DOI 10.3233/ISU-130712
IOS Press

‘Big Data’ collaboration: Exploring, recording and sharing enterprise knowledge

Sreenivas R. Sukumar * and Regina K. Ferrell
Oak Ridge National Laboratory, 1 Bethel Valley Road, TN 37830, USA
Tel.: +1 865 241 6641; Fax: +1 865 241 3191; E-mails: {sukumarsr, ferrellrk}@ornl.gov

Abstract. As data sources and data size proliferate, knowledge discovery from ‘Big Data’ is starting to pose several challenges. In this paper, we address a specific challenge in the practice of enterprise knowledge management while extracting actionable nuggets from diverse data sources of seemingly related information. In particular, we address the challenge of archiving knowledge gained through collaboration, dissemination and visualization as part of the data analysis inference and decision-making lifecycle. We motivate the implementation of an enterprise data discovery and knowledge recorder tool called SEEKER based on a real-world case study. We motivate the implementation of an enterprise data discovery and knowledge recorder tool called SEEKER (Schema Exploration and Evolving Knowledge Entity Recorder) based on the queries and the analytical artifacts that are being created by analysts as they use the data. We show how the tool serves as a digital record of institutional domain knowledge and as documentation for the evolution of data elements, queries and schemas over time. As a knowledge management service, a tool like SEEKER saves enterprise resources and time by (1) avoiding analytic “silos” (i.e., a separate set of data that is not included in an enterprise’s data administration), (2) expediting the process of multi-source data integration and (3) intelligently documenting discoveries from collaborating analysts.

Keywords: ‘Big Data’ integration, knowledge management, digitizing domain knowledge

1. Introduction

Actionable discoveries through aggregation, statistical analysis and creative slice-and-dice of data in science, government and private industry are all providing indisputable evidence for the future of data-driven business. The success of the Kroger retail food chain’s loyalty program [4,8,14] (enabled by market micro-segmentation of current and future customers), the “connect and develop” model of strategic innovation (Procter and Gamble) [10], the IBM Smarter Planet campaign [16,18], the intelligence-based war against terrorism [13,17], and the 2012 presidential campaign [1,12] all have one thing in common – the formula of collecting, fusing, and integrating data from multiple sources to derive strategic advantage and actionable intelligence. The evidence and promise of business value from such data-driven success has pushed the market for enterprise data analysis beyond traditional transaction processing. Today, with proliferation in the number of data sources and increasing data sizes (often referred to as ‘Big Data’), the paradigm of knowledge discovery [9] and data-driven enterprise strategy is facing the challenges of data volume, variety, velocity and veracity.

Significant advances made in hardware and software solutions for data storage, retrieval and analysis are addressing some of these challenges. This paper addresses a specific, emerging challenge in

[*] Corresponding author. E-mail: sukumarsr@ornl.gov.

the market practice of ‘Big Data’ analytics: the inability of data scientists/knowledge engineers and analysts to quickly digest diverse sources of seemingly related information across multiple data assets within and across enterprises. The bottleneck begins when datasets are shared in different formats (structured, unstructured), hosted across different infrastructures (cloud, custom hardware, etc.) or in different databases (row oriented, column oriented, file oriented etc.), and/or in different schemas that all have to be quickly integrated and seamlessly explored. The question posed and addressed in this paper is, *how can we expedite or automate the multi-source data analytics lifecycle?* In other words, we address the knowledge management question of *how can the domain and data understanding of an analyst working for Company A be transferred effectively in digital form to another analyst requesting the data in Company B?*

Scenarios that require active collaborative knowledge management tools occur both in the industry and government sectors – e.g. the Census Bureau working with the Centers for Disease Control (CDC) after a natural disaster or two healthcare companies that have recently merged looking to find common patient diagnoses to streamline their operations. The critical actionable insight of value in either case is enabled only after the analyst from one agency educates the analyst at the other agency. Today, the data analytics lifecycle of finding the right data, getting access to the data, exploring/learning about the data, analyzing the data and publishing and sharing the insights gained after analysis, although is a collaborative effort, ends up becoming a tedious and often improperly documented endeavor. So, how can we make the collaborative process of knowledge discovery from multi-enterprise data more productive? How can we make it easy for any analyst to learn about datasets they have never seen before? How can we enable analysts to share knowledge with peers and pose multi-agency interaction a productive experience – not just for the analysts employed today but also for the analysts in future?

As a solution that answers these questions, we describe the implementation of a tool that serves as an evolving knowledge recorder that not only captures schema and data-element relationships across data sources, but also captures queries, notes and the comments of analysts. The tool also tracks the data elements of value within the enterprise based on the queries and the analytical artifacts being created from the data. In addition to being a data exploration mechanism when exposed/published across enterprise warehouses, the tool also acts as a digital record of institutional domain knowledge of the data elements within the enterprise. Based on our preliminary results, we assert that a knowledge recorder tool implemented across data sources as an information service is a collaboratively generated, digitized form of domain and data knowledge that saves enterprise resources and time by (1) avoiding analytic silos, and (2) effectively documenting what other analysts have already discovered from the data.

This paper is organized as follows:

- Section 2 presents a deeper background on the challenges analysts face today when working with ‘Big Data’ (particularly ones that have to work off of multi-tenant data warehouses and shareable cloud architectures), lists the different aspects of an analyst’s needs, and emphasizes the role of collaboration while sharing, collecting, recording, and integrating analytical knowledge within and across enterprise data sources. Section 2 also documents our experiences integrating particularly large datasets from disparate sources: these experiences guided the development of a tool for analysts/knowledge engineers and data scientists.
- Section 3 demonstrates the capabilities of the tool.
- Section 4 summarizes our work and concludes with recommendations and future directions.

