

IBM InfoSphere Discovery: The next generation of data analysis

Skill Level:

[Alex Gorelik \(alexgor@us.ibm.com\)](mailto:alexgor@us.ibm.com)

Distinguished Engineer; Chief Architect, InfoSphere Discovery and Exception Manager
IBM

17 Jun 2010

Updated 29 Jun 2010

Before you can implement any information-centric project like archiving, data privacy, master data management (MDM), data warehousing, data lineage, or application data consolidation, you have to know what data you have, where it is located, and how it is related between systems. For most organizations, the data discovery and analysis process is very manual, requiring months of human involvement to discover business objects, sensitive data, cross-source data relationships, and transformation logic. In this article, learn how IBM® InfoSphere™ Discovery provides sophisticated analysis to automate the analysis process and generate actionable results.

Introduction

Software systems are by far the most complex and largest systems ever created by humans. Since data represents the persistent state of these systems, it's scale and complexity has also grown and reflects the complexity of the software application. Furthermore, data is usually distributed throughout the enterprise, copied many times over, and transformed along the way to meet the requirements of each application that uses it.

Yet we are still trying to make sense of this data manually without much documentation and governance. We assign SMEs (Subject Matter Experts) and expect them to be able to tell us what data means and where it can be found.

Imagine a small database of 100 tables, if each table has 100 columns, the database would have 10,000 columns. How realistic is it to expect a human to memorize the role of 10,000 columns and to keep track of any changes to those roles... in their heads? Now consider that most applications have thousands of tables and sometimes millions of columns, and large enterprises have thousands of applications!

Clearly, the situation has gotten beyond the ability of humans to informally manage it and on ad-hoc bases. Yet, from data archiving, to data warehousing, to application consolidation and master data management, new projects require access to existing data, and much of the effort in most projects is dedicated to figuring out where that data is, how it is structured, and how it can be aligned with similar data from other systems. This work has traditionally been done manually—mostly by guess work, trial and error, and other time-honored but tedious, expensive, and error-prone methods causing significant cost overruns, delays, and project failures.

In order to address the scale and complexity of data discovery in modern software systems, IBM InfoSphere Discovery provides a full range of capabilities to automate the analysis process and represent a new generation of software that goes well beyond data profiling by performing sophisticated analysis that generates actionable results. These capabilities automate single-source profiling, primary-foreign key discovery, business object discovery, cross-source data overlap analysis, matching key discovery, prototyping and testing for data consolidation, and automated transformation discovery. InfoSphere Discovery delivers up to 10x time and cost savings by using heuristics and sophisticated algorithms that automate analysis that profiling solutions force you to perform manually. There are two work flows you can take through the InfoSphere Discovery product. . Both workflows start with profiling, primary-foreign key discovery, and understanding individual systems, and can be used for data archiving, test data management, and sensitive data discovery. However, both paths then proceed to use the resulting information for different kinds of data-intensive projects:

Unified Schema Builder

A complete workbench for the analysis of multiple data sources and for prototyping the combination of those sources into a consolidated, unified target, such as an MDM hub, a new application, or an enterprise data warehouse. Unified Schema Builder helps build unified data table schemas by accounting for known critical data elements, and proposing statistic-based matching and conflict resolution rules before you have to write ETL code or configure an MDM hub.

Transformation Analyzer

This workflow is used when two existing systems are being mapped together to facilitate data migration, consolidation, or integration and delivers the most advanced cross-source transformation discovery capabilities available in the industry. Transformation Analyzer automates the discovery of complex

cross-source transformations and business rules (substrings, concatenations, cross-references, aggregations, case statements, arithmetic equations, and so on) between two structured data sets. It also identifies the specific data anomalies that violate the discovered rules for ongoing audit and remediation.

The InfoSphere Discovery analysis process establishes an understanding of your data sources and how they relate to each, generating actionable output that can be immediately consumed and put into action by other IBM products. A few examples include:

- Archive, test data management, and data privacy - InfoSphere Discovery finds business objects and sensitive data elements that can be immediately used by IBM Optim™ software for data archiving, test data management, and data privacy.
- Data migration, consolidation, and master data management - The Unified Schema Builder and Transformation Analyzer output delivers the matching keys, transformation logic, and consolidation rules that can be used by IBM InfoSphere DataStage® (ETL) and IBM InfoSphere MDM Server to move or consolidate data.
- Data integration - The Transformation Analyzer component discovers the *de facto* business rules that relate two data sources in your existing distributed data landscape and then outputs actionable transformation logic that can be used by IBM InfoSphere DataStage to move data from a source to a target.

IBM InfoSphere Discovery

Single-source analysis

The initial common step of any Discovery project starts with single-source analysis. The initial step in the Discovery work flow is to profile each data source included in your information-centric project. Data profiling is simply the statistical analysis of the data values in each source. Since the volume of data is vast and the data was created by highly structured applications focused on very specific tasks, the various data elements will have specific statistical properties and patterns. Data profiling enables you to discover these properties and patterns. You can then use the profiling results to:

- Check the quality and develop a detailed understanding of structure and format of each source. Are columns fully populated? Is the format of data consistent within a given column? Do you know the primary-foreign keys? What tables are used to construct an entity within each source?

- Provide a consistent baseline from which to compare the data in each source to the other sources. If one data source is well-understood but a second is not, people will tend to be biased in favor of the data source they know. Profiling each source helps to bring the understanding of each source to a common baseline.

