# MarketUpdate

## Data Fabric 2024

### What is it?

Large enterprises have always struggled to get a coherent and consistent view of data that spans the organisation. Answering deceptively simple questions like "who is my most profitable customer?" requires a consistent definition of what that customer really is, the revenues associated with each customer and how business costs are allocated to that customer. This may not sound difficult, but consider if you make four separate sales to companies called "Enterprise Oil Limited" (in UK), "Butagaz Sas", (in France) Equilon LLC" (in the USA) and "Gasnor AS" (in Norway). Every one of these companies is in fact part of oil giant Shell, but none of those companies have Shell in their name. Somehow your corporate systems have to make this connection for you to understand how much revenue you just sold to Shell. Traditionally such connections would (hopefully) be drawn when data is gathered together into a unified reporting system such as a data warehouse application, with the revenue associated with these four seemingly separate companies aggregated into one parent customer record called "Shell". In reality, common data like "customer" (or "product" or "asset") is scattered across potentially hundreds of different enterprise applications, and duplicates need to be identified and resolved for reporting and analytics to be reliable.

The drawback to data warehouses (and their cousins data lakes) is that you now have yet another silo of data in addition to the original source systems, and data has to be physically copied from the source systems into the data warehouse or data lake. If the data volumes are large then that process will take time and resources, and may only be done at intervals, say daily or weekly. The source systems themselves may occasionally be changed in structure or new data sources added, and these changes need to be reflected in the data warehouse and in the reports that run off it. Changes to database structures for operational systems are non-trivial, and if the changes are frequent then the data warehouse can get out of synch with the underlying source systems, eroding trust in the data.

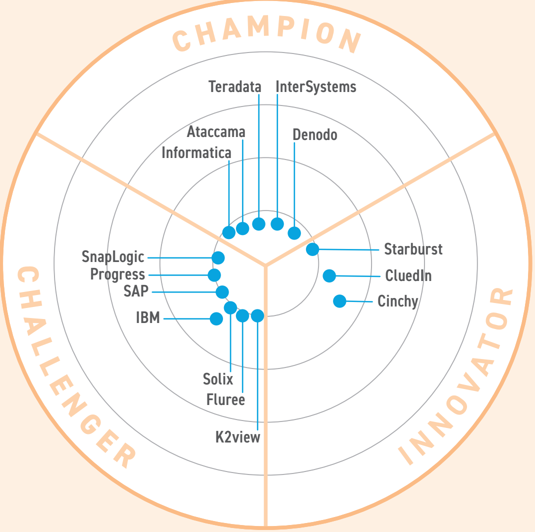
Another approach is to build a data fabric. In this architecture no data is moved about. Instead the structure of the enterprise data is catalogued and a virtual semantic layer is built on top. This layer represents business notions like "sales" and "customer" and shields end users from having to know about the underlying structures and names of the data sources.

### What does it do?

The enterprise data can then be represented in a visual form, perhaps a "knowledge graph", that business users can understand, based on a data catalog that maps the data assest of an enterprise and their context in what is often called a semantic layer. Users can then pose their questions like "who is my most profitable customer?", perhaps via a drag-and-drop interface or even via a natural language interface if AI technology is used. Software will translate the business request into the language needed to access the data (SQL

**Figure 1**

The highest scoring companies are nearest the centre. The analyst then defines a benchmark score for a domain leading company from their overall ratings and all those above that are in the champions segment. Those that remain are placed in the Innovator or Challenger segments, depending on their innovation score. The exact position in each segment is calculated based on their combined innovation and overall score.



in the case of relational database sources) and then pass that query on to the underlying source systems, integrating data sources on the fly and bringing the results back to the user. A variant on this approach is a "data mesh", where the business users are presented with data domains like "customer" or "product", but where ownership of this data is devolved to subject area experts in local business units. Data mesh is fundamentally decentralised whereas data fabric is fundamentally centralised, but both have similar goals: to allow business users to access their data without creating additional data silos and without the need to copy large chunks of data around the enterprise to specialist reporting systems.

## Why should you care?

In practice, this nirvana faces a number of challenges in order to be practical. A key element is building an accurate data catalogue and representing the data structures in a meaningful way. Another, and particularly tricky, issue is how to actually execute the queries that users generate via this semantic (or natural language) interface. It is hard enough for an experienced programmer to build efficient queries that join data from multiple large database tables that have been specifically designed with this in mind, such as a data warehouse. To generate such queries when the data is scattered amongst heterogeneous databases, probably in different underlying formats (SQL, NoSQL, documents, files) requires a sophisticated distributed query engine and optimiser. Without such an engine then a user or AI-generated query could easily result in inefficient queries that result in millions (or billions) of data records being accessed, and response times that will be measured in hours or days. Consequently, a fully functional data fabric solution will need: connectors and integration tools, a data catalogue, a visual representation of data structures, an interface to design queries against that structure, and a distributed query engine.

## Emerging trends

It is not clear that any single vendor offers a complete out-of-the-box data fabric or data mesh solution. Some vendors have integration capabilities but no knowledge graph, some have knowledge graph but no distributed query capability, and others a catalog without integration capabilities. Customers need to piece together the various elements either from vendors or from assorted open-source solutions, most likely with some degree of custom coding. Of course, there will be consulting firms happy to take your money to help you with this, and software vendors that claim to offer everything that you need, at least in PowerPoint slide form if not in actual software.  Given that the terminology has been around for maybe two decades (for data fabric, or since 2019 in the case of data mesh) there is still no universal agreement on the definition and components of a data fabric or mesh, and so in many cases vendors have just relabelled existing products of assorted types as a data fabric solution. Nonetheless, there has been enough frustration with existing approaches that many customers are trying to implement this approach.

## Vendor landscape

*Figure 1* (overleaf) is an overview of the current state of the data fabric market, showing the main participants.

Market sizes are notoriously tricky to estimate since it depends on exactly what you include or exclude, but the data fabric market is probably worth around $1-2 billion and is growing at somewhere between 21% and 31% annually, depending on which market research report you look at and where these firms draw their lines for inclusion. It is particularly hard to really be sure of data fabric market size since there is no widely agreed definition of what is in a data fabric, at least yet. By now everyone knows what a database is and what a data warehouse looks like, but this level of acceptance and maturity is not there yet for data fabric. At this stage of industry maturity, not all data fabric vendors are cut from the same cloth.

## Conclusion

The data fabric market is quite nascent, and even the definition of what a data fabric (or its cousin data mesh) actually is varies significantly from source to source. The underlying goal is clear enough, and that is to achieve an enterprise-wide view of either the whole data landscape (data fabric) or specific data domains (data mesh) without needing to copy source data to other locations like a data warehouse or data lake. The desire to do this is understandable since data warehouses bring their own issues and limitations, though actually implementing a data fabric architecture beyond a demonstration level and on a large scale is a major challenge. Many elements of what is needed are beginning to emerge, but customers need to be aware that they will likely be stitching together separate software components, almost certainly from different vendors, to achieve their data fabric goal.

**Note**

The positioning/scoring of vendors is based on assessments of their standing in the dimensions: financial viability, customer base, revenues, growth, technology breadth, technology depth, geographic coverage and breadth of partner network, within the Bullseye methodology. The overall score determines how near a vendor is to the bull: the nearer the centre, the better. The "clock position" is a secondary measure related to the level of innovation of the software.

**We have a designated webpage for this topic so for the latest research and commentary please visit HERE.**