

# Building a Holistic SQL Chatbot that Solves Real Problems for People in Tech and the Business



Data Council '24

Noy Twerski

Co-founder & CEO @ [sherloq](#)



## **Our Product Evolution in 3 Stages**




## Our Product Evolution in 3 Stages

**The mistakes we've made, and what we learned from them**

---

What is your #1 problem in your  
daily tasks?



85% of companies we talked to said  
this was their #1 problem



How many **active users**  
on our platform?

**17,000** active users

**17,000** active users

VS



**17,000** active users

VS

**27,000** active users

*“I always get **one same answer** for each data question.”*



Said No Data Exec,  
**Never.**

In Data Analytics: the Devil is in the Details 🐈

How many **active users** on our platform?

## In Data Analytics: the Devil is in the Details 🐱

How many **active users** on our platform?

```
WITH user_at_least_one_time_active AS (  
  SELECT  
    u.id AS user_id,  
    U.email,  
    u.account_name,  
    oi.created_at AS account_signup_date  
  
  FROM users_updated_v2 u  
  
  JOIN  
    onboard_information oi ON u.uuid = oi.user_id  
  
  WHERE  
    oi.created_at >= NOW() - INTERVAL '1 month'  
    AND u.id IN (SELECT DISTINCT user_id FROM queries WHERE created_at >= NOW() - INTERVAL '1  
    day')`)  
  
SELECT *  
  
FROM user_at_least_one_time_active u  
  
WHERE  
  account_signup_date IS NOT NULL  
  AND email NOT ILIKE '%gmail%'
```

## In Data Analytics: the Devil is in the Details 🐼

How many **active users** on our platform?

```
WITH user_at_least_one_time_active AS (  
  SELECT  
    u.id AS user_id,  
    U.email,  
    u.account_name,  
    oi.created_at AS account_signup_date  
  FROM users_updated_v2 u  
  JOIN  
    onboard_information oi ON u.uuid = oi.user_id  
  WHERE  
    oi.created_at >= NOW() - INTERVAL '1 month'  
    AND u.id IN (SELECT DISTINCT user_id FROM queries WHERE created_at >= NOW() - INTERVAL '1  
    day')`)  
SELECT *  
FROM user_at_least_one_time_active u  
WHERE  
  account_signup_date IS NOT NULL  
  AND email NOT ILIKE '%gmail%'
```

Relevant assets

# In Data Analytics: the Devil is in the Details 🐼

How many **active users** on our platform?

```
WITH user_at_least_one_time_active AS (  
  SELECT  
    u.id AS user_id,  
    U.email,  
    u.account_name,  
    oi.created_at AS account_signup_date  
  FROM users_updated_v2 u  
  JOIN onboard_information oi ON u.uuid = oi.user_id  
  WHERE  
    oi.created_at >= NOW() - INTERVAL '1 month'  
    AND u.id IN (SELECT DISTINCT user_id FROM queries WHERE created_at >= NOW() - INTERVAL '1  
    day')`)  
SELECT *  
FROM user_at_least_one_time_active u  
WHERE  
  account_signup_date IS NOT NULL  
  AND email NOT ILIKE '%gmail%'
```

Relevant assets

Connections

# In Data Analytics: the Devil is in the Details 🐱

How many **active users** on our platform?

```
WITH user_at_least_one_time_active AS (
```

```
  SELECT
    u.id AS user_id,
    U.email,
    u.account_name,
    oi.created_at AS account_signup_date
```

```
  FROM users_updated_v2 u
```

Relevant assets

```
  JOIN onboard_information oi ON u.uuid = oi.user_id
```

Connections

```
  WHERE
    oi.created_at >= NOW() - INTERVAL '1 month'
    AND u.id IN (SELECT DISTINCT user_id FROM queries WHERE created_at >= NOW() - INTERVAL '1 day')`)
```

```
SELECT *
```

Best practice

```
FROM user_at_least_one_time_active u
```

```
WHERE
  account_signup_date IS NOT NULL
  AND email not ilike '%gmail%'
```

# In Data Analytics: the Devil is in the Details 🐱

How many **active users** on our platform?

```
WITH user_at_least_one_time_active AS (  
  SELECT  
    u.id AS user_id,  
    U.email,  
    u.account_name,  
    oi.created_at AS account_signup_date  
  FROM users_updated_v2 u  
  JOIN onboard_information oi ON u.uuid = oi.user_id  
  WHERE  
    oi.created_at >= NOW() - INTERVAL '1 month'  
    AND u.id IN (SELECT DISTINCT user_id FROM queries WHERE created_at >= NOW() - INTERVAL '1  
    day')`)  
SELECT *  
FROM user_at_least_one_time_active u  
WHERE  
  account_signup_date IS NOT NULL  
  AND email not ilike '%gmail%'
```

