



PGSession¹⁷

Loïc Knuchel
15 jan. 2025

@loicknuchel

**Explorer, documenter et faire évoluer
ses bases de données avec Azimutt**



Loïc Knuchel

Créateur de Azimutt

@loicknuchel

@azimuttapp

Pourquoi Azimutt?

Mieux comprendre et travailler
avec les base de données
à l'échelle



Comprendre une requête complexe

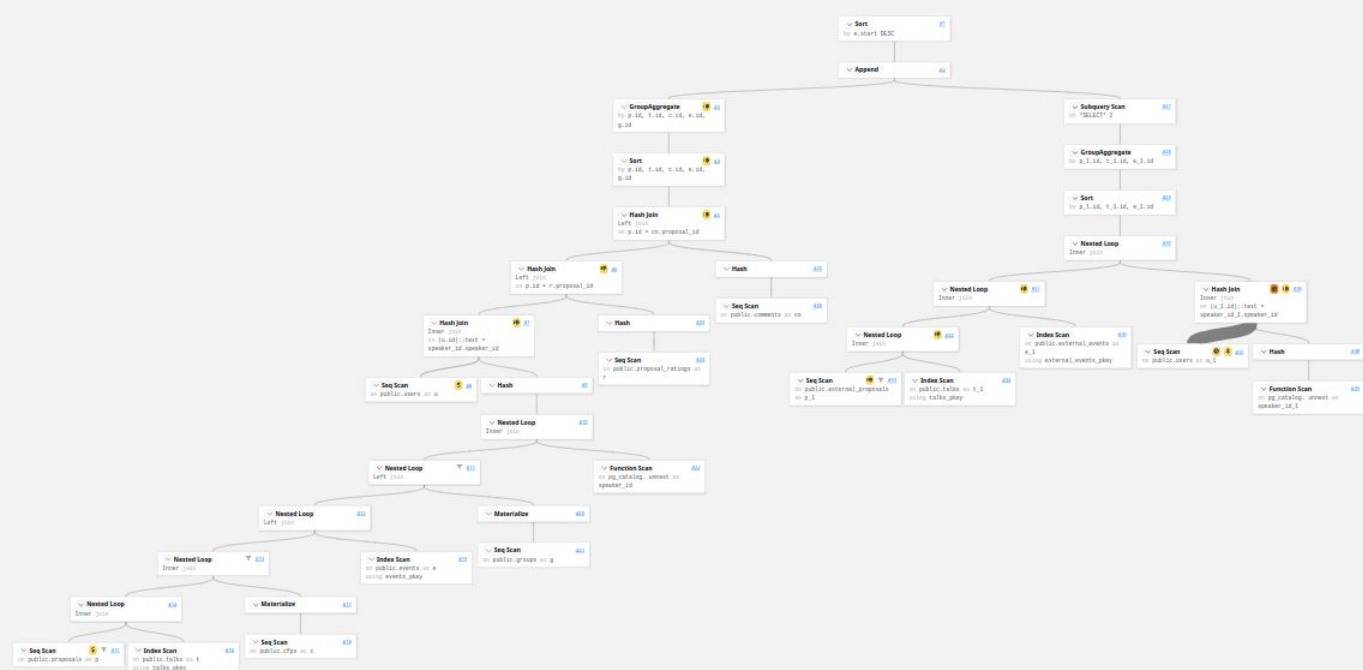
```
(SELECT p.id, 'Internal' AS kind, p.title, p.duration, p.description, p.tags, p.slides, p.video, null AS url
, json_agg(json_build_object('id', u.id, 'slug', u.slug)) AS speakers
, t.id AS talk_id, t.status AS talk_status
, c.id AS cfp_id, c.name AS cfp_name
, e.id AS event_id, e.kind AS event_kind, e.name AS event_name, e.start AS event_start
, g.id AS group_id, g.name AS group_name
, array_agg(r.grade) FILTER (WHERE r.grade IS NOT NULL) AS ratings
, count(co.id) AS nb_comments
FROM proposals p
CROSS JOIN LATERAL unnest(string_to_array(p.speakers, ',')) speaker_id
JOIN users u ON speaker_id = u.id
JOIN talks t ON p.talk_id = t.id
JOIN cfps c ON p.cfp_id = c.id
LEFT JOIN events e ON p.event_id = e.id
LEFT JOIN groups g ON c.group_id = g.id
LEFT JOIN proposal_ratings r ON p.id = r.proposal_id
LEFT JOIN comments co ON p.id = co.proposal_id
WHERE p.status = 'Accepted' AND '8a4ed953-1b25-4c18-8d48-7dd33d48bea1' = ANY (string_to_array(p.speakers, ','))
GROUP BY p.id, t.id, c.id, e.id, g.id
UNION ALL
(SELECT p.id, 'External' AS kind, p.title, p.duration, p.description, p.tags, p.slides, p.video, p.url
, json_agg(json_build_object('id', u.id, 'slug', u.slug)) AS speakers
, t.id AS talk_id, t.status AS talk_status
, null AS cfp_id, null AS cfp_name
, e.id AS event_id, e.kind AS event_kind, e.name AS event_name, e.start AS event_start
, null AS group_id, null AS group_name
, null AS ratings
, 0 AS nb_comments
FROM external_proposals p
CROSS JOIN LATERAL unnest(string_to_array(p.speakers, ',')) speaker_id
JOIN users u ON speaker_id = u.id
JOIN talks t ON p.talk_id = t.id
JOIN external_events e ON p.event_id = e.id
WHERE p.status = 'Accepted' AND '8a4ed953-1b25-4c18-8d48-7dd33d48bea1' = ANY (string_to_array(p.speakers, ','))
GROUP BY p.id, t.id, e.id
ORDER BY event_start DESC;
```



Execution time: 16ms Planning time: 2.42ms Triggers: N/A

time rows estimation cost buffers IO

#1	Sort				
#2	Append				
#3	GroupAggregate				
#4	Sort				
#5	Hash Join				
#6	Hash Join				
#7	Hash Join				
#8	Seq Scan				
#9	Hash				
#10	Nested Loop				
#11	Nested Loop				
#12	Nested Loop				
#13	Nested Loop				
#14	Nested Loop				
#15	Seq Scan				
#16	Index Scan				
#17	Materialize				
#18	Seq Scan				
#19	Index Scan				
#20	Materialize				
#21	Seq Scan				
#22	Function Scan				
#23	Hash				
#24	Seq Scan				
#25	Hash				
#26	Seq Scan				
#27	Subquery Scan				
#28	GroupAggregate				
#29	Sort				
#30	Nested Loop				
#31	Nested Loop				
#32	Nested Loop				
#33	Seq Scan				
#34	Index Scan				
#35	Index Scan				
#36	Hash Join				
#37	Seq Scan				
#38	Hash				
#39	Function Scan				



#1 PRODUCT OF THE WEEK
Developer Tools

1522 stars

Slack community

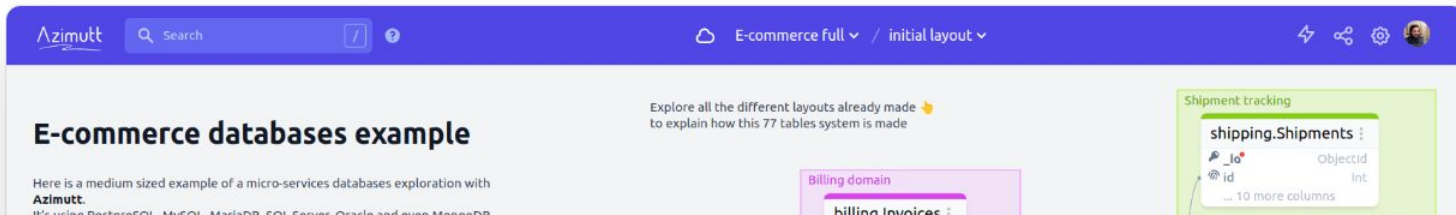
Seamless exploration for large & complex **databases**

\$ npx azimutt explore \$db_url

\$ npx azimutt analyze \$db_url

Explore your database

Explore samples →

For Developers, Architects, DBAs, Data Analysts, Product Managers, Tech Support, **what about you?**

```
→ ~ npx azimutt explore postgresql://postgres:postgres@localhost/gospeak
```



Hello from Azimutt 🙌

Explore your **28 tables** database with Azimutt. Create layouts, show tables you want and share them. Use the [search bar](#) or start with an overview:

Show all tables (28)

AI 🚀

 Click on colored column icons to follow relations (in/out).

If you ❤️ Azimutt, [come and say hi](#). We are eager to learn how you use it and for what. We also love [feedback and feature requests](#).





Hello from Azimutt 🙌


Explore your **28 tables** database with Azimutt. Create layouts, show tables you want and share them. Use the [search bar](#) or start with an overview:

Show all tables (28)

AI 🚀

 Click on colored column i

 Prompt to generate layout

 Generate layout from SQL query

If you ❤️ Azimutt, [come and say hi](#). We are eager to learn how you use it and for what. We also love [feedback and feature requests](#).





Generate layout from SQL query:

If you have sub-optimal or bad results, try again and let us know so we can improve.

If you ❤️ Azimutt, [come and say hi](#). We are eager to learn how you use it and for what.
We also love [feedback and feature requests](#).

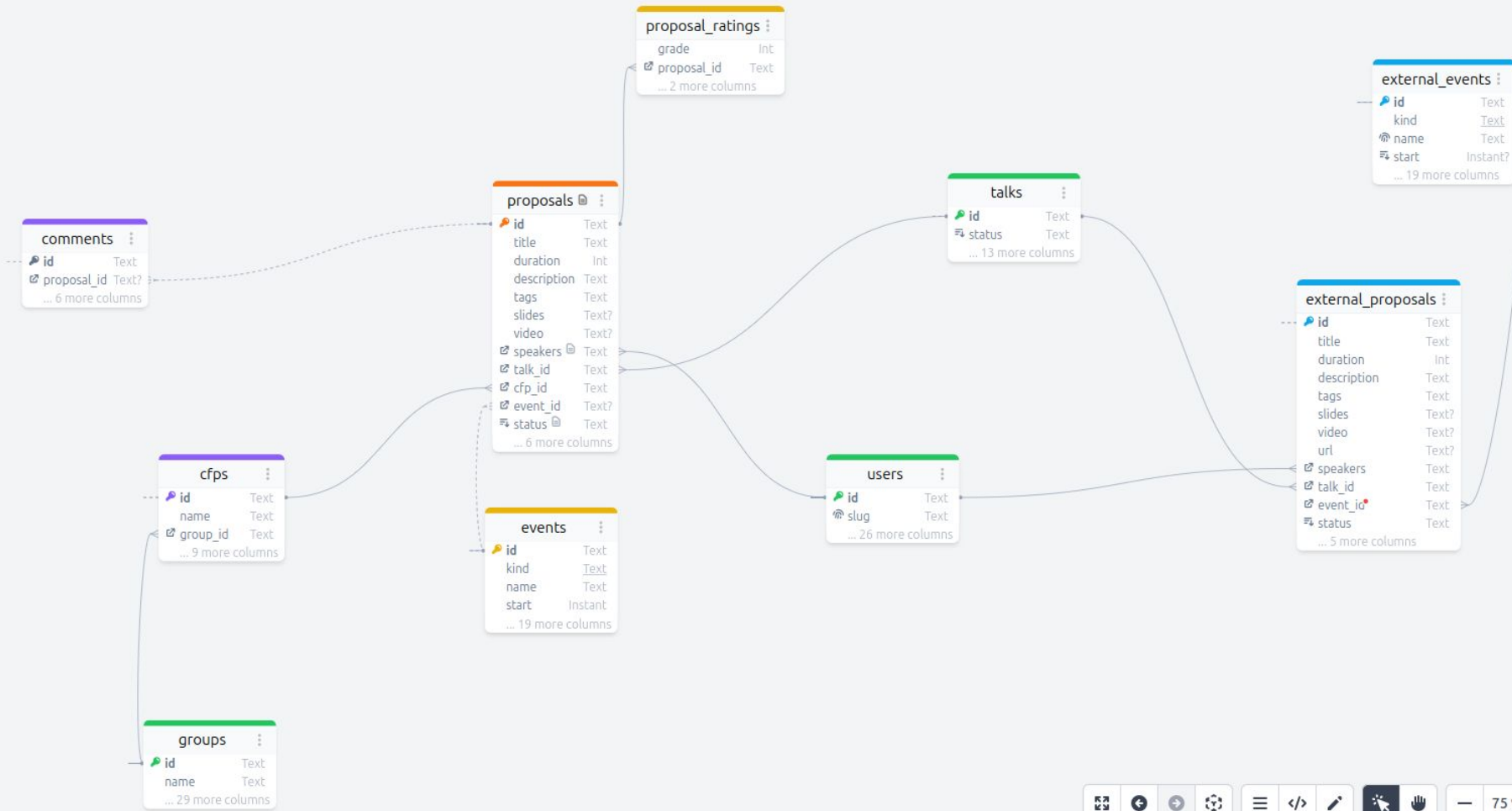


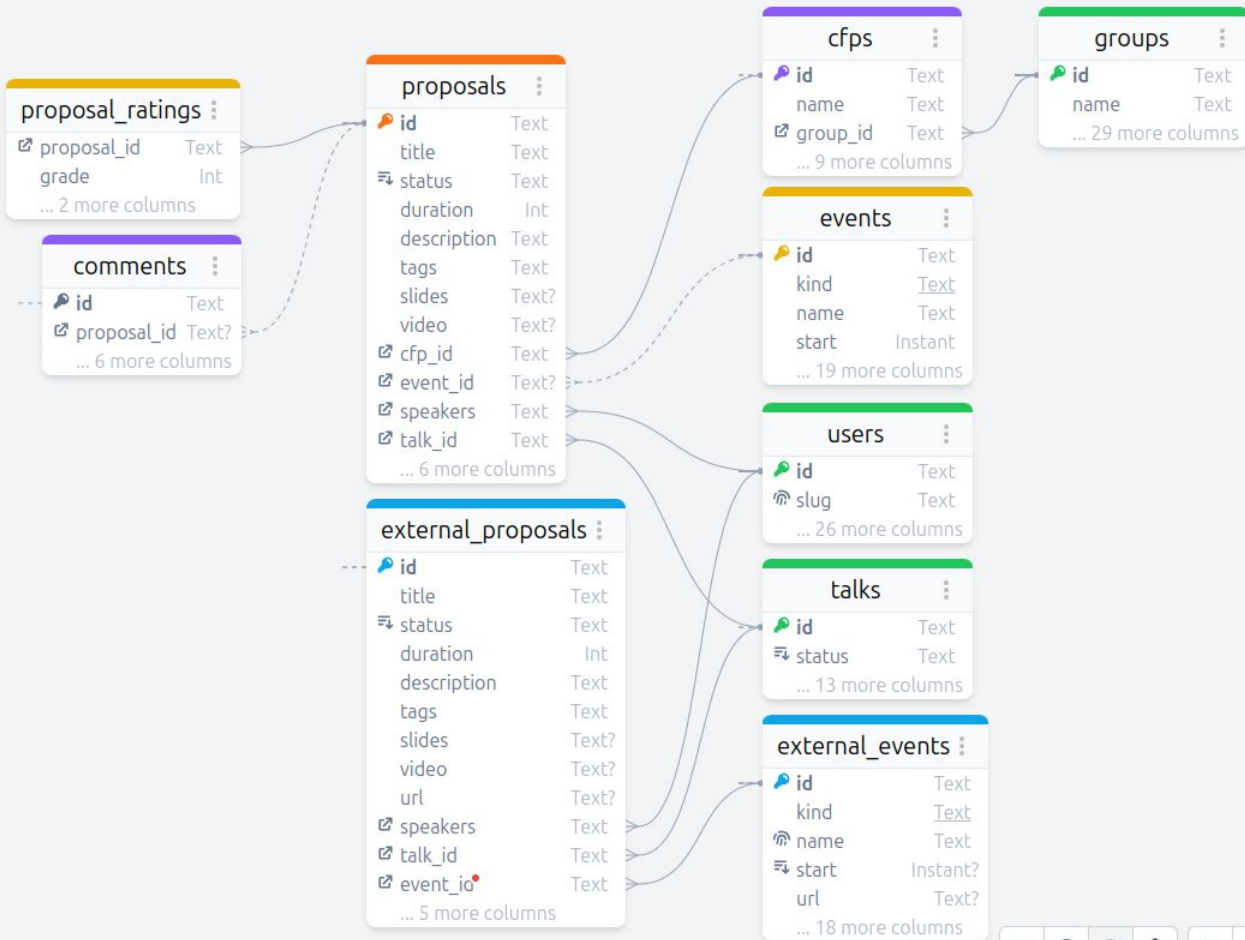
Generate layout from SQL query:

