

The information perspective of SOA Design, Part 7: The execution approach for the data quality analysis pattern in SOA

Skill Level:

[Brian Byrne \(byrneb@us.ibm.com\)](mailto:byrneb@us.ibm.com)
Industry Models and Integration Architect
IBM

[John Kling \(jkling@us.ibm.com\)](mailto:jkling@us.ibm.com)
Consulting and Services Architect
IBM

[David McCarty \(davidmccarty@fr.ibm.com\)](mailto:davidmccarty@fr.ibm.com)
IT Architect
IBM

[Dr. Guenter Sauter \(gsauter@us.ibm.com\)](mailto:gsauter@us.ibm.com)
Information Architect
IBM

[Harald Smith \(smithha@us.ibm.com\)](mailto:smithha@us.ibm.com)
Product Manager
IBM

[Peter Worcester \(pworcest@us.ibm.com\)](mailto:pworcest@us.ibm.com)
Services Solutions Marketing Manager
IBM

08 May 2008

This is the seventh paper in a series called the "The Information Aspect of SOA Design." The purpose of this article is to demonstrate to an architect community the execution approach of detailed data quality analysis in the context of an SOA environment. This article focuses on the implementation of data quality analysis regardless of the specific technology in use, and will be followed by a related article

that describes in more detail how the related IBM® products (WebSphere® Information Analyzer) can be used in this context.

Introduction

Read all the articles in this series

1. [Introduction to the information perspective of an SOA](#)
2. [The value of applying the business glossary pattern in SOA](#)
3. [Use the IBM WebSphere business glossary in SOA design](#)
4. [The value of applying the canonical modeling pattern in SOA](#)
5. [Use of Rational Data Architect in SOA](#)
6. [The value of applying the data quality analysis pattern in SOA](#)
7. [The execution approach for the data quality analysis pattern in SOA](#)
8. [Use of IBM WebSphere Information Analyzer in SOA design](#)

For SOA services to be successful and reusable, it is important that the data they expose is of acceptable quality for all the consumers. Exposing quality data is necessary in any SOA initiative. Whether it is application services being exposed or a simple generated data query service, the resulting data must appropriate to the business context and be accurate to be of any value.

Part 6 of this series outlined the value and approach for reviewing data quality through analytical techniques. The central focus of the data quality analysis practice is the measurement of the data quality level. This measurement is across three core areas:

Analyze source systems

- Apply data quality metrics to data from identified data stores to ascertain data quality levels.
- Interpret data metric results and translate these results to business terminology. Create detailed reports, charts and summaries that portray data quality levels and present recommendations.

Analyze target system

- Identify gaps between analyzed source systems vs. target system. Create recommendations to resolve gaps.

Assess alignment and harmonization requirements for each relevant data element

- Apply data quality measures to identified data elements to assess current standardization and matching support and translate these results to

business terminology. Create detailed reports, charts, and summaries that portray standardization and matching levels and present recommendations.

In SOA analysis and design, these areas are critical as they directly influence service implementation decisions. After service interfaces are identified and designed using top-down business requirements, the next step is to decide how the service can be implemented by leveraging existing systems or by writing new code. Each service exposes function and data to the service consumers, and, in doing so, it must meet agreed service levels including data accuracy, response times, and the like. This can only be achieved if the service designer understands the characteristics of the data stored in each of the systems being leveraged, understands the gaps between the source systems and the data presented by the service implementation, and addresses how to appropriately align and harmonize disparate data elements in complex or composite services. Data quality analysis provides this understanding.

This article introduces the most significant aspects of data quality analysis execution in SOA projects.

The data quality execution plan

As part of the data quality analysis, an execution plan has been formulated as noted in [Part 6](#) of this series. Here the data analyst has identified who is executing the plan, what is being tested, what tools will be used to test with, when tests will occur, and what the expected outputs of the plan are. The targeted deliverables must be clearly stated to ensure effective execution and completion of the plan.

Based upon the defined execution plan, data analysis occurs within three core dimensions. The process of performing data analysis in this SOA context is similar to most data quality initiatives with the goal of reducing project risk and addressing potential gaps.

As each test is executed, results are produced, gathered, and incorporated into the broader data quality assessment. By clearly focusing on the core requirements of the services to introduce, the value of the data quality analysis will be realized.

Table 1 represents the analysis steps performed in the execution phase. The execution process, core questions, and potential issues within each step are detailed below.

Table 1. Steps of data analysis

Source system analysis	Target analysis	Alignment and harmonization analysis
<ul style="list-style-type: none"> • Technical 	<ul style="list-style-type: none"> • Attribute gap 	<ul style="list-style-type: none"> • Attribute alignment

dimensions — Domain-level (metadata, domain, structural, and relational integrity, plus business rule assessment)	analysis <ul style="list-style-type: none"> • Data gap analysis • Field length analysis • Data migration scoping 	analysis <ul style="list-style-type: none"> • Standardization analysis • Matching remediation analysis
<ul style="list-style-type: none"> • Technical dimensions – Entity-level (entity integrity) 		

Where significant problems are discovered, there are three options available to the SOA designer:

- Change the service implementation to mitigate the data problems.
- Request that a change is made outside of the SOA project team. This may be a catalyst for initiating an enterprise data quality initiative in the organization if one does not already exist.
- Choose not to expose the service on the grounds that it cannot meet business service level requirements.

