A Pragmatic Approach to Enterprise-wide Data Lineage
Part 2: Implementation

A Global IDs Technical Whitepaper
November 21, 2019 (revised)

Arka Mukherjee, Ph.D., Kirk Kanzelberger, Ph.D., Bill Winkler, and Stefanos Damianakis, Ph.D.
info@globalids.com

This document is the second in a series of educational technical papers that we plan to release in the coming months in order to share our experiences and lessons learned in enterprise data management with a wider public.[1]

Introduction

In Part 1 of this whitepaper, we defined *data lineage* as the tracing of the *movement of information* throughout the data ecosystem of an enterprise, from its points of origin to terminal consumption points such as reporting artifacts. As we discussed, data lineage enables key business functions in the following areas:

- Data lineage provides objective evidence of *data provenance*, which is key to regulatory compliance.

- Data lineage underlies an objective and systematic approach to *data quality*, since it uncovers all of the critical points in the flow of information and thus guides the most effective placement of data quality checks and controls.

- Data lineage also underlies an objective approach to *impact analysis* of changes to the data ecosystem owing to data harmonization, cloud migration, organization mergers, application rationalization, and so forth. Not understanding how information actually flows from end to end through the enterprise undercuts rational decision-making in all these areas.

In order to support these key functions, data lineage must provide a view of information flow that is simultaneously *comprehensive* over the whole enterprise*, granular* down to object and attribute levels, and *logically framed* in terms of business concepts. The implementation of

[1] An upcoming technical whitepaper will address the topic of transformations that data may undergo at each stage of its journey through the enterprise.

data lineage must access enterprise data and metadata and apply advanced algorithms to map the flow of each type of business concept or object through the ecosystem.

Data lineage: the foundation for solutions to mission-critical business problems

The solution to many business problems — from regulatory compliance to data quality validation to impact analysis in the service of cloud migration, ecosystem rationalization and other forms of data modernization — depends critically upon the ability to answer two fundamental questions about enterprise data: *Where did the data come from? What path did it take?*

These answers are precisely what data lineage provides. Seen in this light, data lineage is the indispensable foundation for addressing a multitude of ever more pressing needs and legal requirements of medium to large enterprises with expansive data ecosystems. Data lineage provides comprehensive knowledge of the flow of information mapped to business concepts, as depicted schematically in figure 1:

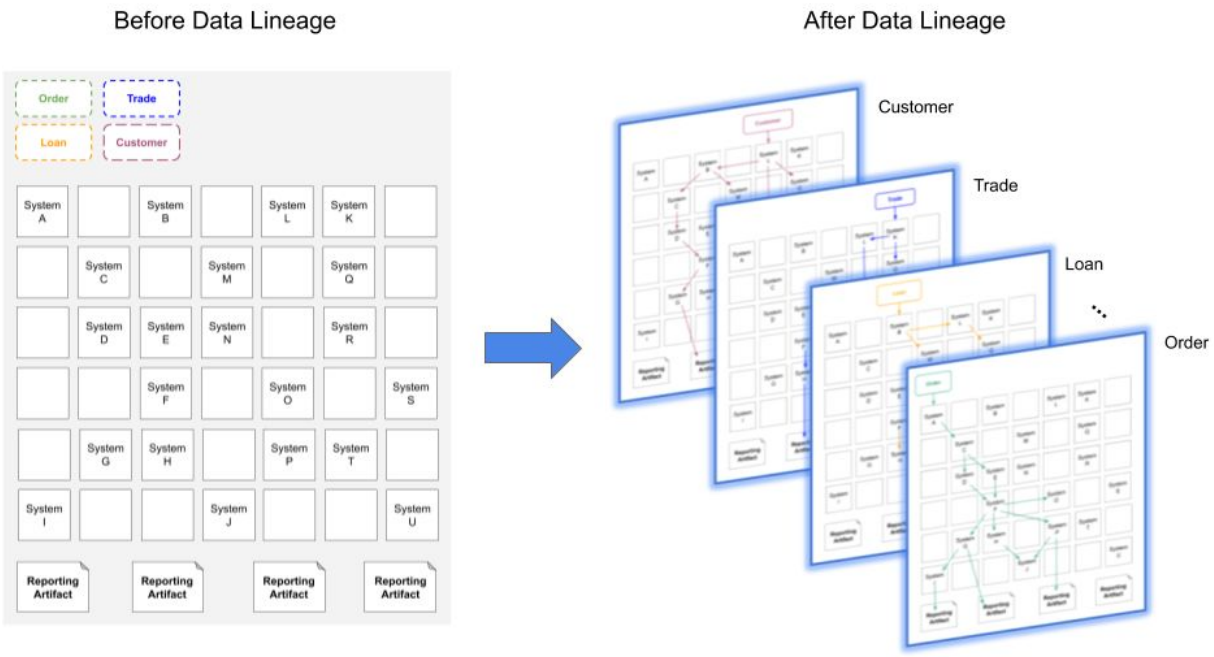

Figure 1. Schematic depiction of the data ecosystem before and after the implementation of data lineage.

As depicted in the figure, the implementation of data lineage generates comprehensive views of the flow of information for every business concept. Data lineage allows the data appearing in reporting artifacts (such as reports) to be traced back through many systems to its points of

origin. The comprehensiveness of the view enables visualization of critical flow points for every business concept, and accurate forecasting of the impact of changes to the data landscape. Moreover, thanks to the granular view available for every validated flow segment, data lineage offers an automatic window into data quality wherever data moves from one place to another within the ecosystem.

The implementation of data lineage

The implementation of data lineage, following the approach proposed in part 1, leverages the strengths of automation and advanced algorithms, along with human inputs, to make enterprise-wide data lineage tractable and effective.

The implementation of data lineage involves a number of stages, as depicted in Figure 2:

