self-service data: the key to analytical agility

Jim Gallo, VP Business Analytics and Strategy

g2o

what's your next?

For years, self-service reporting and analysis were touted as the keys to driving business productivity. More recently, the idea of citizen data scientists has emerged as the next big thing to unlock business value.

This concept of self-service has been the foundation for business intelligence tools since the 1990s. While the adoption of these tools has been widespread, the ability to **quickly** and **economically** connect these solutions to new datasets continues to be a challenge for business users and data teams alike. The issue often has little to do with the tools themselves but with data integration, an ongoing hurdle in the IT industry.

The reality is that business analysts and data scientists cannot always predict their data needs ahead of time, and the existing data stores may not contain the information needed for new and emerging use cases. As a result, teams are left scrambling to provide users with the data they need and are often forced to use platforms and processes that are too slow and cumbersome to deliver the data as quickly as needed.

This paper explores the opportunity to bridge this gap so that companies can provide the means to support unanticipated data needs in an agile and cost-effective way.

how did we get here?

The desire for a single version of the truth has been the holy grail for organizations since the dawn of computing, yet the ability to deliver on this promise still confounds companies to this day. Before diving into today's challenges, let's first take a look at the past.

Creating a web of extracts to connect disparate data

By the late 1980s and early 1990s, technology innovation was focused not only on advancements in hardware, but on deploying IT systems and software to digitize core business processes. This transformation helped companies realize tremendous productivity gains, but it also created new data challenges. These early solutions had self-contained databases that created a fragmented data environment. In order to connect and report on all this fragmented data, it was necessary to create a web of extracts that was time consuming and error prone.

Enterprise Data Warehouses to the rescue

To address the issue of fragmented data, many companies turned to enterprise resource planning (ERP) systems to serve as a single source for business-critical data. But ERP systems alone were not capable of housing all the disparate data, so companies began building enterprise data warehouses (EDWs) which proved to have shortcomings of their own.

While EDWs provided a means to clean up a company's data house, they often were not designed to address the root cause of the problem—poor data governance of the transactional systems where the data is created. If companies had adopted these upstream data governance practices, then building an analytics data store would have been far easier and faster than it is today. Yet, these best practices were often ignored as organizations chose what they believed to be a more economical path—focusing on the functionality and reducing the time and cost of each project.

According to IDC,

90% of the unstructured data is never analyzed.

The issue was compounded by the fact that no one anticipated the need to incorporate unstructured and semi-structured data which has added to the complexity and time required to derive value from the data housed in operational and analytical systems.

Standardizing business-critical data sets with MDM

In the early 2000s, Master Data Management (MDM) emerged to standardize and bring together business-critical customer and product data sets. For some, MDM solutions were able to deliver on this promise, but many struggled with the same issues that they faced with their EDW initiatives. As a result, many were plagued by duplicate records and poor data quality. For those MDM programs that did succeed, their effects were limited by the fact that MDM tools are suitable for only a small subset of core data. In addition, systems were typically hands-off to business analysts, making mastered data inaccessible.

Data lakes: the next big data thing

In the early to mid-2000s, companies began exploring a new way to address the issue of managing the exponential growth in the volume of data and the increasing variety of file formats. The idea of a data lake emerged as a way to store raw, unprocessed, data in a centralized location without the upfront heavy lifting required by data warhouses to develop a schema to process and refine the data for a predetermined use case.

The thought was that data lakes running on Hadoop could leverage advancements in storage capacity and processing speeds to provide rapid access to all raw data without preparing it beforehand. The reality was that the upfront effort of developing a schema was just pushed to the data consumption team, where the data now needed to be prepped and refined. While the speed in which the raw data could be accessed did improve, data preparation became the new bottleneck for enabling self-service analytics. As a result, a slew of specialized programming tools emerged to help accelerate the process—but for many organizations, it just added to the complexity.