If you have sub-optimal or bad results, try again and let us know so we can improve.

```
, g.no AS group_no, g.name AS group_name
, array_agg(r.grade) FILTER (WHERE r.grade IS NOT NULL) AS ratings
, count(co.id) AS nb_comments
FROM proposals p
  CROSS JOIN LATERAL unnest(string_to_array(p.speakers, ',')) speaker_id
  JOIN users u ON speaker_id = u.id
  JOIN talks t ON p.talk_id = t.id
  JOIN cfps c ON p.cfp_id = c.id
  LEFT JOIN events e ON p.event_id = e.id
  LEFT JOIN groups g ON c.group_id = g.id
  LEFT JOIN proposal_ratings r ON p.id = r.proposal_id
  LEFT JOIN comments co ON p.id = co.proposal_id
WHERE p.status = 'Accepted' AND '8a4ed953-1b25-4c18-8d48-7dd33d48bea1' = ANY(string_to_array(p.speakers, ','))
GROUP BY p.id, t.id, c.id, e.id, g.id
UNION ALL
(SELECT p.id, 'External' AS kind, p.title, p.duration, p.description, p.tags, p.slides, p.video, p.url
, json_agg(json_build_object('id', u.id, 'slug', u.slug)) AS speakers
, t.id AS talk_id, t.status AS talk_status
, null AS cfp_id, null AS cfp_name
, e.id AS event_id, e.kind AS event_kind, e.name AS event_name, e.start AS event_start
, null AS group_id, null AS group_name
, null AS ratings
, 0 AS nb_comments
FROM external_proposals p
  CROSS JOIN LATERAL unnest(string_to_array(p.speakers, ',')) speaker_id
  JOIN users u ON speaker_id = u.id
  JOIN talks t ON p.talk_id = t.id
  JOIN external_events e ON p.event_id = e.id
WHERE p.status = 'Accepted' AND '8a4ed953-1b25-4c18-8d48-7dd33d48bea1' = ANY(string_to_array(p.speakers, ','))
GROUP BY p.id, t.id, e.id)
ORDER BY event_start DESC}
```

[Back](#)[Generate](#)





proposal_ratings	
proposal_id	Text
grade	Int
... 2 more columns	

comments	
id	Text
proposal_id	Text?
... 6 more columns	

proposals	
id	Text
title	Text
status	Text
duration	Int
description	Text
tags	Text
slides	Text?
video	Text?
cfp_id	Text
event_id	Text?
speakers	Text
talk_id	Text
... 6 more columns	

external_proposals	
id	Text
title	Text
status	Text
duration	Int
description	Text
tags	Text
slides	Text?
video	Text?
url	Text?
speakers	Text
talk_id	Text
event_id	Text
... 5 more columns	

cfps	
id	Text
name	Text
group_id	Text
... 9 more columns	

events	
id	Text
kind	Text
name	Text
start	Instant
... 19 more columns	

users	
id	Text
slug	Text
... 26 more columns	

talks	
id	Text
status	Text
... 13 more columns	

external_events	
id	Text
kind	Text
name	Text
start	Instant?
url	Text?
... 18 more columns	

groups	
id	Text
name	Text
... 29 more columns	

```

SELECT ...
FROM proposals
JOIN users ON ...
JOIN talks ON ...
JOIN cfps ON ...
JOIN events ON ...
JOIN groups ON ...
JOIN proposal_ratings ON ...
JOIN comments ON ...

```

UNION ALL

```

SELECT ...
FROM external_proposals
JOIN users ON ...
JOIN talks ON ...
JOIN external_events ON ...

```

proposals table details



- public
- proposals

proposals

Click to write notes

No tags

Constraints

- Primary key: id (proposals_pkey)
- Unique: talk_id, cfp_id (proposals_talk_id_cfp_id_key)
- Index: status (proposals_status_idx)

References

- cfps
- events
- talks
- users

Referenced by

- comments
- proposal_ratings

In layouts

- overview
- onboarding / 0. Introduction
- queries / user proposals full

From sources

- gospeak (537 rows)

Stats for gospeak source:

4 rows, table size: 1.2 Mo, index size: 360 ko

18 columns

id	e7b623d0-c4b7-4401-b70c-b2a5e6a27b88
talk_id	7cf50a0c-75ee-4798-874e-ca0077360b4a
cfp_id	7b82e961-dca0-4531-aa03-e77f1f2f1046

proposal_ratings
proposal_id Text
grade Int
... 2 more columns

comments
id Text
proposal_id Text?
... 6 more columns

proposals
id Text
title Text
status Text
duration Int
description Text
tags Text
slides Text?
video Text?
cfp_id Text
event_id Text?
speakers Text

cfps
id Text
name Text
group_id Text
... 9 more columns

groups
id Text
name Text
... 29 more columns

events
id Text
kind Text
name Text
start Instant
... 19 more columns

users
id Text
slug Text
... 26 more columns

talks
id Text
status Text
... 13 more columns

external_events
id Text
kind Text
name Text
start Instant?
url Text?

url Text?
speakers Text
talk_id Text
event_id Text
... 5 more columns

Accès instantané
aux indexes
et contraintes

proposals.status column details



- public
- proposals
- status

4. status

character varying(10) NOT NULL

Click to write notes

No tags

Constraints

proposals_status_idx: btree (status)

In layouts

onboarding / 0. Introduction

overview

queries / user proposals full

From sources

gospeak

Stats for gospeak source:

Common values:

- Accepted (79%, 426)
- Declined (18%, 98)
- Pending (2.9%, 13)

Cardinality: 3, 0 nulls, Avg size: 8 bytes

proposal_ratings	
proposal_id	Text
grade	Int
... 2 more columns	

comments	
id	Text
proposal_id	Text?
... 6 more columns	

proposals	
id	Text
title	Text
status	Text
duration	Int
description	Text
tags	Text
slides	Text?
video	Text?
cfp_id	Text
event_id	Text?
speakers	Text
talk_id	Text
... 6 more columns	

external_proposals	
id	Text
title	Text
status	Text
duration	Int
description	Text
tags	Text
slides	Text?
video	Text?
url	Text?
speakers	Text
talk_id	Text
event_id	Text
... 5 more columns	

cfps	
id	Text
name	Text
group_id	Text
... 9 more columns	

groups	
id	Text
name	Text
... 29 more columns	

events	
id	Text
kind	Text
name	Text
start	Instant
... 19 more columns	

users	
id	Text
slug	Text
... 26 more columns	

talks	
id	Text
status	Text
... 13 more columns	

external_events	
id	Text
kind	Text
name	Text
start	Instant?
url	Text?
... 10 more columns	

Contraintes et statistiques de la colonnes



Explorer sa base de données



Hello from Azimutt 🙌

Explore your **28 tables** database with Azimutt. Create layouts, show tables you want and share them. Use the [search bar](#) or start with an overview:

Show all tables (28)

AI 🚀

 Click on colored column i

 Prompt to generate layout

 Generate layout from SQL query

If you ❤️ Azimutt, [come and say hi](#). We are eager to learn how you use it and for what. We also love [feedback and feature requests](#).





Which layout do you want to build?

If you have sub-optimal or bad results, try again and let us know so we can improve.

BackGenerate

If you ❤️ Azimutt, [come and say hi](#). We are eager to learn how you use it and for what.
We also love [feedback and feature requests](#).



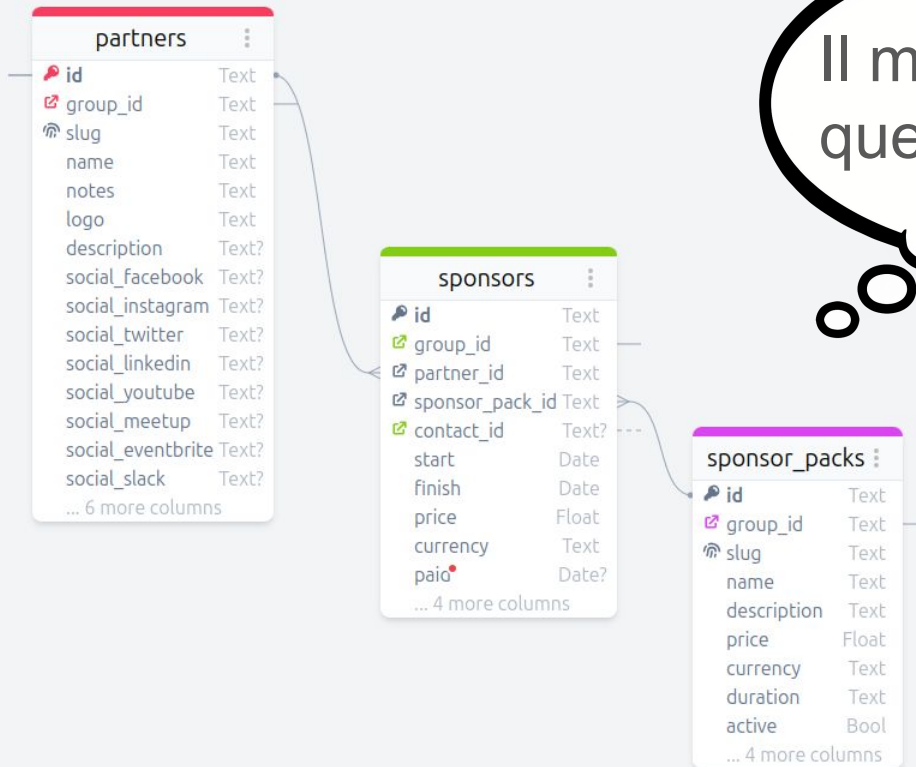


Which layout do you want to build?

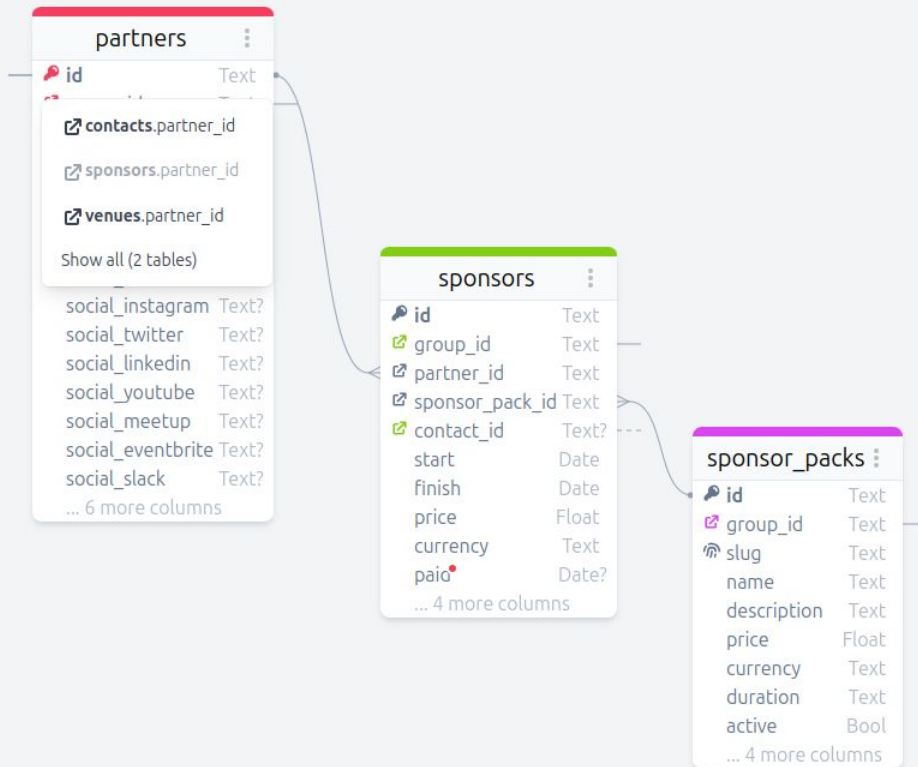
If you have sub-optimal or bad results, try again and let us know so we can improve.

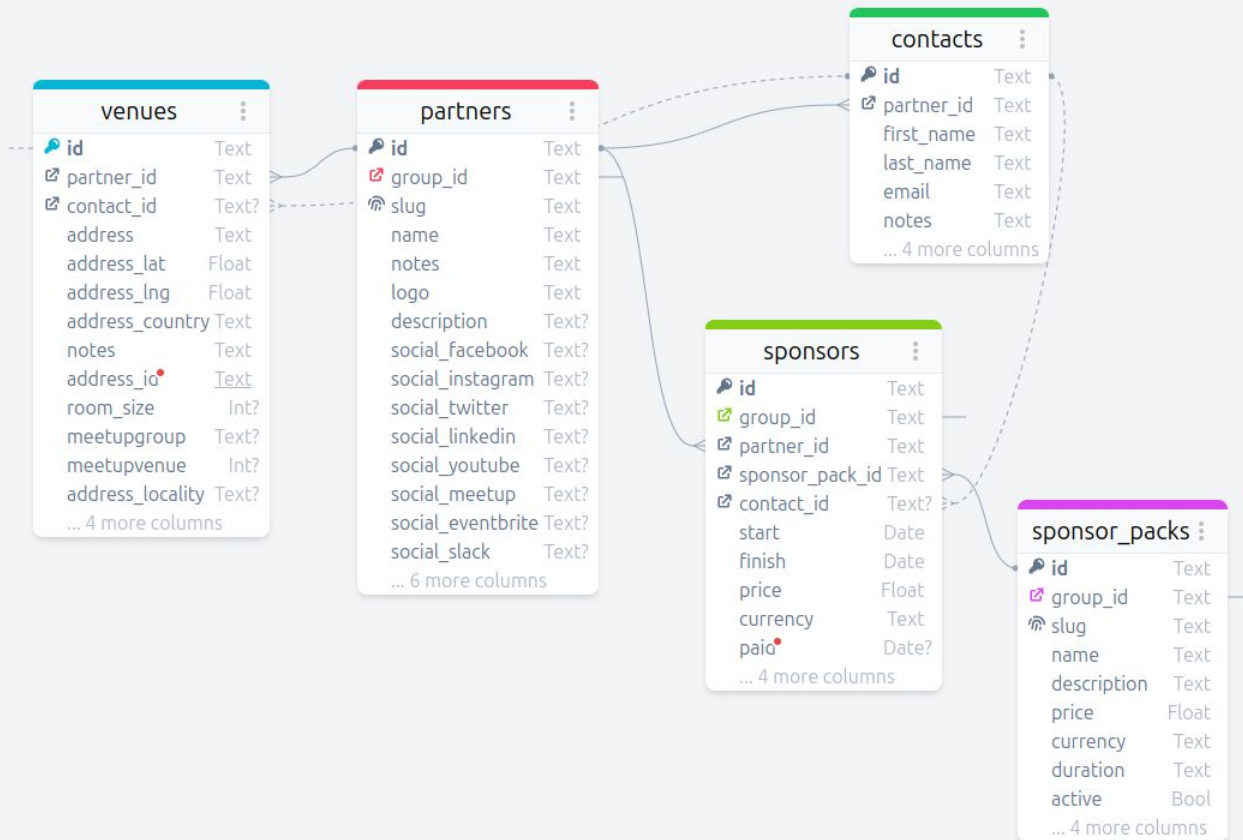
If you ❤️ Azimutt, [come and say hi](#). We are eager to learn how you use it and for what.
We also love [feedback and feature requests](#).