2. Background: Understanding the analyst's needs for collaboration

The next-generation analyst who is posed with the challenge of discovering actionable nuggets from data stored across enterprise data-warehouses begins work with the following question: *how should data sources be staged for analysis?* More specifically, the question is, *how should one data source from one agency be conveniently posed in context to a different data source from another agency to answer a specific question?*. To an analyst, the process of answering this question often translates to (1) learning about new data stores (Oracle, Postgres, Greenplum, Hive on Hadoop, Cassandra, etc.), (2) developing expertise in a new query language (SQL, SPARQL, etc.), (3) obtaining query and data access to new interfaces for data, (4) installing and configuring tools he or she is comfortable with (MATLAB, SAS, etc.) and (5) engaging in several conversations with another analyst (or a data custodian) with whom he or she is collaborating. If the analyst's role is specialized in data retrieval (analysis for enterprise hindsight), exploratory analysis expertise (analysis for enterprise insight), or predictive model building (analysis for enterprise foresight), the complexity and thereby the need for collaboration increases. Today, this cumbersome, iterative process of knowledge exchange toward understanding the domain and the data before proceeding to the multi-source data integration and analysis is the most time-consuming aspect of 'Big Data' analysis.

Based on the authors' experiences as analysts having daily interactions with fellow data scientists, we broadly classify the collaboration lifecycle along the following three dimensions: (1) understanding domain knowledge, (2) understanding the data model, and (3) understanding the data. Sections 2.1, 2.2 and 2.3 discuss common issues of understanding domain knowledge, the data model, and the data, and how each gets resolved through collaboration. This anecdote-driven approach helps us understand the underlying difficulty of designing an automation tool to expedite the process.

2.1. Understanding domain knowledge

Understanding domain knowledge and mapping it to the data of interest to the analyst is the most critical yet time consuming aspect of multi-source data integration and analysis. It is the process of understanding the facts, procedures, and processes involved in generating the data. A sound understanding of domain knowledge helps the analyst control the search by writing smarter queries. Lack of understanding makes it difficult to identify the key views/columns and navigate through definitions in the data that contribute to resolving business problems of interest to the enterprise. When analysts collaborate to understand domain knowledge and when they exchange data dictionaries, they exchange notes on "why" and "how" a particular data element is being collected or measured. Understanding the "why" ensures that important information about the processes and systems that generated the original data is not lost or subsumed in the integration process. Understanding the "how" ensures that accurate summaries/comparisons across data assets do not lead to poor conclusions resulting from a lack of key information about different processes involved in generating the original data.

For example, when two healthcare insurance companies are sharing their data, the analysts would exchange notes about the different entities in the system: beneficiaries, providers, policy and insurance plans and transaction claims. They would then explain the process of how beneficiaries become eligible for a service, seek a service from a provider, how the provider files a transaction claim for a particular cost based on the coverage defined under the beneficiaries insurance plan, and how claims are adjudicated and later archived. They would then discuss the attributes/data elements associated with a beneficiary, the provider, the insurance plan, etc. They would also discuss the rules and business logic under which such transactions are made within their respective domains. Even though the two companies may have many of the same kinds of data entities and business transactions in their respective data domains, their naming conventions, the basis of their data structures, and their relationships between data structures will be unique. Each company’s perspective on business and data transactions will be different from the other’s. The discussion to understand the domain begins when the analysts have shared data dictionaries but do not yet have access to the data records. Understanding the domain is an iterative learning process that progresses as analysts spend more time exploring the data.

2.2. Understanding the data model

The issues and concerns discussed in this paper are specific to the traditional data warehousing practice of relational database systems. The new, emerging “NoSQL” data stores pose a different kind of integration and analysis challenge to the analysts who must deal with hybrid storage and retrieval models. Data model integration with hybrid storage SQL–NoSQL models are beyond the scope of this paper. We restrict our study to relational data models, where data are organized into a structure that is optimal for storage, indexing and querying.

An analyst must have a strong understanding of the data structure (that is, how fields within the structure relate to each other and how the process being measured or described by the data is collected) to ensure that a dataset is accurately analyzed. When analysts collaborate, the first pieces of information that get exchanged are as follows:

1. Which table is the starter table?
2. What fields join within tables?
3. What database normalization form is the model?
4. What fields have to be indexed and pre-joined?
5. What is the business logic applied within tables to capture the domain processes?

More often than not, the data model encodes a lot of information about the domain. While some of this information is revealed by careful study of the data structures specified and their relationships, other highly useful information may not be as readily apparent. With extremely large datasets, seemingly simple calculation queries that compute sums and averages can take more time, and often, a simple query may return seemingly reliable data which is grossly inaccurate due to an incomplete understanding of the construction of the data model. Sometimes the lack of a good understanding can stem from the fact that relational databases often go through many changes in real-world deployments. For example, mistakes may have been made in the initial schema. These mistakes need to be corrected, and at times, these corrections do not happen until long after the system is deployed. Other times, business needs change; new data elements have to be added, or old ones have to be removed, and even if the database was designed correctly in the beginning, it becomes vital from an analyst’s perspective to track the schema evolution. In addition, analysts and developers may add new datasets and create relational linkages between this data and the original data structure. Finding and exploring potential data links between similar or related systems can greatly expand understanding of the business and may provide new ways of exploring and utilizing data already available with no additional costs. As new actionable data are added to a data repository and linkages can be made between the original data and the enhanced information, new queries will be generated to analyze the impact of considering the old data in light of the added information. These new queries may often lead to a new understanding of data relationships and may result in the generation of further relational data.