Table 1: A subset of big data tools used for analysis and reporting

- Abdera
- Airavata
- Ambari
- Apex
- Avro
- Axiom
- Axis2
- Batik
- Beam
- Bigtop
- BookKeeper
- Calcite
- Camel
- CarbonData
- Cassandra
- Cayenne
- Click
- Cocoon
- Cordova
- CouchDB
- Crunch
- Curator
- CXF
- Daffodil
- DataFu
- Derby
- DeviceMap
- DirectMemory
- Drill
- Edgent
- Empire-db
- Falcon
- Flink
- Flume
- Fluo
- Fluo Recipes
- Fluo YARN
- FOP
- Forrest
- Fortress
- Giraph
- Gora
- Hama
- Hbase
- Helix
- Hive
- Ignite
- Jackrabbit
- Kafka
- Kerby
- Kibble
- Knox
- Kudu
- Lens
- Lenya
- Lucene Core
- Lucene.Net
- Lucy
- MetaModel
- mod_ftp
- ODE
- OFBiz
- OODT
- Oozie
- OpenJPA
- ORC
- ORO
- Parquet
- Phoenix
- Pig
- PLC4X
- PredictionIO
- REEF
- Regexp
- Samza
- Sandesha2
- Santuario
- Scout
- ServiceMix
- Spark
- Spatial Information System
- Sqoop
- Storm
- Synapse
- Syncope
- Tajo
- Taverna
- Tez
- Tika
- Torque
- Trafodion
- VXQuery
- Woden
- Xalan for C++ XSLT Processor
- Xalan for Java XSLT Processor
- Xerces for C++ XML Parser
- Xerces for Java XML Parser
- Xerces for Perl XML Parser
- Xindice
- XML Commons External
- XML Commons Resolver
- XML Graphics Commons
- XMLBeans
- Zeppelin
- ZooKeeper

In addition to the added complexity, data lakes and file-based systems touted as business-friendly universal data stores introduced new security and data quality issues that have turned many data lakes into data swamps. The result has been many underutilized or abandoned data lakes.

Self-service reporting tools and shadow IT

As self-service reporting and analysis tools directly marketed to business users took off in the 2000s, the need to quickly deliver a single version of the truth once again created a huge challenge for data teams. This resulted in the continued growth of shadow IT teams that would circumvent the data teams to quickly, but often ineffectively, acquire the data needed by business users.

Coming full circle

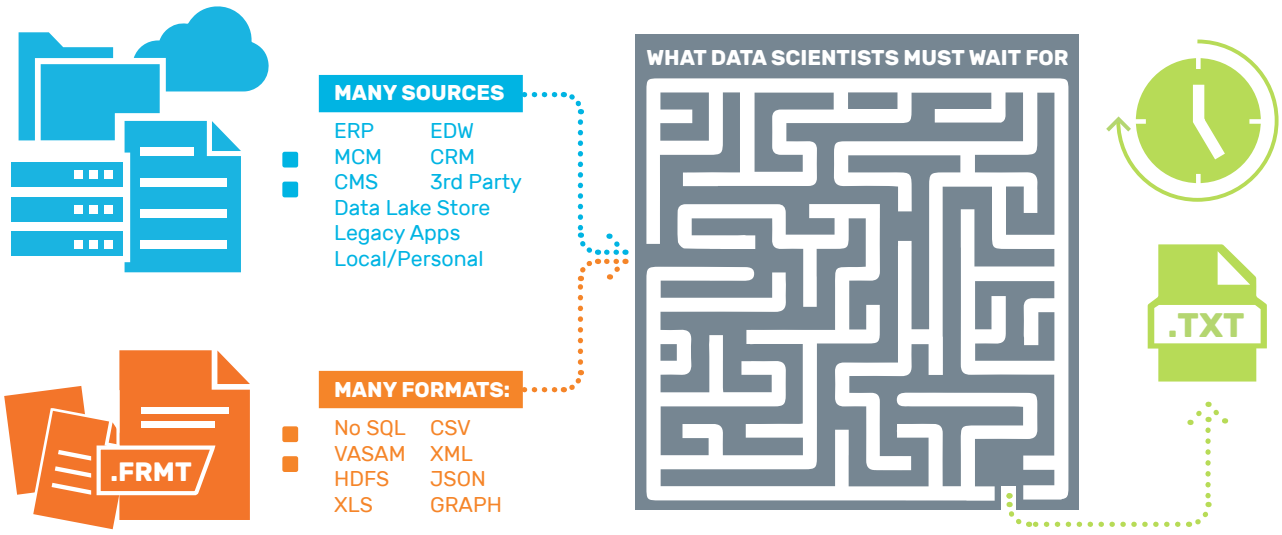
The need for quality, trusted, homogenized, and business-friendly data continues to be a primary goal for organizations today. But the idea that this can provide a single version of the truth may not be the ultimate answer.