Conducting the data quality analysis within the project framework

The center point for this project-based approach is the actual execution of the data quality analysis. The specific paths for any project vary based on project-level decisions. The tools available and in use also influence the scope of effort that can be achieved. Part 8 of this series discusses a very specific product-based approach.

1. Source System Analysis

As noted above, there are three levels of source system analysis that come into play during the SOA project: the domain-level technical dimensions, the entity-level technical dimensions, and the business process dimensions. Why do we want to do a source system analysis? Few SOA solutions are developed as completely new information systems – they are almost always built to leverage existing applications and data. By developing reusable services that expose the capabilities of existing systems in a carefully controlled way, a business can evolve its IT capabilities to be more responsive to change. Only by understanding the nature of the source data being exposed can the SOA service designer meet this goal. For each data source you can perform the following types of analysis. The results provide insight into the

value and integrity of the existing data that supports the SOA solution.

Assessing the domain-level technical dimensions

Technical dimensions represent the physical data content of a given data source. The domain-level addresses explicit fields or attributes of the physical data. The domain-level technical dimensions which are of concern are listed in Table 2:

Table 2: Technical data quality dimensions - Domains

Name	Description
Valid	The data element passes all edits for acceptability and are free from variation and contradiction based on the condition of another data element (a Valid Value combination).
Unique	The data element used as a primary key or alternate identifier is unique —there are no duplicate values.
Complete	The data element is: (1) always required to be populated and not defaulted; or (2) required based on the condition of another data element.
Consistent	The data values persist from a particular data element of data source to another data element in second data source. Consistency can also reflect the regular use of standardized values particularly in descriptive elements
Timely	The data element represents the most current information resulting from the output of a business event
Understood	The metadata of the data element clearly states or defines the purpose of the data element or the values used in the data element can be understood by metadata or data inspection.
Precise	The data element is used only for its intended purpose, that is, the degree to which the data characteristics are well understood and correctly utilized

At the domain-level, the level of the individual data element, the techniques utilized for analysis start with an understanding of the actual underlying data content and build on that to see what that content implies for data structure and data quality.

The following techniques may be applied:

- Frequency distribution

- Domain integrity (or Column Content Analysis)
- Structural Analysis (or Key Analysis)
- Relationship Analysis (or Cross Domain Analysis)
- Business rule assessment
- Metadata integrity (or Business/Technical documentation)

Frequency distribution

This is a count of the number of distinct values and the frequency of their occurrence within a data element or an entity. In other words, what the frequency distribution shows is how often a value occurs. It is the foundational step for most subsequent techniques of Source System Analysis.

Frequency distributions may be looked at in two ways. First, the actual occurrence of a specific value may indicate whether there are issues such as incomplete data by the presence of a Null value, a Space, and so on. Second, the frequency of occurrence of a specific value gives focus to the extent of a data condition — how common or how rare the event is and whether outliers exist. In other words, if you have 90% of your values between a certain range, the frequency distribution would tell you what the outlying values are and may indicate a quality issues in those outliers.

Frequency distributions may be generated in any number of ways—manually from a list of all values, by issuing a SQL query to give a distinct count of values, or through the use of a data profiling tool that automatically generates many frequency distributions for all columns within a table. The results of the frequency distribution need to be accessible for additional analysis.

Domain integrity (or Column Content Analysis)

There is no implicit analysis in a frequency distribution—it is simply the statistical representation of what exists in a particular data element or domain. *Domain integrity* is the technique of analyzing the column contents for specific conditions.

The domain integrity addresses the technical dimensions of completeness, precision, validity, uniqueness, and part of the question of understanding. Column content also addresses structural consistency, but not value consistency, which requires a focal point of comparison, or timeliness, which requires a reference point.

The column content analysis results in a detailed and complex output showing the content of any particular column in a source store. It should answer the following questions:

- **Is there missing data?** This may be in the form of null values or other blank value conditions (for example, spaces). Missing data is indicative of its lack of importance or lack of integrity checks to enforce defaulting.
- **Is there defaulted data?** This may be in the form of specific set values (for example, all 9's or 'Do Not Use') or blank values (where a blank represents a keystroke distinct from a null value). Defaulted data indicates either a lack of importance, lack of integrity checks, or unknown criteria.
- **Is the data constant?** This indicates that most of all data in a column is the same. It may indicate a homogeneous population exists but allows for future system growth, or it may reflect that the element is largely a default condition.
- **Is the data unique?** This indicates that all or most of the data has a single occurrence and is unique. A unique domain is most commonly an identifier that is a potential key to the data, but also includes text-based descriptions.
- **Is the data skewed?** Where the frequency distribution shows data values decreasing in occurrence from some common condition to a rare condition, a skew exists. Such skews may be normal (as in a range of product prices), may indicate outlier conditions, or may represent issues of validity.

