International Journal of Information, Control and Computer Sciences
ISSN: 2517-9942
Vol:12, No:10, 2018

Big Brain: A Single Database System for a Federated Data Warehouse Architecture

X. Gumara Rigol, I. Martínez de Apellaniz Anzuola, A. Garcia Serrano, A. Franzi Cros, O. Vidal Calbet, A. Al Maruf

***Abstract*—**Traditional federated architectures for data warehousing work well when corporations have existing regional data warehouses and there is a need to aggregate data at a global level. Schibsted Media Group has been maturing from a decentralised organisation into a more globalised one and needed to build both some of the regional data warehouses for some brands at the same time as the global one. In this paper, we present the architectural alternatives studied and why a custom federated approach was the notable recommendation to go further with the implementation. Although the data warehouses are logically federated, the implementation uses a single database system which presented many advantages like: cost reduction and improved data access to global users allowing consumers of the data to have a common data model for detailed analysis across different geographies and a flexible layer for local specific needs in the same place.

***Keywords*—**Data integration, data warehousing, federated architecture, online analytical processing.

I. Introduction

SCHIBSTED Media Group is a leading online marketplaces company with presence in 22 countries and reaching more than 200 million users around the world. Schibsted operates in established markets in Western Europe, as well as in emerging markets in Europe, Latin America and North Africa. Some of the well-known brands across these regions include Finn (Norway), Blocket (Sweden), Leboncoin (France), Subito (Italy), Segundamano (Mexico), Yapo (Chile) or Avito (Morocco), to name a few.

Schibsted's online classifieds business has grown rapidly over the years through launching in new countries or by merging and acquiring already existing brands in other markets. This growth strategy has given Schibsted a portfolio of decentralised operations. Although speed of execution is one of the benefits of this strategy, it has proven difficult to compete at a global view and make global data-informed decisions.

According to Analytics Maturity Model [1] most Schibsted established markets are in stage two and three, with fully functioning data warehouses, Business Intelligence and Data Science teams. This was not the case in emerging markets like Avito, Segundamano or Tori a few months ago; instead, they were in early stages of analytics maturity and data engineering capabilities were scarce and local data warehouses inexistent.

Xavier Gumara Rigol, Iker Martínez de Apellaniz Anzuola, Antonio Garcia Serrano, Albert Franzi Cros, Oscar Vidal Calbet, Abdulla Al Maruf are with the Schibsted Media Group, Barcelona, Spain (e-mail: xavier.gumara@schibsted.com, iker,antonio.garcia@schibsted.com, albert.franzi@schibsted.com, oscar.vidal@schibsted.com, maruf@schibsted.com).

In those markets, analysts' questions were sent to engineers that had to translate them into queries to be performed on top of the operational database. Analysts' demands would have to follow engineering planning and priorities making the whole process very inefficient.

From a central perspective, great efforts were made to compile and make available a vast collection of Key Performance Indicators (KPIs) using third party web analytics tools and custom internal tools. These tools provided KPIs at a highly aggregated level and without sufficient granularity to derive the insights that the central marketplaces organisation at Schibsted was starting to need to compete on a global scale.

In this document, we address how these two needs, local and global, were taken as the fuel to architect and develop a novel federated platform for data analysis, named Big Brain. This platform consisted of one global data warehouse and a local data warehouse for each of the seven local brands under scope; using a single physical database system.

It is important to notice that we do not include the technical details of the implementation in terms of choices in the database storage, or the details of the data pipelines used to transform and aggregate data. Instead, we focus on the architecture of Big Brain as a solution to integrate data from different federated operational systems.

II. Related Works

This platform for local and global data analysis at the maximum level of granularity was a debut in Schibsted. We had to specifically look for literature in the field. It always pointed towards federated data warehouses as a possible solution; mainly because they are the best fitted solution when data to be integrated is scattered geographically and also because supplying autonomy for the decentralized operations is needed [2].

