

Anti-patterns for scaling analytics



'Delivering powerful analytics to our business partners' - is likely your current company goal. In this paper, we'll describe 3 anti-patterns you should watch out for when building analytics as well as how GoodData can solve them.

Limited metrics reusability

Your team doesn't have to take care of how complicated it is writing queries as long as you have just a few static reports with a few consumers. However, when building an analytics solution your business can rely on, with self-service for your users and insights integrated right at the point of work, you will need to keep your time investments at a reasonable cost.

Metrics that cannot be reused sufficiently slow down the overall performance and productivity of your teams. For every simple variance of an existing report - and they will rise exponentially as your user base grows - you need to change as few lines of code as possible. The less you need to change, the less you can break and the less complicated maintenance will be. So, if your metrics have to be written again just to accomplish simple report requests, your solution will quickly become difficult to maintain as your user base grows.

With GoodData, the limited metrics reusability anti-pattern is solved thanks to the MAQL query language. One MAQL metric can replace plenty of SQL query scripts. With less query code, it's easier to write and maintain than a large SQL script. MAQL allows you to assemble a new metric from existing ones without the need to know how underlying data are structured and how related metrics are connected.

Let's take a look at an example of MAQL and compare it to a standard SQL based query.

Your goal is to get the sales data from your database. In SQL, the query might be the following:

```
SELECT SUM(Amount) FROM sales_data;
```

The result will be 1 000 000 - the total value of products you have sold. Now, you need to know, how much was generated from shoe sales:

```
SELECT SUM(amount) FROM sales_data JOIN
product_data ON
sales_data.product_type_id=product_data.
product_type_id WHERE
product_data.name = "shoes";
```

The result is 500 000. To get the number, you need to understand the table and column structure, understand the relational algebra and how data is stored in your database.

With MAQL, it's much easier to deliver the same results:

```
SELECT SUM(Amount)
```

So where is the sales_data table? With MAQL, It's not necessary, all relevant amounts are summed from the appropriate datastore, according to the logical data model. For the second, much more complicated query, MAQL is simple again:

```
SELECT SUM(Amount) WHERE Product Type = shoes
```

As you can see, you don't have to remember how to join tables to get the results; you can easily reuse the metrics as you want with business semantics. Using MAQL with GoodData will help you to overcome the issues with Metrics usability and prepare your solution for future growth.

You can learn more about MAQL here:

<https://help.gooddata.com/doc/en/reporting-and-dashboards/maql-analytical-query-language>

One query per insight anti-pattern

Your team doesn't have to take care of how complicated it is writing queries as long as you have just a few static reports with a few consumers. However, when building an analytics solution your business can rely on, with self-service for your users and insights integrated right at the point of work, you will need to keep your time investments at a reasonable cost.

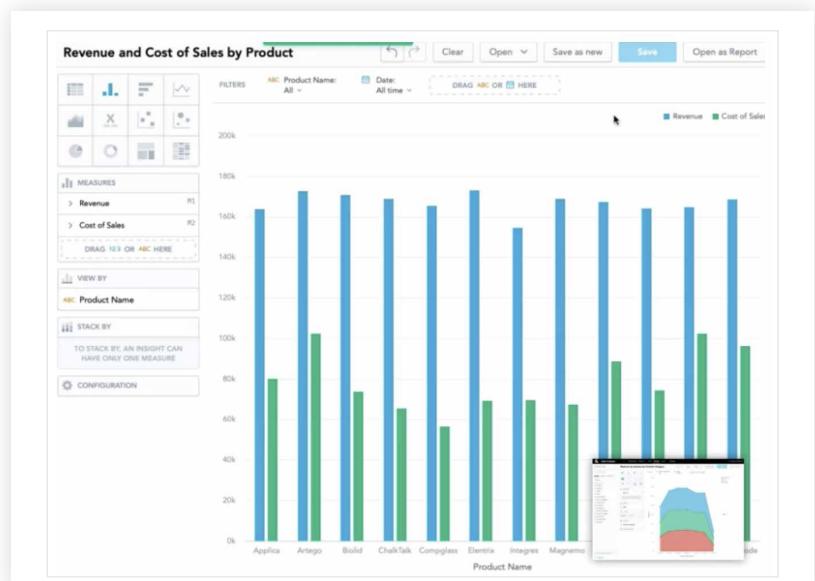
A similar anti-pattern relates to writing a query each time you need to slice or filter an existing metric. Writing another query or changing existing query code means taking up more team resources as well as additional waiting time for the user requesting the change. As a result, this anti-pattern leads to millions of scattered queries with little chance of maintaining them properly. Any change in the model, data or report definitions will require your team's time. Therefore, when building your analytics solution, you should focus not only on the technical requirements but also on how your workforce will need to grow in relation to your user base. In an ideal state, the analytics solution scales with minimal requirements for additional headcount. Building metrics that can be reused in different contexts is an important factor that helps you to keep your metrics uncluttered and, thus, less demanding for your team.

