box.* [REMOVE] [ADD FILES] [ADD FOLDER]
- Table of files to upload:

<input type="checkbox"/>	Name	Graph	Frame	Size	Suppress Errors	Delimiter	Header Mode	Duplicate Keys	Block Size
<input type="checkbox"/>	aml_Account_1M.csv	AML_Transactions	Account	42.03 MiB	Yes	,	Normal	Skip	100.00 MiB
<input type="checkbox"/>	aml_Transaction_1M.csv	AML_Transactions	Transaction	95.66 MiB	Yes	,	Normal	Skip	100.00 MiB

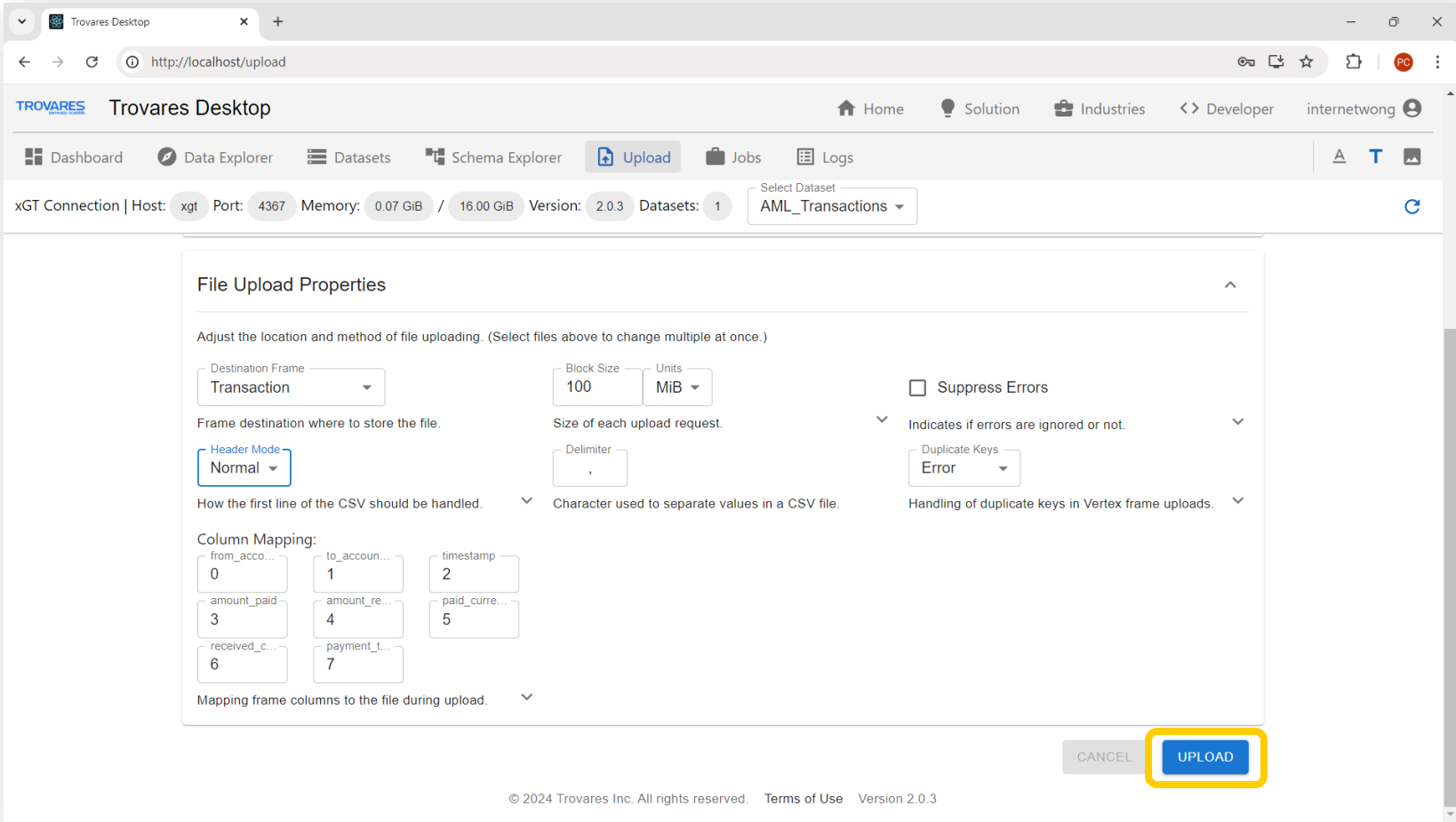
File Upload Properties

Adjust the location and method of file uploading. (Select files above to change multiple at once.)

- Destination Frame: Transaction
- Block Size: 100 Units: MiB
- Frame destination where to store the file. **Normal** (highlighted)
- Size of each upload request.
- Delimiter: ,
- Character used to separate values in a CSV file.
- Suppress Errors: Indicates if errors are ignored or not.
- Duplicate Keys: Error
- Handling of duplicate keys in Vertex frame uploads.



Upload Data



File Upload Successful

Upload Status: Finished

Loading: aml_Transaction_1M.csv into Transaction | Completed: 95.66 MiB/95.66 MiB

Total Progress: 03.2s | Speed: 42.57 MiB/s | ETA: 00.0s

Uploaded Files (2 total, 137.68 MiB, 2,101,709 rows)

Name	Status	Graph	Frame	Size	Duration	Speed	Rows Added
aml_Transaction_1M.csv	Completed	AML_Transactions	Transaction	95.66 MiB	02.0s	42.57 MiB/s	1,000,000
aml_Account_1M.csv	Completed	AML_Transactions	Account	42.03 MiB	01.2s	34.73 MiB/s	1,101,709