The reality is that organizations have deployed ERP and CRM systems, EDWs and MDM solutions—each built with the stated purpose of becoming a 'single source of truth' for the enterprise. Adding data lakes, other OLTP and content management systems, and third party and locally stored datasets into the environment into the mix creates confusion, risk and shadow IT organizations that are booming.

According to Forrester, **between 60% and 73% of all data within an enterprise goes unused for analytics.**

Keeping this in mind, and considering the variety of file formats that one must understand (relational, NoSQL, VSAM, Hive, graph, CSV, XLS, XML, JSON, etc.) to be effective, it's no surprise that InfoWorld reported that data analysts and data scientists spend only 20 percent of their time on actual data analysis and 80 percent of their time finding, cleaning, and reorganizing data. Ironically, this is the same challenge that has been around since the advent of data warehousing, and some would argue that the complexity of today's environment (see Figure 1) has only added to the problem.

Figure 1: Today's complex data environment is slowing down self-service analytics



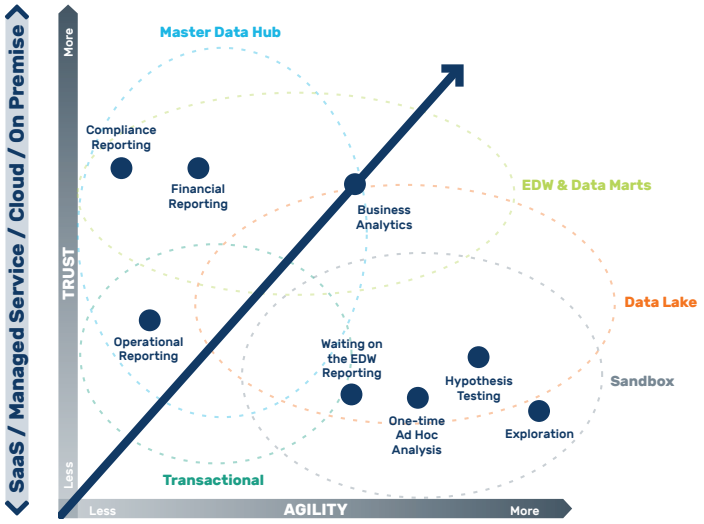
thinking differently about data

In today's responsive and real-time world, data bottlenecks become business bottlenecks. So how can organizations provide access to quality data quickly and easily without adding one more layer of complexity? It requires thinking differently about data and acknowledging that not all data is created equal, and the level of agility and trust required in the data is relative to the specific analytic use case.

Moving from where to what is needed

When thinking through different use cases along the trust and agility axes, it seems that the data industry has historically thought about them in the context of "where" based on the available data stores created by IT (Figures 2).

Figure 2: Aligning analytics use cases by IT-created data stores



In the past 18 months, there has been an emergence of products that make access to datasets, regardless of source, format and complexity, available to mere mortals—easily, quickly and without breaking the bank.

These new capabilities provide an opportunity to think about data and analytics use cases differently. This path forward is based on three new rules to consider when evaluating use cases.

1. Not all data needs to be integrated
2. Data quality is in the eye of the beholder
3. Combining datasets does not always need to be an IT project

Sometimes being directionally correct is good enough

Often, business analysts want to get a hold of a couple of files needed to get to an answer quickly or to explore datasets in different ways. This is very true in the data science community where the scientist simply wants to leverage BFFs (big flat files) to explore data, test a hypothesis or to run models. This does not imply that that data needs to be formally integrated by IT nor does it imply that the data needs to be of the highest veracity. Does it matter if street addresses and customer names are not standardized when trying to conduct market basket analyses using eCommerce sales information?