Relevant assets

Connections

Best practice

Logic



# In Data Analytics: the Devil is in the Details 🐱

How many **active users** on our platform?

```
WITH user_at_least_one_time_active AS (
```

```
  SELECT
```

```
    u.id AS user_id,  
    U.email,  
    u.account_name,  
    oi.created_at AS account_signup_date
```

Relevant assets

```
  FROM users_updated_v2 u
```

```
  JOIN
```

```
    onboard_information oi ON u.uuid = oi.user_id
```

Connections

```
  WHERE
```

```
    oi.created_at >= NOW() - INTERVAL '1 month'  
    AND u.id IN (SELECT DISTINCT user_id FROM queries WHERE created_at >= NOW() - INTERVAL '1 day')`)
```

```
SELECT *
```

Best practice

```
FROM user_at_least_one_time_active u
```

Logic

```
WHERE
```

```
  account_signup_date IS NOT NULL  
  AND email NOT ILIKE '%gmail%'
```

Semantic business term

# Why do LLMs specifically help in this type of problem



Semantic Context

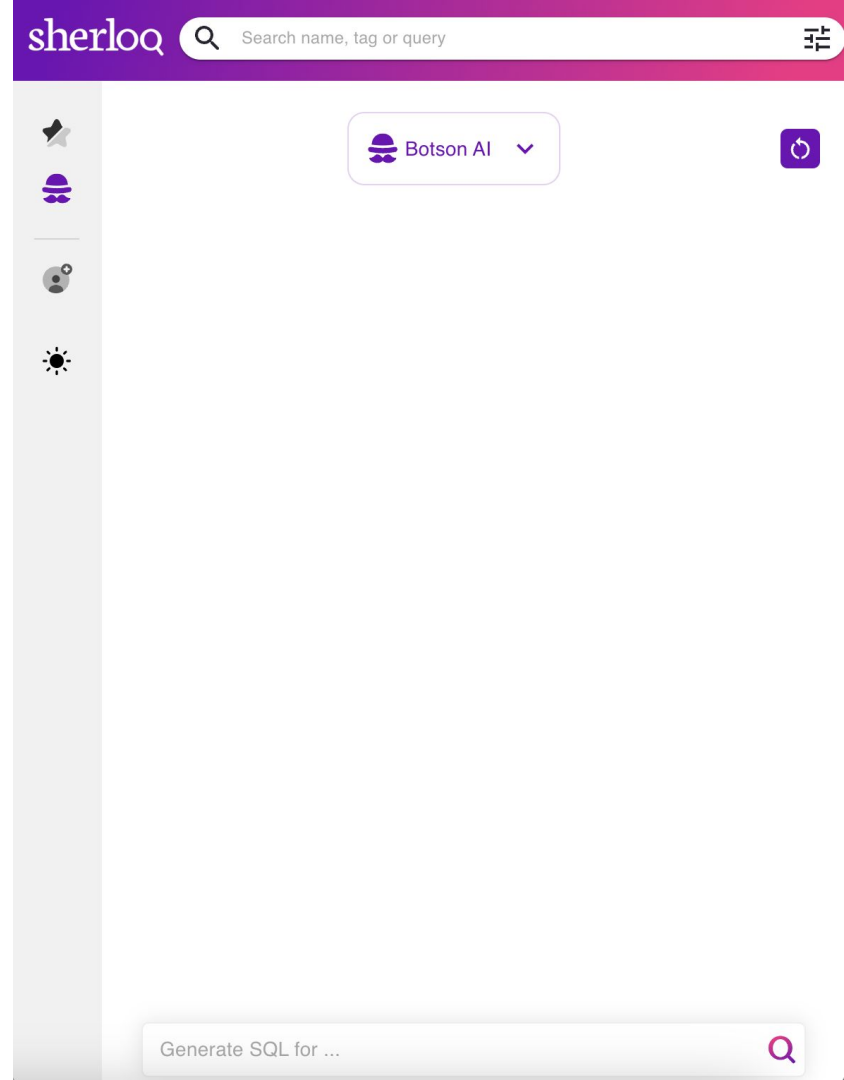


Existing Flow



Text + SQL = <3

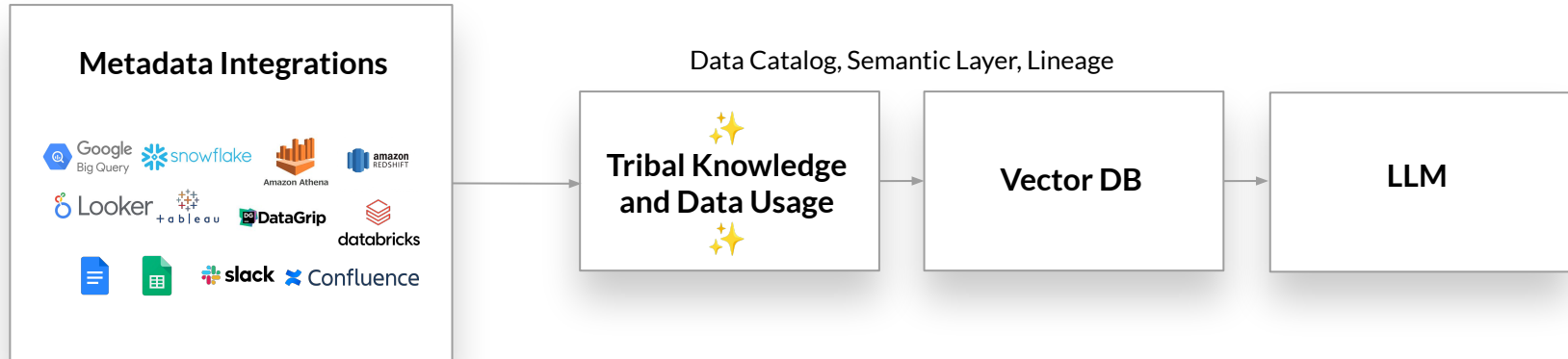
Like ChatGPT and Co-pilot  
customized to any organization's  
internal database



The Goal 🏆

**Generating answers for data teams questions  
about their internal structured DB  
to create alignment**

# The basics



## Stage 1: **Magic**

Find the most relevant metadata connected to the user's question

## Stage 1: Magic

*“What data do I need to use to understand an account’s industry?”*

The screenshot shows the Sherloq Botson Discovery interface. At the top, there is a search bar with the text "Search name, tag or query" and a magnifying glass icon. Below the search bar, there is a navigation menu with icons for home, search, add, Botson Discovery, list, shield, user, and settings. The