Continuing with the health insurance example, the analysts trying to understand the data model would pose questions such as the following:

1. What are the primary keys and foreign keys in the data model?
2. What are the unique identifiers for a beneficiary (customer ID, name, address, etc.)?
3. What are the unique identifiers for the providers (provider ID, license number, name, etc.)?
4. How do we differentiate between a physician and a hospital based on the data?
5. Why does the data model have separate tables for provider practice location and billing location?
6. What are the 1–1 relationships and 1–many relationships in the data model?

2.3. Understanding the data

An examination of the data itself with respect to the structure may yield other important clues as to what combinations of field values might indicate particular groupings of related data or what conditions indicate a particular type of action has taken place. Finally, when insight into data relationships is combined with business process knowledge of the data collection and representation, a greatly enhanced capability to analyze and understand data begins that is often expanded upon while querying. Understanding the data before beginning to execute queries and generating reports is important not only to ensure completeness or cohesiveness of the reports, but also to discover data quality and provenance issues. While data quality issues can plague any dataset, such issues may be more problematic with 'Big Data'. A significant subset of "bad" data easily hides undetected in a large data repository, particularly in fields with less structure or in architectures such as Hadoop, where adherence to the data model is not mandated before retrieval. Although understanding the process and the model has confirmed the importance of a data element, the assumption that the data element is actually filled in correctly can be a bad assumption. Missing data, corrupt data, and/or data filled with default or miscellaneous characters that hamper interpretation of analysis results often show up in the real world. Knowing about these fields is important in order to be able prune the data elements of value to the analyst. This not only saves query execution time, but it also contributes to the accuracy of the analyst's reports. Also, certain business rules that were not implemented during the design of the schema will have to be implemented while querying. We have realized that such queries are difficult to design without understanding the aspects of the domain and the model along with the data. However, once the query is constructed, it captures the results of the analyst's knowledge about the data and the domain, and there is significant business value in sharing the logic and code with fellow analysts.

As an analyst's knowledge of a dataset and the processes of the domain grows in completeness, the value of the analyst's analytic extractions increases, as well. The value of the analyst's knowledge often translates into value to the enterprise both in terms of cost savings and actionable intelligence. The question that we pose and address is, *can this knowledge be recorded in a shareable form? If so, how can we digitize and archive the analyst's understanding of the domain, of the data model, and of the data?* Section 3 presents the case for the need for knowledge management as enterprises move to a data-driven strategy, and Section 4 presents initial results from a tool that we have developed to address these needs.

3. Need for knowledge management

Today's data warehouses have frequently evolved over time and may typically hold a large number of tables with many elements in each table. In addition, in today's corporate world, where business entities both merge and divide, data changes made to accommodate or adapt to these transitions have an impact on the structures and types of data stored. Furthermore, as today’s typical worker tends to have a shorter tenure at any one job, institutional knowledge and memory are frequently lost. This also occurs as older workers retire and their trainees move on to other positions. Any one of these factors can have a detrimental effect on data analysis and full utility of data warehouse resources. When some or all of these factors occur together (large, complex data sources, changing business models, and/or short-lived employee tenures), a major gap can result in knowing and understanding undocumented links and commonalities between data systems and warehouses.

The experiences we have referenced for this paper originate from integrating and querying 30 or more diverse, disparate databases, each database hosting 30 or more tables, and each table having 50 or more data elements and billions of records in each of those tables. We tracked the evolution of a schema over an 18-month period, studying the interactions of knowledge workers with ‘Big Data’ to extract the requirements for an automatic knowledge transfer mechanism for collaboration within and across enterprise warehouse systems. We observed the following trends:

- As more data were received and analysts began to bring their individual interests and experience to the project, there was a proliferation of new third-party reference information and data sources that all analysts were eager to utilize. In addition, several new data sources were arriving, and there was limited or no metadata provided to elaborate on exactly what kind of information was in each data set, how it was generated, and how it might relate to datasets already received. The complex combination of table data and the sheer volume of data to retrieve for business user requirements frequently strained the client system’s capability to manage such requests.
- To manage short-term needs for proof of concept or for testing theories, new data silos were being developed to facilitate difficult or complex analysis concepts. Unfortunately, important metadata about the limitation of the data extracted for these silos was often lost, but these data silos became quite popular for their simplicity and speed. In addition, these silos had the potential to hold stale data because no mechanism was in place to support their maintenance or upkeep.
- Power users who accessed the multi-tenant data warehouses directly through SQL Query Interfaces [5], or analysis tools such as SAS/EG [6] found the disparate data models too complex and/or difficult to integrate and use. They were interested in a metadata search capability and data dictionary management system. They sought commonly used queries and business intelligence tools to help understand the complex data sources to which they had access.
- Analysts were requesting insulation from the underlying complexities of the data model, as well as an easier way to navigate through the data structure to retrieve the data they needed. They were also interested in using both the results and the historical queries made to improve their own query capability and speed of analysis.

4. SEEKER: “Schema Exploration and Evolving Knowledge Entity Recorder” tool

The concept and need for a tool like SEEKER arose from development work performed at the Oak Ridge National Laboratory (ORNL) on a multi-agency, multi-source data integration project. Our customer had a number of related but separated (siloed) data information systems (assets) with discrete data extracts. The challenge posed to ORNL was to bring these datasets together and help analysts explore the disparate data assets in a seamless fashion for knowledge discovery purposes. In the process of analyzing this data and dealing with some frustrations of identifying commonalties among these large data sets, there was a need to identify certain data elements and to determine in what table/database structures the data elements were found. This was a difficult task for analysts because they had to deal with over 10,000 data elements spread across 30 different databases. These elements rarely were named in a consistent manner from system to system. The initial SEEKER implementation was developed to assist in the process, and additional capability grew out of experiences with exploring and analyzing these data sets. This tool played a significant role in several analysts collaboratively coming together in a divide-and-conquer strategy for understanding the siloed datasets, extracting business rules and based on their exploratory insights being able to integrate the diverse datasets seamlessly.

