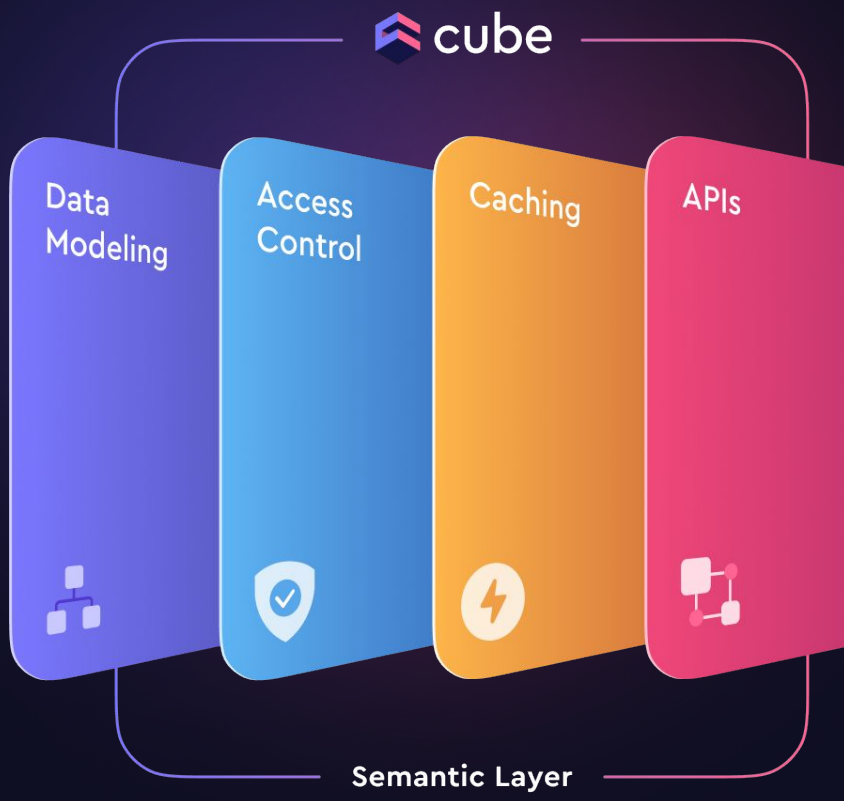
cube | OSA CON 23

The Need for an Open Standard for Semantic Layers

Brian Bickell
VP of Strategy and Alliances



Meet today’s presenter from Cube

Brian Bickell

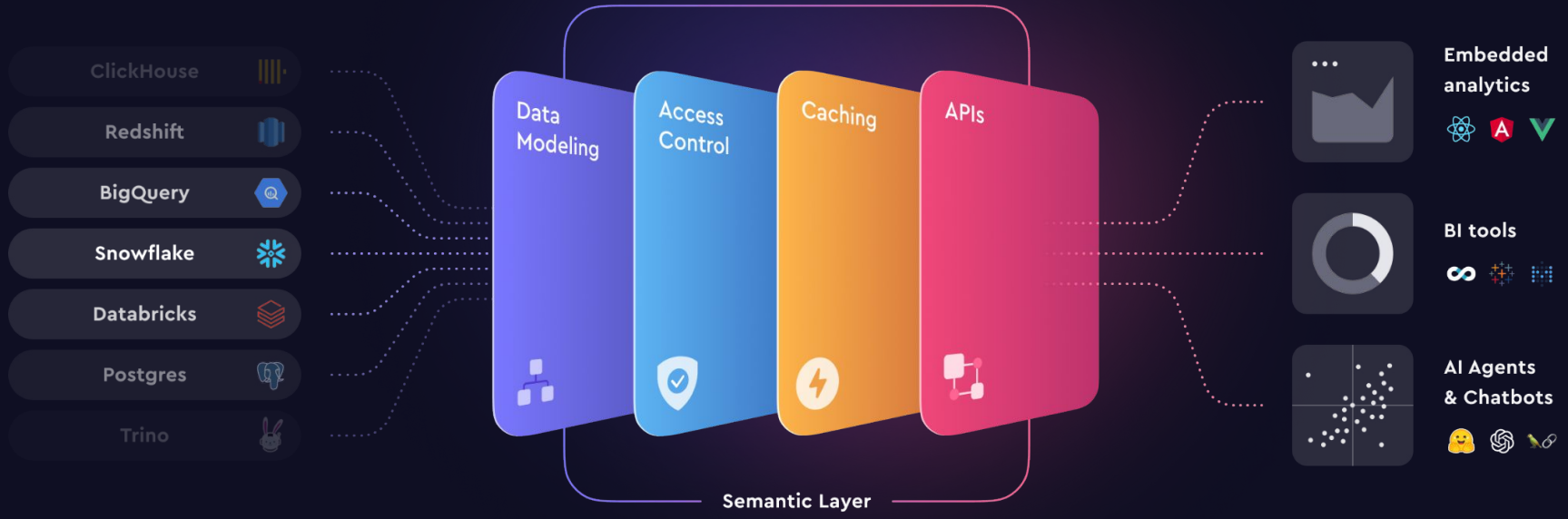