A real world example

A CFO of a regional bank was recently surprised with a sudden downturn in their loan servicing income. This decline occurred during the time the bank's EDW was under construction and the project was delayed due to data quality issues about the attributes of the loans, but not the loan amounts.

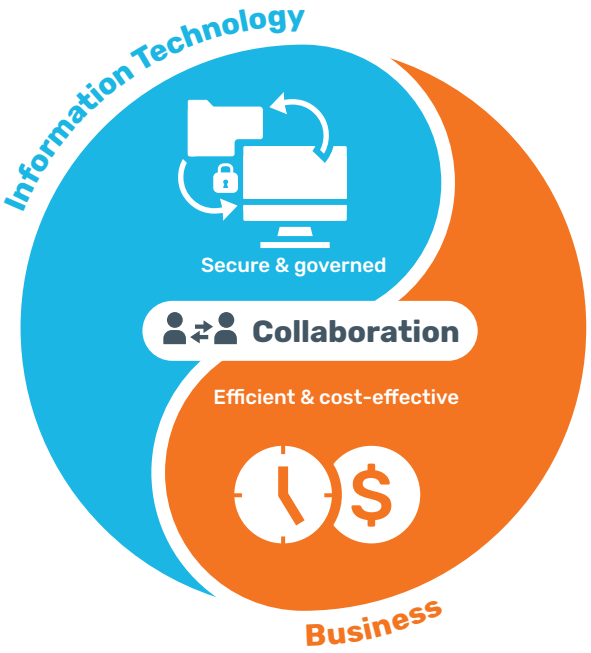
From the CFO's perspective, knowing the number of loans and the total amounts projected to be paid off each month would have provided the relevant information needed to plan better. Given that the monthly loan amounts are in the billions of dollars, it wouldn't have mattered if the calculated total had been off by a couple of million dollars (i.e., tenths of a percent). In this case, less trust and more agility were needed since the macro numbers would have been 'close enough.'

Now, if his use case had involved financial reporting to the street or to the federal government (more trust, less agility), then a higher degree of data quality and precision would be required and would require more formal and rigid development, testing and validation methods.

It's not pandemonium, but more like controlled chaos

This paper does not advocate for the abdication of data management best practices. Rather, it recommends finding the right balance where IT's charter to govern and secure data can peacefully coexist with the business' need for speed.

The reality is that shadow IT will continue to exist and truly does serve a purpose for specific analytics use cases. So rather than swimming upstream, let's embrace it and find appropriate and effective ways to enable self-service data so that IT and the business can truly collaborate.



new ways forward

The remainder of this paper will cover three interesting choices for enabling self-service data. **Note, the intent is not to suggest that self-service data should replace the data platforms of the past but rather work in concert with them when appropriate.** The goal is to describe these products from a practical usage perspective rather than a detailed dissertation on their technical underbellies. The three product categories are:

- User-enabled data movement
- Data marketplace
- Data virtualization

User-enabled data movement

User-enabled data movement tools offer a no-coding, drag and drop experience that can easily be mastered by non-technical analysts. They are especially effective at helping to solve for the inherent gap in analytics tools; namely, the file formats they can use. For example, visualization and data science tools are unable to directly access data stored in graph databases.

In the past, if an analyst needed to move data stored inside a graph database into another type of data store, they would need IT support to move the data using ETL, scripting or coding.

Now with user-enabled data movement tools (see Figure 3), it is possible to enable an analyst to update or create new tables and files from most file types, regardless of size and complexity. And they can move the data from any source to any destination while maintaining performance and being resilient to data changes.

