BLOG

Merging Data from Multiple Sources – Challenges and Solutions

Written by **lbarrera(https://dataladder.com/author/lbarrera/)** November 15, 2021

Merging data from multiple sources

Data merging is a process where data is unified from multiple sources to represent a single point of reference or a single point of truth. Although a seemingly simple goal, data merging is a process as complicated as untangling a ball of knotted yarn. The reason? An average enterprise has around 464 custom applications (https://www.skyhighnetworks.com/cloud-security-blog/every-company-is-a-software-company-today/) deployed to handle their business processes.

In addition to that, most databases have problems with duplication, inconsistencies, and inaccuracy. A customer's name and address may be written in 10 different ways by 10 different people in 10 different data repositories. Moreover, there will be 10 different sources from where this data is obtained – websites, landing pages forms, social media advertising reach outs, sales records, billing records, marketing records, purchase point records and other areas where the customer may have interacted with the company.

But when it comes to deriving useful insights from the data collected, it becomes important to combine all these data sources and get one single point of reference.

In this article, we'll cover essential topics such as:

- Scenarios where merging data becomes necessary
- How to merge data from multiple sources
- Challenges encountered during the data merge process

In this blog, you will find:

Merging data from multiple sources
Scenarios where merging data becomes necessary
1. Digital transformation initiative
2. Driving business intelligence
3. Integrating data after mergers and acquisitions
How to merge data from multiple sources
Pre-merging process
Merging process: Data aggregation/integration
Post-merging process
Challenges encountered during the data merge process
1. Data heterogeneity
2. Scalability
3. Duplication
4. Lengthy merging process
Best practices to enable smooth data merging
1. Know what to integrate
2. Visualize data
3. Try automated, self-service tools

IBM v. Digital Doors, IPR2023-00970, 7,552,482
DigitalDoors EX2031

- Best practices to enable smooth data merging

Let's get started.

Scenarios where merging data becomes necessary

Let's elaborate on the importance of merging data from multiple sources in three different scenarios:

1. Digital transformation initiative

Data merge is required while moving disparate files (such as text files, CSVs, excel sheets, SQL databases, or other file formats) to a full-fledged data hosting and processing system, and enabling automated workflows, enhancing search capability, controlling information access, and more.

How legacy systems and bad data quality hinders a digital transformation plan

Read this whitepaper to find the middle-ground solution – giving up a legacy system and improving master data quality.

Download (https://dataladder.com/whitepapers/how-legacy-systems-and-bad-data-quality-hinders-a-digital-transformation-plan/)

2. Driving business intelligence

Data merge usually happens while combining data residing in different applications (such as CRMs, marketing automation tools, website analytics tools, etc.), and merging it to prepare for further data analysis and processing, and extracting useful insights for future predictions.

3. Integrating data after mergers and acquisitions

Mergers and acquisitions include complex moving parts, and one of the most complicated steps is combining data from different companies into one repository, and then making processes compatible to newly merged projects, structures, and workflows.

How to merge data from multiple sources

Merging data should be treated as a closely monitored and controlled process to avoid data loss or causing irreversible damage to individual data structures involved. Below we have highlighted the three stages of merging data:

Pre-merging process

1. Data profiling

Without profiling individual data sources, it's difficult to understand the data that you are dealing with or the impact of your decisions during the merge process. A data profiling (https://dataladder.com/data-profiling/) process gives insight into two important parts of your data:

a. Attribute analysis

It identifies and highlights the list of attributes (data columns) that each data source contains. This helps you to understand how your merged data might scale vertically, depending on which attributes may be merged and which need to be appended as they represent separate information.

b. Statistical analysis of each attribute

It analyzes data values contained in each column of a source to assess the distribution, completeness, and uniqueness of attributes. Furthermore, a data profile also validates the values against any defined pattern and highlights invalid values.

Precisely, data profiles calculate and present the current state of your data sources, as well as highlight potential cleansing (https://dataladder.com/data-cleansing-software/) opportunities and other considerations before the merge process can actually begin.

2. Data cleansing, standardization, and transformation

It is impractical to merge data sources that contain incomplete, inaccurate, or invalid values. Data attributes in two different sources may conceptually represent the same information, but the format of their data values may be completely different. These structural and lexical differences in data may cause data loss and unfixable errors if the data is merged without being cleaned and standardized (https://dataladder.com/data-standardization-guide-types-benefits-and-process/). By using the generated data profiles as a reference point, a number of steps are taken to standardize data (https://dataladder.com/data-standardization-software/) quality, a few of them are highlighted below:

- Replacing invalid characters with correct values (for example, replacing non-printable characters, null values, leading or trailing spaces, etc.)
- Parsing long data fields into smaller components to standardize data across multiple sources (for example, parsing *Address* field into *Street Number*, *Street Name*, *City*, *Zip Code* and *Country*). Parsing the data fields in this manner ensures that data accuracy (https://dataladder.com/what-is-data-accuracy/) is maintained after the data is merged.
- Defining integrity constraints, pattern validations, and allowed data types for all attributes (for example, max/min number of characters allowed for the *Number* field, that can only contain numbers or dashes in a pattern such as *NNN-NNN-NNNN*).