The major steps of data profiling include:

- [Column analysis](#)
- [Primary key - foreign key analysis](#)
- [Data object analysis](#)

Column analysis

Column analysis provides basic statistics about each column within a data source. The following statistics are automatically discovered by the InfoSphere Discovery product:

- Implicit data type (Do string columns contain numbers or dates in different formats? If so, they are normalized to a common representation to facilitate further processing.)
- Pattern frequency
- Value frequency
- Length frequency
- Scale
- Precision
- Cardinality (How many unique values are in this column?)
- Selectivity (How unique is this column—cardinality/number of rows in a table?)
- Non-null selectivity
- Null count
- Non-null count
- Blank count
- Min
- Max

- Length
- Mode /Mode% (What is the most frequent value in a column—mode? What percentage of rows contain that value?)
- Sparseness (Is column mostly empty? In other words, do most rows contain the mode value?)

Information such as pattern, length, and value frequencies is extremely useful for determining that data stored within a single column is actually stored in multiple formats, some of which may not be valid for the new consolidated target data source you are in the process of building.

Primary key - foreign key analysis

Many production systems do not enforce referential integrity in the DBMS for production reasons. Instead, they enforce it in application logic. Unfortunately, it makes it challenging to understand the table relationships. It also creates potential for dirty data in the system, such as duplicate primary keys and orphaned foreign keys. This is important when you want to archive a database because you need to know the primary-foreign key structures and business objects, if you want a searchable and recoverable archive. In order to understand the schema and to insure data correctness, data analysts are forced to reverse-engineer the schema. Different profiling tools provide some level of assistance in this effort. However, IBM InfoSphere Discovery is unique in its level of automation of primary-foreign key discovery. Traditional profiling tools only find potential primary keys and force the user to manually instruct the profiling tool to then find that same key in another table. This means the data analyst must look at each table and potential primary key one by one, painstakingly working through all of the tables. This approach is impractical when dealing with more than 20 tables at a time. The IBM InfoSphere Discovery product takes primary-foreign key discovery to a new level of automation. Discovery analyzes the data values in all tables and automatically generates an entire ER Diagram prototype. It then allows the data analyst to review the statistics for each automatically discovered key, view good data and dirty data (duplicate primary keys and orphaned foreign keys), consider alternative keys, and modify the results. InfoSphere Discovery can also automatically discover composite keys. This level of automation saves significant time when performing data analysis on poorly documented or undocumented data sets.

The ER diagram generated by this analysis is exportable to data modeling tools such as InfoSphere Data Architect or CA ERwin Data Modeler. Understanding the primary key - foreign key relationships will be critical when you archive data or create consistent samples. Knowing the structure is also important when integrate data from multiple sources because these relationships are used to join tables. In normalized relational databases, you will usually need to join data from multiple tables in order to consolidate it. As a result, knowing these keys will be critical to

performing the joins.

Figure 1 illustrates the automatically discovered primary-foreign key relationships:

Figure 1. InfoSphere Discovery diagram

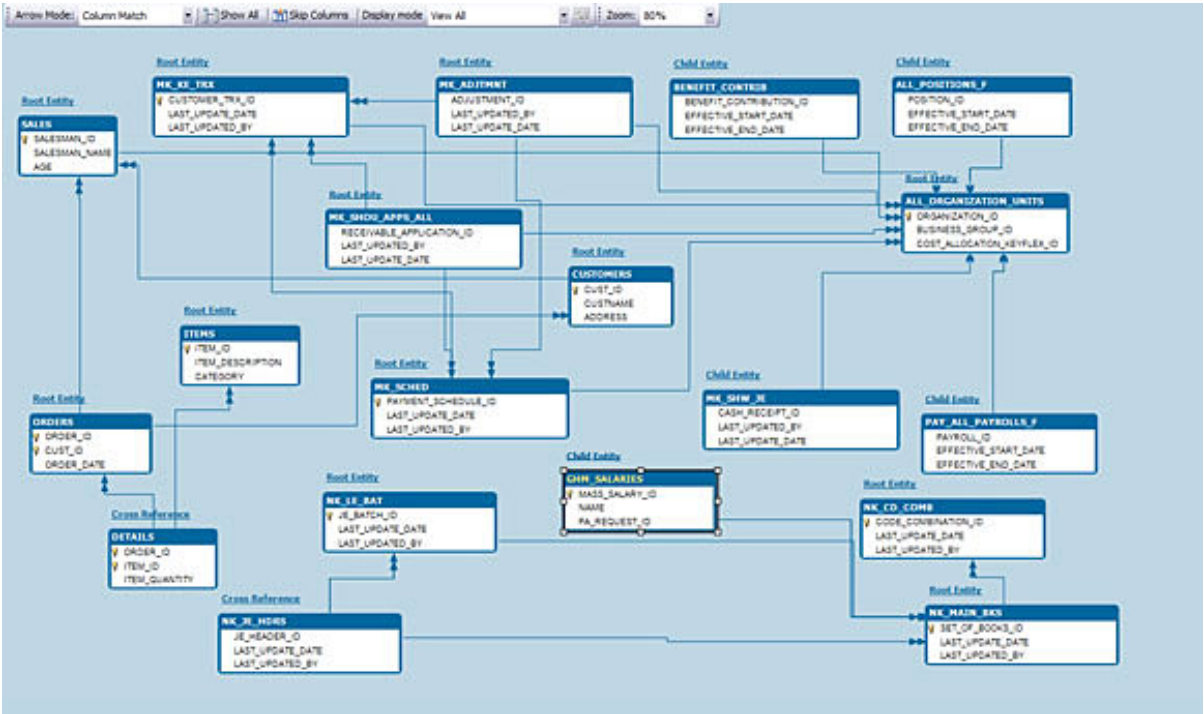


Figure 2 shows the statistics used to determine the primary-foreign key between two tables:

Figure 2. Discovered key statistics

HQ_EMPERS->HQ_EMP		Row Hit Rate		Value Hit Rate		Selectivity	
Expression		HQ_EMP	HQ_EMPERS	HQ_EMP	HQ_EMPERS	HQ_EMP	HQ_EMPERS
HQ_EMP.EMPLOYEE_ID = HQ_EMPERS.EMPID		89% (223/250)	97% (223/230)	89% (223/250)	100% (223/223)	100% (250/250)	97% (223/230)

Data object analysis

This analysis is unique to IBM InfoSphere Discovery and uses the primary-foreign key relationships to group tables into entities comprised of related tables. When you have a large number of tables, the analysis clusters related tables together into business entities (customers, orders, vendors, materials, and so on). When the analysis is finished, you have groups of tables representing specific business objects. These business object definitions can be directly consumed by IBM Optim for data archiving and for creating consistent sample sets for test data management.

Data objects are also useful when you start comparing data across sources. You'll find each source has completely different data structures and formats. Having the ability to focus on each source at the business object level and create consistent samples of data that can be compared across sources allows you to break up large

data sets into smaller related groups of tables and map those groups across data sources.

Overlap and critical data element (CDE) analysis

Overlap and CDE analysis compares data across all the sources you are evaluating. This analysis finds redundant columns and columns that are unique. The results are fact-based and measure the strength of overlap between specific columns by providing the percentage of overlapping data values between columns. This analysis is extremely valuable when you are planning to consolidate multiple data sources together or when you are looking for sensitive data across a group of data sources.

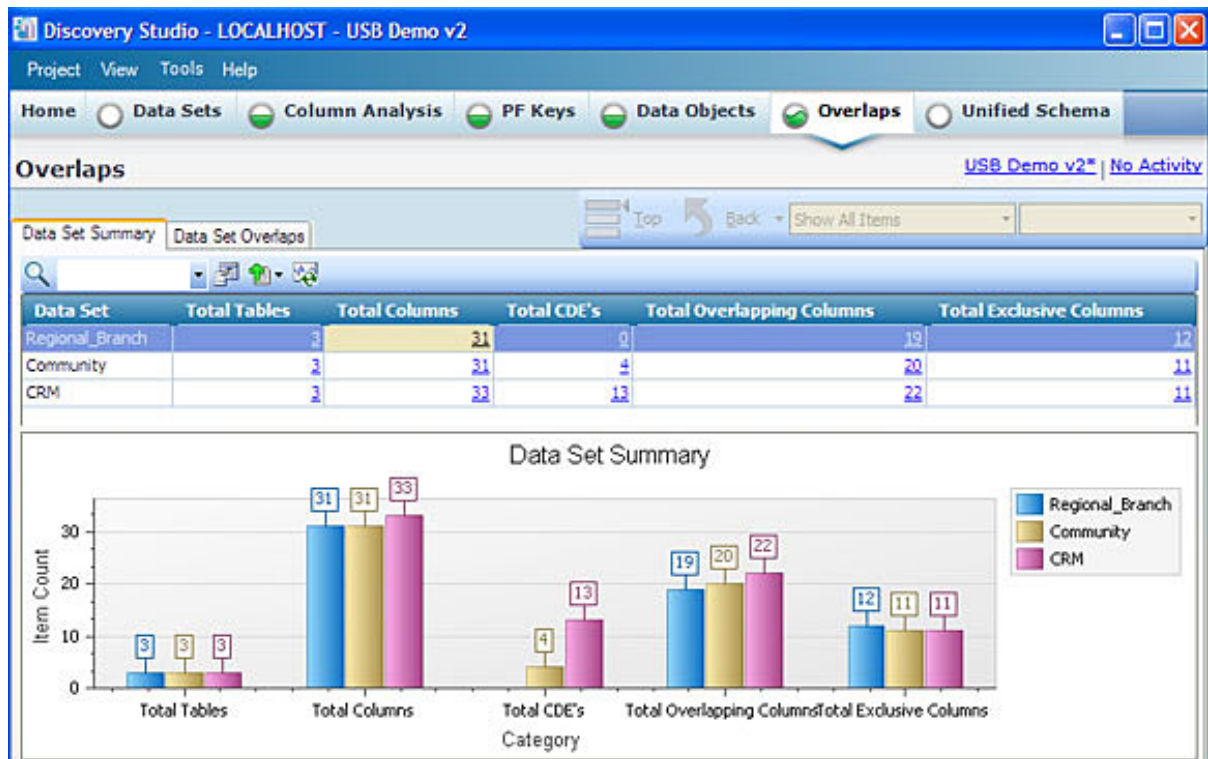
In addition to identifying overlapping and unique columns, this analysis enables you to manage the process of tagging those attributes that you consider critical. Critical data elements, or CDEs, are those attributes that you want to include in your new target schema if you are migrating data, or consolidating data into a new application, MDM hub, or data warehouse. Tagging and classifying CDEs, and performing overlap analysis will help you identify the following:

- Data sources that contain most of the critical data elements, which are often a good starting point for constructing a unified schema that will combine all of the sources
- Data sources that are not overlapping
- Data sources that subsume other data sources
- Level of consistency between overlapping data sources

Overlap analysis can also be used to get quick understanding of the types of data contained inside poorly understood data sources, as well as find columns that contain sensitive data.