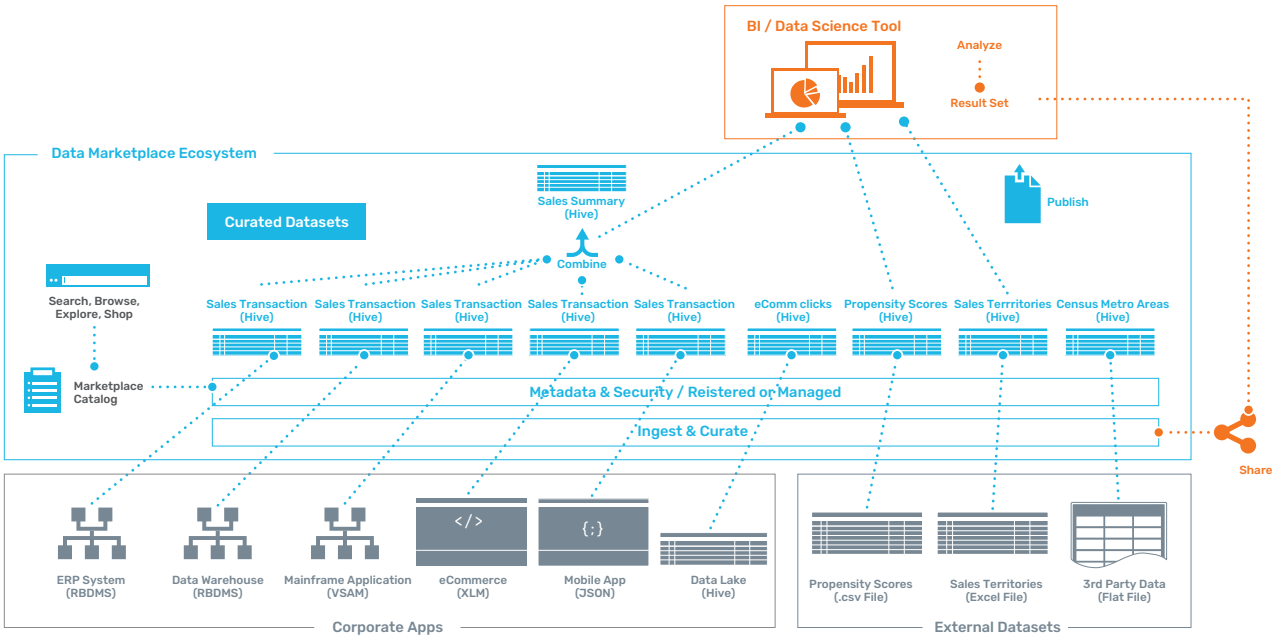
Figure 3: User-enabled data movement tools

Data marketplace

Data marketplace tools and data catalog tools are not the same thing. While there are similarities, the primary purpose of a data marketplace is to pre-position datasets, regardless of origin or quality, and make them available to data consumers through a curation process that enforces governance and security.

A marketplace may contain trusted datasets from transactional systems, EDWs, MDM hubs, etc., but may also include third party datasets and personal or departmental datasets created in Access or Excel. In addition, leading data marketplace tools enable data consumers to combine datasets or create new ones, which in turn, can be placed into the marketplace via ingestion and curation processes (Figure 4).

