

A VA Executive's Guide to Data Transformation

Office of Technology Strategies (TS), Architecture, Strategy & Design (ASD)



Introduction

Have you ever needed to transfer data between computer systems? Perhaps you have performed a data query, a request for data from a database? If so, you probably noticed differences in how data is digitally formatted and stored. To mitigate risks in achieving strategic data initiatives, organizations may need to transform those differences.

This TS Note focuses on data mapping and code generation as tools for transforming data for data queries and data migrations. It explores database transformations for developing data warehouses, with examples of how databases are transformed at VA.

Data Transformation

Data transformation is the process of converting a set of data values from a source, or original, format to the required format of a new destination. It involves two inter-related concepts: data mapping and code generation.

Data mapping shows how data from one information system should match data on another system. It provides a visual representation of the process for linking different configurations of data, or data models. This is useful when an enterprise seeks to extract data from one source to consolidate, or combine, it with another, or to consolidate multiple separate databases.

Code generation is the development of the technical instructions, or code, which is required to transform data. Code is generated by a compiler software application, a program that reads the source language of the data and then translates it into the destination language.

Both data mapping and code generation are critical in data querying and data migration, two common processes that require data transformation.

Data Querying

A data query is the request for the retrieval of data from a database. Users might run a data query to compile and analyze data of a particular subject for research purposes, or to retrieve all files related to a particular person. Data querying usually relies on one of three methods:

- Searching a database menu using system-defined parameters
- Specifying the fields and values of a system-generated blank record
- Writing a query in the stylized coding language of the system

Each method of running a data query varies in complexity and flexibility. The level of complexity is determined by the level of specialized knowledge that the user needs to run the query; flexibility refers to the user's ability to define the parameters.

Data Migration

When an enterprise needs to migrate, merge, or consolidate data, due to server replacement, for example, it may process an *entire* database. Within the hierarchy of data, related data is joined in fields; related fields create records; records create files; and files create a database.

The decision on how the enterprise will transform the data values in each source database is critically influenced by the way the source is organized and the business needs of the enterprise for its destination database.

The TS office within OI&T's Architecture, Strategy & Design (ASD) interacts not only with the ASD pillar offices, but also with multiple stakeholders within OI&T and with strategic offices across the enterprise. TS works closely with IT and business owners to capture business rules and provide technical guidance as it relates to Data Sharing across the enterprise, specifically for interagency operability.

Standard Query Language (SQL) & Not Only SQL (NoSQL)

The most common method to organize a database is Standard Query Language (SQL), pronounced as "sequel." SQL is a programming language used to retrieve or update data with a relational database management system – a system in which data is organized into columns and rows. Each row is identified by a unique key; each column is labeled by a value or attribute that is shared by all rows. For example, a sales ledger would include a column that identifies the number of units sold, with rows that list each unique commodity for sale.

A second common method used to organize databases is an enhanced SQL-based coding language called Not Only SQL (NoSQL), a non-relational, distributed, and scalable database format. NoSQL is also open source, meaning that its source code is available with a license from the copyright holder. Unlike relational database management systems, 1

A VA Executive's Guide to Data Transformation

Office of Technology Strategies (TS), Architecture, Strategy & Design (ASD)

NoSQL is modeled to rely on systems that are not defined by tabular column and row relations. NoSQL systems often use tags associated with the individual data files or key-values that map to the file's location in the database.

Data Warehousing

A data warehouse is created by interlinking separate databases from disparate sources as a central repository of integrated data that allows users to access records, enabling greater data sharing and data analysis.

Creating an enterprise shared data warehouse involves a process of data transformation called Extract, Transform, and Load (ETL). The data is extracted from various distinct sources and transformed through cleaning, filtering, and validating. The ETL process allows for the transformation of heterogeneous database sources into a single, central repository.

