The Dimensional Metrics Approach
Abstract

Informatica Data Quality (IDQ) provides analysts and developers with the ability to implement data quality metrics. These metrics are the results from tests (data quality rules or functions) which can be typed and reported on dimensionally. These metrics depict the number of exception conditions to a data quality rule across one or more data attributes in one or more entities. These results are then graded (scored) to qualify categories relevant for management and remediation.

Supported Versions

• Informatica Data Quality 9.1.0-9.5.1 HotFix 3

Table of Contents

Abstract ............................................................................................................................................ 2
Supported Versions ......................................................................................................................... 2
Table of Contents ............................................................................................................................ 2
Overview .......................................................................................................................................... 2
Data Entries and Attributes .............................................................................................................. 4
Dimensions of Data Quality ............................................................................................................. 6
   Accuracy ...................................................................................................................................... 6
   Completeness .............................................................................................................................. 7
   Conformity .................................................................................................................................. 7
   Consistency .................................................................................................................................. 7
   Duplication ................................................................................................................................... 8
   Integrity ....................................................................................................................................... 9
   Reasonableness .......................................................................................................................... 9
   Timeliness .................................................................................................................................. 10
   Validity ....................................................................................................................................... 10
Data Quality Rules Generation ...................................................................................................... 11
Reusable Rules ............................................................................................................................. 12
Data Quality Monitoring .............................................................................................................. 12

Overview

Informatica Data Quality (IDQ) provides analysts and developers with the ability to implement data quality metrics. These metrics are the results from tests (data quality rules or functions) which can be typed and reported on dimensionally. These metrics depict the number of exception conditions to a data quality rule across one or more data attributes in one or more entities. These results are then graded (scored) to qualify categories relevant for management and remediation.
These tests or rules can be constructed from traditional data quality functions as used for remediation of contact data converted into a check on adherence to standard criteria, or they can be of the new form to categorize any data element by adherence to standard criteria along data quality dimensions.

DQ monitoring is how these results are depicted in a meaningful way to the user for the purpose of data quality evaluation and data remediation efforts. This can include a presentation layer depicting bar charts, pie charts, gauges, and line charts. It also includes the data collection approach with respect to scheduling data quality checks in a practical and reasonable manner so action can be taken.

Data quality functions have evolved over time. The old-school data quality functions were closely tied to contact information and provided immediate and directly calculable business value based on communication costs. These data quality functions included: data scrubbing (name, address, email, and phone content corrections), name and address parsing, address validation, matching/de-duplication, and consolidation/survivorship. Once clear business value was reached by applying data quality to contact information, additional data augmentation built upon it, in the areas of demographics, geography, privacy restrictions, tax jurisdiction, and watch lists helped to further improve business and meet regulatory requirements for many companies and government agencies.
The new-school data quality functions extend the concept of traditional data quality to include a dimensional description that can be applied to any data element, not just contact information. This came about as business automation became more complex with the proliferation of different software and technology to support differing business units. As operations became more complex, more opportunities for data corruption entered into the process. In addition, as data became more valuable to new business, capturing it correctly and integrating it into the enterprise as commonly mapped and shared data was given a higher priority as the direct cost of data conversion failures could be measured in terms of lost opportunity costs, lost customers and the cost of labor for remediation efforts.

The use of enterprise data quality metrics has become a critical integration and reporting issue for many companies due to this explosion of different software applications and data stores. It has been estimated that failures in data quality negatively account for approximately 23% of the average business’s gross revenue.

This business problem is widespread and a natural occurring result from software applications evolving to meet the specific needs of their audience. Given today’s market, business is always looking to innovate to provide it an edge over the competition. As the software application gets “better,” it also gets more specific to the business that it is dedicated to help with.

The problem is that within an enterprise, differing applications need to share equivalent data across applications for differing purposes and with differing representations. When data has multiple meanings or different formats based upon where and how used, problems to ongoing operations with translation, ownership, and relevance are encountered across the enterprise.

The Informatica Data Quality platform provides you the ability to create profiles and scorecards to detect and collect data on integration issues affecting the quality of data. These types of integration issues are relevant for any data type (not only name and address) and are referred to as dimensions of data quality.

Profiles are a summary representation of the data. Typically the most basic profile would be a value based frequency. Additional common checks could include a pattern check(s), null checks, and others to qualify a data element in terms of data quality characteristics such as accuracy, completeness, conformity, consistency, duplicity, integrity, reasonableness, timeliness, and validity (range and precision). Typically, the results of a profiled column, includes a set of additional columns with a discrete set of values that can be scored.