VP of Strategy and Alliances

Agenda

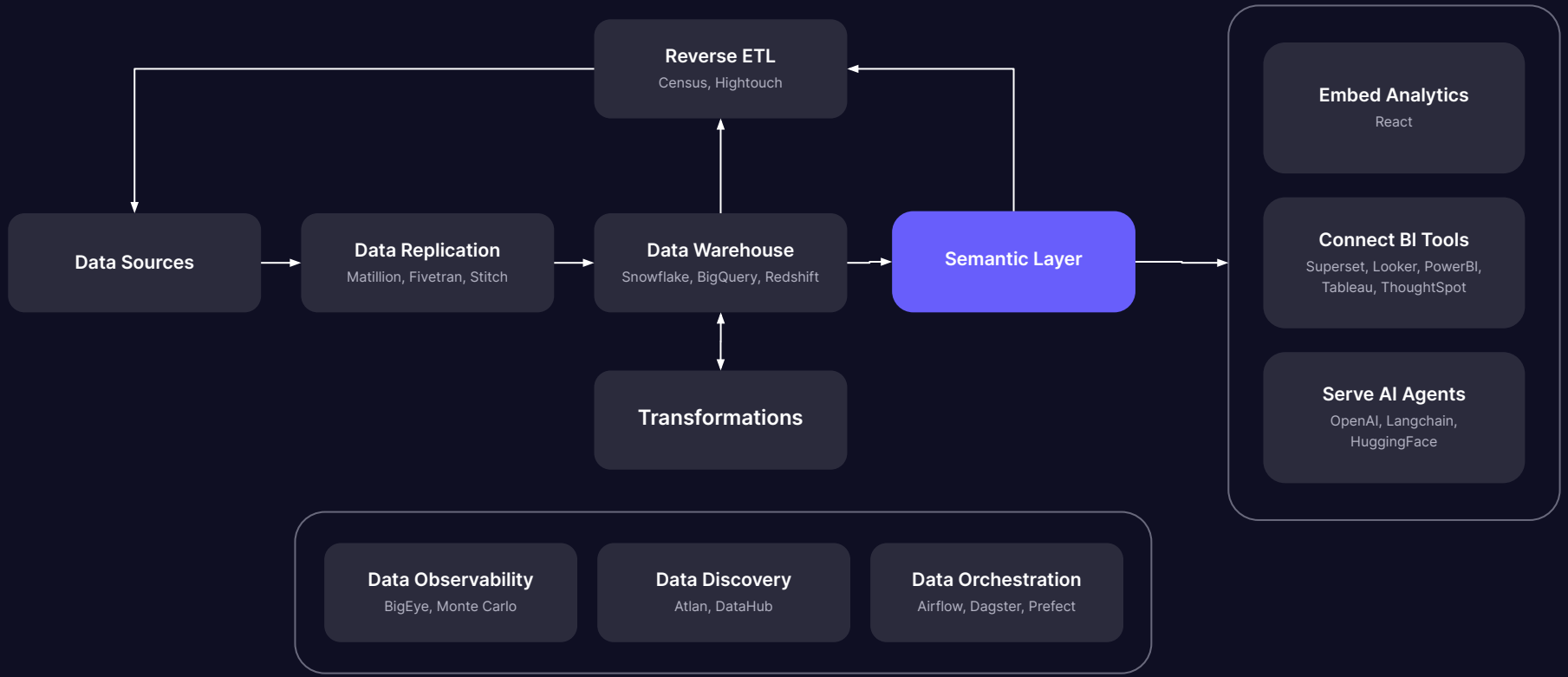
1. What is the semantic layer?
2. Why do we need a standard?
3. What this standard should cover?
4. Metrics-centric vs dataset-centric
5. Querying protocols
6. Metadata API