Il manque quelques tables...





Type to search into tables (🗃️), columns (🔍) and relations (🔗)

Show all tables (28)

Browse table list

Or check other interesting tables:

- 🗃️ users
- 🗃️ groups
- 🗃️ proposals
- 🗃️ events
- 🗃️ external_events
- 🗃️ talks
- 🗃️ cfps
- 🗃️ external_proposals
- 🗃️ pg_stat_statements
- 🗃️ videos
- 🗃️ logins
- 🗃️ user_requests
- 🗃️ external_cfps
- 🗃️ comments
- 🗃️ group_settings

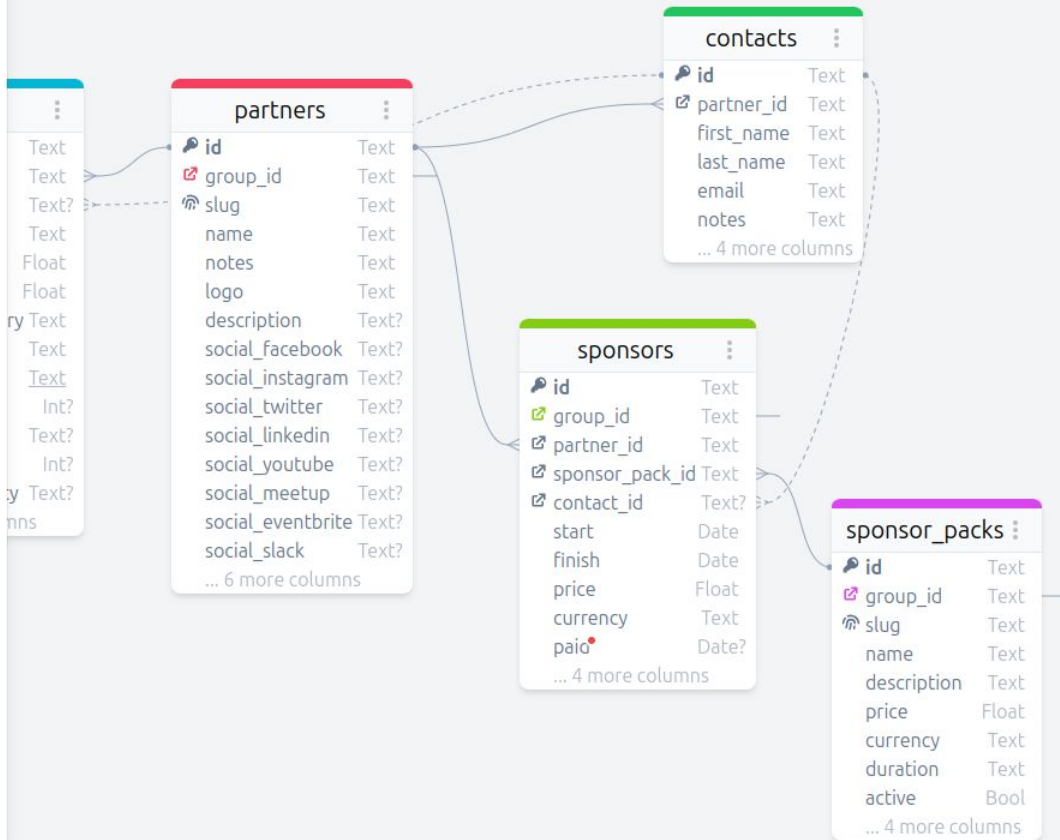
partners	
🔑 id	Text
🔗 group_id	Text
🔗 slug	Text
name	Text
notes	Text
logo	Text
description	Text?
social_facebook	Text?
social_instagram	Text?
social_twitter	Text?
social_linkedin	Text?
social_youtube	Text?
social_meetup	Text?
social_eventbrite	Text?
social_slack	Text?
...	6 more columns

contacts	
🔑 id	Text
🔗 partner_id	Text
first_name	Text
last_name	Text
email	Text
notes	Text
...	4 more columns

sponsors	
🔑 id	Text
🔗 group_id	Text
🔗 partner_id	Text
🔗 sponsor_pack_id	Text
🔗 contact_id	Text?
start	Date
finish	Date
price	Float
currency	Text
paio	Date?
...	4 more columns

sponsor_packs	
🔑 id	Text
🔗 group_id	Text
🔗 slug	Text
name	Text
description	Text
price	Float
currency	Text
duration	Text
active	Bool
...	4 more columns

- events
- event_rsvs
- external_events
- comments.event_id
- event_rsvs.event_id
- events.meetupevent
- external_cfps.event_id
- external_proposals.event_id
- group_settings.event_description
- group_settings.event_templates
- groups.social_eventbrite
- partners.social_eventbrite
- pg_stat_statements.stddev_plan_time
- proposals.event_id
- user_requests.event_id
- user_requests.external_event_id
- users.social_eventbrite
- video_sources.external_event_id
- comments_event_id_fkey
- event_rsvs_event_id_fkey
- event_rsvs_user_id_fkey



events	
id	Text
group_id	Text
talks	Text
cfp_id	Text?
venue	Text?
slug	Text
name	Text
start	Instant
allow_rsvp	Bool
description	Text
orga_notes	Text
tags	Text
kind	Text
max_attendee	Int?
published	Instant?
... 8 more columns	

venues	
id	Text
partner_id	Text
contact_id	Text?
address	Text
address_lat	Float
address_lng	Float
address_country	Text
notes	Text
address_id	Text
room_size	Int?
meetupgroup	Text?
meetupvenue	Int?
address_locality	Text?
... 4 more columns	

partners	
id	Text
group_id	Text
slug	Text
name	Text
notes	Text
logo	Text
description	Text?
social_facebook	Text?
social_instagram	Text?
social_twitter	Text?
social_linkedin	Text?
social_youtube	Text?
social_meetup	Text?
social_eventbrite	Text?
social_slack	Text?
... 6 more columns	

contacts	
id	Text
partner_id	Text
first_name	Text
last_name	Text
email	Text
notes	Text
... 4 more columns	

sponsors	
id	Text
group_id	Text
partner_id	Text
sponsor_pack_id	Text
contact_id	Text?
start	Date
finish	Date
price	Float
currency	Text
paio	Date?
... 4 more columns	

sponsor_packs	
id	Text
group_id	Text
slug	Text
name	Text
description	Text
price	Float
currency	Text
duration	Text
active	Bool
... 4 more columns	



Find a path between tables

Use relations to find a path between two tables. Useful when you don't know how tables are connected but you want to query their data together.

Search settings

From

public.proposals

Starting table for the path

To

public.partners

Table you want to go to

Found 5 paths between tables **proposals** and **partners**:

1. proposals > [events](#) > [groups](#) > [partners](#)

2. proposals > [users](#) > [groups](#) > [partners](#)

sponsors.partner_id → partners.id

3. proposals > [events](#) > [groups](#) > [sponsors](#) > [partners](#)

```
SELECT *
FROM proposals
  JOIN events ON events.id = proposals.event_id
  JOIN groups ON groups.id = events.group_id
  JOIN sponsors ON sponsors.group_id = groups.id
  JOIN partners ON partners.id = sponsors.partner_id
```

4. proposals > [talks](#) > [users](#) > [groups](#) > [partners](#)

5. proposals > [users](#) > [groups](#) > [sponsors](#) > [partners](#)

Not enough results? Check 'Search settings' above and increase max length of path or remove some ignored columns...

We hope you like this feature. If you have a few minutes, please write us [a quick feedback](#) about it and your use case so we can continue to improve 🚀

Done



Documentation

Full User Proposals query

Layout with all and only tables and columns used in the query.



Get user proposals:

```
(SELECT p.id, 'Internal' AS kind, p.title, p.duration, p.description, p.tags, p.slides, p.video, null AS url
, json_agg(json_build_object('id', u.id, 'slug', u.slug)) AS speakers
, t.id AS talk_id, t.status AS talk_status
, c.id AS cfp_id, c.name AS cfp_name
, e.id AS event_id, e.kind AS event_kind, e.name AS event_name, e.start AS event_start
, g.id AS group_id, g.name AS group_name
, array_agg(r.grade) FILTER (WHERE r.grade IS NOT NULL) AS ratings
, count(co.id) AS nb_comments
FROM proposals p
CROSS JOIN LATERAL unnest(string_to_array(p.speakers, ',')) speaker_id
JOIN users u ON speaker_id = u.id
JOIN talks t ON p.talk_id = t.id
JOIN cfps c ON p.cfp_id = c.id
LEFT JOIN events e ON p.event_id = e.id
LEFT JOIN groups g ON c.group_id = g.id
LEFT JOIN proposal_ratings r ON p.id = r.proposal_id
LEFT JOIN comments co ON p.id = co.proposal_id
WHERE p.status = 'Accepted' AND '84ed953-1b25-4c18-8d48-7dd33d48bea1' = ANY (string_to_array(p.speakers, ','))
GROUP BY p.id, t.id, c.id, e.id, g.id)
UNION ALL
(SELECT p.id, 'External' AS kind, p.title, p.duration, p.description, p.tags, p.slides, p.video, p.url
, json_agg(json_build_object('id', u.id, 'slug', u.slug)) AS speakers
, t.id AS talk_id, t.status AS talk_status
, null AS cfp_id, null AS cfp_name
, e.id AS event_id, e.kind AS event_kind, e.name AS event_name
, null AS group_id, null AS group_name
, null AS ratings
, 0 AS nb_comments
FROM external_proposals p
CROSS JOIN LATERAL unnest(string_to_array(p.speakers, ',')) speaker_id
JOIN users u ON speaker_id = u.id
JOIN talks t ON p.talk_id = t.id
JOIN external_events e ON p.event_id = e.id
WHERE p.status = 'Accepted' AND '84ed953-1b25-4c18-8d48-7dd33d48bea1' = ANY (string_to_array(p.speakers, ','))
GROUP BY p.id, t.id, e.id)
ORDER BY event_start DESC;
```

Exemple results from the query:

id	kind	title	duration	description	tags	slides	video	url	speakers	talk_id	talk_status	cfps_id	cfp_name	event_id	event_kind	event_name	event_start	group_id	group_name	ratings	nb_comments
a3d77e31-22ae-433f-8c0e-e8ebafca3f83	Internal	J'ai un XSS, et alors ?	600000000000	Un XSS permet d'exécuter du JS...			https://docs.google.com/presentation/d/1D714wKVL4m9B3ptCvmaR-z6tCzBiv84U8f0Cv9		[{"id": "72ae8931-1b25-4c18-9d48-7dd33d48bea1", "slug": "loicknuchet"}]	e44e84f-d6d2-4b10-7d6f-235c8172e2de	Archived	8b32d4d2-d6e3-4337-ae05-e73ff2f1006	HumanTalks Paris	47f5a32-b4c1-c3b0-4c18f1b3392bdf82	Meetup	HumanTalks Paris Mai 2013	2013-05-14 19:00:00.000000	3bf01e97-3292-77b1-a201-43e3e307246f	HumanTalks Paris	[5,4]	0

GO SPEAK

Gospeak

Gospeak is a SaaS used by meetup organizers to organize events (mostly [HumanTalks Paris](#) 😊).

Partners

venues	
id	Text
partner_id	Text
contact_id	Text?
address	Text
notes	Text
room_size	Int
meetupvenue	Text
...	10 more columns

sponsors	
id	Text
group_id	Text
partner_id	Text
sponsor_pack_id	Text
contact_id	Text?
start	Date
finish	Date
price	Float
...	6 more columns

partners	
id	Text
group_id	Text
slug	Text
name	Text
notes	Text
logo	Text
description	Text?
...	14 more columns

Organizers can manage Partners, basically company they discuss with, either for hosting the meetup (using Venues) or sponsoring (Sponsors).

groups	
id	Text
owners	Text
slug	Text
name	Text
description	Text
tags	Text
status	Text
logo	Text?
banner	Text?
contact	Text?
website	Text?
location	Text?
location_lat	Float?
location_lng	Float?
location_locality	Text?
...	16 more columns

cfps	
id	Text
group_id	Text
slug	Text
name	Text
description	Text
tags	Text
begin	Instant?
close	Instant?
...	4 more columns

Speakers can submit Proposals to CFPs (Call For Papers) and then organizers can select Proposals for Events :D

events	
id	Text
talks	Text
cfp_id	Text?
group_id	Text
venue	Text?
slug	Text
name	Text
description	Text
start	Instant
tags	Text
published	Instant?
allow_rsvp	Bool
...	11 more columns

proposals	
id	Text
talk_id	Text
cfp_id	Text
speakers	Text
event_id	Text?
status	Text
title	Text
duration	Int
description	Text
tags	Text
slides	Text?
video	Text?
...	6 more columns

This is a quick Gospeak overview to understand what it is about. Next step, look at **speaker** side.

proposals table details



- public
- proposals

proposals

A submitted talk to a CFP

Proposals are a version of a Talk submitted to a CFP. Most of the time it has the same data but the speaker can do any change during the submission process so they can have subtle changes or be radically different. From the Speaker perspective, they are grouped inside their talk view.

See also `external_proposals` table. They are proposals but submitted to CFPs not hosted on the Gospeak platform.

owner:group

Constraints

- Primary key: id (proposals_pkey)
- Unique: talk_id, cfp_id (proposals_talk_id_cfp_id_key)
- Index: status (proposals_status_idx)

- References
- cfps
 - events
 - talks
 - users
- Referenced by
- comments
 - events
 - proposal_ratings
 - user_requests
 - video_sources

In layouts

- onboarding / 0. Introduction overview
- queries / user proposals full

organizers to [ks Paris](#) 😊).

Partners
id
name
description
tags
status
logo
banner
contact
website
location
location_lat
location_lng
location_locality
...

Organizers can manages Partners, basically company they discuss with, either for hosting the meetup (using Venues) or sponsoring (Sponsors).

groups
id
owners
slug
name
description
tags
status
logo
banner
contact
website
location
location_lat
location_lng
location_locality
...

cfps
id
group_id
slug
name
description
tags
begin
close
...

Speakers can submit Proposals to CFPs (Call For Papers) and then organizers can select Proposals for Events :D

events
id
talks
cfp_id
group_id
venue
slug
name
description
start
tags
published
allow_rsvp
...

proposals
id
talk_id
cfp_id
speakers
event_id
status
title
duration
description
tags
slides
video
...

This is a quick Gospeak overview to understand what is it about. Next step, look at **speaker** side.

proposals.status column details



- public
- proposals
- status

4. status

character varying(10) NOT NULL

The Proposal status is an Enum to categorize Proposals.