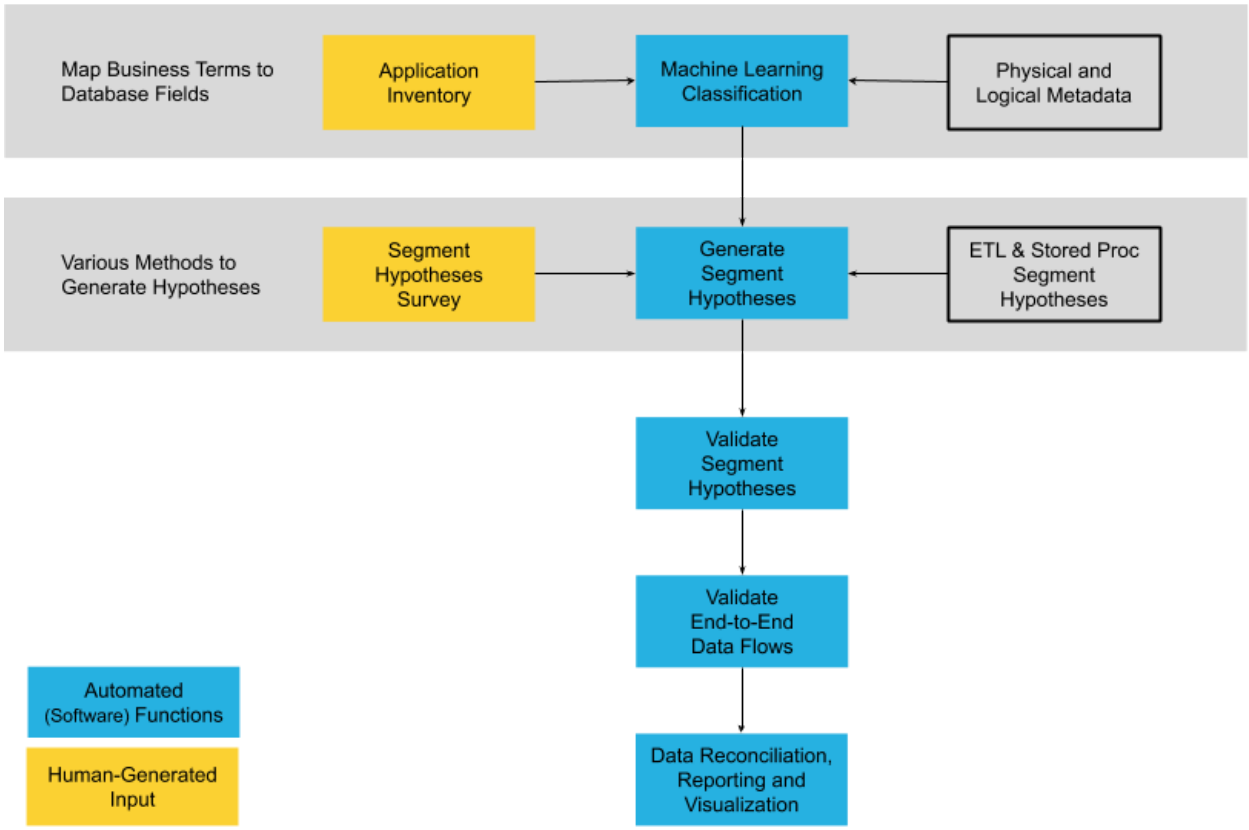

Figure 2. Stages and inputs in the implementation of data lineage.

Following figure 2, we arrange our discussion in five steps:

1. Logical-to-physical mapping using machine learning classification
2. Generate flow segment hypotheses
3. Validate flow segment hypotheses
4. Validate end-to-end data flows
5. Data reconciliation, reporting, and visualization

Step One: Logical to physical mapping using machine learning classification

The first stage of the implementation uses machine learning algorithms to bridge the all-important gap between (a) business knowledge expressed in *logical* terms (business concepts), and (b) the *physical* representation of data, expressed as tables and columns within database schemas.

Logical metadata consists primarily in the *concepts* that are fundamental to the particular enterprise (Customer, Order, Product, Subscriber, Account, Trade, and so forth, depending on the nature of the business). An enterprise may have over a hundred such core concepts. Each concept corresponds to a kind of "thing" or logical *entity type* that can have a physical representation — indeed, many physical representations — in data.

Each kind of entity type needs to be elaborated or refined in terms of its logically constituent *properties.* For example, in a particular business, an entity of type Order logically will always have a unique identifier or Order Number, an Order Date when the order occurred, and an Account Number with which it is associated, as well as other properties.

While this logical metadata is being gathered, *physical metadata* is also being gathered from application owners and by probing the applications themselves. Physical metadata comprises information about systems, applications, databases, database tables, and schemas across the whole enterprise. Physical metadata is granular, reaching the attribute (column) level. This may add up to thousands of applications and databases, tens of thousands of tables, and millions of columns across all systems.

We now perform the key computational step, which uses machine learning algorithms to *link* granular physical data specifications to the corresponding business concepts and their logical properties. It is this linkage of physical representations to logical concepts that enables software to trace the lineage of each type of business entity. The machine learning algorithms

are trained to make human-like classification decisions for millions of physical data columns, simultaneously coping with the variations in physical representation that are inevitable across many databases and schemas. Figure 3 gives a simple illustration of this mapping of different physical representations to the single business concept TRADE:

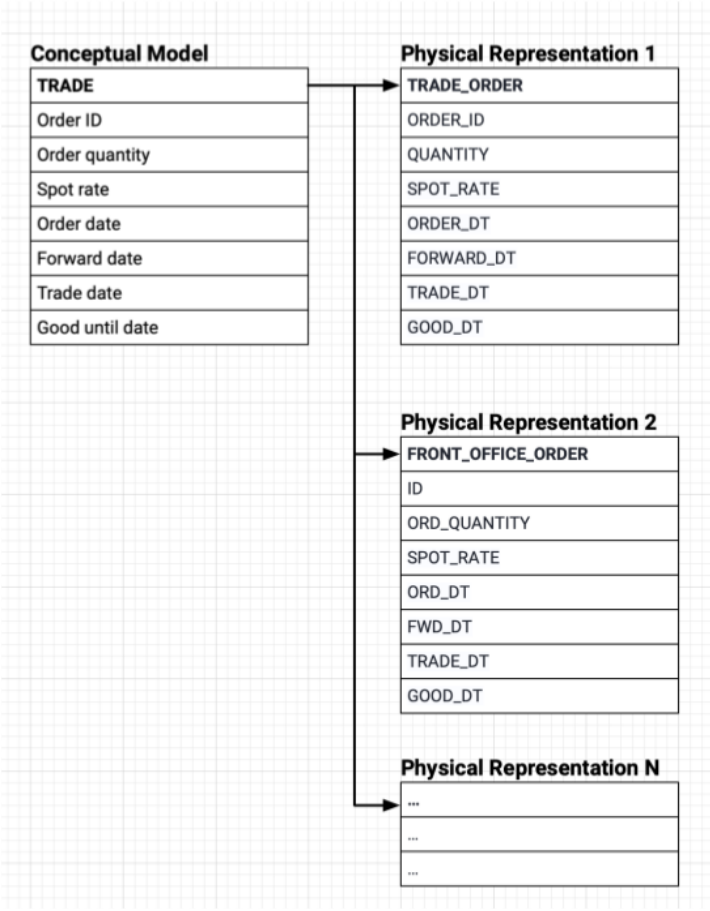

Figure 3. Mapping different physical representations to the same business concept.