3. Data filtering

You may want to just merge subsets of your data sources instead of performing a complete merge. This horizontal or vertical slicing of your data is usually needed when:

- You want to merge data falling in certain time period (horizontal slicing).
- You want to merge data for analysis and only a subset of rows qualifies the conditional criteria (horizontal slicing).
- Your data contains a number of attributes that don't contain any valuable information (vertical slicing).

In case you wish to merge all data without leaving out anything, you can skip to the next step.

4. Data deduplication

Quite sometimes, organizations tend to store a single entity's information across multiple sources. Each of these records represent some valuable data about that entity. Merging data becomes difficult if your datasets contain duplicates. Before starting the merge

process, it is crucial to perform suitable data matching algorithms to identify duplicates, apply conditional rules to delete duplicates, and attain uniqueness of records across all sources.

Merging process: Data aggregation/integration

Data is now ready to enter the merge process. Depending on the purpose, data can be merged in different ways:

- Appending rows
- Appending columns
- Appending rows and columns both
- Conditional merge

Let's cover each of these scenarios in a bit more detail.

1. Append rows

Appending rows is useful when you want to gather and combine records captured from different sources at one place.

An example of appending rows is when you have gathered customer information through multiple contact management systems, but now you need to bring all records at one place.

Considerations

- All data sources to be combined should have the same structure (columns).
- Data types, integrity constraints, and pattern validations of corresponding columns should be the same to avoid invalid format errors.
- In the presence of unique identifiers, ensure that different sources don't contain the same unique identifiers, otherwise it will raise errors during the merge process.

- If an entity's data spans multiple records residing at disparate sources, perform data matching (https://dataladder.com/definitive-guide-to-data-matching/) and deduplication (https://dataladder.com/data-deduplication-software/) prior to the merging process.

2. Append columns

Appending columns is useful when you want to add more dimensions to existing records.

An example of appending columns is when you have your customer's online contact information in one dataset, and their physical or residential contact information in another, and you want to combine both datasets into one.

Considerations

- All columns of disparate sources should be unique (not duplicates).
- Every record should be unique identifiable across all datasets so that records with the same identifier can be merged.
- If a dataset does not contain data for the merging column, then you can specify null values for all records in that dataset.
- If multiple datasets contain the same dimension information, then you can also merge dimensions together in one field (separated by a comma, etc.) in case you don't want to lose data.

3. Conditional merge

Conditional merging is useful when you have incomplete datasets that need to be merged. In this type of merging, you look up values from one dataset and appropriately fill them in the other datasets against the correct record/attribute.

An example of conditional merge is when you have list of products in one dataset, but the average sales per month for each of them is captured in another dataset. Now to merge data, you may need to look up each product sales from the second one and append this data against the correct product record in the first dataset. This is usually done when you don't have unique identifiers in one dataset and so you have to conditionally compare based on another column and merge accordingly.

Considerations

- The dataset from where you are looking up values should contain all unique records (e.g., one average sales number for each product).
- The dataset that you are appending data to can be non-unique (e.g., products are listed by location and so the same product sold at multiple locations may be listed more than once).

Additional note

The type of merging you use highly depends on your specific use case. If your datasets do not contain a lot of null values and are relatively complete, appending rows or columns, or both may fit your requirements. But when you encounter gaps in your datasets, you may need to look up and fill those values first. Quite often, organizations use all data merging techniques to bring their data together. For example, you can perform conditional merge first, and then complete the merge process by appending rows and columns as well.

Post-merging process

1. Profiling merged source

Once the entire merge process is complete, it is crucial to perform a final profile check of the merged source – just like the one performed in the beginning of the process for individual sources. This will raise any errors encountered during the merging, and highlight whether any information is left incomplete, inaccurate, or contains invalid values, etc.

Challenges encountered during the data merge process

1. Data heterogeneity

One of the biggest challenges encountered while merging data is data heterogeneity – the structural and lexical differences present across datasets to be merged.

a. Structural heterogeneity

When these datasets do not contain the same number and types of columns/attributes, this is known as structural heterogeneity. For example, one database could be storing a contact's name as *Contact Name*, while in a second database, it is stored in multiple columns such as, *Salutation, First Name*, *Middle Name*, and, *Last Name.*

b. Lexical heterogeneity

Lexical heterogeneity occurs when the when fields of different databases are structurally the same, but they represent the same information in a syntonically different manner. For example, two or more databases can have the same *Address* field, but one can have an address value: *32 E St. 4*, while the other can have *32 East, 4th Street*.

To overcome this challenge, columns must be parsed and merged to achieve the same structure across datasets. In addition, the column values should be transformed to follow the same syntax.

2. Scalability

Usually, data merge initiatives are planned and implemented by keeping a set number of sources and types in mind, and have no space for scalability. This is a huge challenge as organizational needs transform over time, and they require a system that can integrate more data sources with varying structures and storage mechanisms.