This Enum can have 3 values (maybe more later):

- Pending**: when submitted but not accepted
- Accepted**: when accepted by the CFP owners
- Declined**: when rejected by the CFP owners

Other possible status:

- Confirmed**: when the speaker accepted also
- Backup**: when kept in case of no-show
- Planned**: when assigned to an event

No tags

Constraints

proposals_status_idx: btree (status)

In layouts

onboarding / 0. Introduction

overview

queries / user proposals full

From sources

gospeak

Stats for gospeak source:

Common values:

- Accepted (79%, 426)
- Declined (18%, 98)
- Pending (2.9%, 13)

Cardinality: 3, 0 nulls, Avg size: 8 bytes

organizers to ks Paris 😊).

partners	
id	Text
partner_id	Text
name	Text
description	Text?
more columns	

groups	
id	Text
owners	Text
slug	Text
name	Text
description	Text
tags	Text
status	Text
logo	Text?
banner	Text?
contact	Text?
website	Text?
location	Text?
location_lat	Float?
location_lng	Float?
location_locality	Text?
... 16 more columns	

cfps	
id	Text
group_id	Text
slug	Text
name	Text
description	Text
tags	Text
begin	Instant?
close	Instant?
... 4 more columns	

Speakers can submit Proposals to CFPs (Call For Papers) and then organizers can select Proposals for Events :D

events	
id	Text
talks	Text
cfp_id	Text?
group_id	Text
venue	Text?
slug	Text
name	Text
description	Text
start	Instant
tags	Text
published	Instant?
allow_rsvp	Bool
... 11 more columns	

proposals	
id	Text
talk_id	Text
cfp_id	Text
speakers	Text
event_id	Text?
status	Text
title	Text
duration	Int
description	Text
tags	Text
slides	Text?
video	Text?
... 6 more columns	

Organizers can manages Partners, basically company they discuss with, either for hosting the meetup (using Venues) or sponsoring (Sponsors).

This is a quick Gospeak overview to understand what is it about. Next step, look at **speaker** side.

proposals.speakers column details



- public
- proposals
- speakers

8. speakers

character varying(184) NOT NULL

Concatenated speaker IDs

concatenated IDs are bad as they require `string_to_array` function. Use proper PostgreSQL array or extract them in a n-m join table.

FIXME

References

[users.id](#)

In layouts

[onboarding / 0. Introduction](#)

[overview](#)

[queries / user.proposals.full](#)

From sources

gospeak

Stats for gospeak source:

Common values:

- 5602d5b5-73a7-43be-90bf-4229dd3a4fae (1.7%, 9)
- 2ba139cc-1a0f-4aed-8560-2a16251c3608 (1.5%, 8)
- 8ce0078b-14b8-4be2-a65a-18837beeb1a5 (1.5%, 8)
- 2bf0bdd9-ba1e-4ea5-8742-099da17ee568 (1.1%, 6)
- 4cf0b58b-2b5d-4771-89e3-dd89734f48dc (1.1%, 6)

Cardinality: 3.0798478, 0 nulls, Avg size: 37 bytes

organizers to [ks Paris](#) 😊).

Partners
id
group_id
slug
name
description
tags
begin
close
...

Organizers can manages Partners, basically company they discuss with, either for hosting the meetup (using Venues) or sponsoring (Sponsors).

groups
id
owners
slug
name
description
tags
status
logo
banner
contact
website
location
location_lat
location_lng
location_locality
...

cfps
id
group_id
slug
name
description
tags
begin
close
...

Speakers can submit Proposals to CFPs (Call For Papers) and then organizers can select Proposals for Events :D

events
id
talks
cfp_id
group_id
venue
slug
name
description
start
tags
published
allow_rsvp
...

proposals
id
talk_id
cfp_id
speakers
event_id
status
title
duration
description
tags
slides
video
...

This is a quick Gospeak overview to understand what is it about. Next step, look at **speaker** side.



Gospeak

Gospeak is a SaaS used by meetup organizers to organize events (mostly [HumanTalks Paris](#) 😊).

Speakers can submit Proposals to CFPs (Call For Papers) and then organizers can select Proposals for Events :D

groups	
id	Text
owners	Text
slug	Text
name	Text
description	Text
tags	Text
status	Text
logo	Text?
banner	Text?
contact	Text?
website	Text?
location	Text?
location_lat	Float?
location_lng	Float?
location_locality	Text?
... 16 more columns	

Create new layout Alt + L

- domains (2 layouts)
- initial layout (0 item)
- onboarding (5 layouts)
 - 0. Introduction (12 items)
 - 1. Speakers (2 items)
 - 2. Events (0 item)
 - 3. Partners (0 item)
 - 4. Videos (0 item)
- overview (28 items)
- queries (2 layouts)
- teams (4 layouts)

close Instant? ... 4 more columns

proposals	
id	Text
talk_id	Text
cfp_id	Text
speakers	Text
event_id	Text?
status	Text
title	Text
duration	Int
description	Text
tags	Text
slides	Text?
video	Text?
... 6 more columns	

Partners

venues	
id	Text
partner_id	Text
contact_id	Text?
address	Text
notes	Text
room_size	Int
meetupvenue	Text
... 10 more columns	

partners	
id	Text
group_id	Text
slug	Text
name	Text
notes	Text
logo	Text
description	Text?
... 14 more columns	

sponsors	
id	Text
group_id	Text
partner_id	Text
sponsor_pack_id	Text
contact_id	Text?
start	Date
finish	Date
price	Float
... 6 more columns	

Organizers can manage Partners, basically company they discuss with, either for hosting the meetup (using Venues) or sponsoring (Sponsors).

This is a quick Gospeak overview to understand what it is about. Next step, look at **speaker** side.



Explorer ses données

talks table details



- public
- talks

talks

Click to write notes

No tags

Constraints

- Primary key: id (talks_pkey)
- Unique: slug (talks_slug_key)
- Index: status (talks_status_idx)

References

users

Referenced by

- external_proposals
- proposals
- user_requests
- video_sources

In layouts

- domains / speakers
- onboarding / 1. Speakers overview
- queries / user_proposals_full

From sources

gospeak (587 rows)

Stats for gospeak source:

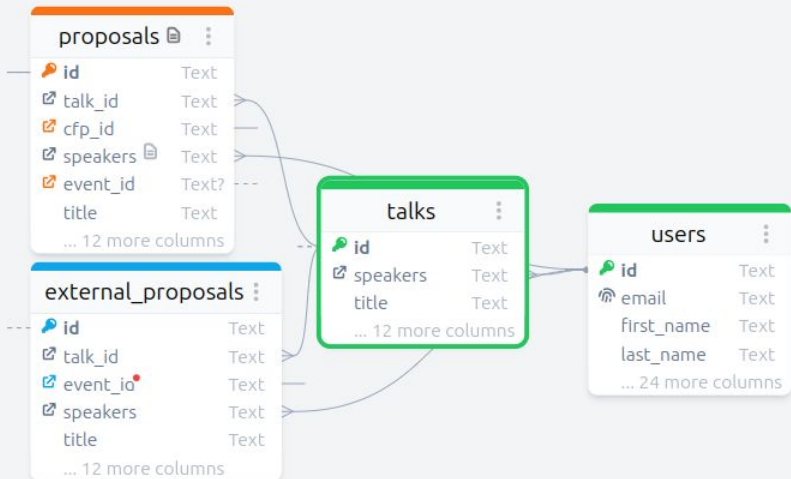
6 rows, table size: 1.2 Mo, index size: 369 ko

15 columns

id c16724e2-9f68-459b-a35c-2b6281ebbbc0

slug daily-english-speaking-practice

status Public



talks table details



public

talks

talks

[Click to write notes](#)

[No tags](#)

Constraints

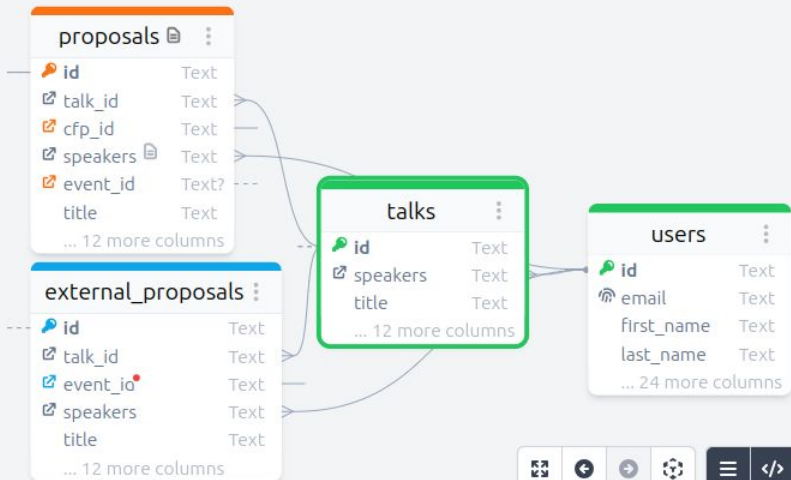
- Primary key: id (talks_pkey)
- Unique: slug (talks_slug_key)
- Index: status (talks_status_idx)

References

[users](#)

Referenced by

[external_proposals](#)



Navigation icons: zoom in, zoom out, home, refresh, menu, code editor, search, hand, close, 97% zoom

Data explorer

Visual editor

Query editor



```

1 SELECT *
2 FROM "public"."talks";
3
  
```

Run query

#1 gospeak (100 rows) 674 ms

Search in results

SELECT * FROM "public"."talks" LIMIT 100;

#	id	slug	status	title
1	1048e75e-bc58-45bf-b80c-cbf488c27b2a	comment-perdre-la-confiance-du-commanditaire-d	Public	Comment perdre la confiance du command
2	c16724e2-9f68-459b-a35c-2b6281ebbbc0	le-developpement-mobile-hybride-sort-du-bois	Archived	Le développement mobile hybride sort du b
3	5c4ab32b-04dd-41ea-9902-361c3d81ec87	blog-tech-vaincre-le-syndrome-de-la-page-blanche	Public	Blog Tech: Vaincre le syndrome de la page b
4	c46f5ac4-aae9-4d29-99ed-453760631332	peut-on-garder-une-url-secrete-	Public	Peut-on garder une URL secrète ?
5	a5aeca0f-5ba0-4907-a261-ce974739ac22	react-seo-un-mariage-impossible	Public	React & SEO : Un mariage impossible ?
6	d28802aa-f908-4481-a34e-1d5805929ec1	daily-english-speaking-practice	Public	Daily English speaking practice
7	7cf50a0c-75ee-4798-874e-ca0077360b4a	modular-coverage-tools-for-the-modern-web	Public	Modular coverage tools for the modern wet
8	c696575a-dbda-4c35-b56c-6767a436b68f	decouvrons-neo4j	Public	Découvrons Neo4J
9	1e1c803f-fe1b-4553-a0b9-d80233e6f233	pourquoi-detestez-vous-tester-	Public	Pourquoi détestez-vous tester ?
10	03f380a3-ddf1-442a-a1c9-612a05fbc06d	manipulation-des-pixels-d-un-canvas	Public	Manipulation des pixels d'un canvas : c'est b

Showing 1 to 10 of 100 results

1 2 3 ... 10

talks table details



public



talks



talks

 [Click to write notes](#)# [No tags](#)

Constraints

Primary key: id (talks_pkey)

Unique: slug (talks_slug_key)

Index: status (talks_status_idx)

References

[users](#)

Referenced by

[external_proposals](#)

🟢 talks: c16724e2-9f68-459b-a35c-2b6281ebbbc0

id c16724e2-9f68-459b-a35c-2b6281ebbbc0

slug le-developpement-mobile-hybride-sort-du-bois

status Archived

title Le développement mobile hybride sort du bois

description Ionic Framework a seulement un an mai...

speakers 8a4ed953-1b25-4c18-8d48-7dd33d48b...

... 9 more columns

from gospeak, a few seconds ago



97 %

Data explorer Beta

Visual editor

Query editor

```

1 SELECT *
2 FROM "public"."talks";
3

```



Run query

#1 gospeak (100 rows) 674 ms

Search in results

SELECT * FROM "public"."talks" LIMIT 100;

#	id	slug	status	title
1	1048e75e-bc58-45bf-b80c-cbf488c27b2a	comment-perdre-la-confiance-du-commanditaire-d	Public	Comment perdre la confiance du command
2	c16724e2-9f68-459b-a35c-2b6281ebbbc0	le-developpement-mobile-hybride-sort-du-bois	Archived	Le développement mobile hybride sort du b
3	5c4ab32b-04dd-41ea-9902-361c3d81ec87	blog-tech-vaincre-le-syndrome-de-la-page-blanche	Public	Blog Tech: Vaincre le syndrome de la page b
4	c46f5ac4-aae9-4d29-99ed-453760631332	peut-on-garder-une-url-secrete-	Public	Peut-on garder une URL secrète ?
5	a5aeca0f-5ba0-4907-a261-ce974739ac22	react-seo-un-mariage-impossible	Public	React & SEO : Un mariage impossible ?
6	d28802aa-f908-4481-a34e-1d5805929ec1	daily-english-speaking-practice	Public	Daily English speaking practice
7	7cf50a0c-75ee-4798-874e-ca0077360b4a	modular-coverage-tools-for-the-modern-web	Public	Modular coverage tools for the modern wet
8	c696575a-dbda-4c35-b56c-6767a436b68f	decouvrons-neo4j	Public	Découvrons Neo4J
9	1e1c803f-fe1b-4553-a0b9-d80233e6f233	pourquoi-detestez-vous-tester-	Public	Pourquoi détestez-vous tester ?
10	03f380a3-ddf1-442a-a1c9-612a05fbc06d	manipulation-des-pixels-d-un-canvas	Public	Manipulation des pixels d'un canvas : c'est b

Showing 1 to 10 of 100 results

1 2 3 ... 10

talks table details



public



talks



talks

📄 [Click to write notes](#)# [No tags](#)

Constraints

- 🔑 Primary key: id (talks_pkey)
- 🔑 Unique: slug (talks_slug_key)
- 🔑 Index: status (talks_status_idx)

References

[users](#)

Referenced by

[external_proposals](#)

🔍 talks: c16724e2-9f68-459b-a35c-2b6281ebbbc0

id	c16724e2-9f68-459b-a35c-2b6281ebbbc0
slug	le-developpement-mobile-hybride-sort-du-bois
status	Archived
title	Le développement mobile hybride sort du bois
description	Ionic Framework a seulement un an mai...
speakers	8a4ed953-1b25-4c18-8d48-7dd33d48b...
... 9 more columns	

from gospeak, a minute ago

🔍 users: 8a4ed953-1b25-4c18-8d48-7dd33d48bea1

id	8a4ed953-1b25-4c18-8d48-7dd33d48bea1
slug	loicknuchel
status	Public
first_name	Loic
last_name	Knuchel
bio	I'm passionate about tech industry and I do a lo...
... 22 more columns	

from gospeak, a few seconds ago

Data explorer Beta Visual editor

Query editor



```

1 SELECT *
2 FROM "public"."talks";
3

```



Run query

#1 gospeak (100 rows) 674 ms

Search in results

SELECT * FROM "public"."talks" LIMIT 100;

#	id	slug	status	title
1	1048e75e-bc58-45bf-b80c-cbf488c27b2a	comment-perdre-la-confiance-du-commanditaire-d	Public	Comment perdre la confiance du command
2	c16724e2-9f68-459b-a35c-2b6281ebbbc0	le-developpement-mobile-hybride-sort-du-bois	Archived	Le développement mobile hybride sort du b
3	5c4ab32b-04dd-41ea-9902-361c3d81ec87	blog-tech-vaincre-le-syndrome-de-la-page-blanche	Public	Blog Tech: Vaincre le syndrome de la page b
4	c46f5ac4-aae9-4d29-99ed-453760631332	peut-on-garder-une-url-secrete-	Public	Peut-on garder une URL secrète ?
5	a5aee0f-5ba0-4907-a261-ce974739ac22	react-seo-un-mariage-impossible	Public	React & SEO : Un mariage impossible ?
6	d28802aa-f908-4481-a34e-1d5805929ec1	daily-english-speaking-practice	Public	Daily English speaking practice
7	7cf50a0c-75ee-4798-874e-ca0077360b4a	modular-coverage-tools-for-the-modern-web	Public	Modular coverage tools for the modern wet
8	c696575a-dbda-4c35-b56c-6767a436b68f	decouvrons-neo4j	Public	Découvrons Neo4J
9	1e1c803f-fe1b-4553-a0b9-d80233e6f233	pourquoi-detestez-vous-tester-	Public	Pourquoi détestez-vous tester ?
10	03f380a3-ddf1-442a-a1c9-612a05fbc06d	manipulation-des-pixels-d-un-canvas	Public	Manipulation des pixels d'un canvas : c'est b

Showing 1 to 10 of 100 results

1 2 3 ... 10

talks table details



- public
- talks

talks

Click to write notes

No tags

- Constraints
- Primary key: id (talks_pkey)
 - Unique: slug (talks_slug_key)
 - Index: status (talks_status_idx)

References

users

Referenced by external_proposals

talks: c16724e2-9f68-459b-a35c-2b6281ebbbc0

id	c16724e2-9f68-459b-a35c-2b6281ebbbc0
slug	le-developp
status	
title	Le dévelo
description	Ionic Framework a seulement un an mai...
speakers	8a4ed953-1b25-4c18-8d48-7dd33d48b...
... 9 more columns	

from gospeak, 3 minutes ago

users: 8a4ed953-1b25-4c18-8d48-7dd33d48bea1

id	8a4ed953-1b25-4c18-8d48-7dd33d48bea1
last_name	Knuchel
bio	I'm passionate about tech industry and I do a lo...
... 22 more columns	

from gospeak, 2 minutes ago

external_proposals (1) »

proposals (0)

See all

Le développement mobile hybride sort du bois (79ad5345-...)