The first approach to be considered was regional federation where local brands have their own data warehouse for local analysis (which generally need more detailed information) and a global data warehouse is built to supply the corporate requirements. This global data warehouse is built as an upward federation, meaning that fact data is moved from regional data warehouses to the global data warehouse and then is aggregated [3], being this aggregated information what constitutes the corporate view of the company.

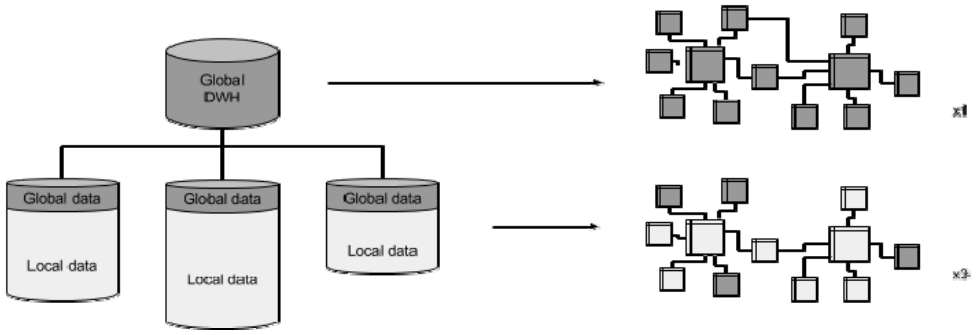

Fig. 1 Federated Data Warehouse high-level architecture

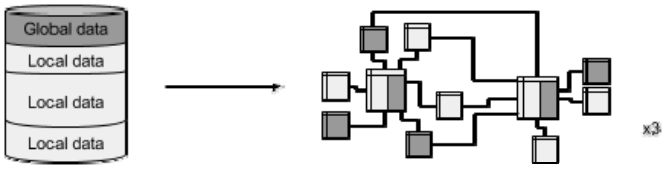

Fig. 2 Big Brain high-level architecture

This model is illustrated in Fig. 1; where local data warehouses are physically located in different geographies, share the same global data, corresponding to common dimensions, but can differ in volume when it comes to storing local data (schema on the left). The global data warehouse is formed by the small common set of dimensions that are duplicated in the local data warehouses plus the aggregation of local business data in a way that it is impossible to go back to the original regional data from the global perspective. In the same Fig. 1 but on the right hand side, the multidimensional schema corresponding to the physical tables is shown. All fact and most dimension tables are local in the local data warehouses. Some common global dimensions exist, typically to store calendar and geographical hierarchies. In the global multidimensional schema (top right) all fact tables are aggregated at global dimension definitions. This type of federation works well when data warehouses already exist in the different geographies since it can take advantage of current existing systems. On the other side, one of the drawbacks is that too much autonomy in the federation risk to make cooperation between local and global warehouses more difficult [2]. The need for a minimum of cohesion and central governance must be enforced and this ought to be provided by a global schema expressed in the common, "canonical" data model [4].

Having a global data model was also a prerequisite in our scenario; however, the fact that we did not have local data warehouses to federate in Schibsted emerging markets implied that we had to look for other solutions and build, at the same time, the regional data warehouses and the global one.

III. Big Brain Architecture

In this section, we describe the architecture of our federation system that allowed us to offer business users both a local and a global data warehouse at the maximum level of granularity.

A. Overview

As stated above, the main goal of the data warehouse architecture introduced in this paper is to provide a tightly coupled data warehouse system for decentralised organisations like Schibsted. The three key success criteria that this architecture supports are:

First, we have a common data model across the different geographies. This facilitates cross analysis and knowledge sharing between the different organisations: both between two or more different local operations and also between any local organisation and global business users. Second, we have common definitions and data processing rules that provide us "one version and unique definition of the truth" for the whole group. And third, we have a flexible analysis layer that local brands and functions can use for their specific needs whether they are in the shape of reports, dashboards or advanced analysis. These three criteria translate into the following functional requirements that can also be visualised in Fig. 2:

- The various local data warehouses share conformed dimensions for reference data (marked in dark grey): they have the same meaning in all fact tables with which they relate allowing for reusability.
- Conformed dimensions represent the dimensions having identical business meaning, identical structure and identical data.
- The common data warehouse has conformed facts and measures: they have identical business meaning and exactly the same values for the same set of dimensions (marked in dark grey in the fact tables).
- The various local data warehouses have a specific set of dimensions for local reference data that have identical business meaning, identical structure but different data (light grey).
- The various local data warehouses have conformed facts and measures: they have identical business meaning but

not the same data values. These are specific to the specific geography the data warehouse belongs to (light grey).