What is the semantic layer?

Four pillars of the semantic layer

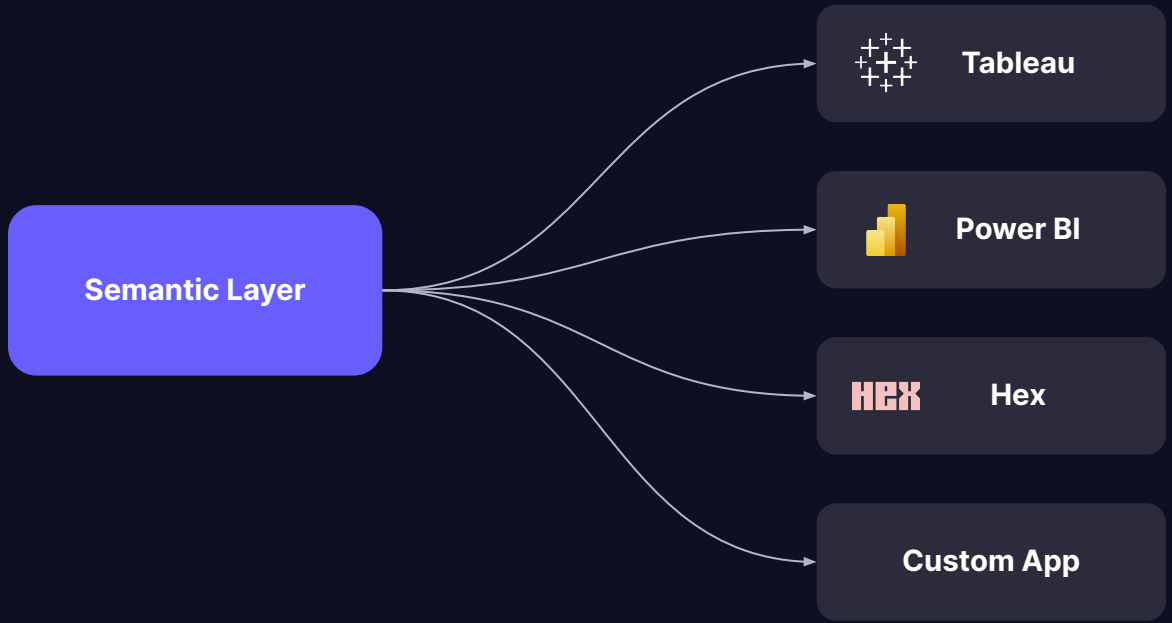


How Semantic Layer fits into MDS



Why do we need a standard?

