Causes and Consequences of Data Quality Problems

  | Business Analysis Essentials

richard rennehan

2017

Contents

[PROCESSES CAUSING DATA PROBLEMS 3](#_Toc498435101)

[Outside Data Processes 3](#_Toc498435102)

[Processes Changing Data from Within 4](#_Toc498435103)

[Processes Causing Data Decay 5](#_Toc498435104)

[POSSIBLE WAYS TO REDUCE DATA ISSUES 6](#_Toc498435105)

[Outside Data Processes 6](#_Toc498435106)

[Processes Changing Data from Within 6](#_Toc498435107)

[Processes Causing Data Decay: 7](#_Toc498435108)

[REFERENCES 8](#_Toc498435109)

## Processes Causing Data Problems

Processes that can cause data problems are grouped into three overall categories. The categories are: processes that bring data from outside, processes changing data from within an organization, and processes that cause data decay.

It’s important to realize the significance of data problems first. A data problem is a piece of data that is inaccurate or is full of errors. Our world is powered by data to a huge degree. Almost every activity you do, such as withdrawing money from a bank to writing a test, is powered by data. Imagine suddenly losing access to your bank account, or getting an incorrect mark back on a costly test. A single data error, at worst, can spread and cause destruction.

### Outside Data Processes

Outside data processes bring data into a database through interfaces and data integration techniques, or through manual entry. The problem with these processes is that errors can be produced by data extraction, transformation, or data loading if caution is not taken. There is also the possibility that the outside data already has errors before migrating to the organization. As data traffic increases, these problems are amplified and become even more of a concern.

Such processes include initial data conversion, system consolidations, manual data entry, batch feeds, and real-time interfaces.

Initial data conversion is often the first step used to fill an empty database, where data is converted from an existing data source. The process of converting the data is sometimes unreliable. When the data is converted, some of the data may be lost, and the data that does arrive can end up changed. This leads to data problems if extensive care, action, and attention to detail is not taken. In fact, the error rate in a newly converted database is often above that of the old system. The first step isn’t always the easiest, and data conversion proves that.

System or database consolidation is the act of combining data from multiple sources into one. It is often used to improve the efficiency of the database or to combine unique databases after a merger between two companies. Like data conversion, data consolidation can lead to a loss of data or undesirable changes in the data. However, consolidation also adds the fact that the new data may not fit together. It can be like squeezing a square peg into a round hole at times. The merger of the data can lead to duplicates, overlaps, and conflicts.

Manual data entry is the process of entering data through user input. The majority of errors coming from manual data entry are on the behalf of the user inputting the data. The user can possibly choose a wrong entry from a list or enter something into the wrong box. To reduce data entry errors, the data entry form or interface should be as easy as possible to fill. The user should not be confused or be forced to enter incorrect data for a workaround. If the user does make a mistake, the form or interface should recognize it and prompt the user to try again.

Batch feeds are large data exchange interfaces that link data systems together. Batch feeds are used to allow communication between the millions of databases around the world and the millions more to be. Batch feeds are typically tied to the largest amount of data quality problems, however. They transfer and share so much data that a single error leads to an accumulation of more and more errors. A data error in a batch feed is almost like a virus, quickly spreading from database to database and going unnoticed among the piles of data. As old data is changed to reflect new data, a single error can potentially trigger an avalanche of inaccurate data.

Finally, real-time interfaces are a separate technique from batch feeds that’s used to exchange data between systems in rapid-fire succession. While fast and efficient, a problem with real-time interfaces is that thorough error checking is simply not possible. The most that can be done is to check the validity of individual attributes. If you ever received an e-mail from a company that you did not do business with, there’s a chance it stemmed from a real-time interface error on the company’s behalf.

### Processes Changing Data from Within

Processes that change data from within can cause accurate data to become inaccurate over time. Typically, it is the result when an outside object connected to the data changes, but the data collection processes do not record the changes.

Examples of these processes include data processing, data cleansing, and data purging.

Data processing is the soul of data. If we do not use the data to process meaningful information, what’s the point of having the data in the first place? Data processing however, like many things in life, can always go awry if enough care goes into it. A majority of errors come from changes in code. Upon making any change large or small to the code processing the data, adequate testing should be performed as a small error can lead to millions of records being negatively affected. Errors can also occur from regular data processing activities. Newly entered data can cause regular processing activities to assemble faulty results, and there’s always the chance of a process running before it’s supposed to happen.

Data cleansing is the activity of correcting data errors. Ironically, data cleansing today can create more data errors. The main concern is that there are many links in data. Fixing one problem can create many others. Considering that the data cleansing process is automated within business, the situation is even worse. By attempting to fix bugs and make corrections all at once, the process of creating new errors is only amplified.

