denodo | DEVFACTO

The Roles of ETL, ESB, and Data Virtualization Technologies in Integration Landscape

Chapter 1

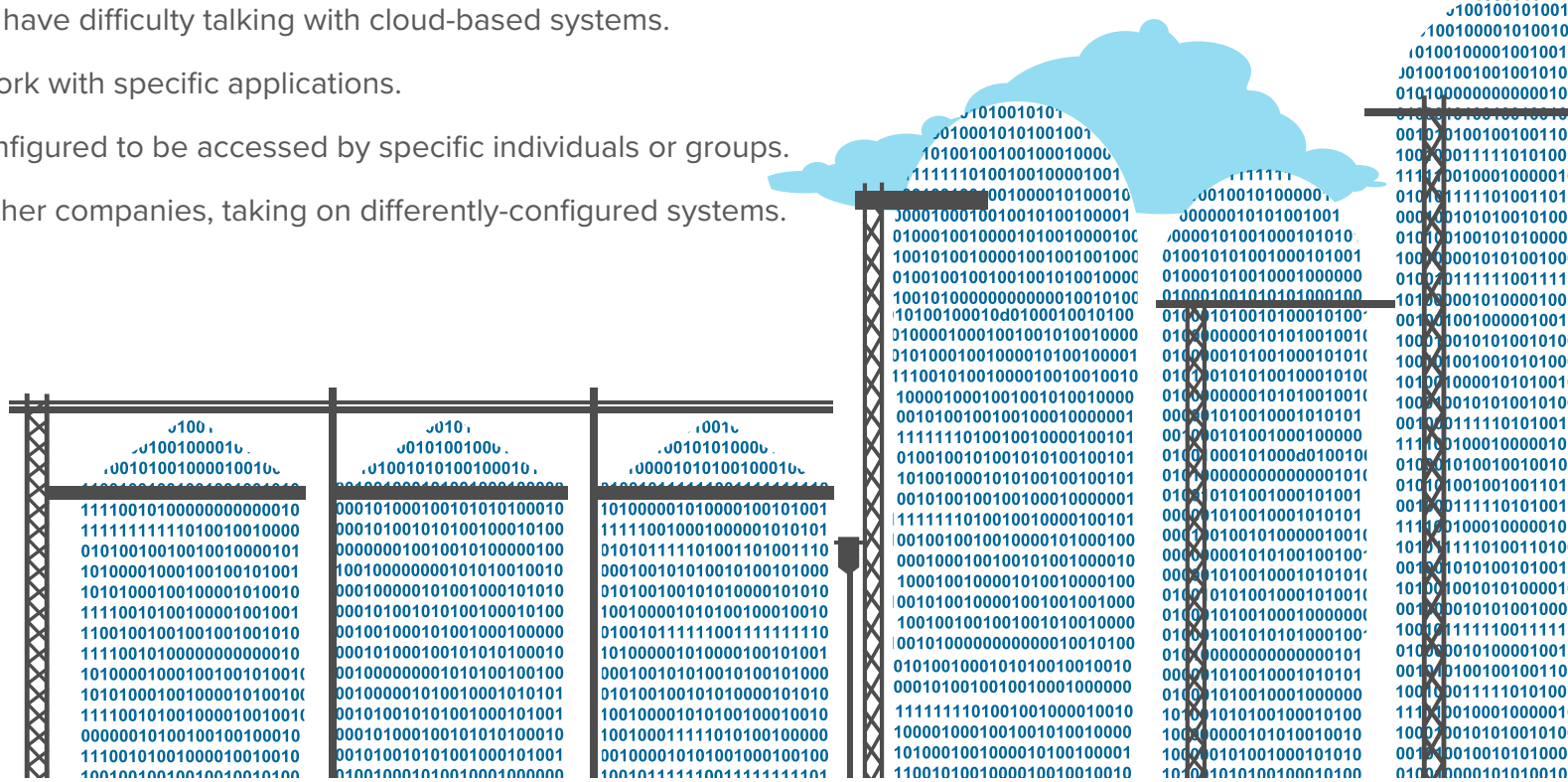
Data Integration

Data Silo Syndrome

The problem of data silos, which are data sources that are unable to easily share data from one to the other, has plagued the IT landscape for many years, and continues to do so today, despite the advents of broadband Internet, gigabit networking, and cloud-based storage.