In sum, the outcome of this first stage is the *fully elaborated business context* for the flow of information within the enterprise. "Fully elaborated" means that all the core business concepts have been specified, their concepts have been refined in terms of their constituent properties, and machine learning algorithms have computed a mapping between these refined concepts and all the different physical representations they may have. Given this computed business context, data lineage software can now detect the occurrence of physical representations of the *same* business concept at either end of a hypothetical flow segment between any two systems in the enterprise.

We note that Figure 3 depicts one of the *simplest* cases of the data classification problem, in which two physical representations of the same concept (TRADE) have the same set of properties, but with differing physical column names. Commonly, while every physical variant may include the same few core properties, many variants will lack some properties that others have (for example, an email address field). It is also common that semantically similar properties will belong to more than one active business concept (for example, an account number may be a property of a customer, an order, a subscription, and so forth). The overlap between concepts, as well as the difference between core and "nonessential" properties, is a dimension of the data classification performed by the machine learning models in this stage.

Step Two: Generate flow segment hypotheses

We now move on to the second stage, which is the generation of flow segment hypotheses.

While Global IDs continues its research on algorithms for automating data lineage more fully, as we pointed out in Part 1, human input helps make data lineage tractable by overcoming the need for an exhaustive search for data connections between systems.

In this second stage, application owners are surveyed regarding the upstream dependencies of their applications. Application owners are generally better-informed regarding upstream dependencies than downstream relationships, since the latter relationships involve the usage of interfaces they have provided to others in the enterprise.

Application owners may also suspect that data is flowing between their application and some other (downstream, for example), but lack the details.

Whether dependencies be known or suspected, these flow specifications, together with the logical-to-physical mappings computed by the machine learning algorithms in the first stage, are sufficient for the automated generation of *flow hypotheses* between source and destination systems.

An example will help to illustrate this process. An application owner tells us that his application, C, is fed by two upstream applications, A and B. Given the logical-to-physical mappings that were computed using the learning algorithms in step one, data lineage software is able to detect which logical business concepts are found in each of the three applications. The software is then able to compute *conceptual flow hypotheses* for each business concept that is shared between A and C, and between B and C. Figure 4 below illustrates two such flow hypotheses for the business concept of an Order.

The automated generation of conceptual flow segment hypotheses is performed for all business concepts, in a process that “works backwards upstream” from destination to source systems across the whole data ecosystem.

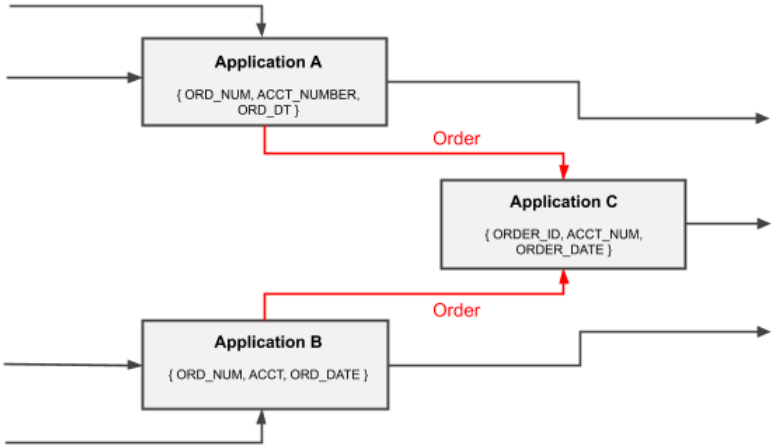

Figure 4. Two flow segment hypotheses for Orders between applications A, B, and C.