The domain integrity also looks at considerations of structural consistency. By generating a value list in the frequency distribution, the value list by itself can be used to infer key properties of the data. These include the length of the actual data (or the precision and scale of numeric values), the pattern or format of the data (for example, the character 'A' followed by four numbers), and the data type (that is, an integer, a decimal, a character type, and so on). These inferences should answer the following questions:

- **Is the data precise?** The shorter the length of the value, the more precise it will be with less opportunity for keystroke or other errors. Precise data is indicative of the requirements for modification of the data as it moves across systems.
- **Is the data structurally consistent?** This may be in the form of mixed data types or mixed formats, but lack of consistency has a high potential to impact the usability of the data.
- **Is the data understandable?** Converse to the question of precision, short pieces of data may be compact but difficult to understand without some external guide or reference. Where issues of

understanding are noted, this indicates the need for further assessment of the available metadata.

By looking at the precision, structural consistency, and understandability, the analysis also determines where existing data sources have broken down and contain embedded logic in the data itself—that they actually contain multiple ‘domains’ of information. Take the case where a column constantly contains a value of -9999; this indicates to programs that consume this data that alternate processing should be performed. This is an example of embedded business logic contained in the column.

Structural (or Key) Analysis

Certain data elements or domains either by themselves or in combination are critical to the understanding of an entire record or entity. These are primary or unique keys, identifiers for the other data elements. They are drivers for linking distinct pieces of data together.

The assessment of uniqueness done during Domain Integrity Analysis forms the foundation for this component. The output from this process provides the elements for not only Relationship Analysis, but also for aspects of Target Analysis and Harmonization Analysis.

The structural analysis results in a detailed output showing the uniqueness, duplication, and support for the other data elements. It should answer the following questions:

- **Is the identifier completely unique?** Any duplicate value compromises the ability of the data to link information. A classic example is the US Social Security Number which, in theory, should identify a specific individual, but where it is used for multiple purposes such as identifying a child who does not yet have a Social Security Number, its use is compromised.
- **If no single data element is unique, does a combination of data elements uniquely identify the data?** Frequency distributions of element combinations may find compound elements that do uniquely determine the data.

Relationship Analysis

When working with data spread across multiple files or tables, or when pulling data together from diverse sources, there must be a means to relate the data together. Relationship Analysis builds off of previous domain integrity and Structural Analysis to find common keys or data elements and domains. Common keys support the levels of entity-level technical analysis. Common domains allows for assessment of

value consistency at both the domain- and entity-levels.

The Relationship Analysis results in a detailed output showing the referential integrity of key identifiers and the consistency (or redundancy) of domains. It should answer the following questions:

- **Is the reference of an identifier between two tables or sources intact?** This referential integrity can be checked in one or both directions depending on the project goal. Where a reference link is missing, it may indicate that an entity simply does not have certain data (for example, a customer may not have a distinct billing address), that the link is broken (for example, there is supposed to be a billing address for every customer), or that multiple records exist (for example, a customer has multiple addresses of differing types).
- **Is one source a reference source?** One source may be a primary reference while the other source utilizes the data. If a review of the frequency distributions for each source points to a many-to-one condition and one source has only unique values, then it is a likely reference source.
- **Is the data consistent across domains?** This indicates that the consistency of data values is intact or compromised. Where values are largely the same, there is also the possibility that the data is redundant in which case a decision on the correct source of the data is required.
- **Are there false overlaps?** Just because two values overlap does not mean they are the same. Code fields may use an abstract set of alphabetic values. Identifiers may use surrogate keys. Dates and quantities are likely to overlap but may not have a relationship. Additional understanding and analysis must be applied to weed out extraneous associations.

Business Rule Assessment

The completeness, validity, and timeliness of a given domain cannot always be assessed in isolation. For example, if an order status field is 'Open', then the shipment date should be null or blank. Here the null or blank value does not represent incomplete data, but reflects that the business condition requiring the data has not yet occurred. Similarly, it is not possible to tell if an address field is timely if there is no check against an Address Modification Date field.

This level of analysis can only be achieved through the evaluation of business or data rule conditions. To do so requires refinement or segmentation of the data to look only at distinct value relationships. Filtering or selection of the data used to produce a frequency distribution, use of an SQL query WHERE clause, or

generation of a frequency distribution for a concatenation of columns are all possible techniques to start this level of assessment.