In this section, we present screen shots of the visualization of metadata and the interactive human-assisted knowledge discovery in action using the SEEKER tool. The visuals present the following features of the tool (due to space constraints for further drill-downs in the visualization tool, the figures are provided as a notional overview of the results):

- Access and schema exploration from multiple 'Big Data' infrastructures;
- Metadata browser to search, browse and visualize metadata schema;
- Virtual schema builder for metadata and data level matching, schema-level hypothesis generation;
- Record of institutional memory.

4.1. Access and schema extraction from multiple 'Big Data' infrastructures

The basic core functions of SEEKER are (1) to extract the schema, table and field structures from databases hosted in most commercial and open source databases, (2) to attempt to find important related fields through fuzzy matching across data schemas in different databases, (3) to display important relationship links graphically for interactive exploration, and (4) capture, share and publish collaborative discussions about data and meta-data. Figure 1 illustrates an example of these functions. SEEKER, when provided with access to the infrastructures where the data reside, automatically scrapes schema information about databases. Our implementation has connectors to commercial servers such as Microsoft SQL Server, Oracle and Greenplum, in addition to open source storage tools like Apache Hadoop and MySQL. When configured with user login information and a frequency to update evolving schemas, SEEKER builds the metadata dictionary that can be interactively browsed, as shown in Fig. 1.

SEEKER acts as a virtual warehouse of data sources within and across enterprise data assets of interest and access to the analyst. The analyst can quickly understand the organization structure of a data asset and then can access a data source of interest within that asset. This assistance towards understanding

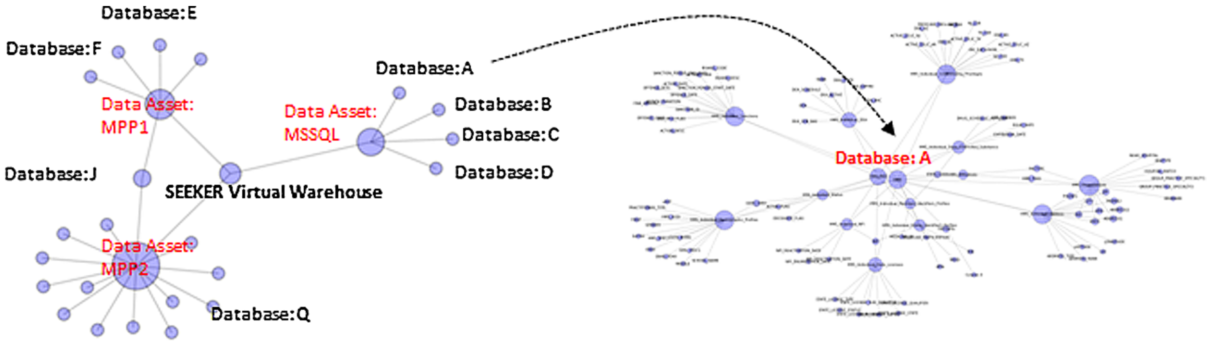

Fig. 1. SEEKER acts as a virtual warehouse of data sources within and across enterprise data assets of interest and access to the analyst. (Colors are visible in the online version of the article; http://dx.doi.org/10.3233/ISU-130712.)

the data model serves as a cursory step to help an analyst write a query for a previously unseen dataset. Furthermore, the ability to probe the virtual data warehouse to come up with insights that other analysts have generated off of it, perusing comments and asking questions to fellow-analysts is a key feature in our solution.

The analyst can interact with all of the databases as though they were all in one centralized virtual warehouse. Furthermore, SEEKER allows the analyst to explore the entity relationship structure of a database. This capability allows knowledge workers to interpret previously unseen schemas that are required for their analysis and to understand where the key elements are within the new dataset. The SEEKER visualization automatically identifies the starter table, the primary keys, and the foreign keys within the schema (if that information was not previously provided) and exposes them to the end user.

4.2. Metadata browser: Search, browse and visualize metadata schema

Often with ‘Big Data’ systems, the number of data elements can be very overwhelming to an analyst. This is particularly true when the analyst has to use datasets spread over multi-tenant data assets. SEEKER implements a search function for that purpose. SEEKER’s metadata repository is exposed as a search tool. This is particularly useful when the analyst is trying to find a field needed for analysis. By indexing metadata, the associated data dictionaries, and the analyst’s and database administrator’s comments, SEEKER serves as a metadata search-engine for the enterprise. For example, the analyst may be interested in writing a query that requires cost fields in “Database: Cost_report” and “Database: Cost_rebates”. The analyst understands Database Cost_report and its elements but does not know where to find the cost-related field in the “Database Cost_rebates” which is from another source. Without having to understand the structure and having to navigate thousands of other elements, the analyst can search for the word cost, thereby listing data elements related to cost in the virtual warehouse, as shown in Fig. 2.