Figure 3 shows a summary of how three data sources (Region, Community, and CRM) overlap with each other:

Figure 3. Discovery Studio data set summary



With traditional profiling solutions, cross-source analysis is mostly a manual process, comparing a single column at time. When comparing three or more sources, traditional profiling tools provide almost no help at all.

With InfoSphere Discovery, overlap analysis can be executed on multiple data sources simultaneously at the push of a button. All columns are rapidly compared to all other columns for overlaps, and then displayed in a spreadsheet format for viewing, sorting, and filtering. This automation makes performing overlap analysis on a large number of data sources extremely easy, allowing you to spend more time analyzing results and less time crafting SQL queries and profiling tasks by hand.

IBM InfoSphere Discovery Unified Schema Builder

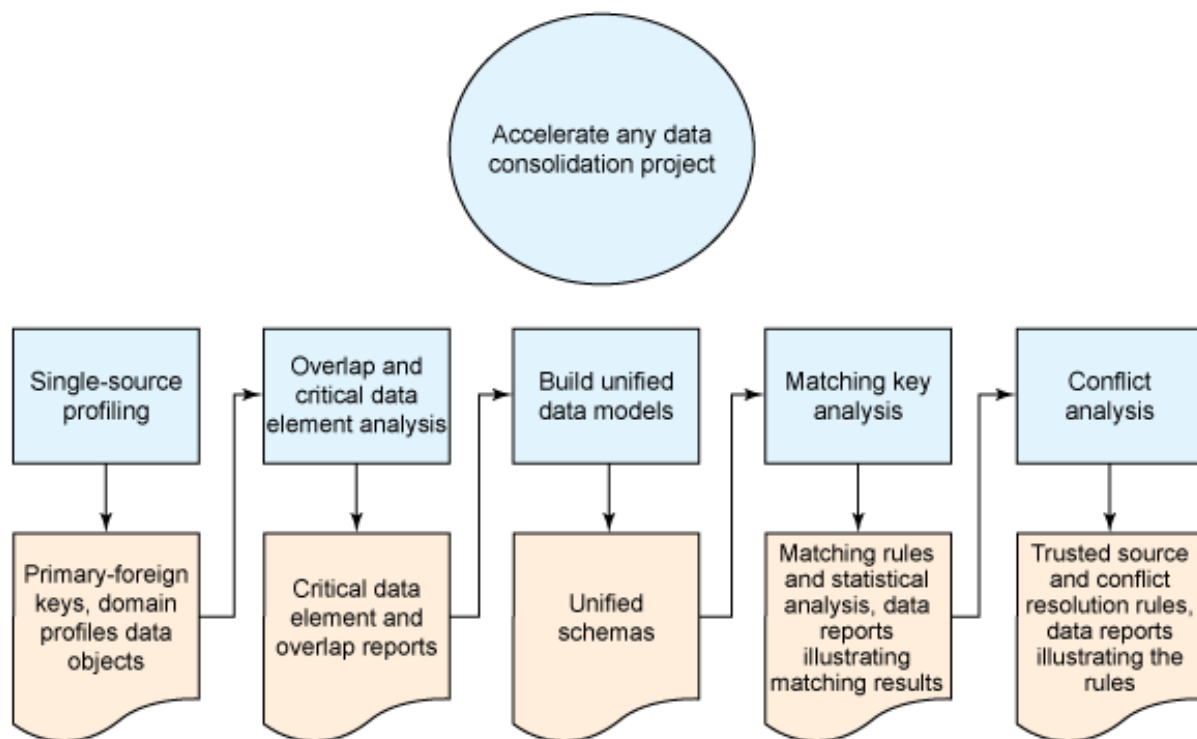
Unified Schema Builder (USB) takes the output of overlap analysis and uses it as input into a process for helping a data analyst determine the rules by which data will be consolidated for data migration, master data management, or a data warehouse, to name a few examples. The USB component delivers automation software with an embedded work flow to help you complete your consolidation project on time and within budget. [Figure 4](#) provides an overview of the methodology and the deliverables of each step of the process.

Steps:

- [Single-source profiling](#)
- [Overlap and critical data element analysis](#)
- [Build unified data models](#)
- [Matching key analysis](#)
- [Conflict analysis](#)

The capabilities in the first two steps are discussed in the "[Single-source analysis](#)" and "[Overlap and critical data element analysis](#)" sections above. The last three steps are described in the following sections of the article.

Figure 4. Steps and deliverables in data consolidation



Building unified data models

Once you have tagged the attributes you want to include in your new data model, as described in the "[Overlap and CDE](#)" section of this article, you can start designing your new unified schema. There are three main approaches to creating a unified schema:

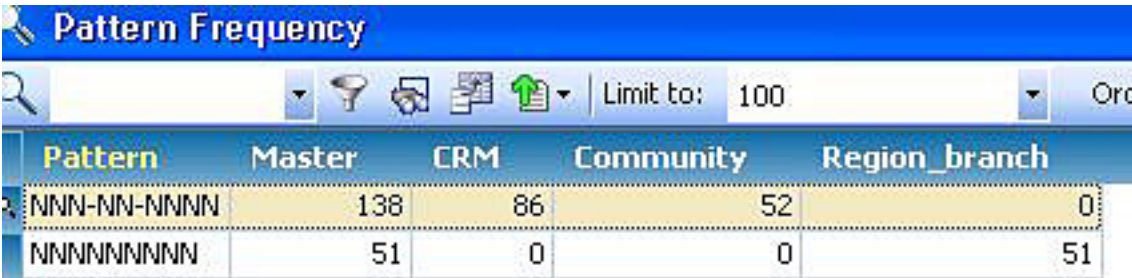
- The first is to start with requirements (for example, a set of reports and analytics to be supported by a data warehouse being modeled), construct a model that supports the requirements, and then map the CDEs to that model.