Data silos exists for a variety of reasons:

- Old systems have trouble talking with modern systems.
- On-premises systems have difficulty talking with cloud-based systems.
- Some systems only work with specific applications.
- Some systems are configured to be accessed by specific individuals or groups.
- Companies acquire other companies, taking on differently-configured systems.



Chapter 1

Bringing the Data Together

The problem with data silos is that no one can run a query across them; they must be queried separately, and the separate results need to be added together manually, which is costly, time-consuming, and inefficient. To bring the data together, companies use one or more of the following data integration strategies:

1. Extract, Transform, and Load (ETL) Processes, which copy the data from the silos and move it to a central location, usually a data warehouse

2. Enterprise Service Buses (ESBs), which establish a communication system for applications, enabling them to share information

3. Data Virtualization, which creates real-time, integrated views of the data in data silos, and makes them available to applications and analysts

Let's take a look at each of these in turn.

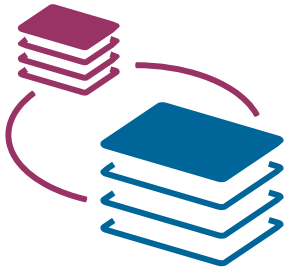
Chapter 2

Extract, Transform, Load - ETL

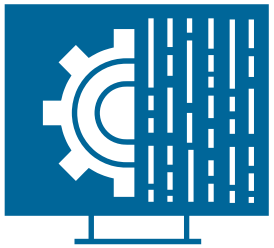
ETL Processes Explained

ETL Processes were the first data integration strategies, introduced as early as the 1970s.

First, the data is extracted from the source.



Next, the extracted copy of the data is transformed into the format and structure required by its final destination.



Finally, the transformed copy of the data is loaded into its final destination, be it an operational data store, a data mart, or a data warehouse.

Some processes do the transformation in the final step, and are therefore called “ELT processes,” but the basic concept is the same.

Chapter 2

Pros and Cons of ETL Processes

- ETL processes are efficient and effective at moving data in bulk.
- The technology is well understood and supported by established vendors.
- ETL tools have features that sufficiently support bulk/batch data movement.
- Most organizations have ETL competencies in-house.

