Integrating Data

Bill Inmon

Patty Haines

David Rapien

Technics Publications

115 Linda Vista
Sedona, AZ 86336 USA
https://www.TechnicsPub.com

Edited by Jamie Hoberman
Cover design by Lorena Molinari

First Printing 2022

ISBN, print ed. 9781634622820
ISBN, Kindle ed. 9781634622837
ISBN, ePub ed. 9781634622844
ISBN, PDF ed. 9781634622868

Library of Congress Control Number: 2022941429

Contents

CHAPTER 1

Integration

If one subject is universally disliked in the world of technology, it is the subject of integrating data. Vendors hate integration. Consultants hate integration. End-users hate integration. Once data arrives in a silo of data, it never comes out.

Data arrives in the corporation from many sources. Once the data exists in digital form, the analyst/end-user believes the data. It is a shock to the end-user that the data they are using is incorrect. And the end-user finds that it is very difficult to make informed decisions on unreliable data.

So what kind of data is subject to the many different inaccuracies that plague data in the corporation? The answer is – all kinds of data are subject to inaccuracy.

Inaccuracy of data

There are many reasons for data inaccuracy, such as entering, calculating, or copying data incorrectly.

Lack of integration

But the biggest cause of inaccurate, unbelievable data is the lack of proper integration across the enterprise. When data is not integrated, it is entered into one location and can't be used in another. Usually, unintegrated data exists in large silos of information. These silos of information allow data to be used and understood only within the context of the silo. The moment data steps outside of the silo, it loses context and meaning.

For example, in a large corporation, a person wants to find out information about a customer for automobiles. The analyst enters one system looking for the customer and finds nothing. Then the analyst enters another system for people who buy tires and finds someone with a similar name. But it isn't the right person. Then the analyst enters yet another system and finds the right person, and it only has data on gasoline purchases. But gasoline purchases are for much smaller amounts than an automobile. By this time, the customer had left and found another dealership and the analyst forgot why they were looking.

Not finding accurate data quickly and easily has very negative consequences.

There is much value in having an enterprise view of accurate and complete data. And this is true for all organizations.

The problem is that integrating data is difficult and complex. Integrating requires hard work and sleuth investigation. Integration requires making mistakes and correcting them. However, integrating data is absolutely necessary to achieve a true enterprise view of data.

You cannot have an enterprise view of data as long as the data is not integrated.

There are no shortcuts. There are no alternatives. There are no silver bullets. There are no easy paths out.

So how did data ever get to be so messed up?

1. We design applications with no thought of integration. We build each application with its own symbology, names, calculations, and encoding algorithms. When the application was built, no thought was ever given to any standardization across the enterprise.

2. There is much data in the form of text. When people write and talk, they don't speak in terms of integrated terms, keys, or attributes. Such a conversation would sound very unusual indeed.

But these conversations still hold tremendous value to the corporation.

3. Organizations change, merge, and split apart. One day a system is aligned with another system that was once a competitor. We never planned for these systems to be integrated, so it is no surprise that the systems don't fit together.

4. Time passes and business changes. There are new laws, new competitors, and new economic conditions. If there is one immutable law of nature, that law is that over time, business conditions change. And this law is true for all businesses.

5. There are mentality silos. When managers make decisions, they rarely look across departments because they are using their budget for their department. They don't want to take money from their employees to give it to someone else. Let someone else fund integration. Most of us understand our jobs only in relation to our work area. Accountants understand accounting, taxes, audits, and rules. Marketing people understand sales, commissions, marketing campaigns, and advertising. Operations managers understand manufacturing, supply chain and queuing theory. Data analysts understand data and sometimes

statistics. Developers understand code and data structures.

6. There is near-sightedness. Not many developers have studied or worked in Marketing, Accounting, or Operations. So, when they design the systems, tables, attributes, and data storage, there is no need to consider integration with the rest of the organization. Often, they do not even understand the jobs of the department for whom they are writing the software. Even within the silo, they often do not personally interact with the real end-users. Instead, they have some manager who is a go between. Data analysts are often handcuffed with similar constraints. The results of new systems being developed within the silos are good for the singular task(s), but detrimental to the integration of organization-wide data.

7. People don't like change. Organizations often build or acquire their software to achieve a particular task. They depend on this software to run their departments. The employees become trained and familiar with the software and people do not like change. As a result, we become more reliant on these systems and they become unchangeable and indispensable. The problem is that the organization is now at the complete mercy of the consultants or software providers.

8. People like power. Organizations often purchase large MRP, ERP, or other vendor packages to run their businesses. We are sold a system that promises to be all things to all people across all departments. The salespeople demonstrate a high level of marketing, finance, accounting, and ops integration. They sell us on best-of-breed and comprehensive reporting. What is often not shown or talked about is that they have built their application from buying many disparate systems that do not natively talk to each other or share integrated databases. So, when we ask for our organization-wide and integrated reports, they come with a caveat that "we have that report in our production list. The system replaced our legacy systems and now we cannot get out from under their power.

The only difference from one business to the next is the rate and scope of change. But business changes all the time. It simply is a fact that one day the organization awakes to find that there is no integrated view of data in the corporation. And this lack of integration puts management in a precarious position. Not having integrated corporate enterprise data is like flying an airplane in the clouds without instrumentation. It is just a matter of time before a disaster occurs.

Integrated data allows us to answer basic and important questions as:

- How many customers do we have?
- What products do we have?
- What revenue do we have?
- Are we growing? Or are we losing customers?
- Are we maintaining customer loyalty?

Running a business without being able to answer these basic questions is very difficult to do.

What is needed is corporate data (or enterprise data), not application data or data gathered in a conversation.

The challenge of data integration is that it is complex in the best of cases and bewildering in the worst of cases.