main content area displays a search result for "Botson Discovery". The result text reads: "You can use *accounts.origin* field. Here are some more details about this field:". Below this text, there is a list of details: "• **Description:** This field represents the specific industry associated with the account. It provides a context for understanding the business and the market in which the account or company operates. This can include industries like technology, finance, health, manufacturing amongst others." "• **Top user:** maya.t@sherloqdata.io" "• **Last used:** Dec 8, 2023 by maya.t@sherloqdata.io in this query:". Below the list, there is a code block containing a SQL query: "SELECT a.origin, a.account\_id, u.user\_id, u.user\_name FROM accounts a JOIN users u ON a.account\_id = u.account\_id". At the bottom of the interface, there is a search bar with the text "Search with Botson" and a magnifying glass icon. Below the search bar, there is a link that says "Botson is fully secure and private".

## Stage 1: Magic

“What data do I need to use to understand an account’s industry:”



Botson Discovery



You can use `accounts.origin` field. Here are some more details about this field:

- **Description:** This field represents the specific industry associated with the account. It provides a context for understanding the business and the market in which the account or company operates. This can include industries like technology, finance, health, manufacturing amongst others.
- **Top user:** maya.t@sherloqdata.io
- **Last used:** Dec 8, 2024 by maya.t@sherloqdata.io in this query:



```
SELECT
  a.origin,
  a.account_id,
  u.user_id,
  u.user_name
FROM
  accounts
JOIN users u ON a.account_id = u.account_id
```