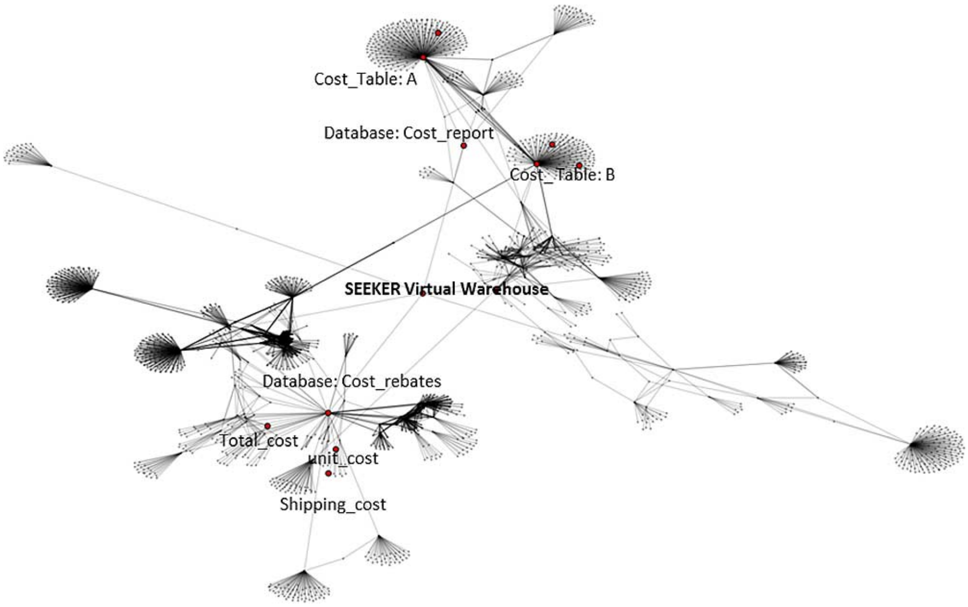

Fig. 2. Screen shot of SEEKER searching through the repository for “cost” related fields. (Colors are visible in the online version of the article; http://dx.doi.org/10.3233/ISU-130712.)

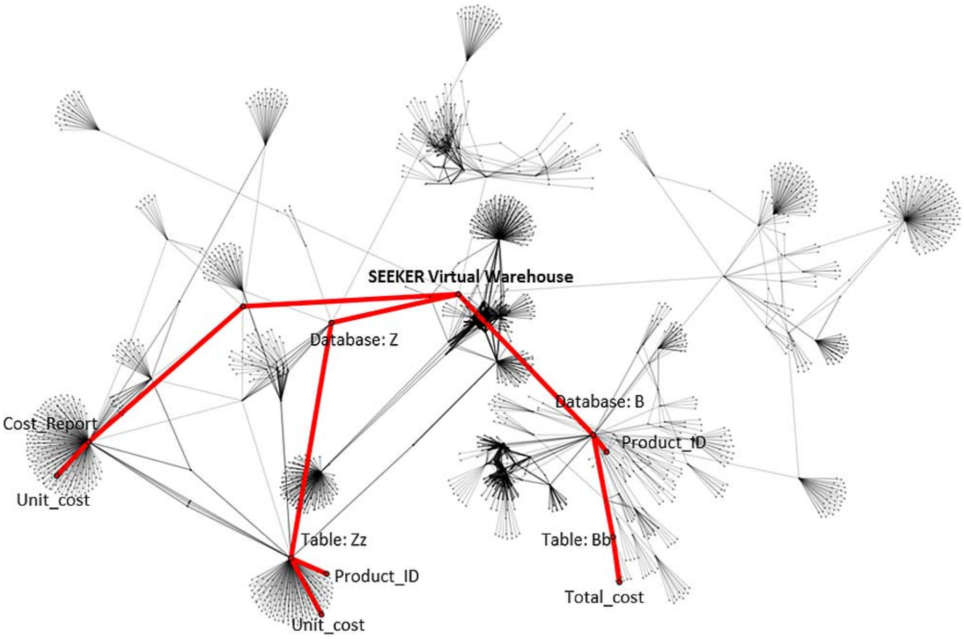

Fig. 3. Example of how SEEKER helps construct queries by showing the access path and join path while querying across data-sources. (Colors are visible in the online version of the article; http://dx.doi.org/10.3233/ISU-130712.)

The graphical structure shows clusters of data elements organized into tables, as well as tables organized into a database. The red circular dots indicate the results of the metadata search.

Once the analyst has pruned through the list of potential fields, SEEKER also can provide the access path to the data to be analyzed. For example, Fig. 3 shows SEEKER automatically identifying the cross-walk keys and the access path to when an analyst expressed interest in bringing two cost elements together. SEEKER automatically finds the primary key in the system ("Product_ID" in the Fig. 3 example), the table and databases to which the analyst needs access, and the table-joins one has to make before the two data elements can be brought together and analyzed.

In this example, the analyst is searching for "unit_cost" and trying to associate the "unit_cost" field with the "total_cost" field for every product. SEEKER finds both of these elements in two different databases hosted in two different assets. The red line marked on the graph is automatically generated. An analyst would interpret that the "Product_ID" field is the key for associating the "unit_cost" and "total_cost" fields across the databases. The analysts appreciate being able to do this without the knowledge of the intricacies of data model.