The business rule assessment results in output that shows whether these conditions are met or if exceptions exist. It should answer the following questions:

- **Is there missing or incomplete data?** This may be in the form of Null values or other blank value conditions (for example, spaces) in situations where the data is required.
- **Is there defaulted or invalid data?** This is usually in the form of specific set values that are within the set of valid values, but outside the valid set for the condition. The defaulted or invalid data is indicative of either a lack of integrity checks, or unknown criteria.
- **Is the data timely?** This indicates the date of creation or update for a specific field is within a tolerable range for usefulness. Unless a date is specifically associated with a specific field, it may not be possible to precisely answer this question.

Business Rule Assessment also touches on the Business Process Quality Dimension of Accuracy. This can mean generating a list of values to cross-check against records on file, but is usually beyond the direct scope of an SOA effort.

Metadata Integrity (Understanding)

This is both an analysis and a process step that helps ensure understanding and documents current data source(s) with all of the above analysis. Furthermore, this step should address the Business Process Dimension of Semantic Definition to validate the business understanding of the data, the metadata, and its usage. It may be enriched by review of business process and system documentation, functional or technical specifications, data dictionaries, subject matter experts, or other sources of data knowledge. This set of knowledge is compared to what has been discovered through earlier steps of Source System Analysis. Where discrepancies exist they should be noted. This assessment should answer the following questions:

- **Are semantic and data definitions available?** This indicates whether there is any knowledge base for understanding of the data element.
- **Is the data understood?** This indicates that the data element or domain can be discussed between multiple individuals with clarity and consistency. If a data element is described differently by business and systems resources, if it is referenced by cryptic codes (for example, 0/1 for female/male) that it cannot be reasonably deciphered, or there is a lack of information, then there is greater risk for error in data

usage.

- **Is the metadata consistent with data content?** Where there is a lack of consistency, there is greater risk for error in delivering an SOA project.

Assessing the entity-level technical dimensions

The entity-level represents a compound of data elements or attributes that form a distinctive unit or entity (for example, a person, a location). Here the techniques start with an understanding of the actual data content but considered as a group and building on that to see what that content implies for data structure and data quality. The entity-level, whose dimensions are noted in Table 3, is often overlooked but is critical to bringing data together from multiple sources and is a foundation for subsequent alignment and harmonization analysis.

Table 3: Technical data quality dimensions - Entities

Name	Description
Unique	Entity is unique —there are no duplicate values
Complete	The required domains that comprise an entity exist and are not defaulted in aggregate
Consistent	The entity's domains and domain values either persist intact or can be logically linked from one data source to another data source. Consistency can also reflect the regular use of standardized values particularly in descriptive domains
Understood	The metadata of the entity clearly states or defines the purpose of the entity and its required attributes/domains.
Timely	Entity represents the most current information resulting from the output of a business event

For example, in the case of complex or composite services, the databases from which data is gathered may have two distinct views of customer, one including prospective customers, one not. Further, one source parses customer names into distinct fields and always applied postal address verification, while the other has an unparsed name field and non-standardized addresses. What is the overlap of entities? What makes a complete entity? Which data is timelier? These are all questions applied at the entity level.

The following techniques may be applied:

- Frequency distribution across data elements
- Record linkage

- Duplicate assessment
- Entity integrity

Frequency distribution

Grouped data (usually concatenations of identifiers, text, or date fields) stored in multiple data elements may represent alternate paths to match or link entities (for example, customers, products, accounts). As above, this is a count of the number of distinct values and the frequency of their occurrence but now within the context of a set of data elements. Where an entity is considered equal to a table, the count of distinct unduplicated records provides this basic assessment. However, in many cases, an entity of importance is a subset of a record, or spans several tables and requires data to be joined through a common key.

Record linkage

This is a technique that applies a grouping of data on specified elements common to two data sources and then evaluates additional data elements for a level of commonality or distinctness. While SQL queries may achieve the desired result, if the data contains a high-level of text information where standardization is low or the chance of keying errors is high, a tool for record linkage or matching is likely necessary. It should build on the relationships discussed during the Relationship Analysis step for the technical domains. In essence, record linkage allows for the identification of alternate data associations within the entity. Consider evaluating differing groups or combinations of fields as possible access points such as a tax ID with a date of birth and a name with a date of birth as two alternatives to a name and address linkage.

Duplicate assessment

This is a count of the number of distinct values and the frequency of the occurrence of duplicates across the data elements of the entity. Duplicate assessment addresses the technical dimension of uniqueness. This should answer the following questions:

- **Is the entity unique?** Duplication within a single source or across sources as found through either a frequency distribution or a record linkage indicates that a consolidation is needed for presentation of the data.

Entity Integrity

Entity integrity is an extension of domain integrity applied to the multiple elements required of an entity. At a basic level, it rolls up the domain integrity of those data elements comprising the entity and states whether all elements are populated or not.

