Scalable Data Curation and Data Mastering

by Michael Stonebraker, Chief Technology Officer, Tamr Inc.

Introduction

Traditional data management practices, such as master data management (MDM), have been around for decades - as have the approaches vendors take in developing these capabilities. For the longest time, the problem set being addressed was the management of data at modest size and complexity. However, as enterprises mature and start to view their data assets as a source of competitive advantage, new methods to managing enterprise data become desirable. Specifically, enterprises need approaches to data management that can solve critical issues around speed and scale in an increasingly larger and more complex data environment. This paper will explore how data curation technology can be used to solve data mastering challenges at scale.

" Enterprises need approaches to data management that can solve critical issues around speed and scale in an increasingly larger and more complex data environment."

Data Curation

Data curation is typically thought of as a combination of processes used to combine data from disparate sources into a composite whole. These processes include:

(1) **extraction** of data from source data systems into a common place for processing (often called a data lake)
(2) **transformation** of data elements, for example from Euros to Dollars.
(3) **data cleaning**, e.g. -99 often means null.
(4) **Schema integration**, i.e. lining up columns in source data sets. For example, your "wages" is my "salary".
(5) **Entity consolidation**, i.e. producing clusters of records thought to represent the same entity. For example, I might be identified as Prof. Stonebraker in one data set and M.R. Stonebraker in a second one.
(6) **Cluster reduction**. For each cluster, a single record must be constructed to represent the records in this cluster. This process is usually thought of as producing a "golden record" for each cluster.
(7) **Export (Load)**. The composite whole is usually exported to a data warehouse or other repository

Data Mastering refers to steps 5 and 6 of the curation process. Generally speaking, one is assembling masters for common entities such as Suppliers, Parts, Customers, and Employees. Master Data Management (MDM) refers to steps 5 and 6, along with governance and provenance issues in the resulting master data set, and will be the focus of this paper.

An Example of Data Mastering

Let's walk through an example of data mastering to illustrate the steps involved. Consider three data sets with information on Computer Scientists as follows

Name	Location	Age	University	Spouse
Data Set 1:				
Michael Stonebraker	Moultonborough, NH	73	Michigan	Beth
Dr. David Dewitt	Madison, WI	68	UW	Julie
Madden, Sam	Newton, MA	41	M.I.T.	Annie
Data Set 2:				
M Stonebraker	Moultonboro, NH	73	MIT	Beth
Dewitt, Dr	Brookline, MA	68	M.I.T.	Julie
Dr. Sam Madden	Cambridge, MA	41	M.I.T.	Ann
Data Set 3:				
Michael R Stonebraker	Boston,MA	73	M.I.T.	Elizabeth
Dr, Dewitt	South Dartmouth, MA	68	MIT	Julie
S. Madden	Cambridge	41	M.I.T.	Annie

Notice that the data is dirty (Dave Dewitt has three addresses, only two of which are correct) and inconsistent (Dave Dewitt sometimes has a first name and sometimes a Dr.). The objective of entity consolidation is to produce a cluster of records for each entity. This is often called "matching" and the desired outcome is:

Name	Location	Age	University	Spouse
Cluster 1:				
Michael R Stonebraker	Boston,MA	73	M.I.T.	Elizabeth
M Stonebraker	Moultonboro, NH	73	MIT	Beth
Michael Stonebraker	Moultonborough, NH	73	Michigan	Beth
Cluster 2:				
Dr. David Dewitt	Madison, WI	68	UW	Julie
Dewitt, Dr.	Brookline, MA	68	M.I.T.	Julie
Dr, Dewitt	South Dartmouth, MA	68	MIT	Julie
Cluster 3:				
S. Madden	Cambridge	41	M.I.T.	Annie
Dr. Sam Madden	Cambridge, MA	41	M.I.T.	Ann
Madden, Sam	Newton, MA	41	M.I.T.	Annie

Next, we want to produce a "golden record" for each entity. In effect, we want to merge the three records in each cluster into a single golden record. The desired outcome is shown below.

Name	Location	Age	University	Spouse
Michael Stonebraker	Boston, MA	73	M.I.T.	Elizabeth
Dr. Sam Madden	Cambridge, MA	41	M.I.T.	Annie
Dr. David Dewitt	Brookline, MA	68	M.I.T.	Julie

If there are a small number of data sources, a small number of records, or the records are very regular, data mastering is fairly straightforward. The next section illustrates the issues in data mastering at scale.

Scalable Data Mastering

Let's look at a real world use case to show the issues that are involved in mastering data at scale. Our example concerns procurement at a top 10 industrial manufacturer. A procurement system is used whenever an employee wants to purchase something, such as a paper clip. It will print the employee a purchase order that (s)he can take to the neighborhood Staples to obtain the desired goods. An ideal enterprise has a single procurement system; however, in large enterprises, a single system is rarely the case.