Scorecards take a profile and convert it into a set of scores for the entire column to be categorized as acceptable, unknown (optional), or unacceptable with pointers to the corresponding relevant records/rows. In addition, with IDQ OOTB there is the ability at the scorecard level to drill down to the acceptable, unacceptable, and unknown score-carded datasets and further qualify them through the interface.

The Informatica Analyst tool is used to report on the result of data quality metric collection at the rule and data attribute level. It can include trend reporting as well as provide drill down into specific entities. It does not include dimensional reporting across rules nor can it be used to create some of the more complex rules. The Informatica Developer tool can be used to create more complex rules.

More sophisticated business intelligence tools exist (although not part of the Informatica product suite) that provide gauges, bar charts, line charts, drill down, drill through, pie charts, and other widgets which can be used for presenting the data to management in a manner that simplifies decision making. These tools require building a dimensional schema (star or snowflake model) data mart for reporting purposes. Such an approach enables management to look at overall data quality at many levels in terms of common characteristics and thus be able to delegate remediation efforts to relevant parties.

**Data Entries and Attributes**

Any data that is used by multiple applications within an enterprise is a potential candidate for data quality metric collection and monitoring. This is true whether the data is entered manually or the
flow of data between the systems is automated. In general, any reference data that would be a candidate for enterprise master data management is likely a good candidate for data quality metric collection and monitoring. Some of the more common and familiar data includes customer and vendor contact data such as name(s), address(es), email(s), phone number(s). Other common operational data would include product, service and material descriptions and terms.

Additional critical data relevant for data quality metric and monitoring that would be shared between applications but not included under the master data management umbrella can include transactional data. This data can also be affected by data quality issues.

As an example of such critical transactional data consider a purchase expressed in dollars and cents ($3,512.60) versus a purchase expressed as a percentage of a total price (20% of $17,563) versus a purchase expressed in thousands of dollars ($3k) versus a purchase expressed in a foreign currency ($3,456.14 CAD). In this case the representation for expressing the same data varies based on the application. These differences in representation equate to a difference in precision. These values need to be reconciled to one version of the truth, regardless of any one application’s use of the data, so that the enterprise can accurately account for its business activity and report on the transaction.

Another example of critical transactional data between businesses could involve non-standard payment terms such as a trade. For example, consider an agreement to stay at a certain hotel chain in return for being awarded an agreement to furnish services to that hotel chain. The value of the services exchanged is not fixed but imposes a future obligation relevant to current business that should be included in the determination of a company’s value. It would be similar to a debt incurred that can change over time based on the lender’s business (not unlike an adjustable mortgage).

In any case, the first step and best practice in any successful enterprise data quality engagement is to conduct an enterprise data quality assessment.

This involves a) meetings with the business users, subject matter experts and technical staff in order to identify all shared data elements, their meaning and use in context for every facet of the enterprise. It involves b) asking where the pain points of data exchange are. In most organizations, there are so many high profile cases where corrupted data exchanges have caused companies to lose money and embarrass all involved that preventing them from re-occurring is a high management priority.

The other side of the data quality assessment involves c) looking at the actual data. This includes profiling both sources and targets at every interface where data is exchanged or represented in multiple or decentralized silos of data. Profiling will identify statistics such as the domain of actual values and frequency of occurrence. It will help to succinctly identify outliers, missing and differing representations for the same values.

Many times, profiling will surface additional critical and even more serious data conversion problems before (hopefully) they damage the company’s bottom line. Some of the issues to look for include referential integrity violations, orphans, same keys pointing to unrelated data, missing data values, discrete domain violations, values out of range, double duty fields, precision variations, incomplete data, unreasonable data combinations, inconsistent representations, duplication, and late (or early) data.

The reasons for these issues are manifold. Often the business users do not know they exist or if they do that they can negatively impact their business. Once the actual condition of the data is determined, Subject Matter Experts are further engaged to determine what the correct data values should be and why. In many cases, existing issues trace back to conversions both between systems and from old to new systems, as well as one-time data fixes performed by previous SMEs, and ongoing interface data governance problems.

This is a significant effort and d) all of this work and research needs to be documented.
Dimensions of Data Quality

The result of a data quality assessment is a) the identification of all relevant enterprise wide data entities and attributes in the context of the business and b) the specification of rules to detect known data quality issues.

During any data quality assessment there is tendency on the part of business users and subject matter experts to blur data quality issues in with the lack of application features or existing application issues. To the business users there remain problems they want solved but these may or may not fall into the category of data quality.