Search with Botson





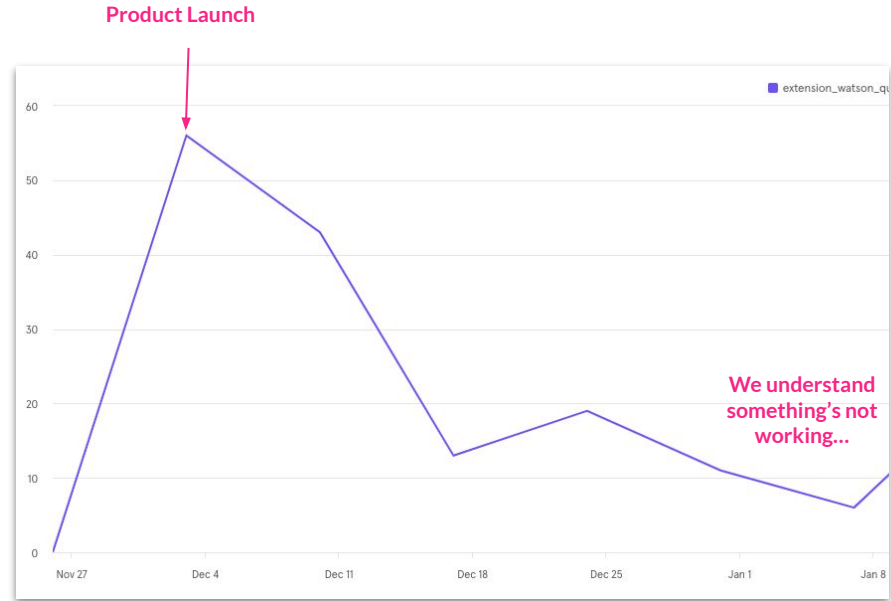
## Stage 1: Magic Naive



It seems we couldn't find a good answer to your question. Please try again with different wording or explore our suggested prompts



# Stage 1: Magic Naive



What did we learn from the **Naive** stage?

1. Answers weren't accurate

The screenshot shows the Sherloq interface. At the top, there is a search bar with the text "Search name, tag or query" and a search icon. To the right of the search bar are icons for a list, a refresh, and a close button. Below the search bar is a navigation menu with icons for home, a star, a plus sign, a Botson Discovery icon, a list, a shield with a plus sign, a person with a plus sign, and a sun. The main content area shows a chat conversation. At the top of the chat is a dropdown menu with the Botson Discovery icon and the text "Botson Discovery". To the right of this menu is a refresh icon. The chat message is: "what is the meaning of active user" with a "DS" tag. The response from Botson Discovery is: "From a data perspective, the term 'active user' can be related to a specific field in a dataset or a table's name. Could you please clarify if you are referring to a field or a table? And if possible, could you provide its specific name?". Below the response are thumbs up and thumbs down icons. At the bottom of the chat is a search bar with the text "Search with Botson" and a search icon. Below the search bar is a link: "Botson is fully secure and private".

What did we learn from the **Naive** stage?

2. Too many types of questions



What did we learn from the **Naive** stage?

3. Outputs weren't useful

The screenshot shows the 'sherloq' interface. At the top, there is a search bar with the text 'Search name, tag or query' and a magnifying glass icon. To the right of the search bar are icons for a list, a refresh button, and a minus sign. Below the search bar, there is a dropdown menu showing 'Botson Discovery' with a downward arrow and a refresh icon. On the left side, there is a vertical sidebar with icons for home, a star, a plus sign, a Botson icon, a list, a shield with a plus sign, a person with a plus sign, and a sun. The main content area shows a chat interface. A message from 'DS' asks: 'What data do I need to use to understand an account's industry?'. Below this, a response from 'DS' shows a code block with the following text: 

```
- Field 'origin' - 468 queries  
- Field 'domain' - 351 queries  
- Field 'acc_id' - 328 queries
```

 At the bottom of the chat, there is a search bar with the text 'Search with Botson' and a magnifying glass icon. Below the search bar, there is a link that says 'Botson is fully secure and private'.

## Stage 2: **Aware-of-itself**

Creating a SQL Chatbot that actually works and gives value

## Stage 2: **Aware-of-itself**

1. Answers weren't accurate

**Create a process and user validation**

The screenshot shows the 'sherloq' interface. At the top, there is a search bar with the text 'Search name, tag or query' and a search icon. Below the search bar, there is a navigation menu with icons for a home page, a bot icon, a user profile, and a settings icon. The main content area features a dropdown menu for 'Botson Discovery' and a search input field containing 'table?'. Below this, there is a form with the following fields:

- Table Name:** Full or partial table name. Input: accounts
- Field in the table:** Full or partial field name. Input: id
- Schema Name:** Full or partial schema name. Input: Optional
- Queried by users:** In the last 30 days. Input: Optional
- In dashboard:** Input: Optional

At the bottom of the form, there are two buttons: 'Edit info' and 'Submit'.

## Stage 2: *Aware-of-itself*

2. Too many types of questions

**Define a closed scope of users, use-cases and question types**

The screenshot shows the Sherloq interface. At the top, there is a purple header with the 'sherloq' logo and a search bar containing the text 'Search name, tag or query'. Below the header, on the left, is a vertical sidebar with four icons: a bell, a person, a gear, and a sun. The main content area features a user profile for 'Botson Discovery' with a refresh icon. Below this is a chat bubble containing the text 'Hey, just making sure I understand what you're looking for'. A list of question types is displayed in purple buttons: 'Field', 'Table', 'Dashboard', 'Join', 'Calculated Fields From A Dashboard', and 'Partition'. A scroll-down arrow is visible on the right side of the list.