Many organizations decide not to integrate their data due to the high level of complexity and effort.

Spider web systems

What happens to organizations that decide not to integrate their systems? Ultimately the organization ends up with a "spider web" environment.

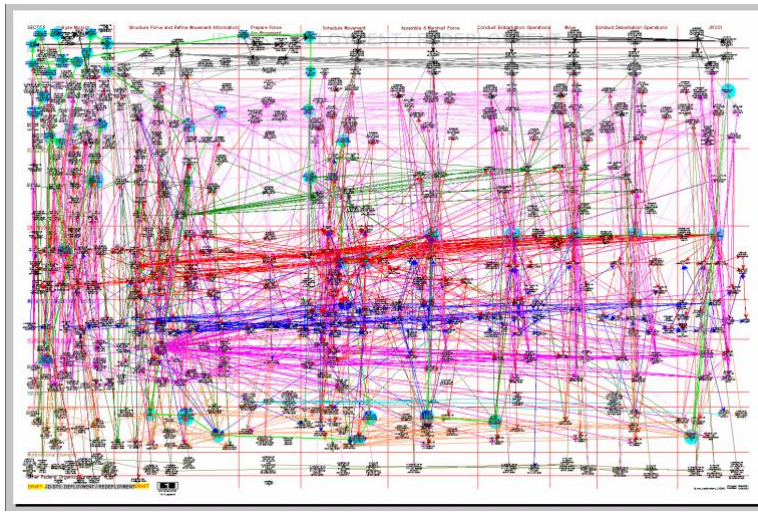

Figure 1-1. In the spider web environment, applications and extract programs toss data from one application to the next. Soon there will be many extract programs connecting many applications.

So what is so terrible about a spider web architecture?

There are lots of things wrong with the spider web architecture:

- **Lack of data integrity**. In a spider web environment no one knows the actual value of any given element of data. The data ends up being shuffled around from one application to the next through the process of extraction. The result is that the same attribute ends up existing in multiple places. The problem is that the attribute has a different value in all the places in which it exists. In one place, the attribute has a value of 650. In another place the attribute has a value of 5000. In another place the attribute has the value of 50.

- **Inability to find the real value of an attribute**. Not only are attributes scattered around like leaves in the fall time, but the data cannot be reconciled. In other words, finding an attribute's actual value is very difficult, if not impossible.

- **Difficult to fix**. It is challenging to repair the spider web environment. Most managers add hardware, software, and consultants to solve the problems of the spider web environment. This makes the spider web environment even worse.

- **Easier to ignore textual data**. What happens to a corporation when it ignores its textual data? The organization is ignoring 90% of its information. This includes the voice of the customer, corporate contracts with all of their liability, and warranty fulfillment which can improve the manufacturing process by understanding defective parts, insurance claims processing, and medical records.

The net result of an organization not integrating their data and ignoring textual data is that the organization is flying its airplane in the clouds with no instrumentation. When that happens, it is a short time until disaster strikes. Stated another way, when an organization integrates its structured data and incorporates its textual data, the organization ends up being in a proactive position. Once an organization finds itself in a proactive position, it can anticipate customers and adjust to changing market conditions.

Reasons for complexity

There are many reasons for the complexity the analyst uncovers when integrating data.

For structured data:

- There is no documentation of what data in a system means
- There is documentation, but it is out of date
- The only documentation is in old code in a language that no one uses or understands anymore
- Documentation of systems must be done at several levels
- The semantic level
- The physical level
- The inclusion/exclusion level
- The calculation level

For textual data:

- Different languages must be accounted for
- Slang must be taken into account
- Misspellings must be considered
- The context of text must be understood
- The dialects of language must be taken into consideration

Of all of these considerations, the most important is the context of text. You need context to understand what is being said. Merely trying to decipher and understand text

by itself is simply not adequate. Instead, we must consider text and context when integrating text into a database.

Transformation of data

In its simplest form, integration is nothing more than data transformation. We transform data from application data to corporate data or text to corporate data. The data is then in a singular and integrated state, creating a true organizational picture.

The challenge with context is that it is normally found outside of the text.

To understand what is meant by context existing outside of text, consider the following simple example:

Two men are talking about a lady, and one says to the other, "She's hot."

Now, what is meant by "she's hot"?

One possibility is that he finds the lady attractive. Another possibility is that it is Houston, Texas in July, and the lady is sweating and physically hot. A third possibility is that the men are doctors. One doctor has just taken her temperature, and she has a temperature of 104 degrees Fahrenheit. In addition, she has a fever and is internally hot.

To make sense of the words "she's hot," you need to know:

Are the men doctors?

Is the conversation occurring in Houston, Texas in July?

Do they find the lady attractive?

Those factors are external to the text, "She's hot." To understand the context, you must know many things that are not directly related to the spoken or written text. The external factors allow the words to have meaning.

The understanding of the external context of the words allows the words to be understood.

Trying to understand the context by looking internally (spoken word), will not lead to a proper interpretation of what is being said. And without a proper interpretation, it is impossible to properly integrate the text into the corporation's data on which it makes decisions.

Summary

Once data is integrated, the data becomes a foundation on which the corporation can make decisions and have confidence in those decisions. A foundation of integrated data allows the corporation to:

- Have confidence in the data on which decisions are made
- Allows the data to be accessed consistently
- Allows the data to be accessed quickly
- Allows anyone in the corporation access to the data

In a word, integrated corporate data forms the foundation for reliable decisions. A corporation can do meaningful business intelligence when it has integrated data. Stated differently, without an integrated foundation of data, an organization only has guesswork to guide decision-making.

Integrating data leads the organization to an interesting place. As long as data is unintegrated, the organization has to make decisions reactively. But once integrated data becomes available, organizations can operate proactively.