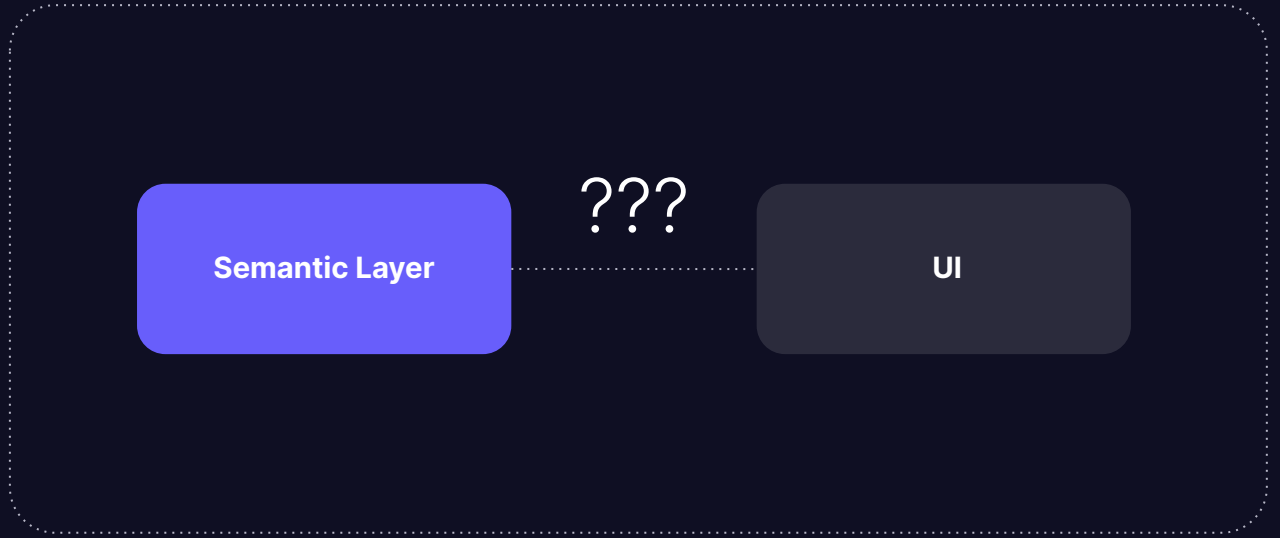
The promise of the semantic layer is to bring consistency to all downstream tools



- How should Tableau communicate with Semantic Layer?
- How Tableau would display all the metrics defined in the semantic layer in its own UI?

It is straightforward when semantic layer is part of the BI

But how to make sure they are in sync when they are decoupled?



> The standard will help to align visualization tools and semantic layers to achieve a user experience similar to integrated semantic layers.

What this standard should cover?

What semantic layer standard should cover?

Specification of objects to describe semantic layer

- Metrics-first
- Datasets-first

Querying protocols

- How BI or visualization tool would query semantic layer?

Metadata API

- How BI or visualization tool would synchronize with a semantic layer to properly map objects in semantic layer to its native constructs.

Specification of objects to describe semantic layer: metrics-centric vs dataset-centric

Metrics-first

Metrics are first-class objects

- customers_with_orders
- new_customers
- order_totals
- orders_count
- revenue
- revenue_growth_mom