Data Transformation at VA

There are several specific applications that VA relies on to transform data. One is the eCRUD, the electronic Create, Read, Update or Delete service designed as a part of the Veterans Lifetime Electronic Record (VLER) Data Access Service (DAS) project that provides users with an interface to perform operations on data access layers. It enables in-house COTS, commercial off-the-shelf applications, to gain access to authoritative data sources through the use of an application programming interface (API). These APIs provide a level of abstraction to developers so that they don't require physical access to databases to obtain data. For DAS, eCRUD is responsible for transforming data placed into DAS data stores. The use of the eCRUD functionality will expand to serve as a data access mediator for many different VA databases and data file types.

Another recent development at VA is the adoption of Hybrid Data

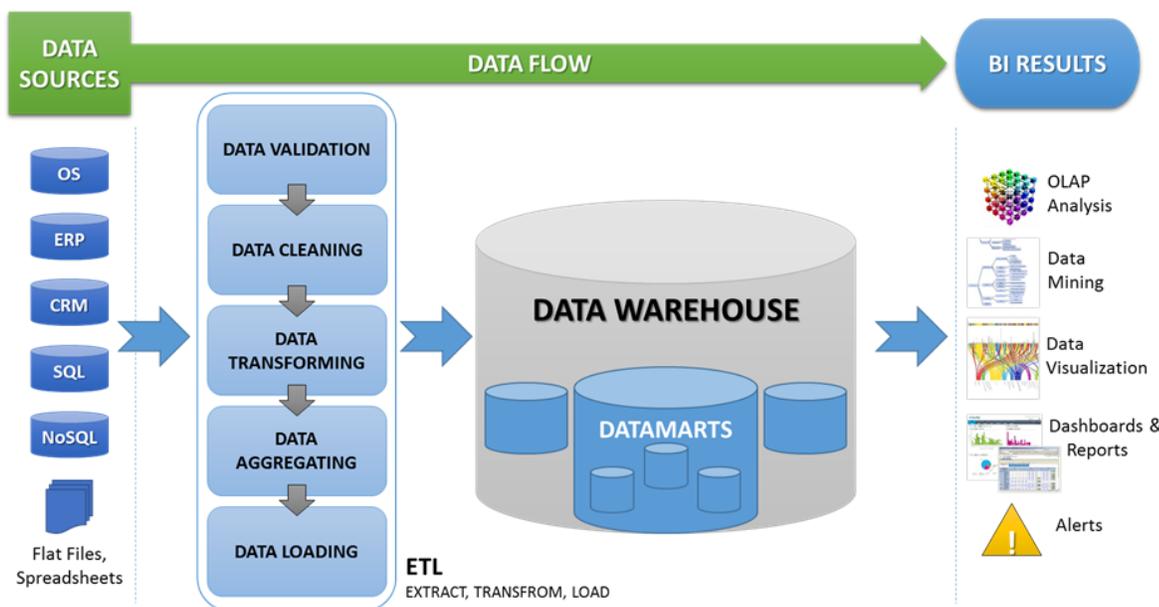


Figure 1 – Data Transformation with Business Intelligence Results

Access (HDA). HDA is a data layer that VA applies to its enterprise shared services – the cost-efficient centralized business operations that are used by multiple divisions within the enterprise. HDA will allow VA to configure a standardized enterprise schema, or model, for data aggregation, whereby data is gathered and expressed in summary form for sharing. Inside the HDA infrastructure, an application called a data ingester performs transformation on data as it migrates from local databases. For more information on HDA at VA, please check out the [Hybrid Data Access Enterprise Design Pattern](#).

At VA, most applications and programs rely on data transformation processes to migrate or query data across different source and destination formats. As in industry, this is necessary because most separate business lines have historically built, maintained, and stored their data on local databases. Through data transformations, the public and private sectors are able to achieve goals of data integration and interoperability - the ability of different information technology systems and software applications to communicate, exchange data, and use the information that has been exchanged. The more data-informed an enterprise, the more enabled it is to make effective decisions.

Read more on data transformation and related topics in the Office of Technology Strategies' [TS Notes](#) and [Enterprise Design Patterns](#). If you have any questions about data transformation, don't hesitate to [ask TS](#) for assistance.