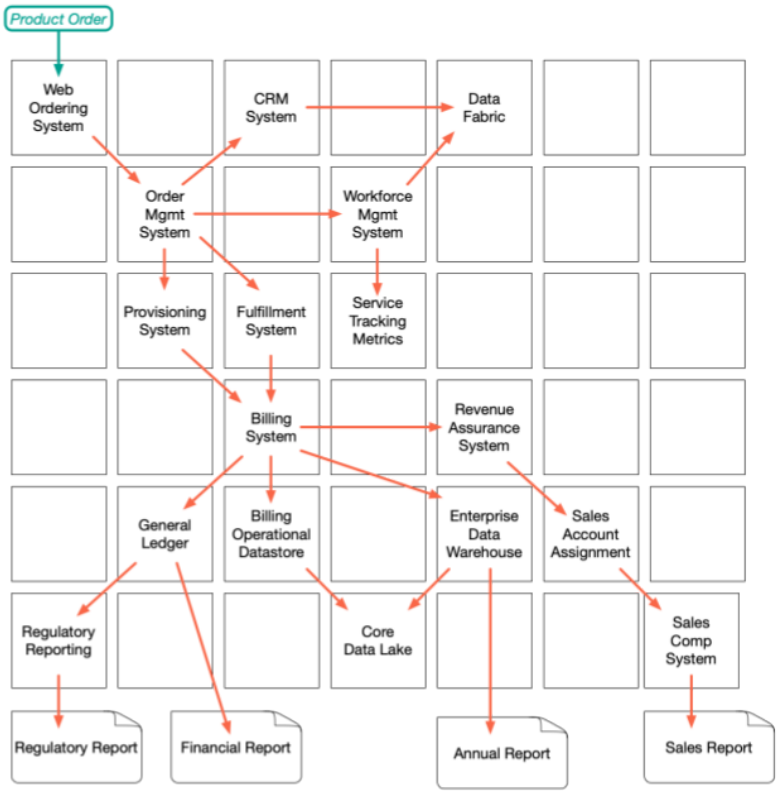

Figure 5. The lineage hypothesis for Product Orders across systems in a data environment.

Figure 5 depicts the automated generation of flow segment hypotheses, tracing the segments of an overall enterprise-wide lineage hypothesis, or "data supply chain", for each and every business concept. Keep in mind that these supply chains define the flow of data in *conceptual terms*. They are maps of the flow of logical business entities through the ecosystem, made possible by the algorithmic data classification performed in the first stage that linked physical data columns to logical business concepts.

Step Three: Validate flow segment hypotheses

The third stage consists in automated validation of the flow segments hypothesized in the second stage. As described in Part 1 of the whitepaper, data lineage software uses sampling techniques (timestamp and stroboscopic analysis) to access the physical data and associated metadata that is actually flowing between pairs of systems.

Given the logical-to-physical mappings generated in step two using machine learning, the software is able to analyze the data and metadata pertaining to the flow connection hypothesized to exist between a pair of applications A and B. Many levels of automated validation can now be carried out, including:

- *Logical attribute* detection (per business concept) at either end of the connection.
- *ID column* based validation of the flow of data across the connection.
- *Non-ID column* based validation of the flow of data across the connection.
- *Row* (data value) based validation of the flow of data across the connection.
- *Temporal pattern* validation of the direction of data flow across the connection.

The result of automated validation at all these levels is not only a confirmation or negation of the existence of data flow between A and B, but a comprehensive picture of exactly which logical attributes are involved, the arrival (or not) of actual instances of the particular business entity, the integrity of the values transmitted, and the timing and directionality of the flow.

Figure 6 below offers a schematic depiction of these dimensions of automated flow validation for entities of type TRADE from an upstream application A to a downstream application B.