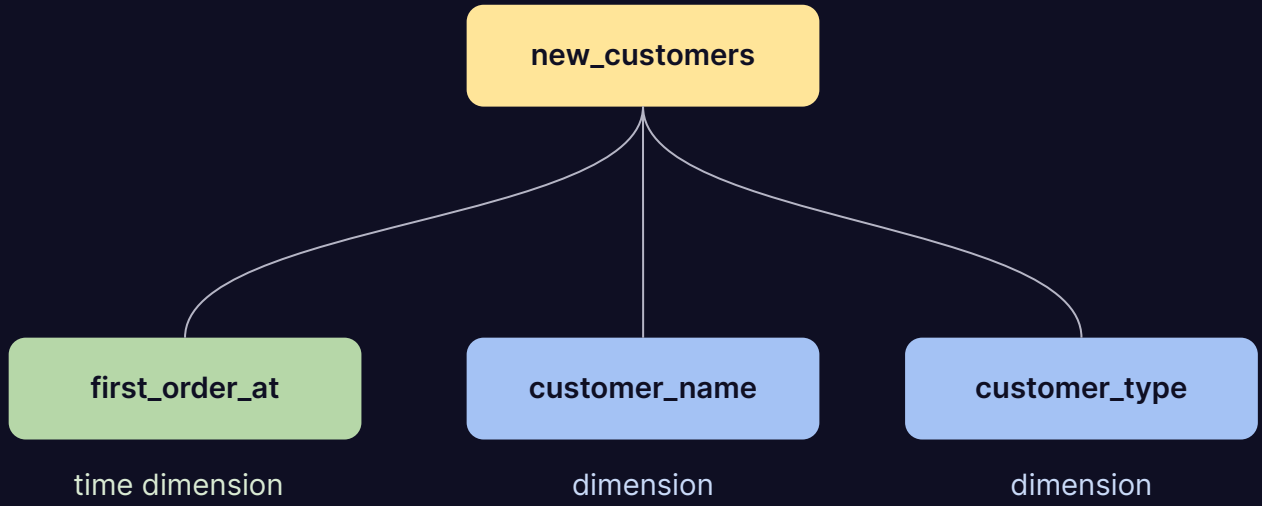
vs

Datasets-first

Dataset-centric semantic layer exposes tables containing measures (metrics) and dimensions as first-class objects.

- customers
- orders
- line_items

Metrics-first



Metrics-first: pros and cons

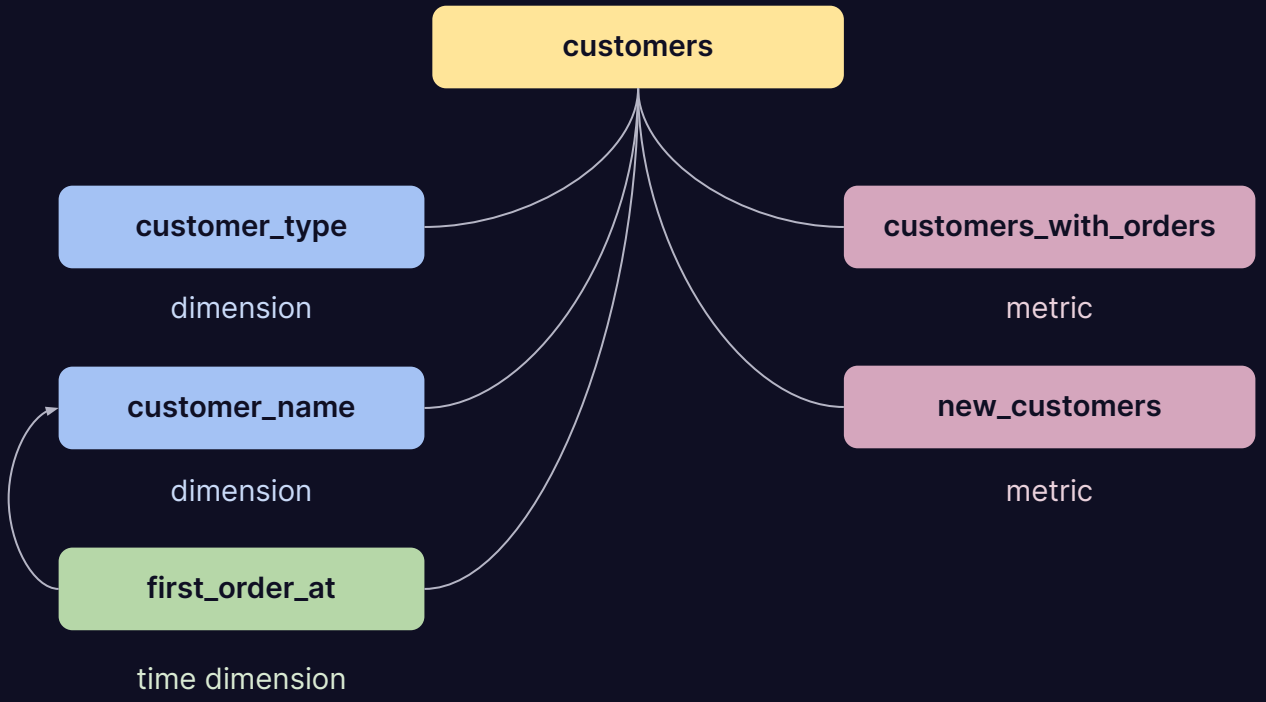
✅

Closer to how people talk about data: We look at metrics, analyze them, and set them as KPIs.

❌

The challenge is that most data consumption tools, including BIs, don't have a notion of metrics. However, some BIs recently started to introduce metrics as native objects.