4.3. *Virtual schema builder*: *Metadata and data level matching*, *schema-level hypothesis generation*

So far we have shown SEEKER fetching and storing a database's table and column specifications (metadata) from most commercial and open-source data stores and allowing a user to perform a full or partial text match on column names to see if a particular column or related columns are found in tables. We enhanced this basic metadata search capability of SEEKER by implementing algorithms that can automate this process. We have leveraged concepts from several seminal publications [2,3,7,11,15] to implement the metadata and data level matching module. Once a database is loaded into the warehouse, SEEKER runs a set of packaged queries through all the tables and extracts. The queries sift through each column to find the number of distinct values collected per column, the percentage of the data that is distinct [11], data type (number, date, etc.) structure (10-character string, mm-dd-yyyy) etc. The query also profiles a histogram of distributions of a random sample of data for each column [2]. These values are used as features to match columns from two different schemas to create a virtual combined schema [15]. This crawling module in SEEKER addresses the need that analysts expressed to generate schema-level hypotheses to make connections for previously unseen data sources. Figure 4 presents one such discovery result. In the figure, an analyst is trying to understand and integrate three databases labeled as "Database:P", "Database:N" and "Database:T". The analyst can pick a few elements in each of these databases if he or she has sufficient information already, or the analyst can allow SEEKER to explore the dataset automatically.

Figure 4 shows how SEEKER generates link hypotheses between potential field elements based on the match by the field name, data type and the histogram of a random sample of data compared against data sources. This capability is very similar to the methodology described in Miller [15].

SEEKER extracts the elements that are potentially linkable across databases and creates those links for the analyst. In the example illustrated, the dark, dashed lines are hypothesis-generated by SEEKER. SEEKER is able to hypothesize an ID–ID link type (e.g., "Product_ID" fields) from one database to another and attribute–attribute link types (e.g., fields like zip codes and phone numbers). SEEKER can also link based on column naming conventions and a fuzzy string matching of column names. This

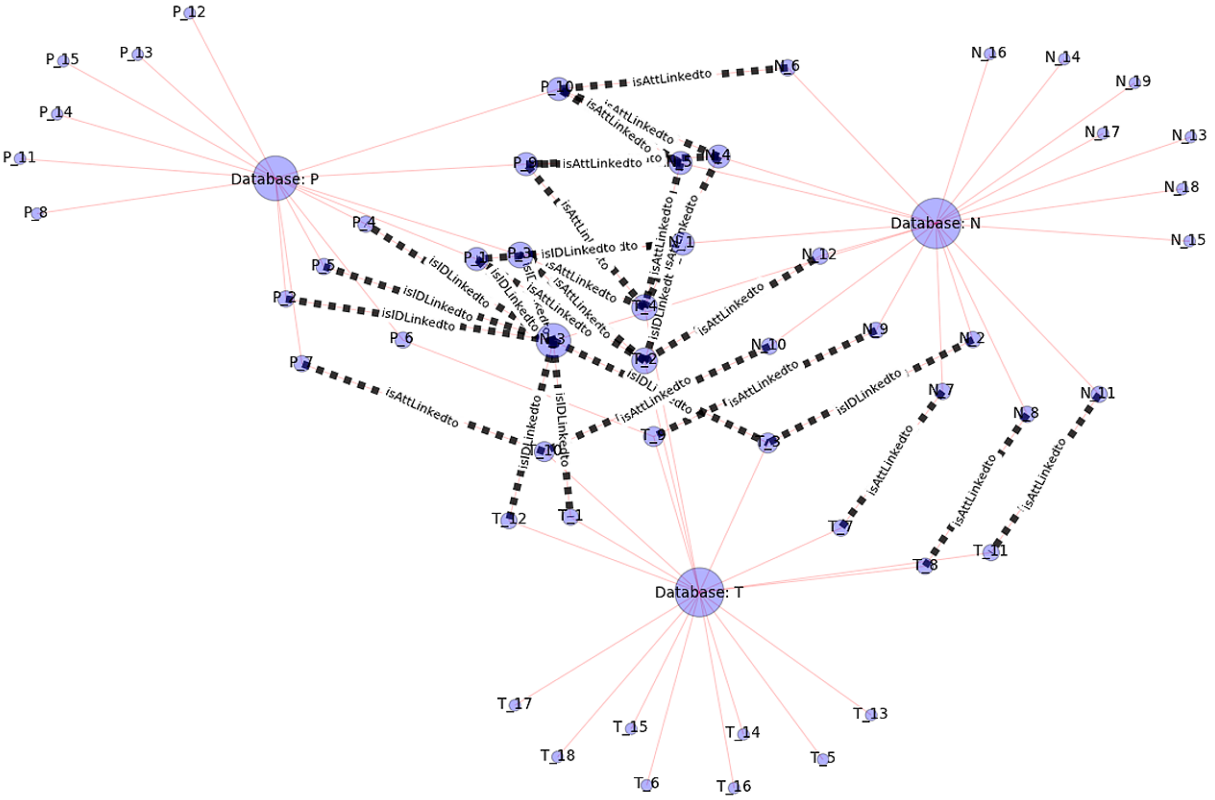

Fig. 4. SEEKER generates link hypotheses between potential field elements based on the match by the field name, data type and the histogram of a random sample of data compared against data sources. (Colors are visible in the online version of the article; http://dx.doi.org/10.3233/ISU-130712.)

schema-level hypothesis generation functionality within SEEKER enables the analyst to build a virtual schema based on metadata and data-level matches. The associations and the business logic behind those associations can be shared and archived for collaboration with fellow analysts.