For example, a report on how many widgets are produced daily may not be a data quality issue. However, if the widget completed status is not reliably recorded in a manner understood by downstream applications thus making it possible for a downstream business unit to ship unfinished product out the door or stack up completed product that is marked incomplete that would be a costly data quality problem affecting the business.

To distinguish what rules to include, you need to understand how to categorize the rules in terms of data quality. This categorization is termed the dimensions of data quality.

Data quality dimensions are identified in several books on the subject but based on real-world experience with clients, the Informatica Data Quality practice has identified the following set of data quality dimensions as a best practice for which data quality metrics and monitoring can be applied to:

**Accuracy**

Data is accurate when it is verifiable by an authoritative source for its intended purpose. In many cases the authoritative source can be an objective third party but it can also be an internal controlling authority. There may also be different levels of accuracy as well as varying purposes for accuracy.

For a common example of such consider an address in the United States. It can be compliant with CASS certified software making it sufficiently accurate for postal automation purposes.

It can also be compliant with DPV certified software, making it sufficiently accurate for USPS discount eligibility requirements.

However, an address may fail both CASS and DPV and may still be “correct” but not deemed deliverable by the USPS after being run through Address validation. Examples of such could include a legal address (such as a deed identified property address), or a shipping address (like a loading dock, or an alternate entrance for the public), or a temporary worksite.

For another example, just because an address is deliverable by the postal service does not mean the company or person designated as a recipient still resides there. For that information you may require additional augmentation services (dependent on matching). It may also be the case that the property address may not be deliverable or even geographically locatable using traditional address validation tools. An assumed level of accuracy may be acceptable.

An additional example of accuracy could be that the latitude and longitude provided by the geocoding software was detailed enough to point to the actual house and not just the centroid of the postal code.

Note: If the address is intended to reflect a shipping address it may or may not be deliverable by a shipping company. Some companies do not deliver to residences.

There are different levels of accuracy for the purpose of mailing, and different authorities may be required for the purpose of shipping or property definition.

Other data types such as product definitions may be required to adhere to international trade standards on product naming (aka harmonization).

Once an enterprise standard is identified, accuracy may be determined by compliance (or at least as a known mapping) to the enterprise standard.
**Completeness**

Data is complete when it contains sufficient value to be used by the business for its singular purpose. There can be varying degrees of completeness and varying purposes for different data elements. Functions that check for completeness typically look to make sure all elements that are required to be populated are actually populated.

For an example of missing data, a customer places an order for shoes over the internet. As part of that order he fails to put in his shoe size. The webpage does not reject the order but refers it to a customer service application to send an email requesting the show size. The shoe size is left null or blank until it can be entered. The order is not complete and may not be fulfilled until the shoe size is added.

Another example could be that the original loan amount on the loan application should not be blank.

For an example of partial data, a customer purchases compact discs but

- provided an address that does not reflect a residence (such as Mailboxes Etc) or
- provided an address that could reflect multiple residences (e.g. default high rise address) or
- added more information to the address than sufficient in order to confuse order processing and defraud your company of product or
- failed to include the city name.

In a) the address appears good but reflects a CMRA after Address Validation.
In b) the address the address appears legit but Address Validation flags it as a default address.
In c) the address looks possibly correct by including APT A, APT B, etc.. but Address Validation indicates there is no legitimate secondary address component.
In d) after address validation the address may be considered ambiguous as the postal code could serve two different towns.

In all partial data cases, the address may be considered incomplete for delivery purposes of your product.

In yet a third example, as part of a data fix, foreign keys to a linked table are set to null by one time external SQL logic to reduce visibility from the application to certain privacy restricted historical data.

In this case, the foreign keys may be deemed incomplete for determining when the data may be archived.

**Conformity**

Data is conformant when it maintains type and value integrity. Different types of data should not be stored in the same data element - no “double-duty.”

For example, phone numbers should contain only phone numbers, not email addresses or contact names. Organization name attributes should not contain individual names. Attributes should not serve double duty.

Numeric values should be retained at a precision level that retains all that is needed to correctly specify the amount across the enterprise. If the same data needs to be represented at different precision levels there should be a keyed map to the correct value.