Datasets-first



Datasets-first: pros and cons

✅

The benefit is better flexibility and compatibility with the existing suite of tools; however, it seems like a more complicated solution/framework.

✅

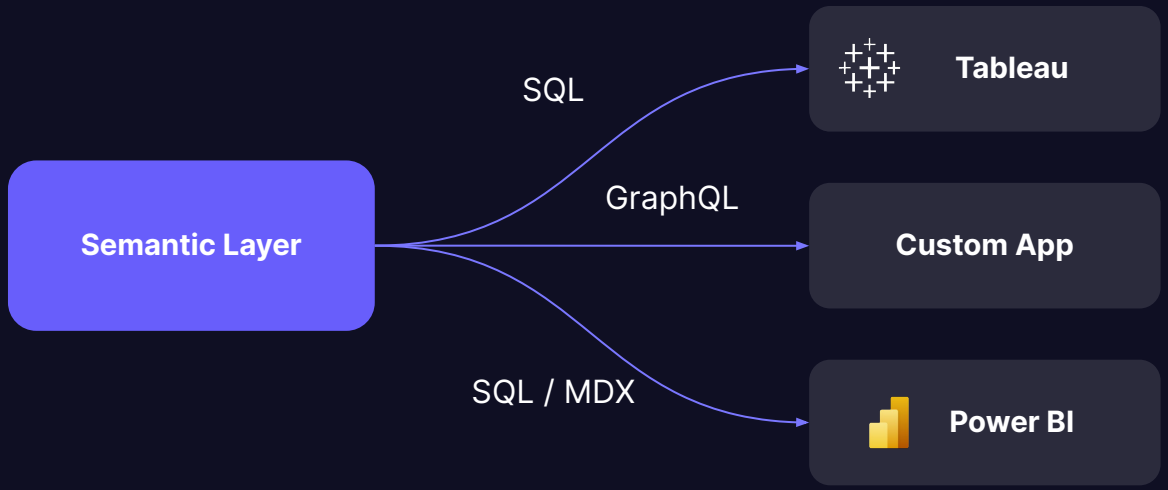
It can support the entity-first semantic layer by exposing datasets about entities with attached metrics and dimensions or metrics-first - by exposing datasets with single metric and connected dimensions.

❌

Metrics are hidden within entities. While sometimes relationships are clear, e.g. orders_total metric can be inside the orders dataset; sometimes it is hard to give it a reasonable structure, e.g. where the revenue metric should belong.

Querying protocols

SQL is a universal querying protocol for BI tools, and GraphQL is taking over the web.



Querying metrics (measures) with SQL

Unlike MDX, SQL doesn't have a notion of metric or measure. To use it as the querying protocol for a semantic layer, SQL needs to be extended to support querying metrics.

Small-scale extension

MEASURE type and special aggregate function to query it.

✅ Easier to support within existing ecosystem

❌ It is relying on SQL structure, which isn't designed to query metrics

Big-scale extension

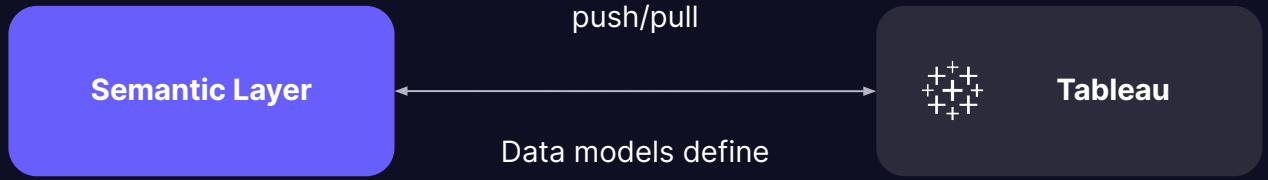
From more operations and commands to a dedicated language for querying metrics embedded inside SQL.

✅ Native way to query metrics

❌ Hard to support in BI/visualization tools

Metadata API

We need to make sure that BI native objects are in sync with data model defined in the semantic layer.



- data sources
- relationships
- measures
- dimensions

Thank you

Continue the conversation at
cube-js.slack.com


cube