5. Record of institutional memory

Figure 5 illustrates an example of how SEEKER acts as a record of institutional domain knowledge, showing screen shots of a database taken several months apart. Figure 5(a) is a visualization of the data schema when the database was loaded into the warehouse for the first time. Several analysts leveraged this data resource to conduct their analysis over time, and the schemas associated with the database transformed into Fig. 5(b). The schema changes, the analysis artifacts created, and the new calculations for reports were all recorded and tracked by SEEKER. This capability to visualize and interact with data from the day it was loaded and accessed and to trace how value is being extracted from the data resource is very valuable. It documents the provenance trail of reports and analysis artifacts, and it exposes data elements of value. For example, in Fig. 5(a) the structure of the graph reveals that there are three kinds of related processes captured in the database. The primary keys, foreign keys and crosswalk keys are revealed in this view. However, it is not obvious as to what the business value elements are. Figure 5(b), which captures the interactions and collaborations of several analysts, shows which other elements are of value. A new hire or an analyst new to this data resource can quickly scan this snapshot to understand the key elements used in previous analysis. This approach saves computational time, as the analyst would have otherwise spent several hours to learn about the dataset. The analyst's inconvenience is also reduced, and productivity is increased. We also observed that analysts knew whom to contact to gain access to an analytical artifact or business logic by browsing through metadata associated with the artifacts and then beginning to collaborate on the analysis artifacts. Figure 5(a) shows the structure of a dataset that was just loaded into the warehouse. Figure 5(b) illustrates the evolved schema several months after analysts were allowed to collaborate.

6. Summary, conclusions and future directions

With today's rapidly changing business environment, data warehouses constantly have to adapt to changes by accommodating new data structures and modifications to old ones and by merging different systems' data. In addition, the average time that an employee remains in one job is decreasing. Together, these factors can create a gap in detailed knowledge of data sets and their relationships. The SEEKER software suite introduced in this paper can provide an analyst with a powerful toolset for beginning to explore and visualize different data sets together in a single view. The added capabilities to help search for potential linkages in data sets provides an important first step in putting data from different repositories together. The fuzzy matching capability serves as an aide in searching through schema for related information even if it is not directly linkable. In addition, the graphic display of the data linkages can provide a more intuitive look at relationships that can be more immediately grasped by a human than some of the more traditional database layout tools.

This tool has already proven to be a powerful aide for the research we have performed as we put data together from disparate data sets for which we had no prior knowledge. Even for well-documented systems, this tool can provide a powerful visualization and learning tool for personnel new to the system.

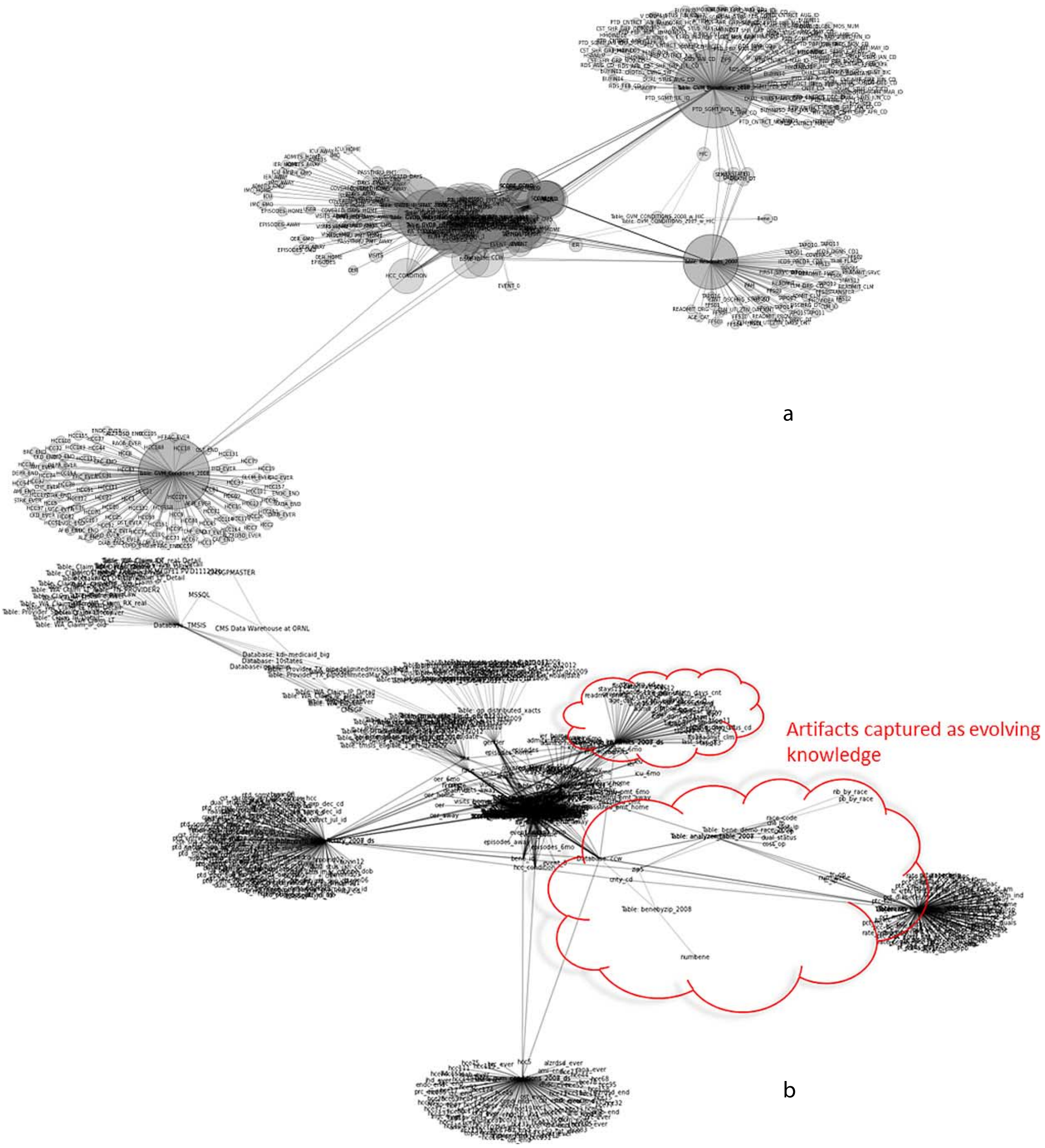
a
MSSQL
Database: TMSIS
CMS Data Warehouse at ORNL
Database: kdi-medicaid_big
Database: 10states
Artifacts captured as evolving knowledge
Table: analyzee table 2007
Table: bene_demo_race_2008
Database: CCW
Table: benebyzip_2008
numbene
b

Fig. 5. SEEKER as an evolving knowledge recorder of enterprise analytics. (Colors are visible in the online version of the article; http://dx.doi.org/10.3233/ISU-130712.)