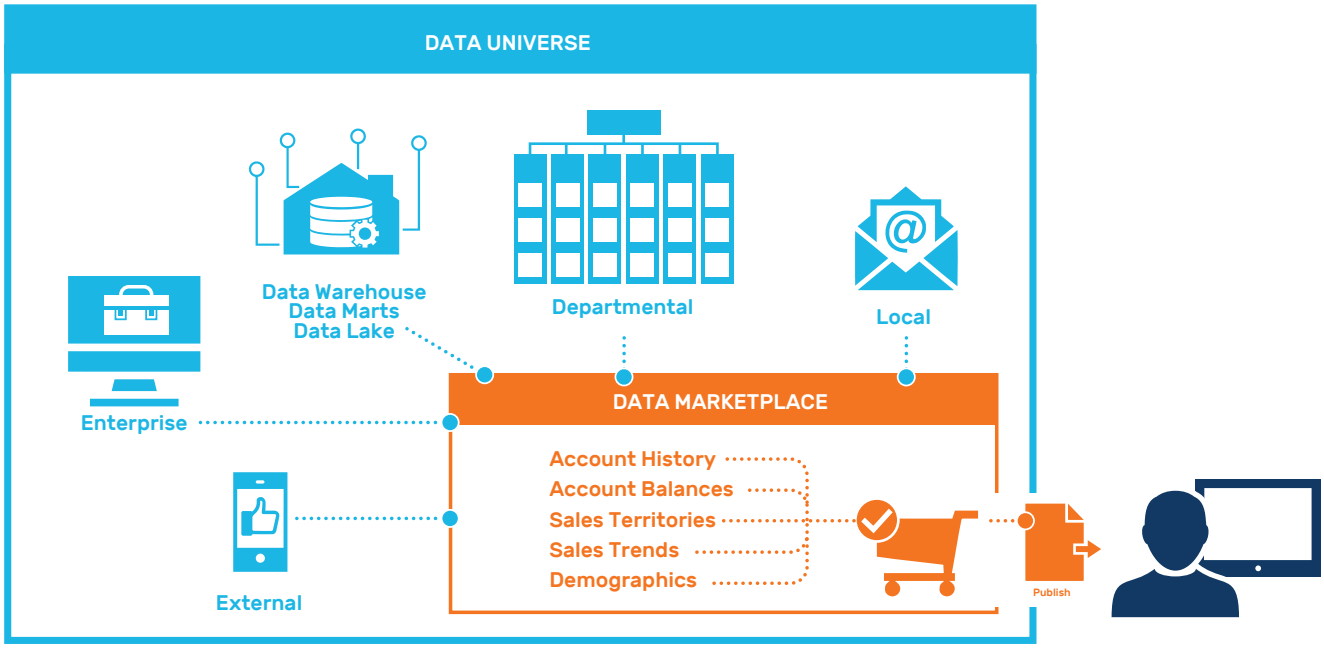
Figure 4: Data marketplace overview



The marketplace allows non-technical users to search, browse, explore and "shop" for needed datasets and to combine and/or store them in the format(s) they find most useful. For example, a data scientist may wish to combine several datasets and publish them as a BFF that can be used by advanced analytics tools or by a marketing analyst who wants to store the data inside an Access database.

In the example below (Figure 5), sales datasets from inside and outside the organization have been made available via the marketplace so the analyst can publish them to a workspace in the preferred format and then combine them however the analyst wants.

Figure 5: Data marketplace example using multi-source sales data



Another big benefit of some data marketplace tools is that they provide the ingredients needed to pay off the investment made in Hadoop data lakes. That is, business friendly metadata and metatags can be generated and applied during the ingestion and curation processes while also securing the data, and if needed, obfuscating sensitive information. The result is readable schemas needed to make sense of the data without having to learn Hadoop-oriented programming constructs. Similarly, different file types can be stored in a common, understandable format such as Hive tables.

One common objection to implementing data marketplaces is that they force organizations to replicate datasets. There are two proven ways of dealing with this issue. The first is to employ a registration construct where only the metadata gets created and stored inside the marketplace while the data itself remains in place in its source. The second way is to use the data marketplace tool to re-ingest datasets already being stored in a Hadoop data lake and deleting the original schema-less datasets. In other words, technical concerns can and should be mitigated in favor of enabling self-service data.