Entity integrity addresses the technical dimensions of completeness and consistency and, in the case of complex or composite SOA systems, feeds directly into questions of alignment and harmonization by providing a view into how the data fits together. It should answer the following questions:

- **Is there missing data?** This may be in the form of Null values or other blank value conditions (for examples, spaces) for one or multiple data elements of the entity.
- **Is there consistent data?** Looking at duplicate or linked entities, are the domains included consistent, or do they require alignment or harmonization for use?
- **Is the data aligned?** If data is expected to be delivered directly between tables or systems, an overlap of data with differing frequencies or with no overlap may indicate issues in identifying or linking specific data instances.
- **Does the data aggregate correctly?** Where an overlap of data occurs, but both have frequency greater than 1, this may indicate an issue in the ability to aggregate or consolidate the data.

Results from Source System Analysis

The results of running this source system analysis are outlined below:

- Complete set of reports of all aspects of each source system that show initial business and technical communities an initial overview of source data
- As results are analyzed, documentation can be captured and used in many contexts
 - Mapping rule specifications that feeds the functional and technical specifications
 - Documentation of follow-up items and issues identified that need to be addressed by the business
 - Information that can be used in future projects and other areas of the organization
- Project scoping and initial risk assessment of data conversion effort
- Data extraction accelerators can be re-used for baseline assessment and for development

2. Target System Analysis

There are several possibilities as to what the “target system” is for the data analyst working in an SOA project:

- For service design, the target is the canonical message model representing the superset of data structures that are exposed by all services created in the SOA project. For this case, no data is stored in the target format.
- If an operational data store (ODS) or a master data management (MDM) system is to be created to support the SOA solution, then the target is the physical data model for this data store. In some cases, the data analysis includes not only validation of the structures being used, but also an investigation into existing data that populates the target system.
- If services are being created to feed data into downstream systems, the target system is the consumer of the service. Once again, in this case, the data analyst may have to investigate not only the structure of the target but also any data already resident there to ensure the compatibility of the incoming data.

For the target system, you can perform the following tasks.

- Attribute Gap Analysis
- Data Gap Analysis
- Field Length Analysis
- Data migration scoping

Attribute Gap Analysis

This is an analysis of whether or not you can adequately define target entities and attributes and whether or not those can be mapped from existing source systems.

At the most basic level, the analyst places the list of target attributes on a spreadsheet or within a mapping tool and then maps the known list of source attributes (from one or multiple systems) against the target attributes. The semantics or definitions of the attributes are most critical at this level. Often source systems lack definition information. The metadata integrity analysis as well as the domain analysis should be used to assess which source attributes fit with the identified target attributes.

Where a target lacks a mapping from the existing source systems, attribute gaps are identified. The source system analysis is then reviewed for potential mitigation or remediation of the gap.

Where attributes are mapped, the set of source attributes must be compiled and

provided for scoping including identification of any semantic alignment requirements.

Data Gap Analysis

This analysis determines whether or not the target model can be fulfilled with existing source data and whether or not extensive transformations may have to be performed.

Similar to the Attribute Gap Analysis, the analyst places the list of target data elements on a spreadsheet or within a mapping tool and then maps the known list of source data elements (from one or multiple systems) against the target data elements. The Data Gap Analysis moves beyond the basic semantic mapping to consider requirements for data domains, data formats, mappings, transformations, standardizations, and aggregations in the target requirements. Where the target model is spread across multiple tables, the Data Gap Analysis must also consider data keys and key relationships to effectively link data elements together.

Where there is no satisfactory mapping from the existing source systems, data gaps are identified. The source system analysis is then reviewed for potential mitigation or remediation of the gap.

Where data is mapped, the set of source data elements must be compiled and provided for scoping including identification of requirements for formatting, transformation, standardization, and aggregation or matching.

Field Length Analysis

The Field Length Analysis is an analysis of whether or not current source systems can map correctly at the field length level of our target model. If not, then transformations have to be performed in the mapping phase. This may also provide feedback to the data architects to adjust the canonical data model if necessary.

Working with the Data Gap Analysis and the set of mapped domains from source-to-target, the analyst reviews the mappings for issues in the data properties. Most commonly this looks to ensure that field lengths are consistent (or precision and scale for numerical data), but may also address issues in misaligned data types.

Data migration scoping

This is a process step that helps scope the magnitude of any required data migration effort. If the SOA solution creates an ODS or MDM system, there is likely to be a data migration effort required to load the initial data into the target database. In other cases, service consumers may also require an initial data population or synchronization effort before the new solution is put into production. This scoping exercise is intended to help put some perspective on how much effort may be required, whether or not we have all the available source data that is needed and what kind of transformations may be necessary.

There are two primary inputs for the scoping exercise. The Gap Analyses (the first input) identify both the source elements required and the target elements with no appropriate mapping. By cross-comparing the required source elements to the Source System Analysis (the second input), the analyst identifies a set of domain or entity issues that must be addressed to support the SOA solution. From the set of identified gaps, the analyst must determine the criticality of the gap, what data sources may address the gap (if any), and incorporate a remediation or mitigation plan for the gap.