As can be concluded from these requirements, the federation is happening in the same physical storage and even in the same tables. Fact tables contain both local and global values allowing benchmarking with a simple query at all levels of granularity of the data.

B. Layers of the Data Warehouse

Layers are another scalability feature of this new approach of federated data warehouse architecture. In Big Brain, we followed the traditional three layer structure introduced by Bill Inmon [5] and we centralised all information in a data warehouse layer that is used to later aggregate on a data mart level for each one of the relevant subject areas of the classified ads business: Content, Accounts, Behaviour.

The novelty in this implementation is the introduction of a new layer in between the more traditional Staging Area and the Data Warehouse layers. We called this new layer, the Time Based Layer, and it acts as a special kind of integration layer.

In the following sections we explain and justify the need of every one of these layers:

1) Staging Area Layer

This layer is a storage facility that allows the integration of multiple data sources and formats (database data, raw text files and logs). One of its main purposes is that it helps to free operational origins quickly because once operational data is loaded to the Staging Area (by quickly copying it), all needed transformations can take place without interfering the operational systems.

Only data that will be needed for analysis is going to be moved then to the Data Warehouse, and lots of detailed raw information is going to be left largely untouched, at first.

2) Time Based Layer (TBL)

This integration layer is placed before the data warehouse layer and it is used as a contract between the operational systems and Big Brain. This layer allows us to convert daily deltas of information into full snapshots of conformed and consolidated information to be accessed by analysts and data scientists. This layer has two main features that are key to the scalability of Big Brain:

First, with the TBL we make sure that all transformations and aggregations that come after this layer are common for all operations implementing Big Brain: all the scripts, data models, etc. are the same from this point on. This ensures reusability of the code but, more important, allows us to have a common data model for analysis for all operations.

Second, every entity we store in this layer has a timestamp that describes it (hence its name). Because of the fact of being time based, the TBL has a lot in common with real time event tracking. Nowadays, the TBL is loaded daily (in batch during the night). In the future we plan to incrementally get rid of the TBL and replace it with real-time event tracking.

Every entity in the TBL is named as an event. For example, as can be seen in Fig. 3, the Accounts subject area is described with the events AccountInsertion, AccountEdit, AccountLogin and AccountStatusChange. With these events, the Account entity can be reproduced in future layers.

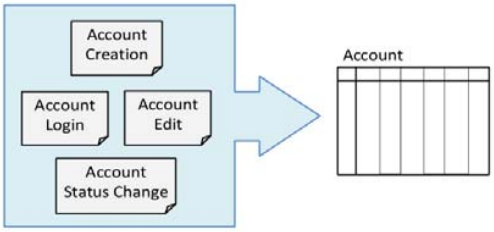

Fig. 3 Relationship between the TBL and the Data Warehouse Layer

For each one of these events we receive local and global mappings to the desired dimensions of analysis and each one of them is responsible for maintaining consistency in specific fields of the Account snapshot data. This means that AccountStatusChange is the only event responsible for modifying the Status field in the Account table in the data warehouse layer; done otherwise, we exposed the solution to have a lot of inconsistencies in the data.

The events stored in this layer have a temporary existence, they only need to be stored here waiting to be loaded in the next Data Warehouse Layer, so it also can be seen as a stream data flow layer where data is not persisted.

3) Data Warehouse Layer

The Data Warehouse layer is an entity-relationship model (in 3NF) with business rules already defined and all the business indicators calculated. In this layer, the data has been transformed, normalized and qualified and in contrast with the previous layer, here we store all the available historical data.