Navigation icons: Home, Back, Forward, Refresh, Menu, Code, Edit, Mouse, Hand, Zoom (97%)

Data explorer Beta Visual editor Query editor

Run query

```

1 SELECT *
2 FROM "public"."talks";
3

```

#1 gospeak (100 rows) 674 ms

SELECT * FROM "public"."talks" LIMIT 100;

#	id	slug	status	title
1	1048e75e-bc58-45bf-b80c-cbf488c27b2a	comment-perdre-la-confiance-du-commanditaire-d	Public	Comment perdre la confiance du commandi
2	c16724e2-9f68-459b-a35c-2b6281ebbbc0	le-developpement-mobile-hybride-sort-du-bois	Archived	Le développement mobile hybride sort du b
3	5c4ab32b-04dd-41ea-9902-361c3d81ec87	blog-tech-vaincre-le-syndrome-de-la-page-blanche	Public	Blog Tech: Vaincre le syndrome de la page b
4	c46f5ac4-aae9-4d29-99ed-453760631332	peut-on-garder-une-url-secrete-	Public	Peut-on garder une URL secrète ?
5	a5aee0f-5ba0-4907-a261-ce974739ac22	react-seo-un-mariage-impossible	Public	React & SEO : Un mariage impossible ?
6	d28802aa-f908-4481-a34e-1d5805929ec1	daily-english-speaking-practice	Public	Daily English speaking practice
7	7cf50a0c-75ee-4798-874e-ca0077360b4a	modular-coverage-tools-for-the-modern-web	Public	Modular coverage tools for the modern wet
8	c696575a-dbda-4c35-b56c-6767a436b68f	decouvrons-neo4j	Public	Découvrons Neo4J
9	1e1c803f-fe1b-4553-a0b9-d80233e6f233	pourquoi-detestez-vous-tester-	Public	Pourquoi détestez-vous tester ?
10	03f380a3-ddf1-442a-a1c9-612a05fbc06d	manipulation-des-pixels-d-un-canvas	Public	Manipulation des pixels d'un canvas : c'est b

Showing 1 to 10 of 100 results



external_proposals: 79ad5345-9eac-425e-baa9-... ▾

id	79ad5345-9eac-425e-baa9-ea6619f34226
talk_id	c16724e2-9f68-459b-a35c-2b6281ebbbc0 @
event_id	c53fda84-6ada-43bd-9070-6afa4a84c039 @
title	Le développement mobile hybride sort du bois
speakers	8a4ed953-1b25-4c18-8d48-7dd33d48b... @
... 12 more columns	

from gospeak, 2 minutes ago

talks: c16724e2-9f68-459b-a35c-2b6281ebbbc0 ▾

id	c16724e2-9f68-459b-a35c-2b6281ebbbc0 *
slug	le-developpement-mobile-hybride-sort-du-bois
status	Archived
title	Le développement mobile hybride sort du bois
description	Ionic Framework a seulement un an mai...
speakers	8a4ed953-1b25-4c18-8d48-7dd33d48b... @
... 9 more columns	

from gospeak, 7 minutes ago

talks: f7cc7a0d-0f5a-4086-b3fb-60a9e9d3feb8 ▾

id	f7cc7a0d-0f5a-4086-b3fb-60a9e9d3feb8 *
slug	pourquoi-tout-soriente-vers-la-programmati...
status	Archived
title	Pourquoi tout s'oriente vers la programmation...
description	C'est généralement mieux quand on ne...
speakers	8a4ed953-1b25-4c18-8d48-7dd33d48b... @
... 9 more columns	

from gospeak, a few seconds ago

users: 8a4ed953-1b25-4c18-8d48-7dd33d48bea1 ▾

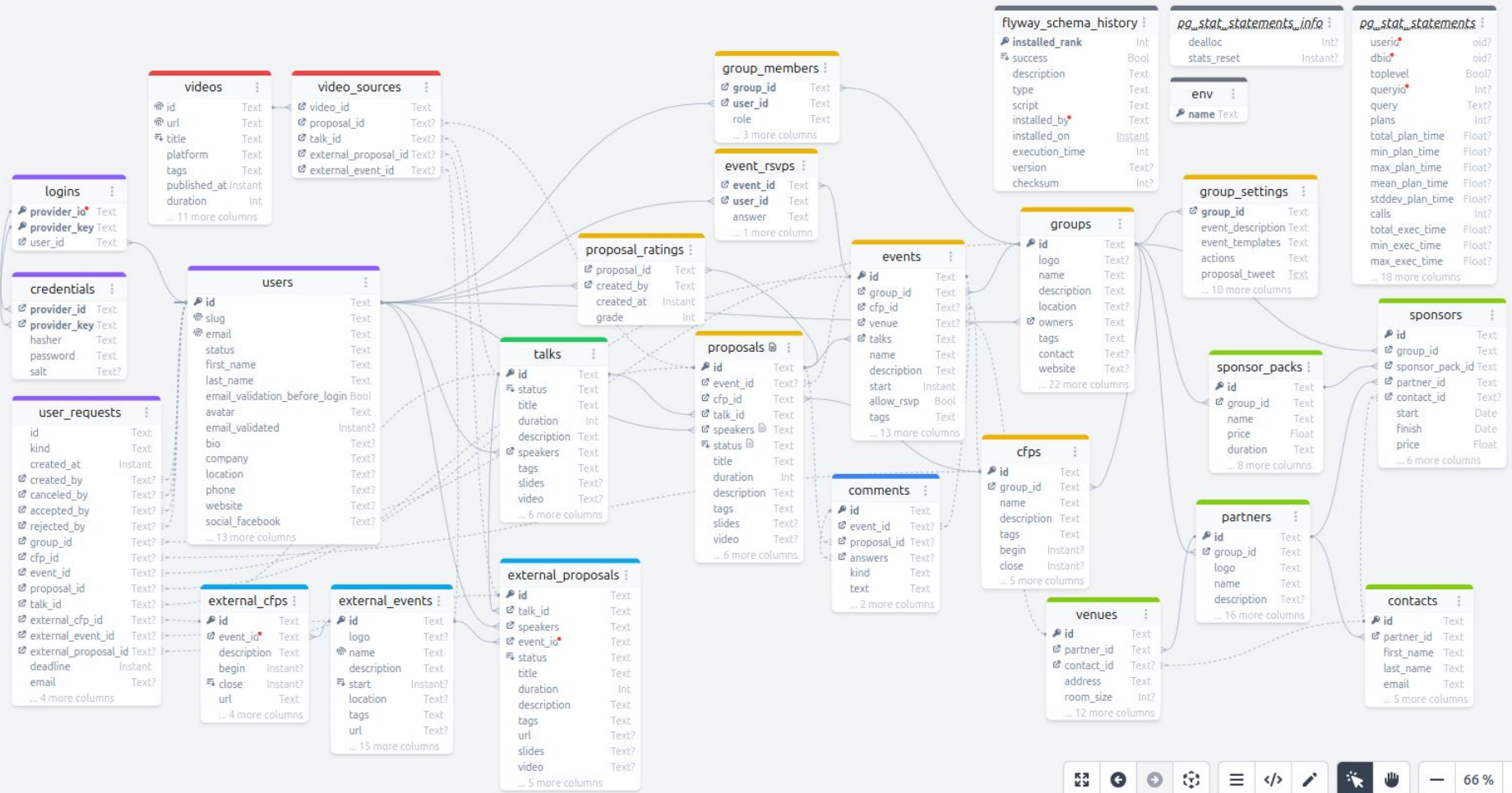
id	8a4ed953-1b25-4c18-8d48-7dd33d48bea1 *
slug	
status	
first_name	
last_name	
bio	I'm passionat
... 22 more columns	

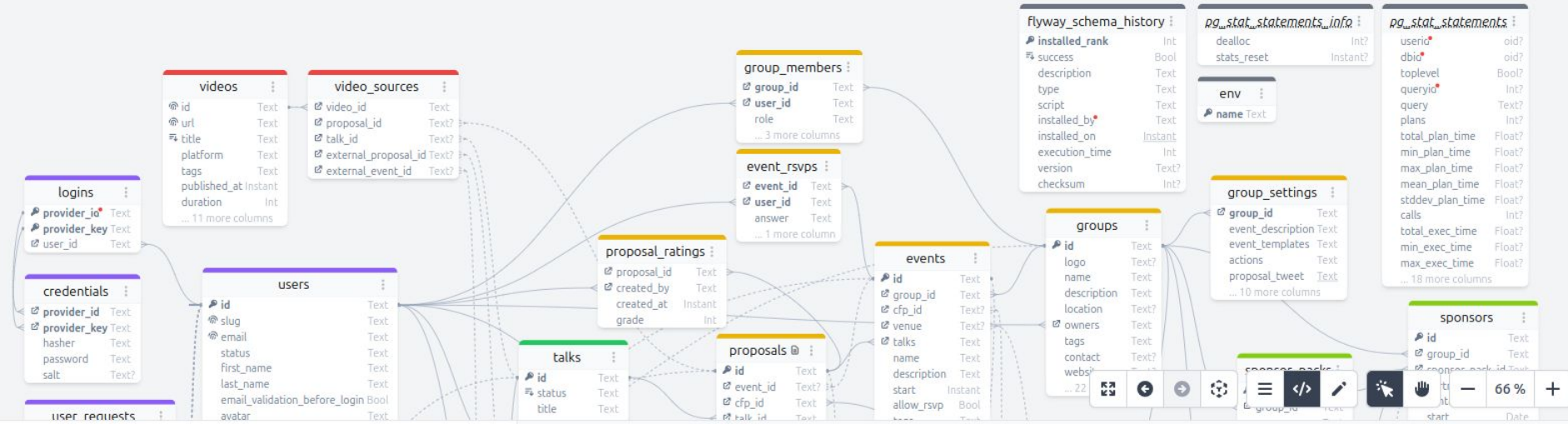
- cfps (1) »
- comments (20+) »
- contacts (20+) »
- event_rsmps (5) »
- events (20+) »
- external_cfps (20+) »
- external_events (20+) »
- external_proposals (20+) »
- group_members (2) »
- group_settings (3) »
- groups (2) »
- logins (5) »
- partners (20+) »
- proposals (20+) »
- sponsor_packs (6) »
- sponsors (20+) »
- talks (20+) »
- venues (20+) »

- HumanTalks Paris Avril 2015 (d80ce35e-...)
- HumanTalks Paris Décembre 2017 (d9bbc58b-...)
- HumanTalks Paris Février 2018 (09ef1e29-...)
- HumanTalks Paris Novembre 2018 (e86230d8-...)
- HumanTalks Paris Décembre 2018 (3f09bd56-...)
- HumanTalks Paris Janvier 2019 (8d6a4a0d-...)
- HumanTalks Paris Avril 2022 (a7b8df4e-...)
- HumanTalks Paris Octobre 2019 (79901028-...)
- HumanTalks Paris Avril 2013 (30c36697-...)
- HumanTalks Paris Juin 2013 (bf833cd1-...)
- HumanTalks Paris Mai 2013 (43fb5a58-...)
- HumanTalks Paris Septembre 2016 (874f22df-...)
- HumanTalks Paris Juillet 2016 (a02f2832-...)
- HumanTalks Paris Juin 2016 (906811a1-...)



Requêtes IA





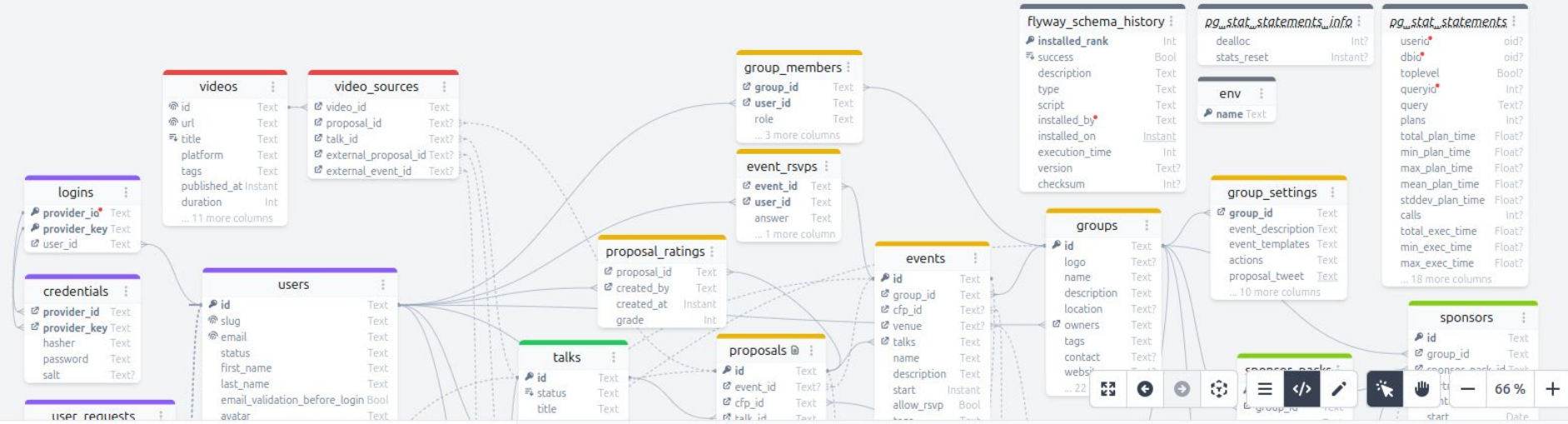
Navigation and zoom controls: back, forward, search, refresh, zoom in, zoom out, zoom reset, zoom 66%, zoom in (+)

Data explorer Beta Visual editor Query editor

```
1 |
```

Run query

Query results



Navigation icons: Grid, Home, Arrow, Refresh, Menu, Code, Edit, Mouse, Hand, Zoom (66%)

Data explorer Beta Visual editor Query editor

1

Generate SQL from text

Run query

Query results



Generate SQL Beta

Write in plain english the query you want, Azimutt will generate it for you.

What do you want?

Tip: for best results, describe precisely what you need. Ideally with exact table and column names, use schema exploration to find them.

Close



Generate SQL Beta

Write in plain english the query you want, Azimutt will generate it for you.

What do you want?

Compute the daily user creation for the year 2024, fill the missing days with 0

Tip: for best results, describe precisely what you need. Ideally with exact table and column names, use schema exploration to find them.

Close

Generate SQL



Generate SQL Beta

Write in plain english the query you want, Azimutt will generate it for you.

What do you want?

Compute the daily user creation for the year 2024, fill the missing days with 0

Tip: for best results, describe precisely what you need. Ideally with exact table and column names, use schema exploration to find them.