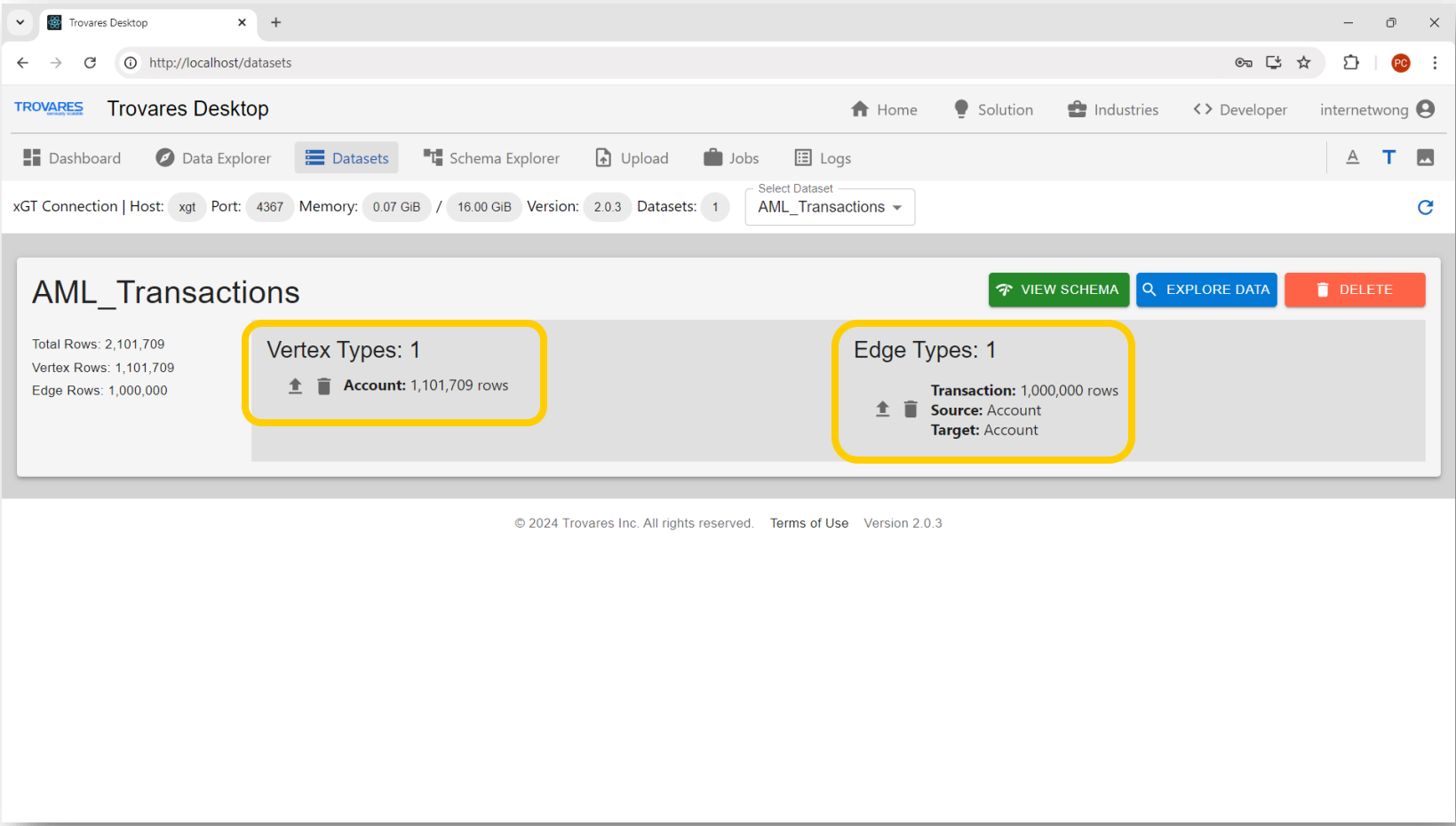
Upload Files

Choose which files to upload. (Select properties below to adjust frame location and other settings.)

Files (0) *Drag and drop files to this box.* REMOVE ADD FILES ADD FOLDER

<input type="checkbox"/>	Name	Graph	Frame	Size	Suppress Errors	Delimiter	Header Mode	Duplicate Keys	Block Size
No files									

Data Is Now in Trovares Graph Server



Value Prop

- A “no-code” concept is possible:
 - No need to use Python or Cypher/GQL
 - Drives a high-performance server that works on large data
 - 2.0 Includes both Trovares Graph Server and new Desktop
 - Developer versions are free on docker hub
 - AWS Marketplace product, *Trovares Basic*, is available for exploration
 - Launch an EC2 instance
 - Aim a browser to that instance
 - Plans for a *Trovares Enterprise* offering are underway --- imagine running on an EC2 instance with 32TB and 896-threads.
- Trovares 2.0 rollout deal:
 - \$10K for one year license on a single server (up to 256 cores)
 - Valid until December 31, 2024

TROVARES

seriously scalable

Example Code for On-Prem LLM Inference

→ Create file:

- `site_local_config.py`

→ Insert this into the
desktop_backend docker
image

→ docker-compose restart

```
def call_onprem_llm(llm_config:dict, llm_credentials:dict={}, question:str=None,
                    prompt:str=None, history:list=[], **kwargs) -> str:
    import requests
    model = llm_config.get('model', "")
    headers = {"User-Agent": "LLM Client"}
    collection_name = "trovares"
    message = f"<s> {prompt}\n[INST]{question}[/INST]</s>"
    payload = {"query": message, "collection_name": collection_name, "n_results": 5,
              "temperature": 0.05, "max_tokens": 5000,
              }
    response = requests.post('http://onpremllm.internal.org:8502/post_query/',
                             headers=headers, json=payload)
    response_json = response.json()
    return response_json['text'].strip()

# Add these site-local LLM configurations
LLM_CONFIG_local_add = {
    'OnPremLLM': {
        'fields': [ ],
        'model': 'Trovares',
        'callback': 'call_onprem_llm',
    },
}
```