As part of the scoping, the analyst may determine that additional review of how data can be aligned and harmonized (particularly in a composite or complex service) is required. This additional work is also factored into the scoping effort.

3. Alignment and Harmonization Analysis

In complex and composite services, multiple components must be brought together in a consistent fashion at the same level and integrated. These components must meet the expected Business Process Data Quality dimensions to ensure appropriate integration. Alignment and Harmonization focus on five of these dimensions as outlined in Table 4.

Table 4: Business Process Data Quality Dimensions – Integrated sources

Name	Description
Common semantic definitions	Joined data elements with the same name have a commonly agreed upon enterprise business semantic definition, usage and related calculations.
Overlapping populations	Interoperability is possible between two systems sharing a partially common population. Sometimes the cross-constraints might be unsustainable.
Comparable generalization levels	Interoperability is possible between two systems sharing a partially similar generalization level. However the mapping rules must be clearly defined to avoid potentially costly impedance mismatch errors.
Comparable aggregation levels	Two data systems may use the same data structure to store semantically different representations of the same real life entities.
Agreed scale, codes and units	Scales, units and measures, as well as type codes (such as country codes, part numbers, etc.) must be matched or mapped for all different sources providing data to a service or process.

What do alignment and harmonization really mean? The previous 2 sections have established an understanding of source and target data.

Alignment is the method by which we ensure common scale and semantic definition between source data and target data, whether that is through direct 1:1 mapping, parsing and standardization of domains, or achievable through transformations.

Harmonization is the method to ensure population overlap and comparable levels of generalization and aggregation by the removal of duplicate data or overlapping definitions. If you are dealing with multiple sources, then you harmonize until you have a single definition or survivable data entity to map to. At this stage, a source-to-target mapping is critical.

Alignment and harmonization ultimately state how the Business Data Quality Dimensions are satisfied when using data from one system in a new service or joining data across systems for a composite or complex service. Common semantic understanding is achieved through standardization and attribute alignment. Consistent generalization and aggregation are achieved by standardizing to consistent levels, matching to consistent levels, and aligning and aggregating attributes to the same level. Standardization ensures that common scales and measurements are applied. Matching ensures that overlapping populations are joined in the appropriate manner to meet the business requirements.

As part of Alignment and Harmonization Analysis, you have to determine how to standardize and transform record structures to a given format, and how to match record instances. You want to ensure that you are always dealing with the right record, given that multiple versions of the same information may exist. For this case, the data analyst must establish the policies governing “survivorship” (which record survives). It is even possible that you may have to build a composite record based on different attributes from different sources.

During this process, you typically perform the following analysis.

- Standardization Analysis
- Matching Remediation Analysis
- Attribute Alignment Analysis

Standardization Analysis

Standardization Analysis is used to understand the operations necessary on source data to standardize it to the correct format. This is typically most prevalent with terms like addresses and names but can also apply to other business terms. This may be immediately apparent based on the source system analysis conducted, particularly the output of the Relationship Analysis. For example, in comparing two sets of area codes, the gap between the sources may indicate that standardization is necessary.

The Standardization Analysis process leverages earlier domain and cross-domain analysis to understand the default and invalid values, to assess where domains do

not overlap or overlap on default values, to review requirements for standardized spelling formats and abbreviations, and to validate the ability of available standardization routines to support the requirements. Usually, the standardization analysis is performed over a single domain or field specific process as identified during earlier Source System Analysis. This requires a tool or technique for testing the known standardizations. The evaluation of the standardization requirements should be conducted after the business requirements have been completed and checked, and the source system analysis and the gap analysis are complete. This helps to ensure that standardization rules are not added to the application for domains that are not required. Output from a standardization tool or process should be assessed as if the parsed and standardized domains were any other domain or field from the original data (that is, you can apply source system analysis techniques to this output to confirm the standardization requirements).

Core questions to address include:

- **Which fields need to be standardized?** Through reports or spreadsheets, match the identified input fields to the list of established corporate standards to determine which input fields need standards defined. The requirements for the target field to which the input information will be mapped should also help define how to standardize a given field.
- **Will new corporate standards be applied in the target?** Many systems carry older legacy standards that need to be updated in new target systems. If new standards exist, data should be mapped to those new standards through mapping values, against a lookup table, or based on a defined pattern or algorithm.
- **How will default or missing data be standardized?** For single source data, standardization of these default values may be through assigned mapping values, against a lookup table, or based on a defined pattern or algorithm within the data. For composite and complex services where data must be linked or matched, default values should be removed to prevent linkages of records with the same default value. Values can be propagated across linked records for missing or conflicting data so that a common representation of name or tax ID will be present on all linked records. Data fields can also be enhanced with an additional match of source files to external third-party files [for example, CASS (Postal Validation), Zip-4 processing (Zip Code enhancement), Dun & Bradstreet, or other information providers]. The goal here is to optimize match results and different handling of these default/conflicting values may be desired for target purposes. If different handling of a given field is required for service matching and target system update, then two versions of the field should be created to satisfy those different objectives. Another

option in handling these different objectives is to carry the desired match value into the match process, but retrieve the original source field value from the extract (by joining back to the extract on the unique Source ID) prior to loading the data into the target system.