Generated SQL

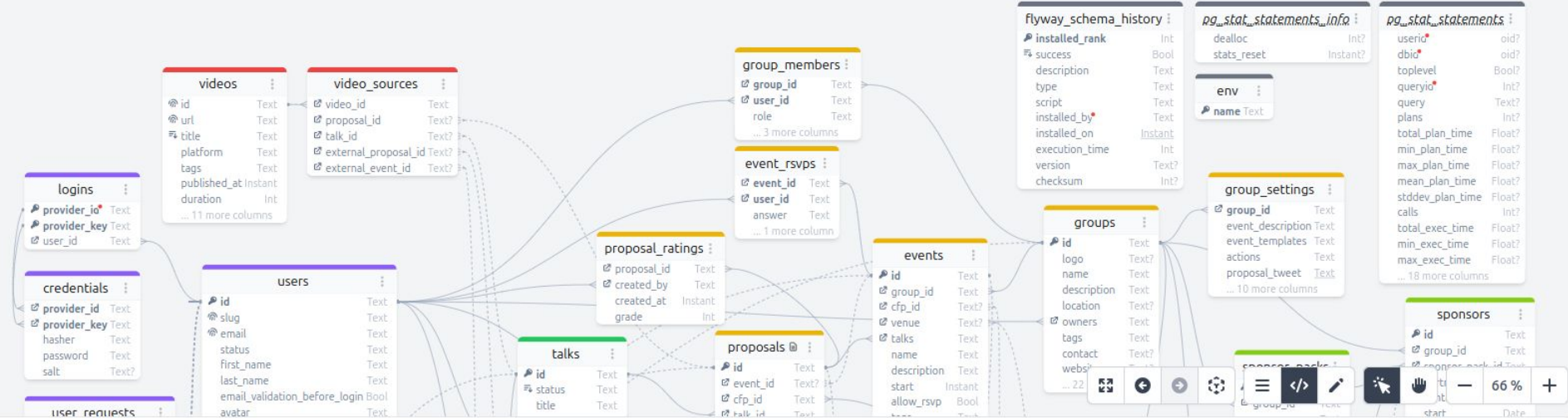
```
WITH recursive date_series AS (  
  SELECT  
    generate_series(  
      '2024-01-01'::date,  
      '2024-12-31'::date,  
      '1 day'::interval  
    )::date AS date  
)  
SELECT  
  ds.date,  
  COALESCE(COUNT(u.id), 0) AS user_creation_count  
FROM  
  date_series ds  
LEFT JOIN  
  public.users u ON ds.date = u.created_at::date  
GROUP BY  
  ds.date  
ORDER BY  
  ds.date;
```



Close

Generate SQL again

Execute SQL

Data explorer Beta Visual editor Query editor

```

1 WITH recursive date_series AS (
2   SELECT
3     generate_series(
4       '2024-01-01'::date,
5       '2024-12-31'::date,
6       '1 day'::interval
7     )::date AS date
8 )
9 SELECT
10  ds.date,
11  COALESCE(COUNT(u.id), 0) AS user_creation_count
12 FROM
13  date_series ds
14 LEFT JOIN
15  public.users u ON ds.date = u.created_at
16 GROUP BY

```

Run query

#1 gospeak (366 rows) 518 ms

WITH recursive date_series AS (SELECT generate_series('2024-01-01'::date, '2024-12-31'::date, '1 day'::interval)::date AS date) SELECT ds.date, COALESCE(COUNT(u.id), 0) AS user_cr...

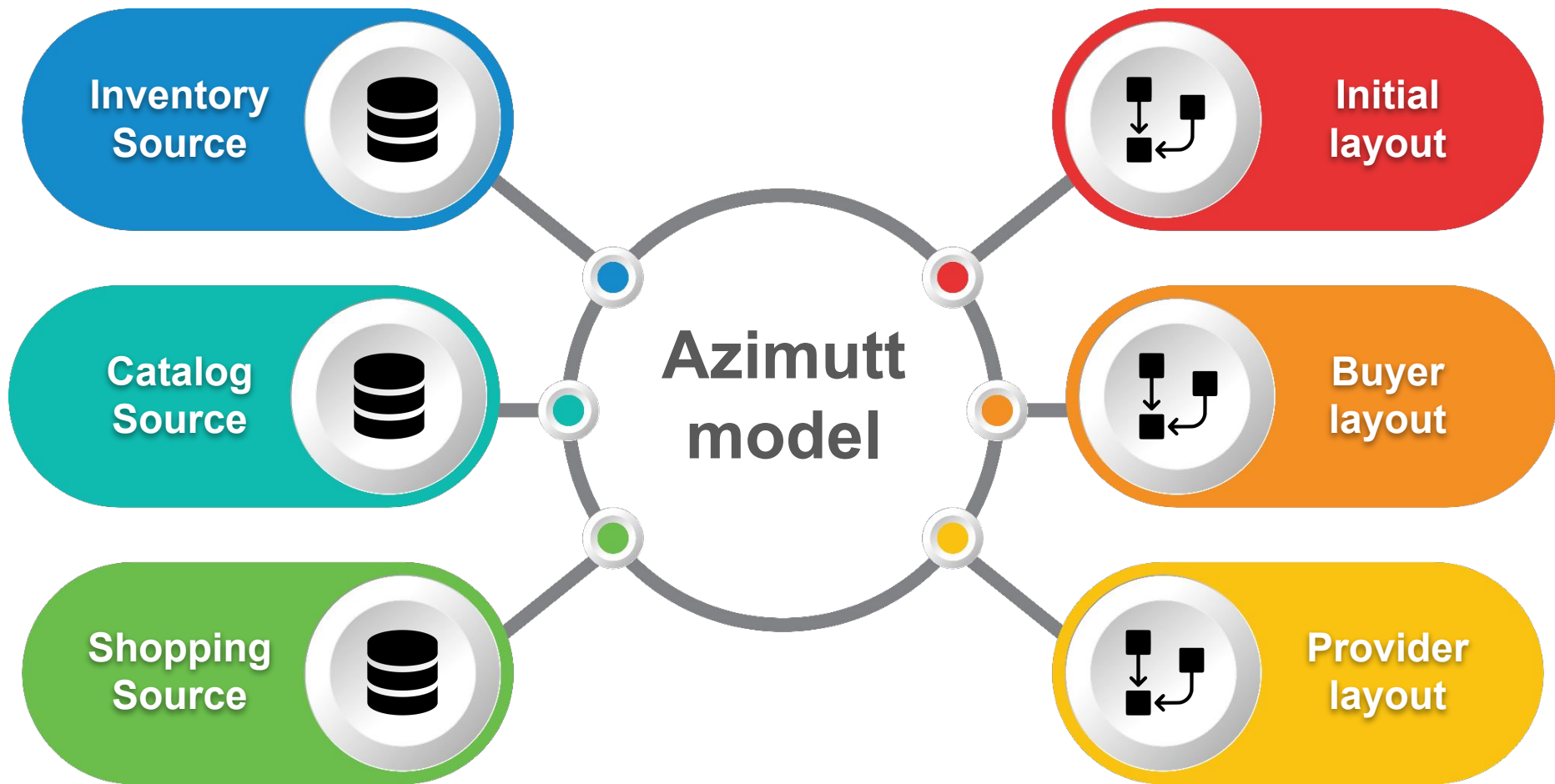
#	date	user_creation_count
1	2023-12-31T23:00:00.000Z	0
2	2024-01-01T23:00:00.000Z	0
3	2024-01-02T23:00:00.000Z	1
4	2024-01-03T23:00:00.000Z	1
5	2024-01-04T23:00:00.000Z	0
6	2024-01-05T23:00:00.000Z	0
7	2024-01-06T23:00:00.000Z	0
8	2024-01-07T23:00:00.000Z	1
9	2024-01-08T23:00:00.000Z	0
10	2024-01-09T23:00:00.000Z	0

Showing 1 to 10 of 366 results

1 2 3 ... 37



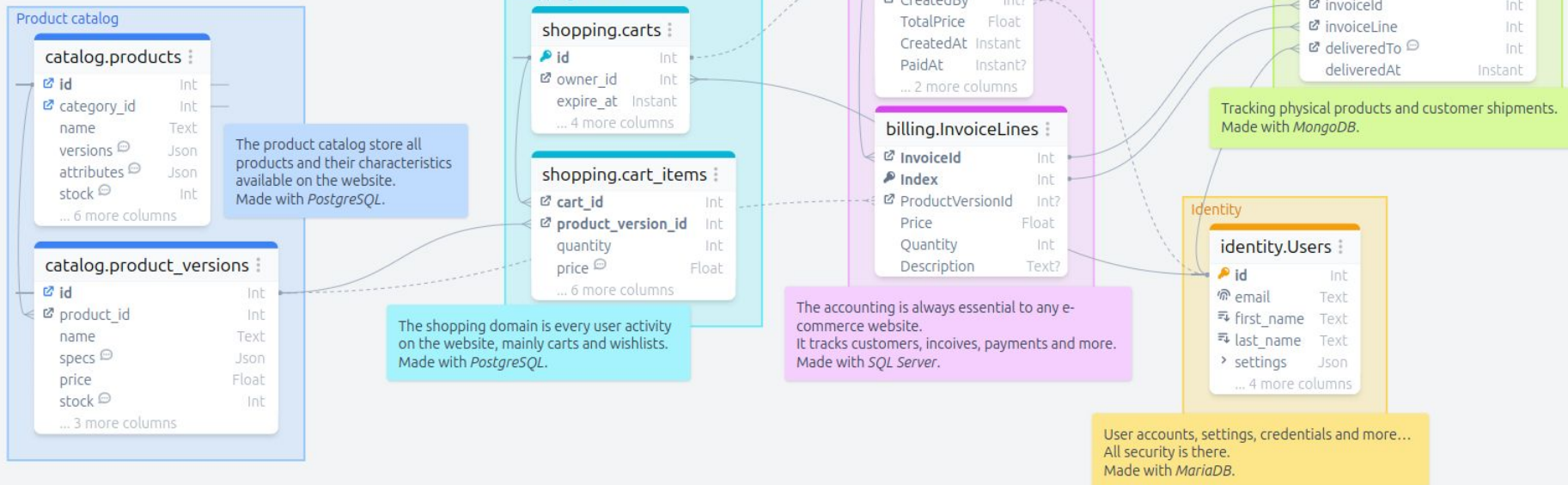
Base de données multiples



E-commerce databases example

Here is a medium sized example of a micro-services databases exploration with **Azimutt**. It's using PostgreSQL, MySQL, MariaDB, SQL Server, Oracle and even MongoDB over several domains, you can explore their schemas but also the data inside if you set them up.

Setup databases locally: github.com/azimuttapp/azimutt/tree/main/demos/ecommerce

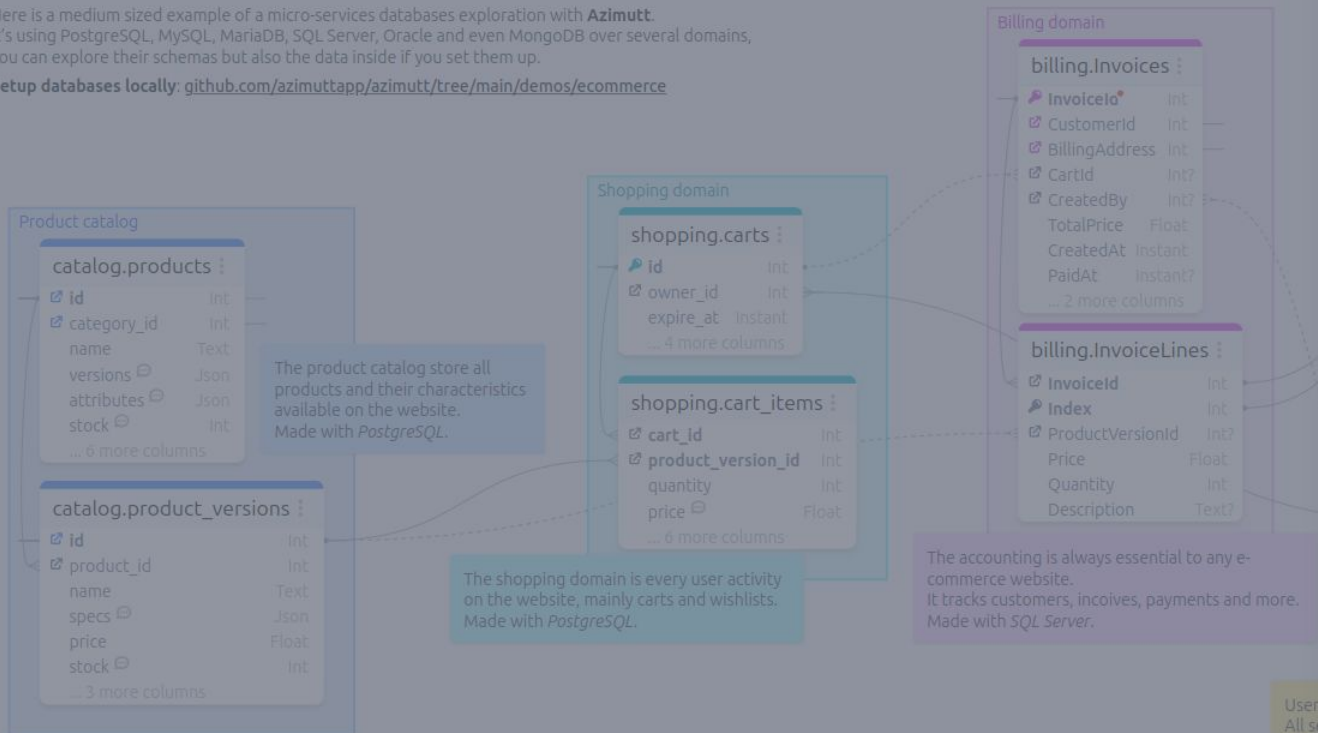


Explore the **initial layout data** to find how you can explore data seamlessly. Then look at **tutorials** layouts or dig directly in **domain** or **use case** ones...

E-commerce databases example

Here is a medium sized example of a micro-services databases exploration with **Azimutt**. It's using PostgreSQL, MySQL, MariaDB, SQL Server, Oracle and even MongoDB over several domains, you can explore their schemas but also the data inside if you set them up.

Setup databases locally: github.com/azimuttapp/azimutt/tree/main/demos/ecommerce



Explore the **initial layout data** to find how you can explore data seamlessly. Then look at **tutorials** layouts or dig directly in **domain** or **use case** ones...

Project settings

Project sources

Active sources are merged to create your current schema.

<input checked="" type="checkbox"/>	design	85 tables, 0 view & 238 relations		8 Nov 2024
<input checked="" type="checkbox"/>	Referential project	3 tables, 0 view & 2 relations		24 Sep 2024
<input checked="" type="checkbox"/>	Identity project	7 tables, 0 view & 7 relations		24 Sep 2024
<input checked="" type="checkbox"/>	Inventory project	25 tables, 0 view & 44 relations		24 Sep 2024
<input checked="" type="checkbox"/>	Catalog project	12 tables, 0 view & 18 relations		24 Sep 2024
<input checked="" type="checkbox"/>	Shopping project	13 tables, 0 view & 9 relations		8 Nov 2024
<input checked="" type="checkbox"/>	Billing project	7 tables, 0 view & 8 relations		24 Sep 2024
<input checked="" type="checkbox"/>	Shipping project	3 tables, 0 view & 0 relation		24 Sep 2024
<input checked="" type="checkbox"/>	CRM project	11 tables, 0 view & 8 relations		24 Sep 2024
<input checked="" type="checkbox"/>	Analytics project	2 tables, 0 view & 0 relation		24 Sep 2024
<input checked="" type="checkbox"/>	additional relations	0 table, 0 view & 138 relations		8 Nov 2024

+ Add source

Project schemas

Allow you to enable or not SQL schemas in your project.