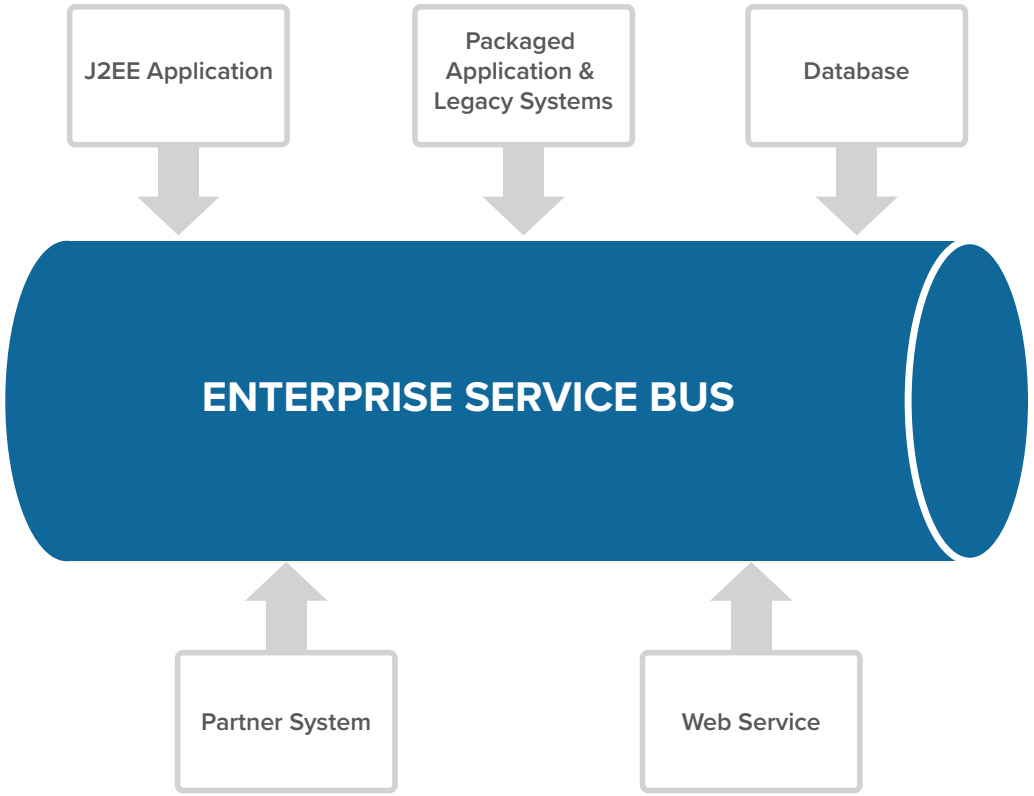
- Moving data is not always the best approach, as this results in a new repository that needs to be maintained.
- Large organizations can have thousands of ETL processes running each night, synchronized by scripts that are difficult to modify if needed.
- Typically, ETL processes are not collaborative; the end user needs to wait until the data is ready.
- ETL processes cannot handle today's data volumes and complex data types.

Chapter 3

Enterprise Service Bus - ESBs

ESBs Explained

ESBs, introduced in 2002, use a message bus to exchange information between applications. With a communication bus sitting between the applications, they can talk to each other by talking to the bus. This decouples systems, and allows them to communicate without depending on, or even knowing about, other systems on the bus. This forms the underpinnings of service oriented architecture (SOA), in which applications can easily share services across an organization. ESBs were born out of the need to move away from point-to-point integration, which, like ETL scripts, are hard to maintain over time.



Chapter 3

The Pros and Cons of ESBs

- Applications are decoupled.
- They can be used to orchestrate business logic using message flows.
- ESB technology is mature, and is provided by established vendors.
- ESBs can address operational scenarios by using messages to trigger events.

- However, ESBs cannot integrate application data to deliver on analytical use cases.
- Queries are static and can only be scheduled; ESBs do not easily support ad-hoc queries.
- Database queries are restricted to one source at a time; joins and other multiple-source functions are performed in memory, which drains resources.
- However, ESBs are only suitable for operational use cases, which involve small result sets.