- **Is it a requirement that free-form fields (names, addresses, descriptions, comments, and so on) be parsed and their components loaded into pure single domain fields for the target data structures?** Pure single domain fields make detailed queries and candidate matching during on-going applications easier, but will usually require re-assembly for generating outputs such as customer statements. In some cases it is preferable to carry both free-form and pure single domain fields to simplify both output and query/match purposes. At minimum, however, some form of match key file should be created if on-going match processing will occur to prevent the need for the re-parsing of these domains from source database records when only free-form formats (name, address line 1 & 2, and the like) are loaded to target systems.
- **When applying specified standardizations to data, is all data handled successfully?** Particularly for free-form text fields, ensure that all data is not only handled and parsed, but that the resulting standardization into specific parsed components occurs successfully.

Matching Remediation Analysis

This is the analysis of what transformations have to be specified to match and survive data in order to create the correct record – the single version of the truth – when integrating and transforming the data. This should leverage the Entity Integrity Analysis from Source System Analysis to understand what linkage is feasible and what the drivers of such matching should be.

The Match Remediation Analysis needs to drill further into the data, particularly text fields, to understand where subtle variations of data exist. Spelling variations, keystroke errors, variations in dates within a range (for example where a clinical test or pharmacy record date is after an outpatient encounter date), and so on all represent conditions that may limit or restrict the ability to match and associate data. In Match Remediation Analysis, the analyst takes the outputs from the Entity Integrity Analysis as well as frequency distributions for specific fields and develops a core list of issues to address. These may include required standardizations or data corrections, feeding back into the Standardization Analysis, or may simply be the defined rules for data matching and survivorship, particularly for composite or complex services.

Match Remediation Analysis usually requires a tool that can perform complex record linkage. These tools typically produce match output that identifies how records are matched or linked and allow insight into variations in domains between matched

records. Match outputs can also be analyzed further through source system analysis techniques to further understand issues at the domain level or compared to Duplicate Analysis results to understand issues at the entity level.

This analysis should help drive answers to the following questions:

- **Is there a specification describing the business rules to join key entities such as customer and location?** Ultimately, Match Remediation Analysis focuses on standard business practice for recognizing and joining data. These business rules must elaborate from a business perspective when and how to state that an entity is the same. For example, for an individual customer there may be two primary rules: 1) if the name and tax ID are the same, then this is the same entity; 2) if the name, address, and date of birth are the same, then this is the same entity. However, there may be additional distinctions allowing for subtle variations such as misspellings. These rules are the guidelines from which subsequent questions arise and are answered.
- **What data will be used to drive matching and survivorship?** For example, tax ID is an excellent identifier of a unique entity. However, if tax ID is only present on one source file, then tax ID is not useful for matching to a data file that does not contain a tax ID. tax ID, however, may be one of the fields that will be copied to the target system for records with cross-reference matches. Other fields that can help to define an entity are: demographic information [name, birth date, gender, other identifiers (D&B #, etc.)]; geographic information [address, phone #, coordinate locations (geocodes)]; product or part information [product or part #, standard product codes or categories, product or part description], and so on.
- **How should blank or missing values be handled in matching?** A common cause of incorrect matching is the lack of enough common or differentiating values to determine whether two records are truly related. For example, if the tax ID and birth date are missing from a record, and the only remaining fields name (match) and a PO Box address (do not match) then two records are most likely not considered matches. In reality, those records may be for the same person, but the different mailing addresses and lack of a common tax ID prevent the linkage of those records.
- **How will conflicting key values be handled in matching?** If a tax ID is different for two records that in reality belong to the same individual, this conflict may prevent those records from matching because tax ID is usually a critical match field. Another potential conflict is when individuals with different names have the same tax ID.

Again, because of the strength of these key fields in identifying individuals, these records may be erroneously brought together with a common key.

- **Will any records be excluded from matching?** If the records do not contain a name, then the business may have no reason to link records for an “unknown” entity. This linkage is also suspect, because without a name there may be no way to confirm that these are indeed related records. If linked records have no address, these records may also provide no business value (especially if it is for direct mail marketing) when linked because there is no address to use for a direct mail campaign. These records usually are identified and removed from matching until the source systems can be updated to include some of this critical match information.
- **When two or more records are related (joined), but have conflicting values for one or more domains, how will the conflict be resolved, consistent with the business rules?** Sometimes instead of creating a single representative record, all linked records are kept and the highest quality data is propagated across the matched records in order to fill in missing data or standardize populated field values. Survivorship rules for these population overlaps will need to be created to determine which source (or sources) will be the higher quality “supplier” of the field values that will be propagated to the target.
- **Will duplicate records be consolidated into single records?** Typically when records are integrated, some source specific information exists on a given record that does not need to be carried to the representative target record. Identify which fields, from which sources, should be populated on the target record.
- **What criteria will be used to consolidate records?** The survivorship rules that are to be used to create the representative records will need “tie-breaker” criteria as part of each of the rules. This may be necessary because there may be multiple linked records that satisfy the highest priority selection criteria, for a given field.