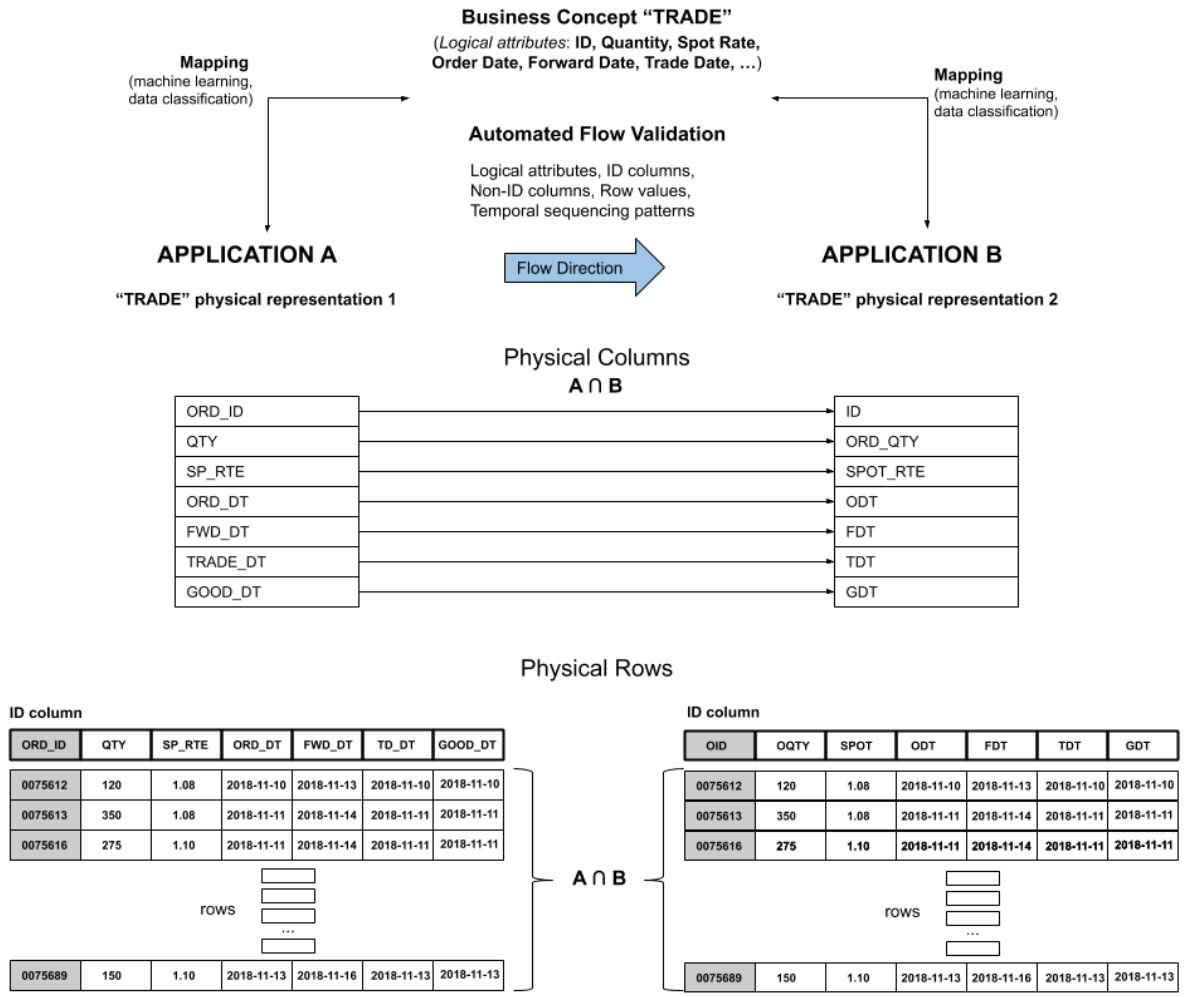

Figure 6. Automated validation of the flow of instances of concept "TRADE" between two applications A and B.

In Figure 6, note the two different physical representations of TRADE across the two different applications, that are both linked to the business concept TRADE via the data classification methods of step two.

Automated validation of flow segments across the entire ecosystem generates reports that are reviewed and confirmed by an Information Architect or other responsible official. At the heart of the reports are flow diagrams for each business concept/entity, one of which is depicted in Figure 7. The figure shows the data lineage of the concept PRODUCT ORDER, from its point of origin (Web Ordering System) to its ultimate destinations in reporting artifacts. The green arrows in the figure represent flow segment hypotheses that were *confirmed* by the automated

analysis. The red arrows in the figure represent flow segment hypotheses that were *not confirmed* by the automated analysis.

The many levels of automated validation allow the viewer of the report to expand any individual flow segment to view the details of information flow.