With GoodData, your metrics will be context-aware. Your users will learn to use a single metric in their insights regardless of the context, with GoodData handling it for them.

Your team can simply document a set of context-aware business metrics that will be understood by everyone. Changing such metrics in one place, instead of changing plenty of scattered queries, makes the difference. Moreover, your users will be able to create their own insights, based on the metric you have prepared, in a drag and drop interface, without the risk of them reporting a wrong result.

The screenshot shows the GoodData Sales Analytics interface. At the top, there's a navigation bar with a logo, 'Sales Analytics', and links for 'Dashboards' and 'Reports'. Below the navigation is a search bar labeled 'Search data...'. Underneath the search bar are three tabs: 'all data' (which is selected), 'measures', and 'attributes'. To the right of these tabs is a grid of icons representing different data types like Date, Project Data, and Measures. The 'MEASURES' section contains two items: 'Revenue' (R1) and 'Cost of Sales' (R2). Below the measures is a 'DROP TO ADD' area. The 'VIEW BY' section contains a 'Product Name' item with a 'DROP TO ADD' button. The 'STACK BY' section has a note: 'TO STACK BY, AN INSIGHT CAN HAVE ONLY ONE MEASURE'. At the bottom is a 'CONFIGURATION' section. The main workspace is titled 'Untitled insight' and is currently 'Unsaved'.

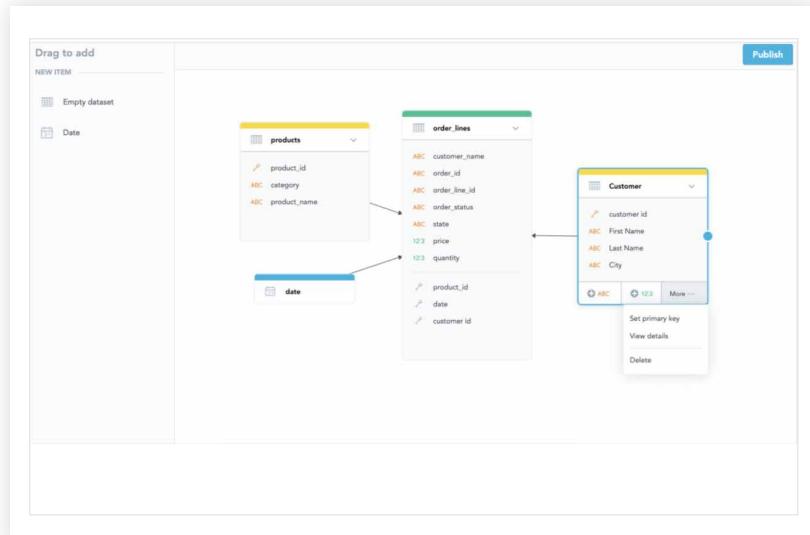
Unlike in SQL-based tools, when joining multiple tables data is protected from a mistake by GoodData's semantic layer. By simply dragging and dropping the Product name to the bucket, the Revenue and Cost of Sales metrics are automatically calculated.



Limited dataset extensibility

Most solutions are built with a set of data sources and specific datasets, when required, right at the point of the project. However, in the words of Heraclitus “Ta panta rhei”, and so too should your analytical solution; changing over time along with your real-world data analytics needs. Imagine having to change a data model due to the requirement for a new data set to be added, describing your new business process performance. In SQL-based solutions you will have to rewrite queries and correct join statements. Doing that for thousands of your franchises or customers would be a long-term project. It goes without saying, to always check how resilient your solution’s data model is and how easy it is to make changes as you grow.

With the GoodData analytics platform, you will be able to connect new datasets without the fear of breaking existing dashboards and insights. Thanks to the semantic layer and logical data model, changes in the model won’t impact existing metrics or insights. Even those that have been created by end-users themselves. You can also split model dimensions and facts, or update relationships between entities, from a single place centrally.



Why the world's top companies choose GoodData

The GoodData advantage

Business:

1. One platform for ALL: internal teams, client companies, external partners

Technical:

1. Automated scaling to different departments and companies

- 
- 2. Self-service visualization for business managers
 - 3. Your own branding
 - 4. Sustainable pricing that fits your business at all times (no paying per user)
 - 5. The highest data privacy and security certifications
 - 2. Embedded dashboards in your application or software product
 - 3. Streamlined multi-tenant change management
 - 4. 150+ data-source options
 - 5. Fully hosted or deployed as a container in your application

Dive deeper into the GoodData platform

[Request a demo](#) and let our experts take you on a guided tour of the GoodData platform.