- The second approach is to leverage the schema of one of your existing data sources, bringing in just the CDEs identified in the previous step from that schema, and then extend that model with CDEs from other sources.
Note: The first and second approaches are usually used together, where an analyst would take a schema of the existing source, rework it to meet the requirements of the new application or data warehouse, and then extend it with CDEs from the other sources.
- The third approach is to use an industry schema like the IBM Industry Models (or the schema of your target application in the case of an application migration) that you may have purchased and map the CDEs to that schema.

Once the schema is created, you need to map each data source to the unified schema. Unified Schema Builder gives you powerful tools to help you develop your unified schema, including the ability to drag and drop elements from the data sources into your target schema and leverage the overlap information to automatically match attributes from multiple sources together. You can also check for domain compatibility across combined attributes using unified pattern, length, and value frequency, as well as complete column profile analysis for the combined attributes.

Figure 5 shows the pattern frequency for SSN for each individual data source (CRM, Community, and Regional Branch) and the statistics for the union of those sources (Master). There are two patterns—one with dashes and one without dashes. The data for the Region-branch has no dashes, while the other sources do have dashes. Region_branch SSNs will need to have dashes added to be consistent when combined with other data sources.

Figure 5. Pattern frequency



Pattern	Master	CRM	Community	Region_branch
NNN-NN-NNNN	138	86	52	0
NNNNNNNNNN	51	0	0	51

You can even view all of the combined source data in the unified schema format, all from the same interface. These capabilities give you the opportunity very early in the consolidation process to see how your data sources will look once they are combined. This initial prototype of your combined target happens long before you have to write any ETL, or code your merge and match rules for an MDM hub or a consolidated data warehouse. This means more errors are caught early in the

process, which results in less rework later on. As you move on through the rest of the methodology, there will be more opportunities to refine this initial prototype and add more detail to it.

Matching key analysis

One of the biggest challenges in combining multiple data sources is in determining the matching key attributes that will be used to align the rows across the various data sources. Sometimes this is easy, as you might have a transaction number that is consistently used in all of your data sources, or you may already know that the combination of First Name, Last Name, date of birth, address, and, where available, a government identifier like a Social Security Number across all of your sources allows you to align customer information. However, companies often don't know the columns that align all data sources together. Furthermore, different keys frequently have to be used to match rows from different sources and, to further complicate matters, expressions may have to be used to match, normalize, and compare data. In these cases, you need to be able to rapidly prototype and iterate to determine the best-matching condition (an expression involving matching key columns that determines whether rows are aligned).

There is a common misconception that MDM hubs solve this problem. An MDM hub will match rows of data across data sources only after you have told it what columns to use as keys for the matching and de-duping process.

This step in the methodology is all about rapidly prototyping and statistically analyzing different matching keys to determine the best matching key combination that you can then use in your MDM Hub or ETL process to align data from multiple sources when you populate your target.

InfoSphere Discovery provides analysis specifically for identifying the best-matching condition across multiple data sources and for assessing the semantics of a matching condition using statistics and data views. Automated statistical data analysis helps you determine if a matching condition for multiple sources results in:

- Over-matching (Are we grouping apples and oranges together?)
- Under-matching (Are two apples that should be together grouped separately?)

The analysis will also help you to identify better-matching conditions by providing views to help you understand the matching behavior (for example, what happens if I add a new attribute to my matching condition or take one away?) and quickly experiment with a modified-matching condition.

Additional automation is also available with the InfoSphere Discovery Transformation Analyzer component (described in the "[InfoSphere Discovery Transformation Analyzer](#)" section) that can automatically discover the matching key

between any two data sources, even if the key is a composite key that involves many columns.

Conflict analysis

At this point in the process, you will have created your new unified target schema, mapped source columns to that target, and identified a matching condition to align rows. The next step will be to determine how to choose a value for a given attribute in the target being constructed, when the values from different sources are in conflict.

One common approach is to determine trustworthiness of each data source, assign precedence to each source for each attribute, and use the values from the source with the highest precedence. For example, do I trust my contact database, the CRM system, or the data warehouse for "street address" when the information in the three sources is not consistent? In what order should I trust each system when they do conflict? Very often, determining trust comes down to the personal experience of the subject matter experts. But this does not always accurately reflect the accuracy or consistency of the actual data. However, we can now back up this personal experience with cross-source data analysis that provides for the following kinds of information and capabilities for each attribute:

- Shows the number of sources mapped to each attribute
- Performs consensus analysis to determine the source that is the most consistent with other sources for each attribute
- Allows the analyst to prototype conflict-detection rules using fuzzy and approximation matching (for example, only consider dollar amounts a conflict if they have greater than two cents difference)
- Automatically performs conflict resolution for each attribute based on use of the most recent value or the most trustworthy source, and allows the analyst to prototype custom conflict-resolution rules (for example, always choose the latest date for last_update column)

IBM InfoSphere Discovery Unified Schema Builder automatically generates trust rules based on statistical analysis. And while you may still want to consider personal experience in designating trust rules, the statistical information is now available to back up what was formerly only gut-feel analysis.