Chapter 4

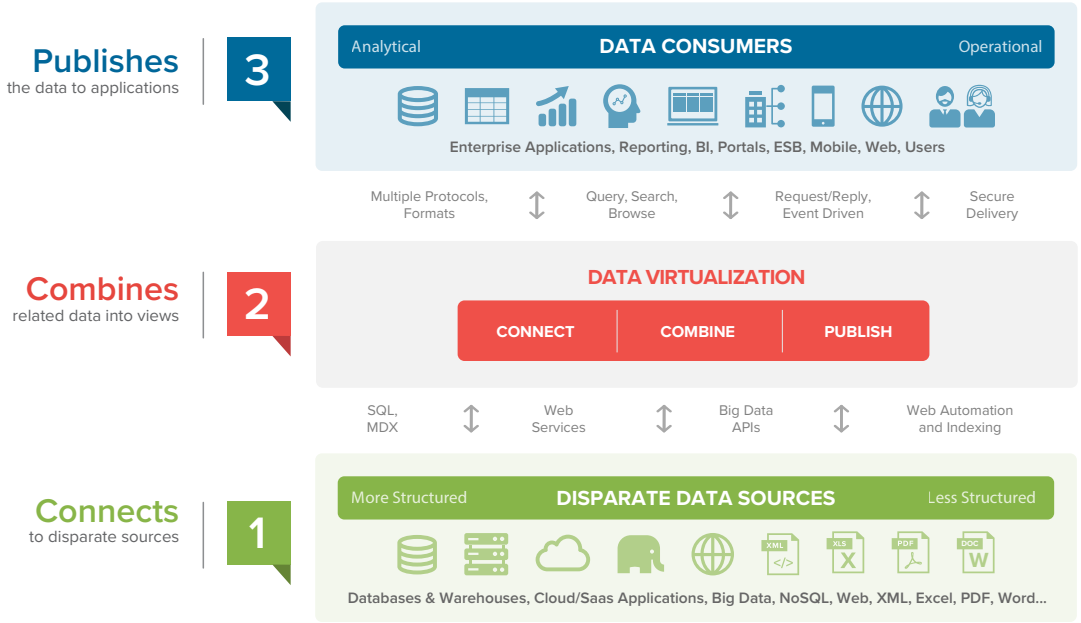
Data Virtualization - DV

Data Virtualization Explained

Data virtualization creates integrated views of data drawn from disparate sources, locations, and formats, without replicating the data, and delivers these views, in real time, to multiple applications and users. Data virtualization can draw from a wide variety of structured, semi-structured, and unstructured sources, and can deliver to a wide variety of consumers.

Since no replication is involved, the data virtualization layer contains no source data; it only contains the metadata required to access each of the applicable sources, as well as any global instructions that organizations may wish to implement, such as security or governance controls.

Users and applications query the data virtualization layer, which in turn gets the data from the various sources. The data virtualization layer abstracts users and applications from the complexities of access, and to all consumers, the data virtualization layer appears as a single, unified repository.



The Pros and Cons of Data Virtualization

PROS

- Seamlessly federates two or more disparate data sources (makes them appear and function as one), including a mix of structured and unstructured sources.
- Adds value added features such as intelligent real-time query optimization, caching, in-memory processing, and custom optimization strategies based on source constraints, application needs, or network awareness
- Via an API, any primary, derived, integrated or virtual data source can be made accessible in a different format or protocol than the original, with controlled access, in minutes
- All data is accessible through a single virtual layer, which quickly exposes redundancy, consistency, and data quality issues, and enables the application of universal, end-to-end governance and security controls.

CONS

- Lack of support for bulk/batch data movement which might be required by a few use cases.

Chapter 5

Data Integration Strategies Compared

Below is a summary of several data integration use cases, indicating which strategies are best suited to each task. In the next chapter, we'll discuss how two strategies can work together in support of various use cases.

Use Case	DV*	ETL**	ESB***
Moving data into EDW or ODS		✓	
Migrating EDW (to Cloud)	✓	✓	
Data Unification	✓		
Cutomer 720°	✓		
Real-time insights	✓		✓
Virtual Data Marts	✓		
Physical Data Marts		✓	
Agile Reporting (from EDW + other sources)	✓		
Logical Data Warehouse	✓		
Data Warehouse Offloading	✓	✓	
Application Synchronization		✓	✓
Metadata Discovery and Enrichment	✓		
Self-Service Analytics	✓		
ETL "seeding" (decouple ETL from sources)	✓		
Event-Driven Workflows			✓

*DV: Data Virtualization.
**ETL: Extract, Transform, Load.
***ESB: Enterprise Service Bus.

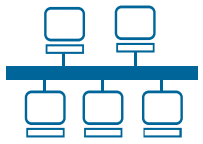
Chapter 6

Integrating Data Integration Strategies

Of each of the three strategies, data virtualization is most adaptable in working with other strategies, since it supports such a wide variety of sources and targets. Let's take a look at how data virtualization works with ETL processes and ESBs.