**Consistency**

Data is consistent when there exists a unique representation for each value in its domain. Maintaining consistency is one of the more common problems in data integration. Attributes should not contain more than one value that mean the same thing. The meaning of a value should also be unique within the target domain. Common causes for these types of problems include language barriers, application requirements, and alternate common expressions for the same items; double duty fields; and a lack of data governance.
For example:

- **Amount attribute containing:**
  - One versus Uno (Spanish) versus Ichi (Japanese) versus 1 versus 1.0 versus 1.00 versus 0001;
- **Generation attribute containing:**
  - “III” versus “3rd” versus 3 versus “the third”
- **Flag attribute containing:**
  - True versus 1, Yes versus Y, Null versus False;
- **State attribute containing:** “DC” versus “Dist. of Col” versus “District of Columbia”
- **Status attribute containing:**
  - Pass versus Good versus Live,
  - Suspect versus Warning versus Support,
  - Fail versus Bad versus Fatal

Some inconsistencies are more subtle and only apparent in context with other attributes, such as a mailing address attribute containing: both alias address lines and postal preferred address lines, e.g. 50th Ave versus ML King Blvd (in Lanham MD); or a mailing city containing both vanity cities and postal preferred cities, e.g. Hollywood versus Los Angeles (in California);

Some inconsistencies follow a reverse path and are only cleared as consistent in context with other attributes such as PCH (implying Peabody Community Hospital in Peabody Massachusetts) versus PCH (implying Pacific Coast Highway in California).

An amount could be based on different currencies such as: USD, CAD, AUD, JPY, GBP, EUR, INR, CHF, MXN,

However, a status attribute of:

- completed (in context of order received by Sales) versus
- completed (in context of order processed by Accounting) versus
- completed (in context of box provided by Shipping to UPS)

could be the same physical value but take on different meanings depending on the source application within the enterprise. If they were all lumped into the same attribute without distinction in context they would make the data attribute inconsistent.

**Duplication**

Data is duplicated when there is more than one entry to reflect the same underlying instance of an entity. This is typified in many applications by non-normalized data (key values not unique within an entity). It can also be more subtle as the keys could be unique but the remaining attributes within instances of the entity refer to the same real world entity.

For example, key values for name and address data can and often do differ when the underlying entity is the same:

- A vendor organization can be entered multiple times based on differing:
  - points of contact (Mr. Jim Smith Jr., Ms. Nancy Taylor-Swett, Col Juan Lopez Hernandez);
  - sales departments (Eastern Region Sales versus Data Quality Vertical versus Worldwide Sales);
  - DBA names for the vendor (Wheelhouse versus House of Wheels, DELL versus DELL CORPORATION, IBM versus International Business Machines); and
  - name registration and licensing (In the US, businesses register and license at the state level);
- The same address can be entered as a customer address and as a vendor address.
- It may also be entered as shipping address and as a mailing address.
• A new customer is entered every time a purchase is made as the order entry system either does not maintain history or keeps a limited amount;
• A part can be entered multiple times with different part numbers based upon its use in multiple assemblies;
• A service offering can be listed multiple times based upon packaging it with other products and services.
• A loan is financed multiple times for different customers using the same property address as collateral;
• Insignificant differences exist in names, email, phones, and addresses; as well as part, assembly, product and service descriptions due to lack of standardized data.
• Multiple identical orders enter the system due to internet connection volatility and accidental resubmittal on the website.
• Multiple payments are made due to multiple invoices received for the same product or service from a vendor.
• Multiple entries are made in an application to reflect multiple uses for the same entity, e.g. the same person holds more than one job and is entered for each job in the HR system.

**Integrity**

Key data has integrity when referential constraints on the primary and foreign keys are not violated. This means there are no orphans, no corrupted keys, and the cardinality expected reflects reality. Join checks on primary and foreign keys are used to expose these issues.

Examples of data integrity issues such as orphans include primary keys that map to no other records in a relational database schema. These can occur when data is moved from one application to another with differing features.

Corrupted keys can occur when one-time “fixes” are made to the data that meet immediate needs but are not be maintained by the application.

Referential constraint enforced cardinality such as one to one, many to one, one to many, or many to many relationships is frequently dropped in order to bulk load data into a replica of the original schema. Hopefully, the referential constraints are restored, but if restoring them cannot be done because of unexpected duplicates or missing parent keys, the quickest and easiest approach for resolving the problem is to ignore it and not reapply the referential constraints. This can lead to more orphans.

Sometimes referential constraints are removed for compatibility with new software procedures or to capture data with less effort required during data entry.

**Reasonableness**

Data is reasonable when it is relevant to the business in context with other entities and attributes across the enterprise. Checks for reasonableness typically verify that the data makes sense in terms of the business.