Figure 6 illustrates how the software has used statistical analysis to determine that for the attribute <Middle Name>, CRM should be trusted first, Community second, and Region_branch third, per the precedence column.

Figure 6. Conflicts (per data set)

Conflicts (per Data Set)			
Data Set	Expression	Completen...	Preced...
CRM	CRM_BRCH_1A_LT.MIDDLE_NAME	100.00 % (86/86)	1
Community	COMMUNITY_BRCH_LT.M_NAME	100.00 % (52/52)	2
Region_branch	REGION_BRCH_LT.MI	98.04 % (50/51)	3

Once this final analysis is performed, you now have all of the information you need to begin coding ETL to populate a data warehouse, write the migration scripts, or program your MDM hub merge/match rules. This methodology and automation process replaces what was previously a frustrating guessing game, significantly accelerating the cross-source analysis process, improving the quality of the results, and increasing the likelihood of success for data consolidation projects.

IBM InfoSphere Discovery Transformation Analyzer

The Transformation Analyzer component of InfoSphere Discovery automates discovery of complex cross-source transformations and business rules by analyzing data values and patterns across two data sources. This component is used when you know that two data sources are related, but you also know those relationship can't be described by simple overlaps in data values but require figuring out how data is transformed between the two data sources. Data migration, application retirement, data warehousing, and master data management almost always require the mapping and discovery of complex transformation logic between two or more data sources. Transformation Analyzer accelerates this process by automating much of the analysis involved and replacing tedious manual work.

The capabilities of Transformation Analyzer go well beyond what is available in traditional data profiling tools. In traditional profiling, if you know a business rule, the software will validate it for you. With the Discovery Transformation Analyzer component, the software analyzes millions of data values to identify patterns in the data and deduce the de facto rules that currently govern how your data is transformed as it moves between sources. It then tests these rules against the data values to measure the exact accuracy of that rule.

InfoSphere Discovery Transformation Analyzer is the first data-driven data analysis workbench that automatically discovers, documents, and validates cross-source business rules, transformations, and data inconsistencies between structured datasets. The software accomplishes this by analyzing the data values, not the metadata. Innovative data exploration and analysis techniques enable InfoSphere Discovery to automatically discover data relationships, business rules, and complex transformations for most of the structured data in your enterprise.

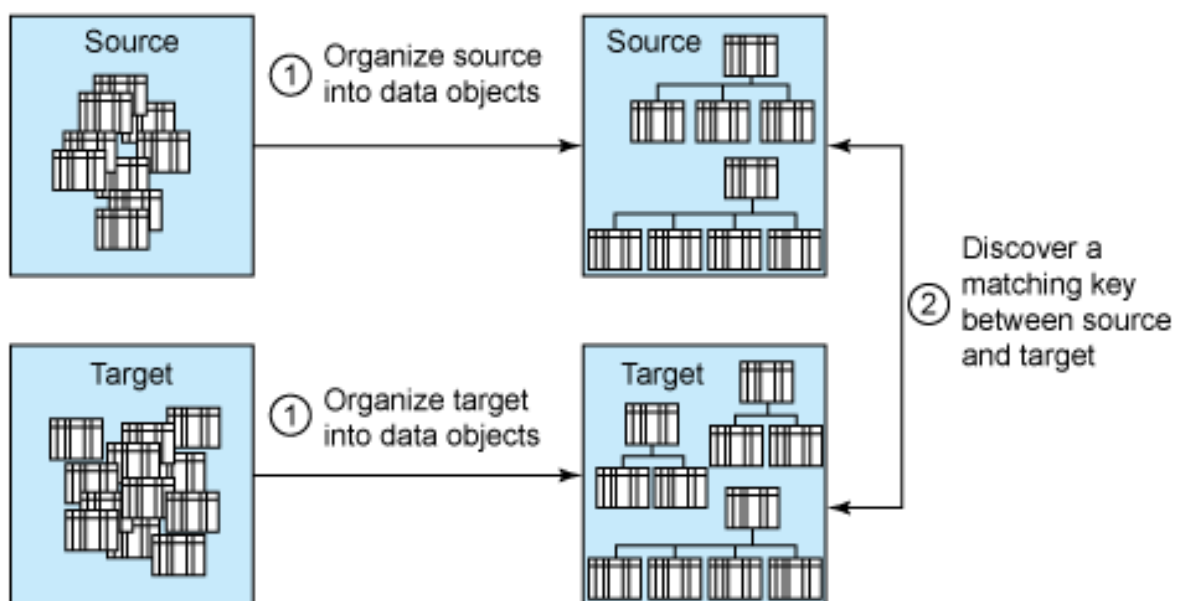
Discovery Transformation Analyzer is the first product to analyze not just data quality, but also **data relationship quality**. The result is lower risk, accelerated time to deployment, and lower integration costs for even the most complex data

governance and integration projects.

The data-driven discovery process

InfoSphere Discovery provides an automated methodology and a step-by-step process for discovering, documenting, and validating cross-source transformations and business logic. The Transformation Discovery process leverages some of the same functions discussed in the "[Single-source analysis](#)" section: profile each source, discover primary foreign keys, and then generate data objects. Let's take a look at the high-level process for cross-source transformation discovery (see [Figure 7](#)).