Data Virtualization and ETL Processes. ETL processes were designed for moving data into data warehouses and similar environments, and they are particularly well suited to this task. But ETL processes cannot easily support cloud-based sources. Data virtualization can complement ETL processes in the following ways:

- Seamlessly connecting on-premises with cloud data sources without the need to consolidate data in a single repository.
- Enabling the migration from on-premises to cloud-based systems without interrupting business continuity.
- Data Warehouse offloading in which data virtualization not only helps with the offloading process, but also unifies data across the traditional data warehouse and the new repository such as Hadoop, AWS S3 or a Cloud-based data store.
- Real-time integration of disparate data sources.
- Replacing ETL processes with data virtualization where faster access to data is necessary.



Data Virtualization and ESBs. Data virtualization can complement an ESB and enhance its performance. Adding new sources to an ESB can be complex; sources like relational databases, Web or cloud-based sources, flat files, or email messages are not immediately enabled for the service oriented architecture (SOA) that the ESB supports. To streamline this process, all sources that the ESB cannot handle can be unified by the data virtualization layer before being passed to the ESB. This architecture exploits the best qualities of both technologies: Data virtualization unifies disparate sources, and ESBs deliver the critical messages to support the business process.

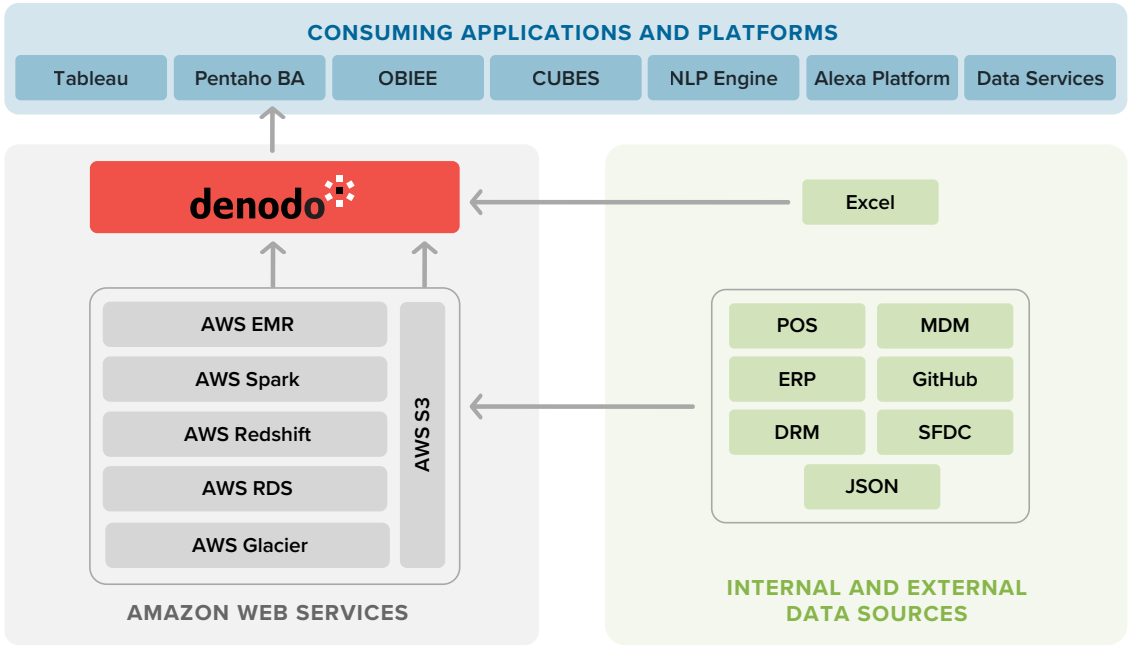
Chapter 7

Case Studies

Logitech: Achieving a Successful Cloud Migration by Complementing an ETL System with Data Virtualization

Logitech is a Swiss global provider of personal computer and tablet accessories. For many years, the company had been developing and delivering data services for analytics using on-premises systems, integrated via ETL processes.