For example, the manufacturer currently has about 80 procurement systems, each of which has an independently constructed supplier database. The average number of suppliers in their procurement system is in the thousands. One might wonder why there are so many systems, and the answer is quite simple. The company has many independent divisions; often these have resulted from acquisitions. An acquisition, of course, comes with one or more procurement systems. Any enterprise could require all of its divisions to use a single system, but this would slow down progress in each division dramatically. Hence, many large enterprises allow business units to operate relatively independently, which leads to the multiple-system phenomenon noted above.

There is huge upside to the organization in performing data mastering on these 80 supplier databases. When the Staples contract comes up for renewal, a procurement officer would love to know the terms and conditions negotiated with Staples by other business units, so that (s)he can demand "most favored nation" status. The manufacturing company estimates that accomplishing this task would save them millions to billions of dollars per year. However, one must unify (master) 80 independent supplier databases with millions of supplier records before gaining the needed visibility. A scalable data mastering system must be able to deal with this kind of scale.

In the next section, we indicate why traditional MDM solutions to the "match/merge" problem fail to scale.

"For example, the manufacturer currently has about 80 procurement systems... A scalable data mastering system must be able to deal with this kind of scale."

Traditional MDM Solutions Struggle To Master Data At Scale

The traditional MDM solution to finding "matches" is to write a collection of rules in a proprietary rule language. For our example data, the following two rules will do the job:

If spouse-1 matches spouse-2, then put records 1 and 2 in the same cluster

If spouse-1 has spouse-2 as an included string, then put records 1 and 2 in the same cluster

Notice that a human will generally have trouble coming up with the correct set of rules. Also, unless the data is very, very regular, then the number of rules will get very large – especially as more sources are added. A few hundred rules is the maximum that a human can wrap his brain around. Note that two rules distinguished 9 records in our example. Unless the data is super regular, then a few hundred rules will distinguish a few thousand records, 2+ orders of magnitude less than the million+ supplier records. In other words, a rule system simply won't scale.

" ...A human will generally have trouble coming up with the correct set of rules. Also, unless the data is very, very regular, then the number of rules will get very large – especially as more sources are added. "

With the result of entity consolidation in hand, we can move on to golden record construction. For our example data, consider the following rules for selecting the values of each attribute in the cluster:

Use majority consensus, if it exists

If there is an included, ordered string, then shorten the longer one to the included shorter one.

This collection of rules will produce the following golden records.

Name	Location	Age	University	Spouse
M Stonebraker	Moultonboro, NH	73	MIT	Beth
Madden	Cambridge, MA	41	M.I.T.	Ann
Dewitt	??	68	MIT	Julie

There are assorted problems with this solution. First it generates incorrect data (for example, Ann is not the spouse of Sam Madden, Moultonboro is misspelled and is not the primary residence of Mike Stonebraker, MIT has two representations, and there is no way to tell which of the three Dewitt addresses is his primary residence). Second, the rules are difficult for a human to come up with, and lastly, the solution does not scale unless the data is very, very regular. Moreover, we cannot have a human check each data element to ensure that it is correct.

In summary, rules systems have the following problems:

(1) They are difficult for a mere mortal to construct
(2) Small problems can be solved with a few hundred rules, withn the scope of human understanding. This is not possible for large problems
(3) A human can check the answers for small problems, but not large ones.

Simply put, rule systems for match/merge do not scale. The next section illustrates a much more scalable approach.

Match/Merge

We do not depend on a rule system to identify clusters. Instead, we use a machine learning (ML) based approach. ML depends on the availability of training data, from which our ML model can learn. This training data can be entered manually, can be the result of a collection of rules or be obtained in some other way. Once training data is available, we fit an ML model to the data and uses the model to identify clusters of records. When we are not sure whether two records are in the same cluster, it asks a human domain expert. The result is added to the training data, and our ML model gets smarter using active learning. In this way, we can scale to arbitrary sized datasets and address large-scale matching challenges.

" In this way, we can scale to arbitrary sized datasets and address large-scale matching challenges."

Our approach to merging uses program synthesis to automatically construct a candidate collection of transformations. These transformations try to refine values into normal forms. We produce a (perhaps large) set of these transformations along with the number of records to which the transformation applies. We then look for templates that multiple transformations fit and generalize the transformation to the multiple cases. Then, we ask a human to validate the transformations in frequency order, stopping when a human budget is exceeded. After applying the transformations, majority consensus is used to create golden records. This approach not only arrives at the correct attribute values for the golden record but is much faster and more scalable than the traditional approach to merging entities.

" This approach not only arrives at the correct attribute values for the golden record but is much faster and more scalable than the traditional approach to merging entities."

Summary

In summary, traditional data management practices such as master data management have proven successful in the art of matching and merging entities for decades. However, scaling using this approach will lead to a variety of complications and, ultimately, not be a viable option for anything beyond small-scale data challenges and/or those with very regular data. Tamr's match/merge capabilities are engineered to work at scale on a large variety of problems and is precisely the type of approach that is to be applied in an environment where enterprises are looking to master all of their data quickly and accurately to power analytic or other organizational initiatives.