Some examples of unreasonable attribute values may include:
• A borrower lists insufficient collateral on a loan application.
• A loan is made for the purchase of a property where the title search does not come back clean.
• A borrower fails to include a sufficiently large enough down payment to be eligible for a lower interest rate.
• A large amount of product (in small multiple orders as per lead-in offer) is shipped to a location (or area) without pre-payment.
• A credit report indicates the customer is a higher payment risk and pre-payment was not made.
• A look at the historical behavior of this customer indicates they require excessive customer service support and have thus not been profitable to your company. A referral to your competitor is warranted.
• More perishable materials have been purchased than you have the capacity to store and use at your facilities.
• More products have been produced than you have the capacity to store or ship at your facilities.
• Too few products have been produced in a batch causing excess overhead requirements for downstream business units.
• Customer commitments have been made for products and delivery times without a sufficient co-commitment from vendors to supply raw materials and dependent parts.

**Timeliness**

Data is timely when it is made available in time for proper use during ongoing business operations and analysis. Timeliness depends on the data being populated and available when it needs to be for the business to use it.

Some examples of exceptions to timeliness include:
• Late payment is received from a customer or made to a vendor contrary to the contract terms.
• Materials, parts, or services are not provided by a vendor when promised. Materials or parts arrive either too late or too early or services are completed too late or too early.
• A loan is not processed soon enough to meet the closing date on the purchase and sale agreement.
• An interest rate is promised but the originator fails to lock the rate until the last minute and the rate is no longer available to the borrower.
• A borrower fails to provide proof of income before the closing date.
• Inventory data is not provided to purchasing in time to place an order with vendors for parts and materials needed for assembly and order fulfillment.
• Sales fails to provide detailed order information soon enough for purchasing to order sufficient supplies and for sufficient staffing to be assigned to meet product commitment dates.
• Production equipment maintenance has not been performed as often as per vendor specifications thus violating any warranty and/or putting product delivery at risk should a breakdown occur.
• The legal department has imposed new contract terms causing them to engage in prolonged negotiations with a critical vendor over said terms and a cutback in service until resolution.

**Validity**

Data is valid when the values fall within the expected domain or range of values that it should contain. Validity is typified by range checks, discrete value checks, and relevant precision level checks.

For example:
• A loan amount for a primary residence could range from: $30,000 to $800,000.00 and should not be negative. Other values may not be valid.
• A currency amount may not include partial cents, e.g. $3,124.33333333 would be invalid.
• A task completion flag may contain: yes or no. Other values such as: True, X, 1, 1.0, -1, false, 0, 0.0, null, blank, or empty may not be valid.
• The account balance for an investor should not be negative.
• The purchase price for a product should not be negative.
• The quantity of products ordered should range from 1 to 9999.
• The ranking of a customer can be: conditional, preferred, silver, gold, platinum, or double platinum. Other colors are invalid.

• The house number on a mailing address can range from 10 to 120. (The house may not exist but for the purpose of mailing the USPS will find it compliant with results from CASS certified software).

• The product color ordered can be beige, black, or olive only.

• The mail-ability score attribute can be 0, 1, 2, 3, 4, or 5. Other values would be invalid.

• The postal directory match code can contain one of the following list of discrete values: (V4, V3, V2, V1, C4, C3, C2, C1, I4, I3, I2, I1, Q3, Q2, Q1, Q0, RA, R9, R8, R7, R6, R5, R4, R3, R2, R1, R0, S4, S3, S2, S1, N1, N2, N3, N4, N5 or Null). Other values would be invalid.

• The value specified should be a date. Other data types would be invalid.

Data Quality Rules Generation

The new-school data quality functions created along the lines as described above can be configured to produce pass/fail output with the column profiled and graded in accord with how many passes or failures occurred. Thresholds can then be established as to how many occurrences indicate remediation efforts are warranted.

The old-school data quality functions as described above typically produce a different set of results because they include some data remediation and data augmentation for output. However, given evaluation criteria for pass/fail they can also be profiled and graded as well as categorized according to dimensional descriptions.

Profiled data quality functions typically produce a binary value (Y/N, 1/0, or pass/fail) and counts. The old-school data quality functions typically produce string output, linkage keys, numeric scores, coded ratings and flags.

With data quality functions such as name, phone, email standardization or address validation, a score can be produced indicating how close the standardized value was to the original value, or a flag can be set to indicate they changed. With address validation, a code is produced indicating how well the address provided matched to the postal directories. In addition, with address validation, field level codes as to the condition of the input data, relevance to the DQ function, and status of the result as well as detailed error messages can be produced to assist data entry personal in remediating the data to avoid the errors detected.