## Stage 2: **Aware-of-itself**

3. Answers weren't useful

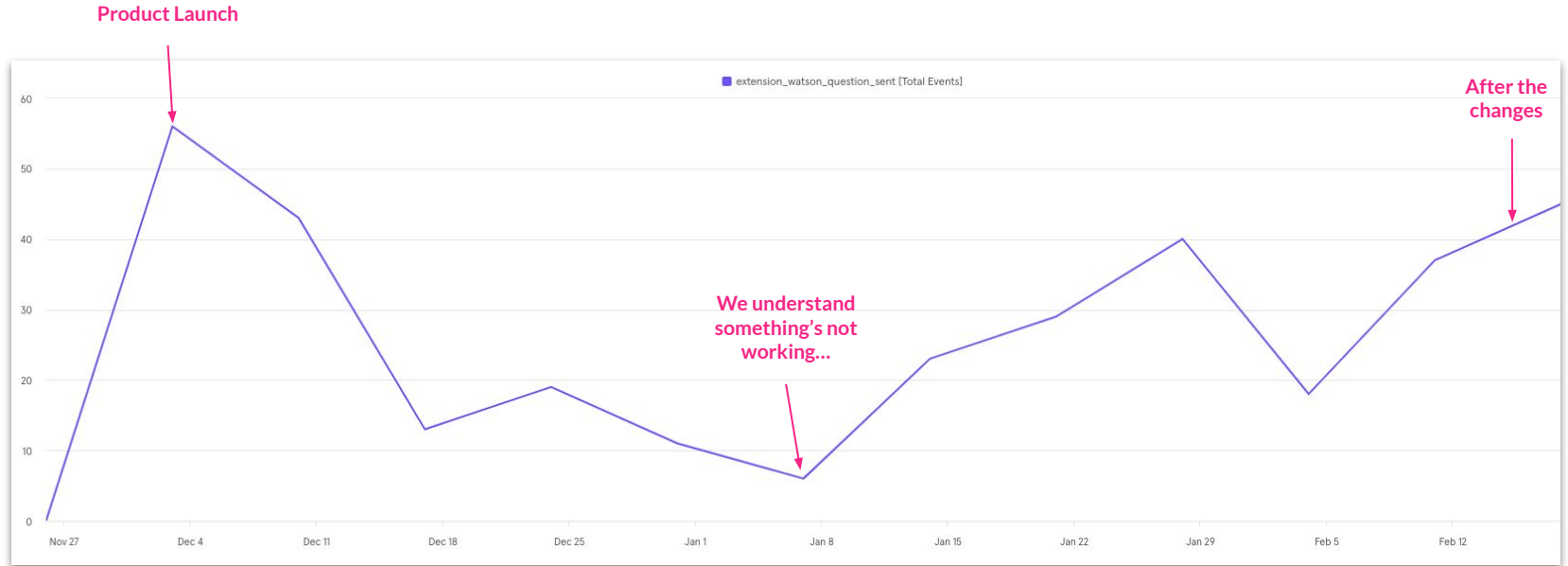
Define outputs for each type of question

The screenshot shows the 'sherloq' interface. At the top, there is a search bar with the text 'Search name, tag or query'. Below the search bar, there is a dropdown menu labeled 'Botson Discovery'. The main content area displays a table with the following data:

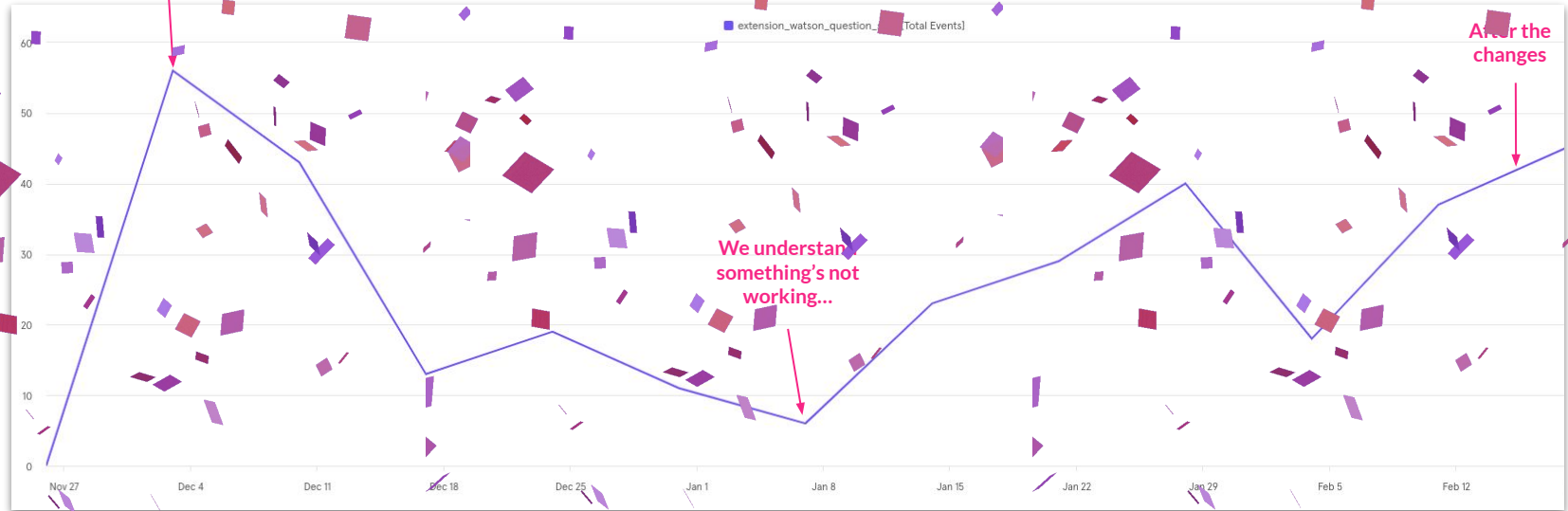
Table Name	Num Of Uses In The Last Month	Last Query
public.events_prod	1867	2 days ago 9/3/2024
public.events_pr	574	2 days ago 9/3/2024
public.events_l3	322	26 days ago 14/2/2024
public.events_v2	64	4 days ago 6/3/2024

Below the table, there is a 'show more' link and a downward arrow icon.

## Stage 2: Aware-of-itself



## Stage 2: Aware-of-itself



—————  
How many **active users**  
on our platform?

**17,000** active users

**17,000** active users

VS

**17,000** active users

VS

**17,000** active users

**17,000** active users

VS

**27,000** active users





**17,000** active users

VS

**27,000** active users

*“I didn’t know that I need to remove the duplicates,  
and take into account the NULL values.”*



**The Concept:**  
They “ask questions” through their SQL

What did we learn from this stage?

*Known Unknowns*

*Known Knowns*

*Unknown Unknowns*

*Unknown Knowns*

## What did we learn from this stage?

*Known Unknowns*

*Known Knowns*

No need to ask

*Unknown Unknowns*

*Unknown Knowns*

## What did we learn from this stage?

*Known Unknowns*

*Known Knowns*

No need to ask ✓

*Unknown Unknowns*

*Unknown Knowns*

## What did we learn from this stage?

*Known Unknowns*

I'll ask but I won't find an answer

*Known Knowns*

No need to ask ✓

*Unknown Unknowns*

*Unknown Knowns*

## What did we learn from this stage?

*Known Unknowns*

I'll ask but I won't find an answer ✓

*Known Knowns*

No need to ask ✓

*Unknown Unknowns*

*Unknown Knowns*

## What did we learn from this stage?

*Known Unknowns*

I'll ask but I won't find an answer ✓

*Known Knowns*

No need to ask ✓

*Unknown Unknowns*

*Unknown Knowns*

I don't know that I'm missing something,  
But someone else does



## What did we learn from this stage?

### *Known Unknowns*

I'll ask but I won't find an answer ✓

### *Known Knowns*

No need to ask ✓

### *Unknown Unknowns*

Don't know I need to ask

### *Unknown Knowns*

I don't know that I'm missing something,  
But someone else does

## What did we learn from this stage?

### *Known Unknowns*

I'll ask but I won't find an answer ✓

### *Known Knowns*

No need to ask ✓

### *Unknown Unknowns*

Don't know I need to ask

### *Unknown Knowns*

I don't know that I'm missing something,  
But someone else does

### **Stage 3: Dealing with the unknowns**

Answering questions before they were asked

## Stage 3: Dealing with the unknowns

### Q Metadata Usage

*“Is this the correct way to connect between these tables?”*

```
SELECT
  u.id AS user_id,
  u.email,
  u.account_name,
  oi.created_at AS account_signup_date
FROM
  users u
JOIN
  onboard_information oi ON u.id = oi.user_id
WHERE
  oi.created_at >= NOW() - INTERVAL '1 month'
  AND u.id IN (SELECT DISTINCT user_id FROM
  queries WHERE created_at >= NOW() - INTERVAL
  '1 day')
```