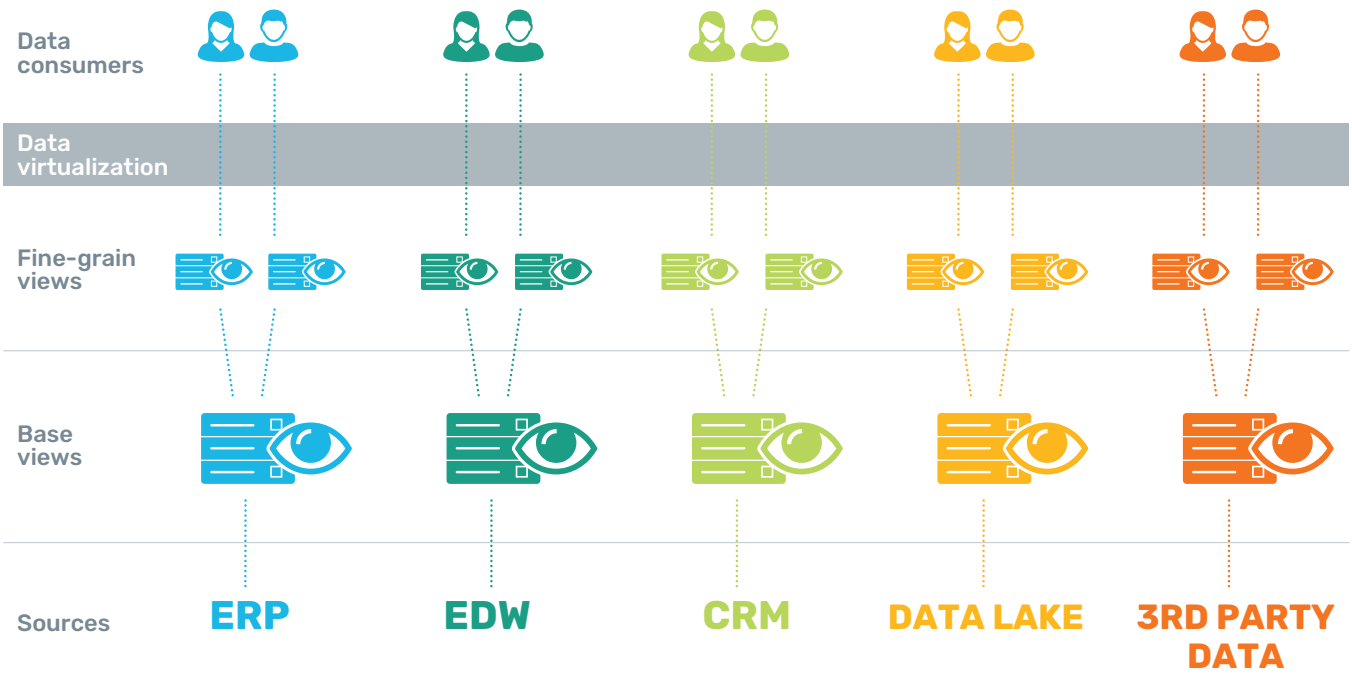
Data virtualization

Data virtualization tools have been around for quite some time, but were, until recently, relegated to what Gartner refers to in their Hype Cycle as the "Trough of Disillusionment." In the past couple of years, these tools have gotten much better with respect to features, functions and performance, and according to Gartner, are now entering the "Plateau of Productivity."

The primary use case that data virtualization tool vendors have been talking about is the "Logical Data Warehouse (LDW)." Mark Beyer, known as the "Father of the LDW,"coined the term in 2011. Gartner, TDWI, and a number of product companies began to promote the LDW as a next-gen architecture in earnest around 2015. Within this construct is the concept of "logical data integration," which is exactly what data virtualization tools are all about.

Data virtualization products have not only matured as an LDW implementation tool, they have also emerged as a self-service data tool. The best analogy is that logical views into data stores can be created much like one would create semantic layers for BI tools. These virtualized layers can provide both base and fine-grain views into all tables within each source system (see Figure 6).

Figure 6: Data virtualization view layers



Base views serve as canonical views into every table within each source system. Why is this important? Because any authorized user can access any table of any system at any time, which removes the barrier of having to anticipate data needs ahead of time. And while the base views provide the foundation for reuse, fine-grain views can be added to address table, row and column-level security.

While many DBAs will be concerned about performance hits because there's no free lunch, even with data virtualization, there are multiple ways to handle this concern. The primary means to ensure performance is to keep a database replicate that is current with change data capture (CDC) or with incremental batch refreshes. The point is that IT must remember to keep the endgame in mind—user convenience and analytical velocity courtesy of data virtualization.

Netting it out

Each evolution in data management and integration has brought progress and new challenges. Since going back in time to solve all the past data problems is not possible, addressing today's challenges requires thinking about data differently and reminding ourselves of the three new rules:

1. Not all data needs to be integrated
2. Data quality is in the eye of the beholder
3. Combining datasets does not always need to be an IT project

It's important to remember that veracity must take precedence over agility and convenience when trusted datasets are a business imperative. Accept that IT cannot keep up with the business's ever-changing demands and perhaps it's better to support shadow IT than to let it go unchecked. At the end of the day, let's remember that self-service data tools are not meant to replace the legacy platforms but rather complement them when appropriate for the use case. Self-service data tools can provide the means for business agility while allowing IT to stay true to its core mission of being good stewards of corporate information.

g2o

what's your next?