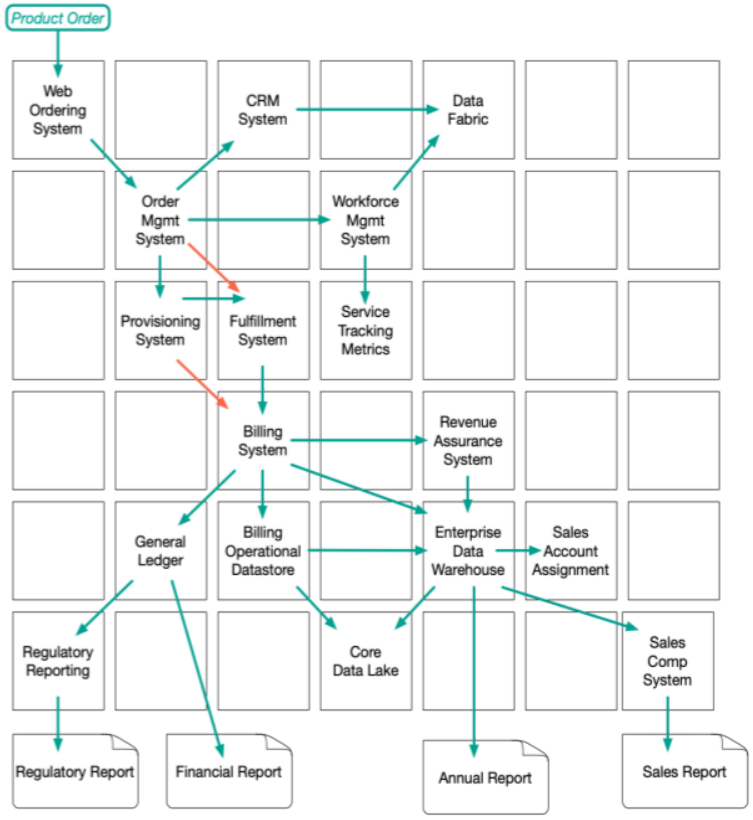

Figure 7. Confirmed lineage diagram for Product Orders across systems in a data environment.

Step Four: Validate end-to-end data flows

In the fourth stage, we test for the *continuity* of flow along paths consisting of multiple flow segments. Up to this point we have confirmed flows on a per-segment basis between pairs of applications, by detecting common instances of a business concept between one application as source and the other as destination. For example, we detect common instances of a business concept between applications A and B, and our temporal analysis confirms A as the source application and B as the destination. In a similar way, we confirm that instances of the same business concept flow from application B as source to a third application C as destination.

Now we want to confirm the flow of instances of this business concept across a longer path, from A to B to C, by computing the common instances across the two *segment flows*. In this

way, we compute the knowledge of *end-to-end flows* that have common instances across each flow path segment from an origin system to a terminal consumption point.

Using the diagram in figure 7 above as an example, we have confirmed in step four that there is in fact a flow of Product Order from the Web Ordering System to the Order Management System, and also from the Order Management System to the CRM System. Now in step five, we confirm continuity across these two segments by performing record level comparisons across the two flow segments.

Step Five: Data reconciliation, reporting, and visualization

In the final stage, we perform exhaustive data reconciliation for every validated flow segment. We can verify the arrival of data and its correctness at the row level, at the column level, and at the data value level (did a value change from source to destination?). Note that *data reconciliation* is itself an important business function that is automatically enabled by the implementation of data lineage.

Data lineage sampling methods may now be performed at desired intervals to periodically validate the flow of data and monitor its integrity. A dashboard for each flow segment allows visualization of flow characteristics and data integrity for the complete lineage of every business concept — i.e., every step in the path that the data pertaining to a given concept takes through the ecosystem. As required, reports can be generated to provide objective evidence of the lineage of a particular business entity.

As the data landscape changes over time, lineage hypotheses are recomputed and validated, so that the reconciliation and reporting processes are kept up to date.

Conclusion

The automated discovery and validation of flow patterns, linked to business concepts and granular down to the object and attribute level, provides the kind of objective evidence needed for demonstrations of data provenance, data quality controls, and impact analyses. Data lineage also immediately enables vital internal audit processes such as data reconciliation.

It has become a cliché to say that today's business enterprises revolve around their data, but for all that, the statement is true. Like blood in the body, data feeds the enterprise through being in motion. On its journey it can become lost, contaminated, or bleed to places it should not. Responsible data governance therefore must embrace the flow of information from end to end through the enterprise, which means the implementation of data lineage as one of the foundations for rational decision-making, quality control, and compliance.