### Q Data Freshness

*“Is this the most up-to-date field I'm supposed to use?”*

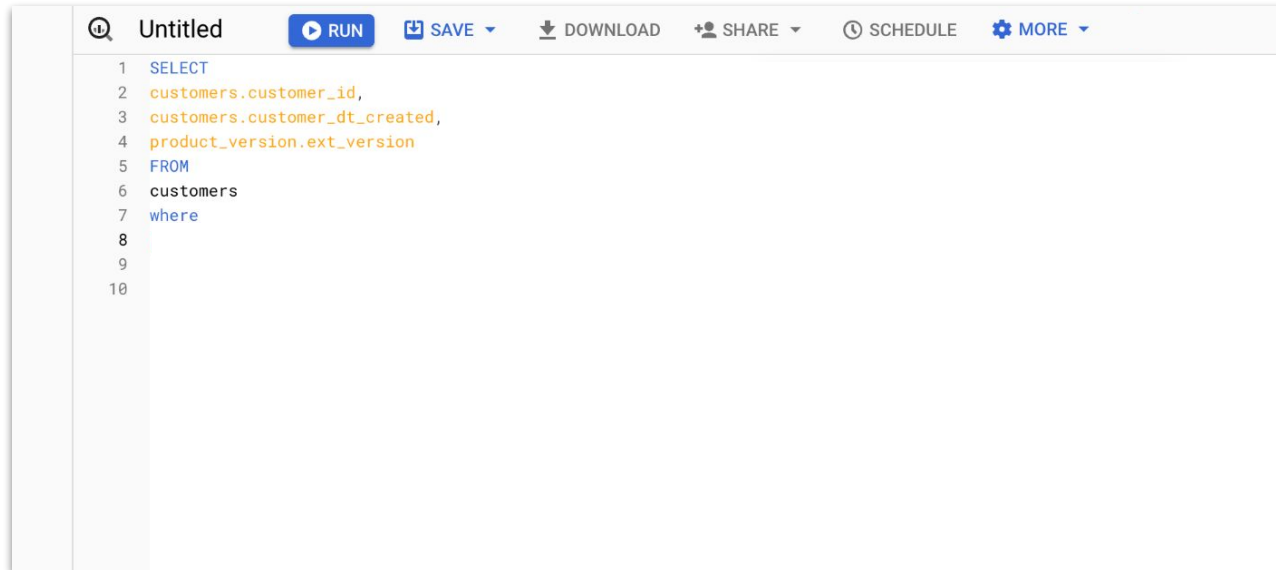
### Q SQL Logic

*“Am I missing any other filters that I need to apply?”*

### Q Syntax Conventions

*“Is this the right syntax in this case?”*

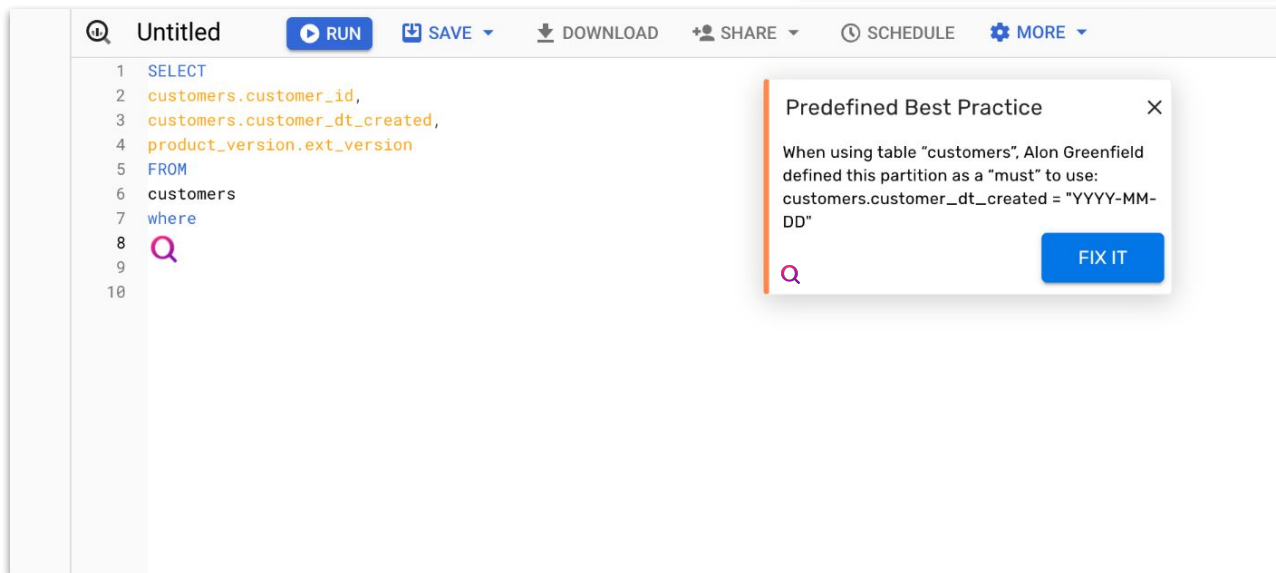
## Stage 3: Dealing with the unknowns



The image shows a screenshot of a SQL editor interface. The title bar at the top reads "Untitled" and contains several action buttons: "RUN" (with a play icon), "SAVE" (with a floppy disk icon), "DOWNLOAD" (with a download icon), "SHARE" (with a person icon), "SCHEDULE" (with a clock icon), and "MORE" (with a gear icon). The main area of the editor contains a SQL query with line numbers 1 through 10 on the left margin. The query is as follows:

```
1 SELECT
2   customers.customer_id,
3   customers.customer_dt_created,
4   product_version.ext_version
5 FROM
6   customers
7 where
8
9
10
```

## Stage 3: Dealing with the unknowns



The screenshot shows a SQL editor window titled "Untitled" with a toolbar containing buttons for RUN, SAVE, DOWNLOAD, SHARE, SCHEDULE, and MORE. The SQL query in the editor is as follows:

```
1 SELECT
2   customers.customer_id,
3   customers.customer_dt_created,
4   product_version.ext_version
5 FROM
6   customers
7 where
8   Q
9
10
```

A "Predefined Best Practice" dialog box is open on the right side of the editor. It contains the following text:

When using table "customers", Alon Greenfield defined this partition as a "must" to use:  
customers.customer\_dt\_created = "YYYY-MM-DD"

The dialog box includes a search icon (Q) and a "FIX IT" button.

# Alignment Calculator

# of data assets used with Sherloq suggestions

	Before Sherloq	After Sherloq	Increase%
<b>Verified Business Metrics</b>	22% 223 / 1026	60% 824 / 1354	+38%
<b>Filters</b>	29% 423 / 1429	75% 1131 / 1502	+46%
<b>Joins</b>	39% 241 / 609	73% 870 / 1191	+34%
<b>Partitions</b>	17% 107 / 623	46% 437 / 952	+29%