Figure 7. Data-driven discovery



- The data values in each data source are analyzed to discover data objects. These groups are based on discovered primary-foreign key relationships, as well as additional categorization based on the determination if tables are transaction, reference, or attribute tables. Sophisticated sampling mechanisms are available to reduce the amount of data analysis without sacrificing discovery effectiveness.
- Once each data source is organized into data objects, the software determines which data objects (subgroups of tables) in the source relate to the corresponding data objects in the target.
- The next level of discovery takes each matching source data object and target data object pair, and for each table in the target data object determines which source tables are used to generate that table. These source tables and the corresponding target tables are used to generate a

"map." You can think of a map as an SQL query that can be applied to the source tables to generate the target table.

- Once the map is created, Discovery automatically determines the matching keys that will be used to align the rows between the source tables and the target table, and then discovers the transformations and business rules that explain the complex cross-source column level relationships. The analyst interactively verifies and approves the discovered relationships. (More details of this step are contained below in ["Table mapping detailed example"](#).)
- Data maps containing the transformation and business logic and statistics reflecting how well the data adheres to the business logic are output in SQL, XML, or ETL script formats, ready to be used by downstream processes to transform and move the data.

Transformation / Business rule discovery detailed example

The following diagram illustrates the steps that InfoSphere Discovery follows to automatically map the columns in the Product Sales table from Application 1 to the columns in Product Sales table in Application 2. Discovery reads the actual data values, not just metadata (such as column names), in order to identify these data relationships.

Figure 8. Transformation / Business rule discovery detailed example

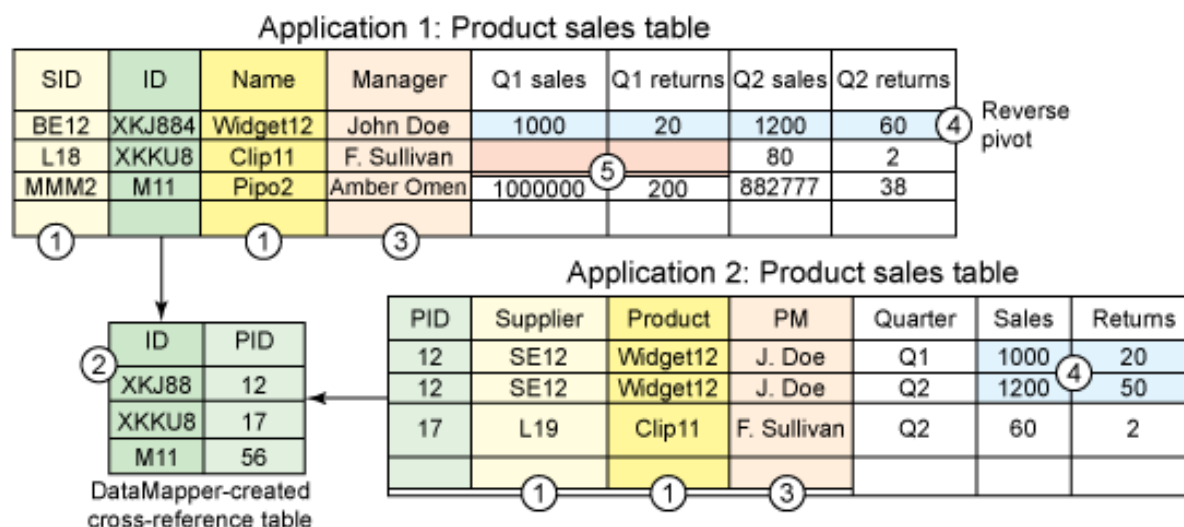


Figure 8 illustrates the steps that InfoSphere Discovery follows to automatically map the columns in the Product Sales table from Application 1 to the columns in Product Sales table in Application 2. Discovery reads the actual data values, not just metadata (such as column names), in order to identify these data relationships.

- First, InfoSphere Discovery discovers a matching key used to align the

rows between the two data sets. In this case, the software discovers that the natural key consisting of supplier id and product name relates the two tables. This key is stored in the SID and Name columns in Application 1, and the Supplier and Product columns in Application 2. Only by reading the data values (not the metadata), can Discovery find this relationship, since the column names Name and Product cannot be logically related by themselves.

2. A cross-reference table is created between the primary keys in the two tables (ID in Application 1 and PID in Application 2). Discovery uses the natural keys discovered in Step 1 to cross-reference the primary keys.
3. Discovery discovers that the PM column in Application 2 consists of the first character of the Manager column in Application 1, followed by a period (.), a space, and the second token of the Manager column.
4. The values in Q1Sales, Q1Returns, Q2Sales, Q2Returns, and so on from Application 1 have been reverse pivoted (turned into rows) in Application 2. Discovery generates a separate mapping for each set of pivoted columns that creates a single row (for example, Q1Sales and Q1Returns).
5. Finally, Discovery discovers a filter on the Q1Sales column—only rows with non-null Q1Sales have corresponding rows in App2.