To overcome this challenge, it is important to implement and utilize a scalable integration design during the merge process, rather than hardcoding the integration to certain sources only. A reusable data integration system considers current and future possibilities and builds a scalable architecture that pulls data from a number of sources and supports different data formats, such as APIs, SQL databases, text files, ETL warehouses, etc.

3. Duplication

No matter whichever data merge (https://dataladder.com/merge-purge-software/) technique you are planning to implement, data duplication is a big challenge that needs to be fixed. Duplication may exist in your database in different forms, some common ones include:

- Multiple records representing the same entity (that may or may not have unique identifier).
- Multiple attributes storing the same information about an entity.
- Duplicate records or attributes stored within the same dataset, or spanning across multiple datasets.

This duplication problem can be solved by:

- Selecting and configuring suitable data matching algorithms (https://dataladder.com/benefits-data-matching/) that identify records representing the same entity. In absence of unique identifiers, a combination of advanced fuzzy matching (https://dataladder.com/fuzzy-matching-101/) algorithms must be used to find accurate matches.
- Defining a set of conditional rules that intelligently assess same or similar columns and suggest which of these attributes contain more complete, accurate, and valid values.

4. Lengthy merging process

Data integration (https://dataladder.com/data-integration-explained/) processes tend to run longer than expected. The most common reason behind this is poor planning and unrealistic expectations set at the beginning. It is easier to start something from scratch than to correct what has already been implemented, and has been in effect for decades.

It is crucial to consider the amount of data you are dealing with, and assess the data profiles of all sources before you can realistically build an implementation plan.

Another reason behind long-running integration projects is last minute additions or changes. The team must spend some time in the pre-evaluation process and collect information from all stakeholders involved, such as business users (that enter/capture the data), administrators (that manage the data), and data analysts (that make sense of the data).

Best practices to enable smooth data merging

1. Know what to integrate

Prior to initializing the data integration process, spend some time evaluating the data sources involved. This helps in identifying exactly what to combine – the sources and the attributes within them. It could be that old records are outdated and should not be considered for the integration process. Knowing exactly what to combine will increase the speed and accuracy of the process.

2. Visualize data

It is always best to understand the data you are dealing with, and the quickest way to do so is to visualize it. It not only is easier to assess, but gives you a complete view of any outliers or invalidations that may be residing in your dataset. Statistical histograms and bar charts showing completeness of attributes can be very useful.

3. Try automated, self-service tools

Manually carrying out the entire data integration and aggregation process seems like a resource and cost intensive process. Try automated, self-serve data integration tools that offer all-in-one solution for quick, accurate, and in-depth data profiling, cleansing, matching (https://dataladder.com/data-matching-software/), integration, and loading.

Data Ladder's DataMatch Enterprise (https://dataladder.com/products/datamatch-enterprise/) is one such tool that seamlessly supports a variety of data types and formats, including local files (text files, CSV, excel sheets), databases (SQL Server, Oracle, Teradata), cloud stores (CRMs such as Salesforce), APIs, and other databases using ODBC connection, as well as building native connectors based on specific user needs.

4. Decide where to host merged data

You may want to merge data from all sources into one destination source, or load the merged dataset to a completely new source. Depending on your requirement, make sure that you have tested, designed, and structured the destination source accordingly, so that it can withstand the incoming merged data efficiently.

A Fortune 500 Company uses DataMatch Enterprise

A Fortune 500 Company fast tracks merger process with company two months ahead of deadline.

Read case study (https://dataladder.com/case-studies/a-fortune-500-company-fast-tracks-merger-process-with-company-two-months-ahead-of-deadline/)

Enter your work email

Try data matching today

No credit card required

DOWNLOAD

WANT TO KNOW MORE?

Check out DME resources

(https://dataladder.com/address-standardization-guide/)

Address standardization guide: What, why, and how? (https://dataladder.com/address-standardization-guide/)

lbarrera - October 31, 2022

(https://dataladder.com/what-is-data-integrity-and-how-can-you-maintain-it/)

What is data integrity and how can you maintain it? (https://dataladder.com/what-is-data-integrity-and-how-can-you-maintain-it/)

lbarrera - October 20, 2022

(https://dataladder.com/guide-to-data-survivorship-how-to-build-the-golden-record/)

Guide to data survivorship: How to build the golden record? (https://dataladder.com/guide-to-data-survivorship-how-to-build-the-golden-record/)

lbarrera - October 10, 2022

Resource Center (https://dataladder.com/resources/)

(https://dataladder.com)

Quick Links

About us

Help Docs

Partners

Changelog

Contact

+1 (516) 468 6879 (tel:+1 866 557 8102)

Data Ladder offers an end-to-end data quality and matching engine to enhance the reliability and accuracy of enterprise data ecosystem without friction.

Services

All Resources

sales@dataladder.com (mailto:sales@dataladder.com)

www.facebook.com/D (https://twitter.com/d www.linkedin.com/com

© DataLadder 2023

Privacy Policy (https://dataladder.com/privacy-policy/)