<b>Aggregated metadata</b>	20% 636 / 3184	53% 898 / 1675	+33%
<b>Up-to-date fields</b>	19% 583 / 2961	56% 1693 / 3024	+37%
<b>Average Increase</b>			<b>+36%</b>

# What helped us?

Integrated into  
their workflow



## Monitoring

**Botson For Slack - Question** APP 10:12 AM  
zapier botson: asked

**Zapier** APP 10:12 AM  
zapier Botson's Answer:

**Botson For Slack - Like** APP 10:13 AM  
zapier Liked the message:

has copied text from Botson 🤖

## 24/7 Support

### After each time user asks a question

1) Nadav, 2) Noa, 3) Alon → checking through the events the user flow and understand what he was looking for, and that we answered.

- If we helped them → 🙌
- If we didn't help them →
  - Check why [UX/DB/Feature/Bug]
    - If it's because a bug →
      - Notify for all: "Bug in production", fix the bug asap, make additional QA and notify them back.
      - Send a message (through Nadav) in private, if we can catch it before they notice us
      - Check if we can help them through our DB and suggest them the results.
    - Ask if that helped them.
  - If it's from because our DB (AKA - we don't have the information they're looking for) →
    - Understand what is missing → tables? parser?
    - Notify the user that we don't have this scope of information and we'll ask from Itai or Haya for this information.
    - After inserting the new DB, checking if we actually answer on the previous user question.
    - Inform the users about the update and suggest they check the answer through HelloSherlock.
  - If it's because UX reason →
    - Understand what the user was looking for, and search for the ideal answer.
    - Notify the user through HelloSherlock about what we understood they were



## Last thoughts and learnings



GenAI & LLMs as a tool  
not as a product



Start with just 1 use-case  
that works



“Start prepping the servers”  
Approach



**Thanks and I'd love to stay in touch!**

Noy Twerski

Co-founder & CEO @ Sherloq

[noy@sherloqdata.io](mailto:noy@sherloqdata.io)