Types of transformations discovered

InfoSphere Discovery Transformation Analyzer discovers simple 1:1 relationships as well as extremely complex types of transformations, as shown in Table 1:

Table 1. Types of transformations

	Type of transformation	Example
Scalar	Identity mapping	Target.Name = Source.Name
	Substring	Target.ProductNumber = Substr(Source.SerialNumber, 1, 7) Source.ProductID
	Concatenation	Target.Name = Source.FirstName ' ' substr(Source.MiddleInitial, 1, 1) ' ' Source.LastName
	Constants	Target.Status = 'S'
	Tokens	Target.FirstName = token(Source.Name, 1)
	Type and date conversions	
Joins	Inner, left outer	
Aggregation	Sum, average, minimum,	Target.Sales =

	maximum, count	sum(Orders.Amount)
Reverse pivot		
Cross-reference	Key, code	
Multi-nested case statements	=, !=, <, <=, >, >= in, not in, null, not null Conjunctions	Target.Code = CASE WHEN Units < 10000 and State in ('NY', 'CA') THEN '1' WHEN State in ('MA') THEN '2' ELSE Source.CFlag END
Column arithmetic	Addition, subtraction, multiplication, division, or rounding of one or two columns	Target.ItemPrice = Source.ItemCost * .08

Cross-references are stored in lookup tables. Discovery can automatically generate a lookup table or use an existing lookup table.

Discovering data inconsistencies and exceptions

Because the Transformation Analyzer evaluates data values to discover transformations, this approach also identifies inconsistencies that can result in lost revenue, customer dissatisfaction, and regulatory fines. In the real life example in [Figure 9](#), the software automatically discovered that the column called AGE (which shows the age of drivers in an insurance application) is related by the case statement to a column called Youthful_Driver in a second application. (Case statement: WHEN AGE <= 25 THEN Youthful_Driver = 'Y' ELSE 'N' END.)

However, not all rows of data followed the discovered rule that the Youthful_Driver column should be a 'Y' when the value of the AGE column is less than or equal to 25. In the example, an 83-year-old driver has a "Y" in the Youthful_Driver column. This row of data is automatically flagged as not following the discovered rule. The data analyst can now research whether the driver was actually 83 or if there was some sort of manual override that caused the business rule to be violated.

Note: The example in [Figure 9](#) shows a subset of the actual data set (about 10000 rows of data) that was used to automatically discover the case transformation.

Figure 9. Transformation example

Transformation	
CASE WHEN AGE <=25 THEN Youthful_Driver = 'Y' ELSE 'N' END	
Hit Rate = 90%	
Application A	Application B
AGE	Youthful_Driver
17	Y
24	Y
55	N
28	N
40	N
33	N
Exception 83	Y
29	N
36	N
42	N

InfoSphere Discovery architecture

The Discovery platform is built upon a unique architecture that combines a scalable, high-performance engine with a graphical user interface that provides guided analysis for the data analyst.

- **Discovery Server:** Coordinates the deployment and administration of the overall Discovery environment.
- **Repository:** Holds the mapping metadata discovered during the source-to-target mapping process.
- **Discovery Engine:** The core component that analyzes data between multiple data sources and generates business rules, transformation, and data maps. Multiple Discovery Engines can be deployed on multiple physical systems allowing for a high level of scalability.
- **Discovery Studio:** Graphical mapping environment that displays information about data sources, structures, and mappings discovered by the Discovery Engine, as well as actual data. This allows analysts to rapidly investigate, design, and validate mappings between disparate systems
- **Staging Database:** Stages data from source and target systems for use by the Discovery Engines and Discovery Studio during the mapping process.
- **Reports:** HTML and Excel metadata reports that document discovered metadata, showing data lineage, mappings, and relationships for all data

analyzed by Discovery.

- **Integration with IBM Products:** The business objects, business rules, transformation logic, and all of the metadata discovered by InfoSphere Discovery are exportable in XML format or in an Excel report. XML describing the schema is directly consumed by other IBM products, including IBM Optim, InfoSphere Information Analyzer, InfoSphere Data Architect, and the IBM Metadata Server. Discovered transformations, matching keys, and conflict resolution rules can also be used by DataStage and the IBM MDM Server. Mappings are exported using FastTrack-compatible CSV files and can be used to generate DataStage job skeletons or provide lineage in Metadata Server. Discovered matching keys and conflict resolution rules can also be used by QualityStage and the IBM MDM Server.

Summary

At customer sites, IBM InfoSphere Discovery has reduced the time and resources required to deploy IT integration projects as much as 10x. As the only software product to examine the data values themselves, instead of relying on metadata or specifications for integration planning, InfoSphere Discovery is a pioneer in the IT integration marketplace.

Discovery can accelerate time to deployment for many IT projects, including:

- Data archiving
- Application retirement
- Application Consolidation
- Master data management
- Sensitive data discovery and masking
- Data lineage discovery and documentation
- Data warehousing

Resources

Learn

- [IBM InfoSphere Discovery](#): Learn more about IBM InfoSphere Discovery.
- [IBM InfoSphere Foundation Tools](#): Learn more about IBM InfoSphere Foundation tools
- [InfoSphere page on developerWorks](#): Get the resources you need to advance your skills on IBM InfoSphere products.
- [developerWorks Information Management zone](#): Learn more about Information Management. Find technical documentation, how-to articles, education, downloads, product information, and more.
- Stay current with [developerWorks technical events and webcasts](#).

Get products and technologies

- Build your next development project with [IBM trial software](#), available for download directly from developerWorks.

Discuss

- Participate in [developerWorks blogs](#) and get involved in the [My developerWorks community](#); with your personal profile and custom home page, you can tailor developerWorks to your interests and interact with other developerWorks users.

About the author

Alex Gorelik



Alex Gorelik is an IBM Distinguished Engineer and Chief Architect with 20 years experience developing cutting-edge data integration technology and a long track record of successful senior management technical roles, including founder and chief technology officer (CTO) of Exeros Inc., which was acquired by IBM in May of 2009. Alex holds a Bachelor's Degree in Computer Science from Columbia University and a Master's Degree in Computer Science from Stanford University.