E-commerce databases example

Here is a medium sized example of a micro-services databases exploration with **Azimutt**.

It's using PostgreSQL, MySQL, MariaDB, SQL Server, Oracle and even MongoDB over several domains, you can explore their schemas but also the data inside if you set them up.

Setup databases locally: github.com/azimuttapp/azimutt

Product catalog

catalog.products

- id
- category_id
- name
- versions
- attributes
- stock
- price

The product catalog products and their attributes are available on the way. Made with PostgreSQL.

catalog.product_versions

- id
- product_id
- name
- specs
- price
- stock

Add a source

A project can have several sources. They are independent and merged together when enabled to build the usable schema. It's a great way to explore multiple database at once or create isolated schema evolutions.

Database SQL Prisma JSON AML



PostgreSQL



MySQL



MariaDB



Microsoft SQL Server



ORACLE DATABASE



Couchbase



mongoDB



snowflake



Azimutt



SQLite

Database url

ex: postgres://<user>:<pass>@<host>:<port>/<db>

Store in browser

Sample PostgreSQL url: postgres://<user>:<pass>@<host>:<port>/<db>



Access databases from your computer using [Azimutt CLI](#) (install `npm` & run `npm azimutt@latest gateway`), otherwise Azimutt will use the hosted gateway to connect. For security prefer to use a *read-only user* and on a *non-production database*.

Close

Project settings

Project sources

Active sources are merged to create your current schema.

<input checked="" type="checkbox"/>	design	85 tables, 0 view & 238 relations	8 Nov 2024
<input checked="" type="checkbox"/>	Referential project	tables, 0 view & 2 relations	24 Sep 2024
<input type="checkbox"/>	Identity project	tables, 0 view & 7 relations	24 Sep 2024
<input type="checkbox"/>	Inventory project	tables, 0 view & 44 relations	24 Sep 2024
<input type="checkbox"/>	Catalog project	tables, 0 view & 18 relations	24 Sep 2024
<input type="checkbox"/>	Shopping project	tables, 0 view & 9 relations	8 Nov 2024
<input type="checkbox"/>	Billing project	tables, 0 view & 8 relations	24 Sep 2024
<input type="checkbox"/>	Shipping project	tables, 0 view & 0 relation	24 Sep 2024
<input type="checkbox"/>	CRM project	tables, 0 view & 8 relations	24 Sep 2024
<input checked="" type="checkbox"/>	Analytics project	2 tables, 0 view & 0 relation	24 Sep 2024
<input checked="" type="checkbox"/>	additional relations	0 table, 0 view & 138 relations	8 Nov 2024

+ Add source

Project schemas

Allow you to enable or not SQL schemas in your project.

E-commerce databases example

Here is a medium sized example of a micro-services databases exploration with **Azimutt**. It's using PostgreSQL, MySQL, MariaDB, SQL Server, Oracle and even MongoDB over several domains, you can explore their schemas but also the data inside if you set them up.

Setup databases locally: github.com/azimuttapp/azimutt/tree/main/demos/ecommerce

Product catalog

catalog.products ::

id	Int
category_id	Int
name	Text
versions	Json
attributes	Json

The product catalog store all products and their characteristics

Shopping domain

shopping.carts ::

id	Int
owner_id	Int
expire_at	Instant
...	4 more columns

shopping.cart.items ::

Billing domain

billing.Invoices ::

InvoiceId	Int
CustomerId	Int
BillingAddress	Int
CartId	Int?
CreatedBy	Int?
TotalPrice	Float
CreatedAt	Instant
PaidAt	Instant?
...	2 more columns

billing.InvoiceLines ::

InvoiceId	Int
-----------	-----

Shipment tracking

shipping.Shipments ::

_id	Objectid
id	Int
...	10 more columns

shipping.ShipmentItems ::

_id	Objectid
shipmentId	Int
physicalProductId	Int
invoiceId	Int
invoiceLine	Int
deliveredTo	Int
deliveredAt	Instant

Tracking physical products and customer shipments.

Data explorer Beta Visual editor Query editor ⌵ ⌶

Referential

-- select a source

Referential

- Identity
- Inventory
- Catalog
- Shopping
- Billing
- Shipping
- CRM
- Analytics



Run query

Query results

catalog.products: 1	
id	1
name	Pixel 8
category_id	3
versions	[[{"key": "color", "label": "Color", "values": [{"n...}]]
attributes	[[{"key": "Brand", "value": "Google"}, {"key": ...}]]
stock	11
... 6 more columns	
from Catalog, 5 months ago	

shopping.carts: 1	
id	1
owner_id	102
expire_at	2024-08-26T12:44:27.126Z
... 4 more columns	
from Shopping, 5 months ago	

billing.Invoices: 1	
InvoiceId	1
CartId	1
CustomerId	1
BillingAddress	1
TotalPrice	1323
PaidAt	null
CreatedAt	2024-08-25T14:44:53.087Z
CreatedBy	102
... 2 more columns	
from Billing, 5 months ago	

identity.Users: 102	
id	102
first_name	SpongeBob
last_name	SquarePants
email	spongebob@bikinibottom.com
settings	{language: "en", theme: "light"}
... 4 more columns	
from Identity, 5 months ago	

shopping.cart_items: 1/1	
cart_id	1
product_version_id	1

shopping.cart_items: 1/2	
cart_id	1
product_version_id	2

shopping.cart_items: 1/15	
cart_id	1
product_version_id	15

shopping.cart_items: 1/20	
cart_id	1
product_version_id	20
quantity	1
price	30
... 6 more columns	
from Shopping, 5 months ago	

billing.InvoiceLines: 1/1	
InvoiceId	1
Index	1
ProductVersionId	1
Description	Pixel 8 Obsidian 128 Go
Price	599
Quantity	1
from Billing, 5 months ago	

shipping.Shipments: 66cb17a0fdd0405e567c193d	
_id	66cb17a0fdd0405e567c193d
id	1
... 10 more columns	
from Shipping, 5 months ago	

shipping.ShipmentItems: 66cb17a3fdd0405e567c193f	
_id	66cb17a3fdd0405e567c193f
shipmentId	1
physicalProductId	1
invoiceId	1
invoiceLine	1
deliveredAt	2024-08-25T14:46:12.309Z
deliveredTo	102
from Shipping, 5 months ago	

Display and navigate table rows across several databases 🐾



Database design



The easiest DSL for database schemas

If you ever designed a database schema on a whiteboard, AML is made for you ❤️. It's **fast to learn and write**, and can be translated to other dialects.

[Documentation](#)[Get on npm →](#)

blog.md

blog.sql

blog.json

```
# sample AML
```

```
users
```

```
id uuid pk
name varchar
email varchar unique
role user_role(admin, guest)=guest
```

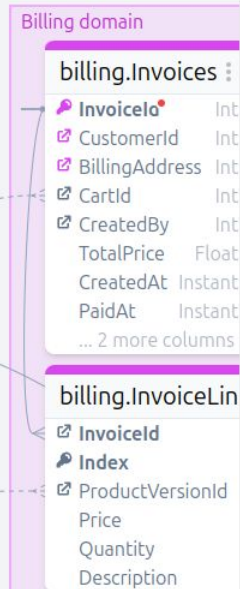
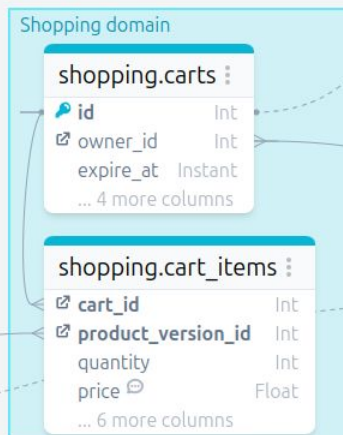
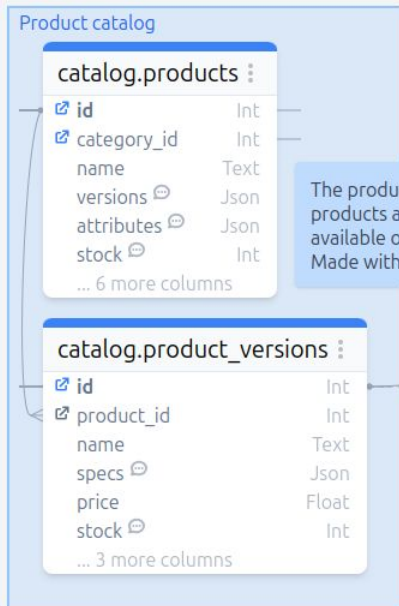
```
posts
```

```
id uuid pk
title varchar
content text | in markdown
created_at timestamp='now()'
created_by uuid -> users(id) # inline relation
```

E-commerce databases example

Here is a medium sized example of a micro-services databases exploration with **Azimutt**. It's using PostgreSQL, MySQL, MariaDB, SQL Server, Oracle and even MongoDB over several domains, you can explore their schemas but also the data inside if you set them up.

Setup databases locally: github.com/azimuttapp/azimutt/tree/main/demos/ecommerce



Update your schema

In Azimutt your schema is the union of all active sources. Create or update one with [AML syntax](#) to extend it.

design

New source

Shopping

```
shopping.carts
id bigint pk
owner_kind cart_owner("identity.Devices", "identity.Users")
owner_id bigint
expire_at timestamp
created_at timestamp
updated_at timestamp
deleted_at timestamp nullable index
```

```
rel shopping.carts(owner_id) -> identity.Devices(id)
rel shopping.carts(owner_id) -> identity.Users(id)
```

```
shopping.cart_items
cart_id bigint pk -> shopping.carts(id)
product_version_id bigint pk -> catalog.product_versions(id)
quantity int
price double
created_at timestamp
created_by bigint nullable -> identity.Users(id)
updated_at timestamp
updated_by bigint nullable -> identity.Users(id)
deleted_at timestamp nullable index
deleted_by bigint nullable -> identity.Users(id)
```

```
shopping.wishlists
id bigint pk
name varchar
description text nullable
```

Write your database schema using [AML syntax](#). You can also use the [VS Code extension](#) for best editing experience.

AML to PostgreSQL converter

```
492 # Shopping
493
494 shopping.carts
495   id bigint pk
496   owner_kind cart_owner("identity.Devices", "identity.Users") | Devices are used
497   owner_id bigint
498   expire_at timestamp
499   created_at timestamp
500   updated_at timestamp
501   deleted_at timestamp nullable index
502
503 rel shopping.carts(owner_id) -> identity.Devices(id)
504 rel shopping.carts(owner_id) -> identity.Users(id)
505
506 shopping.cart_items
507   cart_id bigint pk -> shopping.carts(id)
508   product_version_id bigint pk -> catalog.product_versions(id)
509   quantity int
510   price double | at the time the product was added to the card, prevent price c
511   created_at timestamp
512   created_by bigint nullable -> identity.Users(id)
513   updated_at timestamp
514   updated_by bigint nullable -> identity.Users(id)
515   deleted_at timestamp nullable index
516   deleted_by bigint nullable -> identity.Users(id)
517
518 shopping.wishlists
519   id bigint pk
520   name varchar
521   description text nullable
522   public boolean
523   created_at timestamp
524   created_by bigint -> identity.Users(id)
525   updated_at timestamp
```

Open in Azimutt

AML

A very simple language made by Azimutt to ease and speed database design. In addition, you can use [AML](#) inside Azimutt or as a [standalone library](#) for your own projects.

```
759 -- Shopping
760
761 CREATE TABLE shopping.carts (
762   id bigint PRIMARY KEY,
763   owner_kind cart_owner NOT NULL,
764   owner_id bigint NOT NULL, -- references: identity.Devices.id or identity.Users
765   expire_at timestamp NOT NULL,
766   created_at timestamp NOT NULL,
767   updated_at timestamp NOT NULL,
768   deleted_at timestamp
769 );
770 CREATE INDEX carts_deleted_at_idx ON shopping.carts(deleted_at);
771 COMMENT ON COLUMN shopping.carts.owner_kind IS 'Devices are used for anonymous c
772
773 CREATE TABLE shopping.cart_items (
774   cart_id bigint REFERENCES shopping.carts(id),
775   product_version_id bigint REFERENCES catalog.product_versions(id),
776   quantity int NOT NULL,
777   price double NOT NULL,
778   created_at timestamp NOT NULL,
779   created_by bigint REFERENCES identity.Users(id),
780   updated_at timestamp NOT NULL,
781   updated_by bigint REFERENCES identity.Users(id),
782   deleted_at timestamp,
783   deleted_by bigint REFERENCES identity.Users(id),
784   PRIMARY KEY (cart_id, product_version_id)
785 );
786 CREATE INDEX cart_items_deleted_at_idx ON shopping.cart_items(deleted_at);
787 COMMENT ON COLUMN shopping.cart_items.price IS 'at the time the product was add
788
789 CREATE TABLE shopping.wishlists (
790   id bigint PRIMARY KEY,
791   name varchar NOT NULL,
792   description text
```

PostgreSQL

The world's most advanced open source relational database. Use the [DDL SQL dialect of PostgreSQL](#) to design your database schema and convert it to the other ones.



AML - Fast database design (ERD)

Azimutt | 📄 76 installs | ★★★★★ (1) | Free

Design database schema and view Entity-Relationship Diagram using AML language in your VS Code editor. Convert to SQL, JSON and other languages.

Installation

Launch VS Code Quick Open (Ctrl+P), paste the following command, and press enter.

```
ext install azimutt.vscode-aml
```

Copy | [More Info](#)

[Overview](#) | [Version History](#) | [Q & A](#) | [Rating & Review](#)

AML Support for VS Code

Visual Studio Marketplace **v0.1.2** [Stars](#) | [1.5k](#) [Follow @azimuttapp](#) [Tweet](#)

A VS Code extension to design database schemas using [AML](#), a simple DSL that speed your design by 2x 🌟

```
[Extension Development Host] schema.aml - resources - Visual Studio Code
File Edit Selection View Go Run Terminal Help
EXPLORER
RESOURCES
OUTLINE
TIMELINE
schema.aml
1 type muth kind (password, google, twitter, github)
2
3 users as u (color: blue, tags: [pii])
4 id int pk (autoIncrement)
5 name varchar(64) index: user_name_idx
6 email varchar(256) unique
7 muth muth kind
8 settings json nullable
9 theme theme(light, dark)
10 last_page string | url of the last visited page
11 created_at timestamp=now()
12 updated_at timestamp=now()
13 deleted_at timestamp nullable
14
15 # CMS tables
16
17 cms_posts
18 id int pk (autoIncrement)
19 title varchar check (length(title) > 10)
20 content text | allow markdown
21 status post status(draft, published, archived)=draft
22 author int -> users(id)
23 tags 'varchar[]'
24 created_at timestamp=now()
25
26 created by -> users(id)
27 updated_at timestamp=now()
28 updated by -> users(id)
```

Categories

[Programming Languages](#) [Snippets](#) [Visualization](#)

Tags

[aml](#) [Azimutt Markup Language](#) [database](#) [diagram](#)
[entity](#) [erd](#) [relationship](#) [snippet](#) [sql](#) [uml](#)

Works with

Universal, Web

Resources

[Issues](#)
[Repository](#)
[Homepage](#)
[License](#)
[Changelog](#)
[Download Extension](#)