Data purging is the act of dumping old data from systems so new data can take its place. It is a normal practice for when a system becomes overloaded, where the deletion of older, unnecessary data benefits the business. In business, however, there’s always the risk of important data being deleted by accident. This can lead to a significant loss in data quality and validity when an integral piece of data is removed.

### Processes Causing Data Decay

Processes causing data decay are processes that manage data inside a database. Data decay is simply the result of data going bad. Like how an apple decays if you leave it out for too long, data decays if it is not updated to stay up-to-date. Examples of regular activities that can lead to data decay include system upgrades and process automation. Other factors of data decay include failure to capture changes, finding new uses for data, and a loss of expertise within the organization.

First, system upgrades have a sizable impact on data quality. While a good upgrade can improve data quality, a very poor upgrade can potentially destroy data. The main cause behind a poor data system upgrade is the assumption that data does what it’s expect it to do. Truth is there will always be inconsistencies in data. With these assumptions, new upgrades are sometimes tested only on the basis on a perfect database. Upon release of the upgrade, there’s a good chance the systems will no longer function at all when they actually has to work with more realistic databases.

Process automation is the action of automating more and more tasks, all without human interaction. Computers think very differently than humans. While a human can validate data before using it, computers are very poor at making judgments and are more likely to accept erroneous data. If the automation process is well designed to handle errors, the risk of data decay can be decreased immensely. However, no automation system in large data organizations completely eliminates data decay.

Besides these two processes, data decay is affected by three factors. The first, failure to capture changes, has a large impact on the validity of data. When the object the data describes changes, the failure to capture the changes either by the users or the system leads to incorrect data. Second, finding new uses for the data creates faults and errors if no precaution is taken. The data may suffice for one process but may be too inaccurate for another. Finally, a loss of expertise in a business always means there is one less person who knows how to handle the data correctly to produce quality and up-to-date data.

## Possible Ways to Reduce Data Issues

The following is a list of actions that can be taken to reduce the amount of data issues produced for each process:

(Author’s Note: Some of these are my rough ideas. There’s a chance that they may not work. As I continue to learn more about data, I will be able to better identify how data issues can be reduced.)

### Outside Data Processes

Manual Data Entry: To reduce manual data entry issues, it is important for the entry forms and interfaces to be as easy to use as possible. The user should know exactly what to enter for each input. The amount of data entry errors can be reduced greatly through a clear and easy-to-use entry form or interface.

A good practice would be to minimize the amount the user has to type, as typing errors are common. Whenever possible, input boxes should be replaced with simple drop-down lists, built-in calendars, or any other input method suitable for the kind of input.

Initial Data Conversion: A solution to reduce the amount of data issues would be to create a program that compares the two databases after one is copied from another. The program would highlight changes and errors. However, it still would be important to always check the two databases manually through intensive care and attention to detail. This step should never be ignored after a data conversion.

Batch Feeds: Before the data is sent through, verification steps should be taken on each part of it. The verification steps can be as simple as checking to see if the inputs are valid. In action, it would know if a date is not stored in the correct format or if a value accidentally set as a negative number is not supposed to be a negative number.

### Processes Changing Data from Within

Data Processing: For data processing, the solution is as simple as adequate testing. Test and test and test the database! For the testing phase, try inputting incorrect or invalid data and see what happens.

Data Cleansing: A warning message can be implemented if the data to be changed will have an impact on other data within the database. Even further, the warning message can display the exact link between data to be affected.

Data Purging: A simple solution is to temporarily store deleted data so it can be brought back at any time. If the data quality is not sacrificed afterward, the data can be safely deleted. If the data quality is sacrificed or if errors occur, the data can be quickly brought back.

### Processes Causing Data Decay:

System Upgrades: As with data processing, a system upgrade should be thoroughly tested before being implemented. Do not assume anything. You don’t want an upgrade to turn into a downgrade.

Process Automation: The automation process should be well-designed to minimize data problems. It should be programmed to specialize in making decisions around data. A good automation process views the data like a human would before accepting or fixing it. It should easily be able to track formatting errors. Even better, it should recognize patterns in consistency and inconsistency and use it to make better judgments.

Failure to Capture Changes: The data should be neatly organized so it is as easy as possible to access data that needs to be changed. If the data is hard to access and work with, it decreases the will for workers to fix what needs to be changed.

To improve the process, the time should be taken to organize and group important data together. It’s understandable that some businesses have data coming on at an incredible rate, but taking the time to manage important data pays off. When the data becomes outdated, it can easily be changed thanks to good organization

## References

Causes of Data Problems. (n.d.). Retrieved from http://cdn.ttgtmedia.com/searchDataManagement/downloads/Data\_Quality\_Assessment\_-\_Chapter\_1.pdf