This layer is built by entities that have dimensions and metrics, and is a solid and robust origin with the rules of the analytical models applied. In this layer we guarantee that we have a single point of rule maintenance and a single version of truth. Also, it is the place where we trace changes within entities for the required attributes.

4) Data Mart Layer

This layer is a multidimensional model that has fact tables with metrics and dimensions. Rules are not modified at this point; we do not maintain trace nor change logic and we just read from the data warehouse layer. In this model there are calculated dimensions and metrics, but always based on the Data Warehouse layer. If the business rule changes, the data mart can be recalculated because the valid source data with the single version of truth always exists in the previous layer.

The Data Mart Layer is a report and human friendly place where data is ready to be used by different audiences at the same time.

This layer is the entry point of Data Scientists and Business Analysts to the data warehouse, all queries are performed in this layer and it is also the layer where data visualization tools like Tableau [6] connect to display charts and dashboards previously built by the analysts.

C. Data Flow

There is one flow of data for each of the seven marketplaces implementing Big Brain. Due to the fact that local data warehouses were non-existent in Schibsted's emerging markets prior to Big Brain, the fact that the development was centralized made support and maintenance easier.

Once the layered design was 80% ready for the more relevant domains, the implementation started for a couple of brands to be able to test the feasibility of the architecture. Once the implementation was finished for the two first brands, users started to see the benefits of having data from two operations in the same physical storage. For local operations, it allowed to have a data warehouse in the first place. For central teams, it allowed benchmarking of the operations at levels of granularity never seen before.

IV. Conclusions

At the time of writing, Big Brain has been scaled to seven different operations in Schibsted and dozens of employees enjoy access to steady feeds of quality data to inform their decision making on a daily basis.

One of the most requested features by analysts and data scientists is the ability to directly query the Time Based Layer, which indicates that nowadays analysts demand for all the corporate information available. Persisting the TBL is one of the next steps in the implementation.

Although the solution could be scaled to more markets, the globalization of the company not only affected its organizational structure but also the technology in place. Nowadays, more and more global components are available to the local operations. These components, exposing real-time data, are the source of corporate transactional systems and we will be steering our efforts into the integration of this global data sources.

One of the next natural steps will be to provide downward federation of these global datasets to regional analysts. Since we are already storing data in the same physical storage, the current architecture supports this concept and it will just be a matter of regular data integration and data modelling work. This will affect the TBL; since the fact that it is entirely based on events will help us scale the solution when consuming real-time events from this new global corporate data sources and deprecate current pipelines. These new data sources are allowing us to grow the volume of "global data" (see the diagram in Fig. 2) providing easiness in the transformations and aggregations.

Acknowledgment

We would like to thank all the members of the Data Science and Product Analysis teams at Schibsted for the joint effort made in Big Brain. Without their feedback and collaboration this work would never have come to life.

We would like to thank also all those Schibsted operations that believed in Big Brain, saw the need of it and collaborated without hesitation. Subito, Segundamano, Ekhanei, Avito, Tori, Yapo and Kapaza.

References

[1] T. H. Davenport, D. Cohen, A. Jacobson. Competing on Analytics. BABSON Executive Education (May 2005) http://www.babsonknowledge.org/analytics.pdf.
[2] M. Schneider. Integrated Vision of Federated Data Warehouses. Data Integration and the Semantic Web (Luxembourg, June 2006).
[3] R. Jindal, A. Acharya. Federated Data Warehouse Architecture. WIPRO (accessed September 2017) http://hosteddocs.ittoolbox.com/Federated%20data%20Warehouse%20Architecture.pdf.
[4] S. Berger, M. Schrefl. From Federated Databases to a Federated Data Warehouse System. Proceedings of the 41st Hawaii International Conference on System Sciences (2008).
[5] W. H. Inmon. Building the Data Warehouse. Wiley Publishing Inc. (1996).
[6] Tableau Software http://www.tableausoftware.com.