Attribute Alignment Analysis

This is to determine what kinds of transformations have to be performed to map a source term or entity to a target term or entity. By evaluating the success rates of data matching policies during solution analysis and design, the SOA designer can identify cases where alternative policies or additional data preparation effort will be required to ensure that services deliver the service levels required.

As with Standardization Analysis, the Attribute Alignment Analysis leverages earlier

domain and cross-domain analysis, but in this case, the purpose is to understand the differences in data properties such as data type, length, precision and scale, or underlying data formats to assess where domains do not align, to review requirements for transformations, and to evaluate the transformation routines necessary to support the alignment.

- **Will the source records map one-for-one to the target?** If not, specific transformations, merges, or aggregations may be required to appropriately consolidate the data. Typically, for SOA services this will more commonly be handled through matching strategies noted above in the Matching Remediation Analysis, but does not exclude the use of other transformation strategies.
- **Which fields need to be transformed?** Through reports or spreadsheets, match the identified input fields to the list of established target outputs to determine which input fields need transformations defined. The requirements for the target field to which the input information will be mapped should drive these decisions, but in cases where multiple sources do not align with the target, consideration can also be given to modifying the canonical model and the canonical message model.
- **Is there a requirement to aggregate numeric data?** When data between two source systems or the source and the target exist at different levels of aggregation, numeric data in particular may require aggregation. Earlier domain analysis may give some indication of the different scale of the data. Alignment here will identify which fields require aggregations.

Conclusion

SOA enables opportunities for wide reuse of the functions and data. With this freedom comes additional responsibility to ensure that service implementations can meet the service levels demanded by the consumers of those services.

This article has outlined an approach for executing the data quality plan to complete the Data Quality Analysis such that it can impact the effectiveness of an SOA service. It then listed the basic steps needed to investigate and analyze these issues in detail such that appropriate implementation choices can be made.

Resources

Learn

- [IBM WebSphere InformationAnalyzer and Data Quality Assessment](#): This IBM Redbooks® publication discusses how to implement IBM WebSphere Information Analyzer and related technologies in a typical financial services business scenario.
- In the [Information Integration area on developerWorks](#), get the resources you need to advance your skills on IBM's Information Platform & Solutions portfolio of products.
- Browse the [technology bookstore](#) for books on these and other technical topics.

Get products and technologies

- Download [IBM product evaluation versions](#) and get your hands on application development tools and middleware products from DB2®, Lotus®, Rational®, Tivoli®, and WebSphere.

Discuss

- Check out [developerWorks blogs](#) and get involved in the [developerWorks community](#).

About the authors

Brian Byrne

Brian Byrne has over 10 years experience in the design and development of distributed systems, spending 7 years driving the architecture of Industry Models across a range of industries. Brian is currently an architect within IBM's Information Management organization.

John Kling

John Kling is an architect in the Information Services Practice within IBM's Global Business Services. He is responsible for leading large client engagements that focus on data quality, data integration and master data management. He is currently the data team lead for the SAP implementation of a Fortune 500 industrial company.

David McCarty

David McCarty is based at IBM's European Business Solution Center in La Gaude, France and has 20 years experience designing and developing IT systems with IBM customers. He is currently a member of the Information as a Service Competency Center developing techniques and best practises for leveraging data systems in SOA solutions.

Dr. Guenter Sauter

Guenter Sauter is an architect in the Information Platform & Solutions segment within IBM's software group. He is driving architectural patterns and usage scenarios across IBM's master data management and information platform technologies. Until recently, he was the head of an architect team developing the architecture approach, patterns and best practices for Information as a Service. He is the technical co-lead for IBM's SOA Scenario on Information as a Service.

Harald Smith

Harald Smith is the Product Manager for IBM's data profiling and monitoring products: IBM Information Analyzer and IBM AuditStage. Harald has worked specifically with data quality solutions for the past 10 years and has over 25 years experience focused on system implementations, project methodology and project management, application development, technical services, and business processes primarily in the software, financial services, healthcare, and education sectors.

Peter Worcester

Peter joined IBM three years ago after almost 25 years at institutions like the US Dept. of Defense, GE Corporate and Morgan Stanley where he held technical leadership positions and gained valuable experience in Enterprise Architecture and Enterprise Data Integration. He initially joined IBM as a Sr. IT Architect as part of the architect team for Information as a Service. Currently he is a Solutions Marketing Manager for the IPS Global Services organization, specializing in MDM solutions.