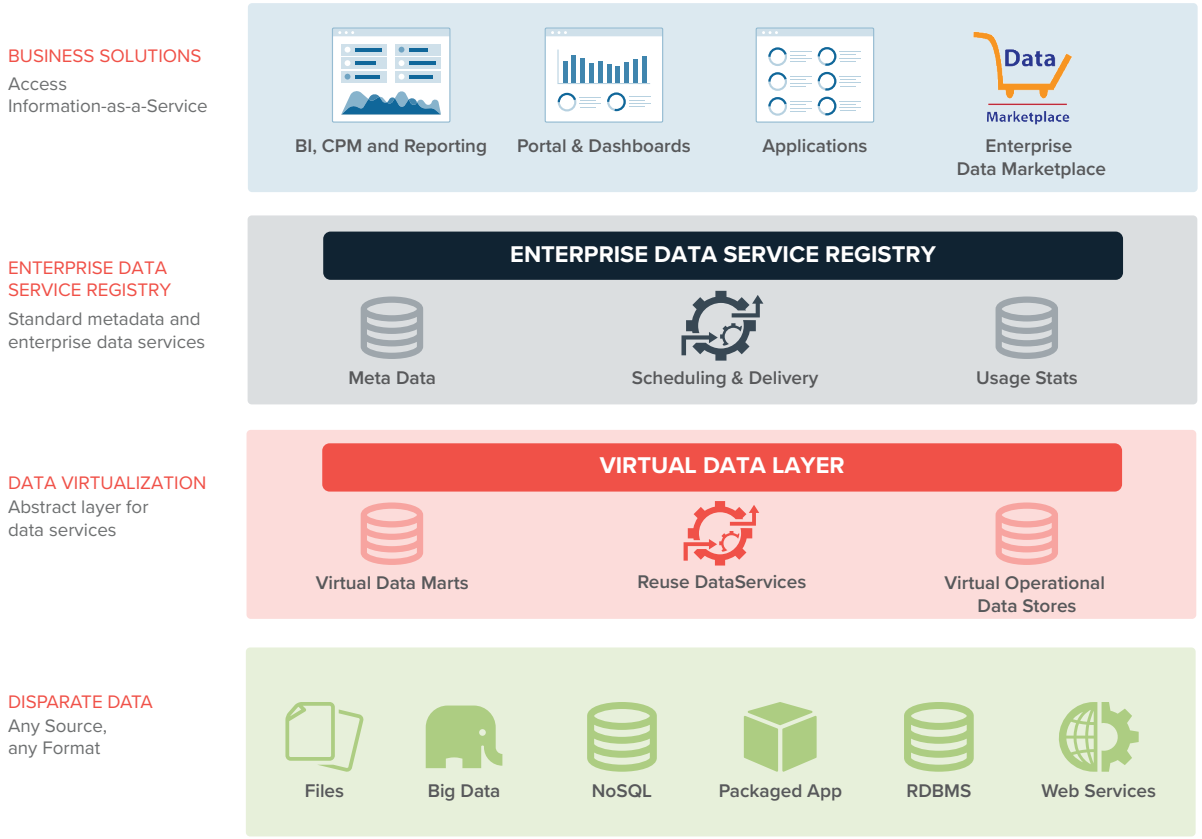
But provisioning data services for business users was proving to be reactive, time consuming, and inefficient. To overcome these limitations, Logitech moved IT operations to the cloud. However, some data sources remained on-premises, so Logitech needed a solution that could seamlessly integrate all of its on-premises, ETL, and cloud components.



Logitech leveraged the Denodo Platform, hosted on Amazon AWS, to establish a data virtualization layer that integrates these sources. After creating a single, consistent data store, the Denodo Platform feeds analytics and reporting applications such as Tableau, Pentaho BA, and web services. In the Logitech infrastructure, the Denodo Platform has become the single source of truth, feeding the entire consumption layer.