Project Details

[azimuttapp/azimutt](#)
Last Commit: a month ago
18 Pull Requests

File Edit Selection View Go Run Terminal Help

EXPLORER

- AZIMUTT
 - .github
 - .idea
 - .vscode
 - assets
 - backend
 - charts
 - cli
 - demos / ecommerce
 - README.md
 - source_00_design.aml
 - source_01_referential_sqlserver.sql
 - source_02_identity_mariadb.sql
 - source_03_inventory_oracle.sql
 - source_04_catalog_postgres.sql
 - source_05_shopping_postgres.sql
 - source_06_billing_sqlserver.sql
 - source_07_shipping_mongo.sql
 - source_08_crm_mysql.sql
 - source_09_analytics_mongo.sql
 - source_10_additional_relations.md
 - extensions
 - frontend
 - gateway
 - libs
 - local
 - node_modules
 - .dockerignore
 - .env
 - .env.example
 - .gitignore
 - .pre-commit-config.yaml
 - OUTLINE
 - TIMELINE

source_00_design.aml x

```

demos > ecommerce > source_00_design.aml > ...
485     deleted_by bigint nullable -> identity.Users(id)
486
487     # Shopping
488
489     shopping.carts
490     id bigint pk
491     owner_kind cart_owner("identity.Devices", "identity.Users") | Devices are used for anonymous carts, otherwise it's Users
492     owner_id bigint
493     expire_at timestamp
494     created_at timestamp
495     updated_at timestamp
496     deleted_at timestamp nullable index
497
498     rel shopping.carts(owner_id) -> identity.Devices(id)
499     rel shopping.carts(owner_id) -> identity.Users(id)
500
501     shopping.cart_items
502     cart_id bigint pk -> shopping.carts(id)
503     product_version_id bigint pk -> catalog.product_versions(id)
504     quantity int
505     price double | at the time the product was added to the card, prevent price changes after a product has been added to a cart
506     created_at timestamp
507     created_by bigint nullable -> identity.Users(id)
508     updated_at timestamp
509     updated_by bigint nullable -> identity.Users(id)
510     deleted_at timestamp nullable index
511     deleted_by bigint nullable -> identity.Users(id)
512
513     shopping.wishlists
514     id bigint pk
515     name varchar
516     description text nullable
517     public boolean
518     created_at timestamp
519     created_by bigint -> identity.Users(id)
520     updated_at timestamp
521     updated_by bigint -> identity.Users(id)
522     deleted_at timestamp nullable index
523     deleted_by bigint nullable -> identity.Users(id)

```

Ln 497, Col 1 Spaces: 2 UTF-8 LF AML

File Edit Selection View Go Run Terminal Help

EXPLORER

- AZIMUTT
 - .github
 - .idea
 - .vscode
 - assets
 - backend
 - charts
 - cli
 - demos/ecommerce
 - README.md
 - source_00_design.aml 5, U
 - source_01_referential_sqlserver.sql
 - source_02_identity_mariadb.sql
 - source_03_inventory_oracle.sql
 - source_04_catalog_postgres.sql
 - source_05_shopping_postgres.sql
 - source_06_billing_sqlserver.sql
 - source_07_shipping_mongo.sql
 - source_08_crm_mysql.sql
 - source_09_analytics_mongo.sql
 - source_10_additional_relations.md
 - extensions
 - frontend
 - gateway
 - libs
 - local
 - node_modules
 - .dockerignore
 - .env
 - .env.example
 - .gitignore
 - .pre-commit-config.yaml
- OUTLINE
- TIMELINE

demos > ecommerce > source_00_design.aml > ...

```

485     deleted_by bigint nullable -> identity.Users(id)
486
487 # Shopping
488
489 shopping.carts
490     id bigint pk
491     owner_kind cart_owner("identity.Devices", "identity.Users") | Devices are used for anonymous carts, otherwise it's Users
492     owner_id bigint
493     expire_at timestamp
494     created_at timestamp
495     updated_at timestamp
496     deleted_at timestamp nullable index
497
498 rel shopping.carts(owner_id) -> identity.Devices(id)
499 rel shopping.carts(owner_id) -> identity.Devices(id)
500
501 shopping.cart_items
502     cart_id bigint pk -> shopping.carts(id)
503     product_version_id bigint pk -> referential.Cities(CityId)
504     quantity int
505     price double | at the time the product was added to a cart
506     created_at timestamp
507     created_by bigint nullable -> identity.Users(id)
508     updated_at timestamp
509     updated_by bigint nullable -> identity.Users(id)
510     deleted_at timestamp nullable
511     deleted_by bigint nullable -> identity.Users(id)
512
513 shopping.wishlists
514     id bigint pk
515     name varchar
516     description text nullable
517     public boolean
518     created_at timestamp
519     created_by bigint -> identity.Users(id)
520     updated_at timestamp
521     updated_by bigint -> identity.Users(id)
522     deleted_at timestamp nullable index
523     deleted_by bigint nullable -> identity.Users(id)

```

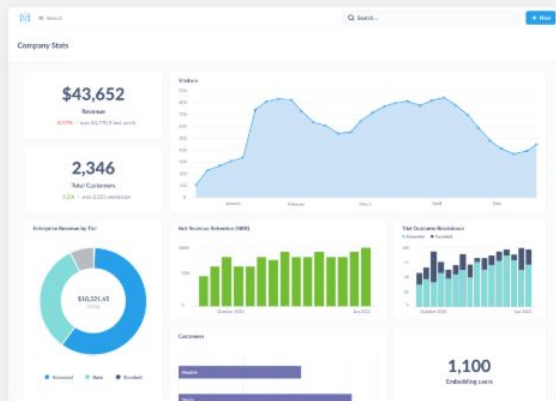


Analyse de base de données



Metabase

Fast analytics with the friendly UX and integrated tooling to let your company explore data on their own.



core_user	
id	Int
email	Text
first_name	Text?
last_name	Text?
settings	Text?
...	15 more columns

collection	
id	Int
personal_owner_id	Int?
name	Text
type	Text?
...	8 more columns

report_dashboard	
id	Int
creator_id	Int
collection_id	Int?
name	Text
...	19 more columns

dashboard_tab	
id	Int
dashboard_id	Int
name	Text
position	Int
...	3 more columns

report_dashboardcard	
id	Int
dashboard_id	Int
card_id	Int?
dashboard_tab_id	Int?
row	Int
col	Int
size_x	Int
size_y	Int
visualization_settings	Text
parameter_mappings	Text
...	4 more columns

report_card	
id	Int
name	Text
dataset_query	Text
display	Text
visualization_settings	Text
result_metadata	Text?
database_id	Int
table_id	Int?
...	21 more columns

Dashboards

They are the **main feature** of Metabase, they have saved queries (saved in `report_card`) associated with visualization and option settings.

Database

metabase_database	
id	Int
name	Text
engine	Text
details	Text
metadata_sync_schedule	Text
cache_field_values_schedule	Text
dbms_version	Text?
...	16 more columns

metabase_table	
id	Int
db_id	Int
schema	Text?
name	Text
display_name	Text?
entity_type	Text?
...	12 more columns

metabase_field	
id	Int
table_id	Int
parent_id	Int?
name	Text
display_name	Text?
base_type	Text
database_type	Text
semantic_type	Text?
last_analyzed	Instant?
...	24 more columns

metabase_fieldvalues	
id	Int
field_id	Int
type	Text
values	Text?
...	6 more columns

table_privileges	
table_id	Int
role	Text?
select	Bool
update	Bool
insert	Bool
delete	Bool

Databases

This is used to store the data-sources schema and data analysis.



Schema analysis

Let's find out if you can make improvements on your schema...

- 💡 Found 3 tables without a primary key ▼
- 💡 Found 99 potentially missing relations ▼
- ✅ No relation with different types found
- 💡 Found 23 columns with heterogeneous types ▼
- 💡 Found 2 tables too big ▼
- 💡 Found 15 tables without index ▼
- 💡 Found 21 relations without index ▼
- 💡 Found 14 duplicated indexes ▲
 - In table **core_user**, index `idx_user_full_name` is included in `idx_user_qualified_id`
 - In table **core_user**, index `idx_lower_email` is included in `idx_user_full_name`, `idx_user_qualified_id`
 - In table **login_history**, index `idx_user_id` is included in `idx_user_id_timestamp`
 - In table **login_history**, index `idx_session_id` is included in `idx_user_id_device_id`
 - In table **metabase_field**, index `idx_field_entity_qualified_id` is included in `idx_field_name_lower`
 - In table **metabase_table**, index `idx_table_db_id` is included in `idx_metabase_table_db_id_schema`
 - In table **permissions**, index `idx_permissions_group_id` is included in `idx_permissions_group_id_object`
 - In table **permissions_group_membership**, index `idx_permissions_group_membership_group_id` is included in `idx_permissions_group_membership_group_id_user_id`
 - In table **qrtz_fired_triggers**, index `idx_qrtz_ft_trig_inst_name` is included in `idx_qrtz_ft_inst_job_req_rcvry`
 - In table **qrtz_triggers**, index `idx_qrtz_t_nft_misfire` is included in `idx_qrtz_t_nft_st_misfire`, `idx_qrtz_t_nft_st_misfire_grp`
 - In table **qrtz_triggers**, index `idx_qrtz_t_state` is included in `idx_qrtz_t_nft_st`
 - In table **qrtz_triggers**, index `idx_qrtz_t_g` is included in `idx_qrtz_t_n_g_state`
 - In table **query_execution**, index `idx_query_execution_card_id` is included in `idx_query_execution_card_id_started_at`
 - In table **timeline_event**, index `idx_timeline_event_timeline_id` is included in `idx_timeline_event_timeline_id_timestamp`
- ✅ All tables have consistent naming case

If you've got any ideas for improvements, [please let us know](#).

Close

Azimutt.app

Version 0.1.21

```
0 entity with too many indexes
0 entity with too heavy indexes
11 business primary key forbidden
- Entity public.qrtz_calendars
- Entity public.qrtz_fired_trig
- Entity public.qrtz_job_detail
... 8 more
0 index on relation (1 ignore)
33 missing relation (1 ignore):
- Create a relation from public
- Create a relation from public
- Create a relation from public
... 30 more
50 high violations (9 rules):
33 duplicated index (1 ignore):
- Index idx_bookmark_ordering_u
- Index idx_card_bookmark_user_
- Index idx_card_label_card_id
... 30 more
0 too slow query
0 degrading query
15 entity not clean:
- Entity public.databasechangel
- Entity public.metabase_field
- Entity public.metabase_fieldv
... 12 more
2 missing primary key (1 ignore):
- Entity public.model_index_val
- Entity public.table_privilege
0 entity without index
0 misaligned relation
0 attribute not found in relation
0 entity not found in relation
```

```
Found 91 entities, 100 relations, 6
Found 271 violations using 27 rules
```

```
Analysis report written to ~/.azimutt/analyze/metabase/report_2024-07-17T10-43-48-600Z.azimutt.json
```

```
Thanks for using Azimutt analyze!
For any question or suggestion, reach out to contact@azimutt.app.
Cheers!
```

● Schéma

- Table sans clé primaire
- Index dupliqué
- Relation manquante
- Relation avec des types différents
- Colonne avec un mauvais type de donnée
- Conventions de nommage

● Snapshot

- Index trop lourd
- Table/Index non utilisé, colonne vide
- Analyze/Vacuum vieux, trop de dead rows
- Requête trop lente, coûteuse ou avec un écart type élevé

● Trend

- Table/Index qui grossit vite
- Requête qui se dégrade

```
er_id, ordering), unique_bookmark_u
d, card_id).
_id).
```

27 règles métier, 30 de plus prévues, + l'ajout d'extension



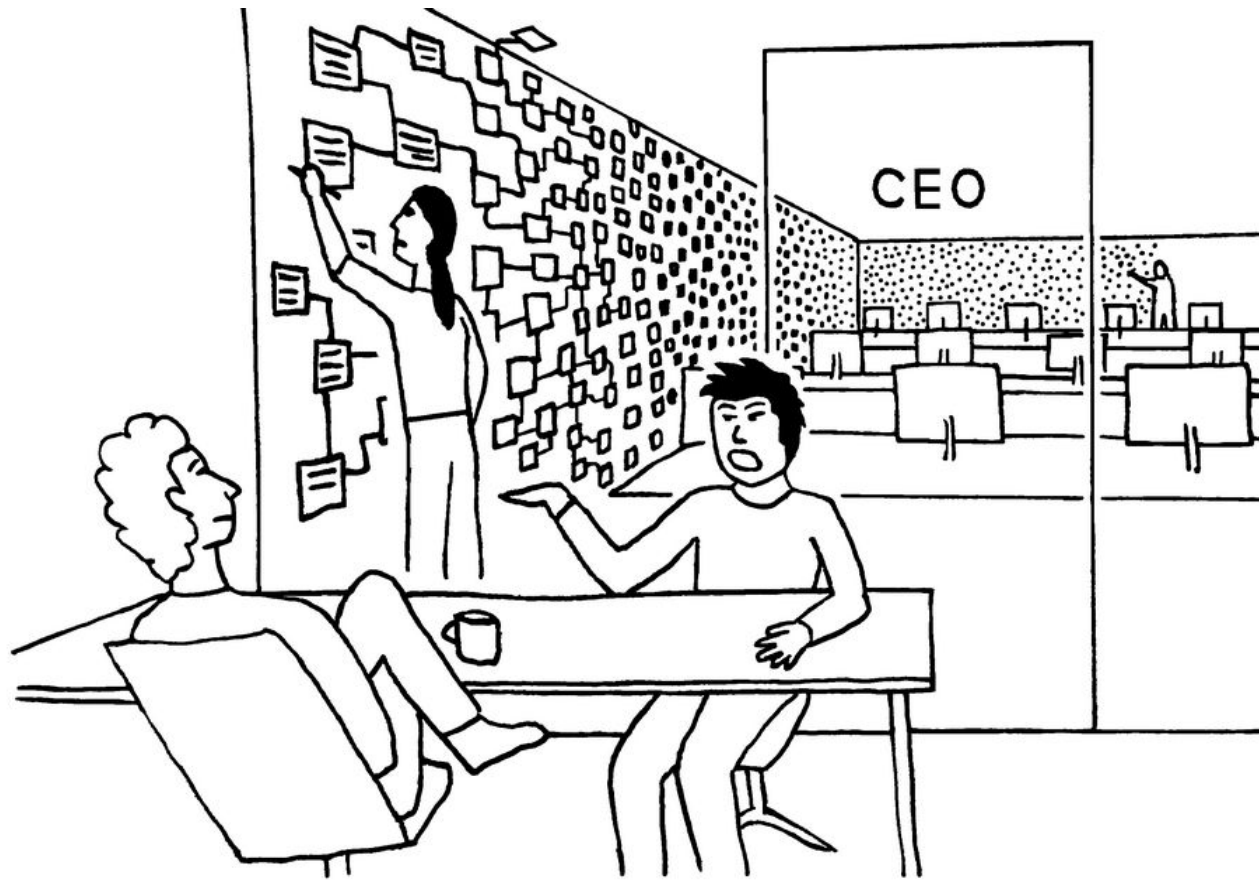
Comment je me suis lancé sur Azimutt?

Azimutt



Doctolib

2020



Après 4 ans d'utilisation

80 personnes par mois utilisent Azimutt
(20% de la tech)

Usages:

- Doctolib academy
- documentation de projets
- identification de ownership (bugs)
- design de datamart
- ...

Utilisé par des:

- développeurs
- data analysts
- product managers
- équipes support...

Bien démarrer avec Azimutt

- Meilleurs cas d'usage
 - Comprendre une base de données complexe
 - Consolider les connaissances de l'entreprise sur ses données

- Recommandations
 - Chaque équipe fait un layout (à minima) sur son scope
 - Définir un template de notes pour les tables et colonnes

Bientôt: Azimutt plateforme 🚀

Objectif: anticiper les incidents de production

- Diagnostic continu de PostgreSQL (pour commencer)
 - Léger, configurable et sécurisé
 - Agrégation de clusters
 - Métriques, procédures et actions métier
 - Agents IA pour l'identification et recommandation de résolutions



Take survey!

<https://bit.ly/pg-reliability-survey>

Merci



<https://bit.ly/pg-reliability-survey>