With data quality functions such as labeling and parsing a flag can be set to indicate that the name or names provided were recognizable and by which parsing rule. Parsing rules can be assigned a relative score based on how simple they are and thus more likely to be accurate. The other outputs from parsing (i.e. multiple records for compound names or compound addresses) would share the same parsing score but could also be assessed later during matching as related together.

With data quality functions such as matching, a key is generated that links the records in the match together based on the match criteria. Multiple matches mean multiple keys. In the case of house-holding, you can have keys within keys generated. Depending on the type of match, for reference matches can be scored as to how well the candidate record matched the reference record or for intra-file how well the record matched the driver record or for inter-file matches how well the record in one file matched to the other driver record in the other file. Matching can also be configured to return one to many matches. The fact they share the same match key is enough to indicate they matched.

A flag can be set to indicate the driver record but this is of minimal business value. It is more value to a data quality expert when debugging and explaining why an unexpected match may have occurred.

Multiple matches need to be resolved. These are usually identified by another key indicating the cross match relationship between records.
The fact that a record matched to another record with a different primary key indicates that there is duplication. If the unique key of a record exists it should not be duplicated even by a surrogate key. When it is so, the business value for the row is corrupted, if not the referential integrity of the database table.

With data quality functions such as consolidation, there is not usually a score or flag assigned indicating how well the records were consolidated, but the string output is used in remediation efforts. It is a requirement for implementing master data management.

Reusable Rules

Given that we correctly applied the dimensions of data quality during the data quality assessment, we should have a specification for what data quality rules to build. The next step and best practice is to identify the reusable rules. Some of these rule definitions may repeat across dimensions. Some of these rule definitions may repeat for different data attributes. Some of these rule definitions may be included in other rules. These are all candidates for reusable rules.

The advantage to reusable rules is that standard logic can be used and adapted based on the needs of the business. Some of the more common simple data quality rules include: null check, empty, blank, numeric, positive, negative, AttributeA <> Standardized(AttributeA), DateA < DateB, amount < limit, etc.

It is important to note that reusable rules do not take the place of the specific rule in context. A DQ Rule may contain a reusable rule. Each DQ rule needs to be uniquely identified for reporting purposes.

Data Quality Monitoring

In most enterprise data quality efforts, data quality is monitored over time and remediation efforts apply to the current datasets (not historical). There are several approaches for executing data quality checks. They depend on the type of reporting you want to depict as well as how you want to use the data. Trend reports, drill-down and drill-through reports, aggregation by differing dimensions all affect how you schedule the execution of data quality metrics.

In the simplest case, you can test a data element, profile, and score the profile at a particular time, i.e. take a snapshot. This gives you the condition of the data at that point in time. You can also make that time right now, as in an on-demand service.

Trend reporting implies you are taking snapshots at certain intervals. Most trend reporting adheres to this strategy. It could also imply a blended score within a time period for a set of time periods.

Blending scores for a given core data element can be done several different ways. You can average the execution results. You can weight certain execution results higher based on the number of records processed, the number of days passed since the last snapshot, how recent was the snapshot, or just take the most recent snapshot. The side effect to blending scores is that less frequent versus more frequent DQ checks can produce different results.

Some clients have implemented a strategy where some snapshots are taken daily, others weekly, and others monthly. Then these are bucketed (or blended) into daily, weekly, monthly, quarterly, and annual periods for reporting purposes. Some snapshots could even be taken multiple times in the same day.

Some reporting depicts not just data quality dimension, but element, entity, business segment, business unit, support function, rule type, and application data quality. These imply a blending of scores of across data quality metric results. A common approach is to average these. The side effect of doing this is that the importance of the metric to the business can be lost at the summarized level.

The best practice for trend reporting is to report snapshots taken at specific fixed intervals (keep it simple). It takes time to remediate data. The frequency of the snapshot should reflect not just the
priority given the data by the business but the practical time it would take to remediate the data if warranted. There needs to be a balance between these two goals for monitoring to be effective. Current conditions can be verified during remediation and an on demand service can also be implemented to show the current improved results to interested upper management. Effective drill down (drill into subset of data depicted) reporting requires a dimensional data mart as well as a business intelligence tool acceptable to upper management that will support it. Drill through (drill into other dataset) reporting is best done with a presentation layer tool that provides alternate presentation widgets.

Author

Mike Gagnon
Principal Consultant, Informatica Professional Services