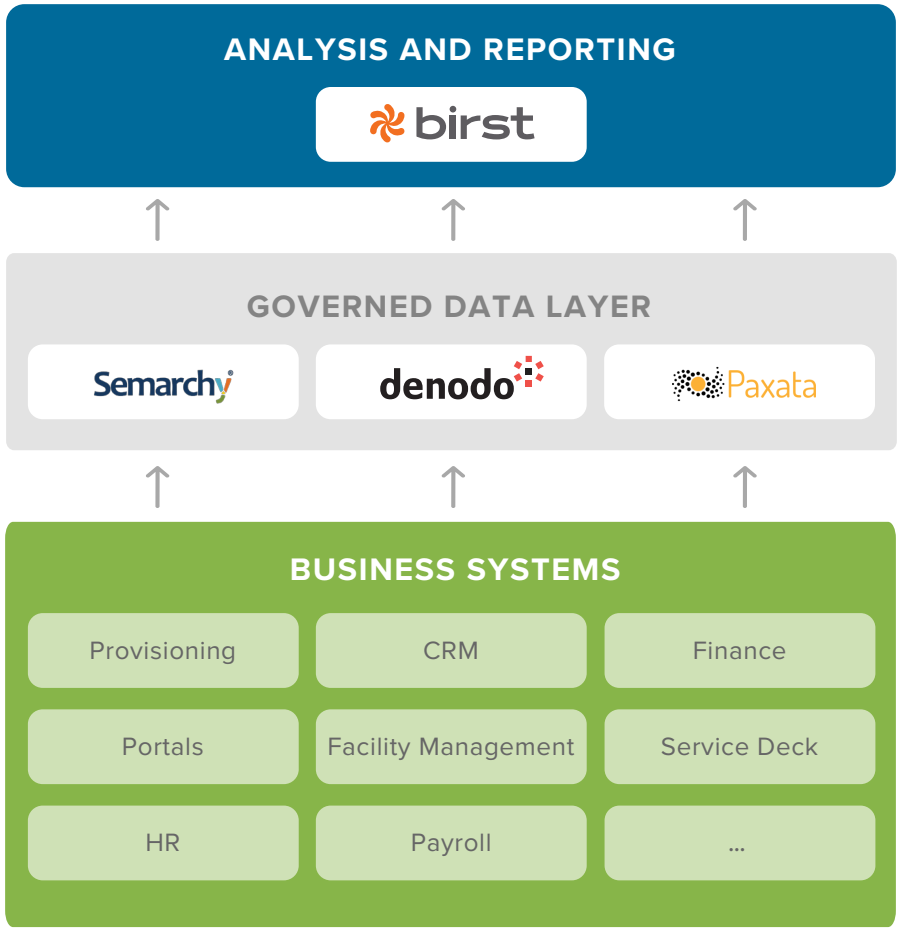
Leading National Life Insurance Firm: Enhancing an ESB with Data Virtualization

A Leading National Life Insurance Firm was integrating data from a variety of heterogeneous sources via an ESB, but stakeholders could not easily change input parameters, adding complexity and latency to the company's Enterprise Data Marketplace, an in-house data mart.



A Leading National Life Insurance Firm deployed the Denodo Platform, which established a virtual data layer that the Data Marketplace UI can access via a web service. The data virtualization layer unifies the data from the heterogeneous sources before passing it to the ESB, in full support of the company's existing workflows, while also enabling stakeholders to dynamically change query parameters and other functions.

Digital Realty: Replacing ETL Processes with Data Virtualization



Digital Realty is a provider of data center acquisition, ownership, development, and operations, as well as of colocation services. For data integration, the company was making extensive use of ETL processes, but felt that these systems were negatively impacting the efficiency and speed with which business users could access information.

Digital Realty replaced the majority of its ETL processes with a single data virtualization layer enabled by the Denodo Platform including ETL processes for MDM. In addition, the data virtualization layer seamlessly aggregates a broad and diverse set of disparate sources to feed Digital Realty's Birst-based dashboards, enabling executives to create financial and operational reports much more easily and quickly.

The data virtualization layer improved data speed-to-delivery fivefold, and enabled Digital Realty to reduce costly ETL processes by more than 90%. By passing all data through a unifying layer, Digital Realty was also able to implement robust governance protocols, with granular control over data lineage.