In addition, we plan to include emerging collaboration tools for analysts to enable conversations regarding the domain hints towards analyzing and querying the data more effectively. We will be leveraging emerging social-media features around datasets, business logic and enterprise data workflows, allowing data analysts, data scientists and other stakeholders to collaboratively tackle their 'Big Data'. Data assets will have discussion forums and blogging capability for annotating and exchanging notes making it both easy to share, index and retrieve in a data-asset centric workspace. The digital archive of exchanges of shared insights on the metadata will enrich the transferrable institutional knowledge over the existing prototype described in the paper.

In the future, additional capability will be incorporated into the toolset. Some of the plans include (1) a data analysis component to display distribution of data values for an element, (2) a field linking based on data matching capability, and (3) a supervised interface to allow analysts to make their own linkages between data sources and to further prune and validate SEEKER-generated hypotheses. As new capabilities are incorporated into this tool, it has the potential to become a core part of any analyst's or data mining expert's software set. In addition, we envision our tool living as a semantic interface layer in multi-tenant massive data centers and data warehouses, with several end-users and analysts attempting to extract actionable intelligence from 'Big Data'. We are also quantifying the value of SEEKER to connect and map massive amounts of institutional/enterprise data across disparate silos to enable analysis for better decision making. By providing the user with this powerful tool for exploring and linking disparate data sets, the time now required to integrate new data constructs for analysis will be greatly reduced. With these additions, an analyst will be able to share insights with team members, who can then verify the insight, make comments, brainstorm about overlooked possibilities, and generate new questions of their own enabling analysts to know what data to put together for strategic business value.

Acknowledgements

This paper has been co-authored by employees of UT-Battelle, LLC, under contract DE-AC05-00OR22725 with the U.S. Department of Energy. Accordingly, the United States Government retains and the publisher, by accepting the article for publication, acknowledges that the United States Government retains a non-exclusive, paid-up, irrevocable, world-wide license to publish or reproduce the published form of this manuscript, or allow others to do so, for United States Government purposes.

References

[1] J. Aaker and V. Chang, Obama and the power of social media and technology, *The European Business Review* (2010), 17–21, available at: http://www.europeanbusinessreview.com/?p=1627.

[2] P. Andritsos, R.J. Miller and P. Tsaparas, Information-theoretic tools for mining database structure from large data sets, in: *Proceedings of the 2004 ACM SIGMOD International Conference on Management of Data*, ACM, 2004.

[3] R. Axelrod, Schema theory: An information processing model of perception and cognition, *The American Political Science Review* **67**(4) (1973), 1248–1266.

[4] M.T. Capizzi and R. Ferguson, Loyalty trends for the twenty-first century, *Journal of Consumer Marketing* **22**(2) (2005), 72–80.

[5] R. Copeland, *Essential SQLAlchemy*, O'Reilly Media, 2008.

[6] J.B. Davis, *Statistics Using SAS Enterprise Guide*, SAS Inst., 2006.

[7] B. Dove and J.A. Handler, Automatic generation of virtual database schemas, U.S. Patent Application 12/715, 409.

[8] G. Hawkins, Will big data kill all but the biggest retailers?, *Harvard Business Review* (2012), available at: http://blogs.hbr.org/2012/09/will-big-data-kill-all-but-the/.

[9] A.J.G. Hey, *The Fourth Paradigm: Data-Intensive Scientific Discovery*, S. Tansley and K.M. Tolle, eds, Microsoft Research, Redmond, WA, 2009.

[10] L. Huston and N. Sakkab, Connect and develop, *Harvard Business Review* **84**(3) (2006), 58–66.

[11] J. Kang and J.F. Naughton, On schema matching with opaque column names and data values, in: *Proceedings of the 2003 ACM SIGMOD International Conference on Management of Data (SIGMOD'03)*, ACM, New York, 2003, pp. 205–216.

[12] P.F. Lazarsfeld, B. Berelson and H. Gaudet, *The People's Choice: How the Voter Makes up His Mind in a Presidential Campaign*, P.F. Lazarsfeld et al., eds, Columbia Univ. Press, 1965.
[13] N.C. Livingstone, *The War Against Terrorism*, Lexington Books, Lexington, MA, 1982.
[14] J. Manyika, M. Chui, B. Brown, J. Bughin, R. Dobbs, C. Roxburgh and A. Byers, *Big Data: The Next Frontier for Innovation, Competition, and Productivity*, McKinsey Global Institute, 2011.
[15] R.J. Miller, L.M. Haas and M.A. Hernández, Schema mapping as query discovery, in: *Proceedings of the 26th International Conference on Very Large Data Bases*, 2000.
[16] S.J. Palmisano, *A Smarter Planet: The Next Leadership Agenda*, IBM, November 6, 2008.
[17] B.R. Posen, The struggle against terrorism: grand strategy, strategy, and tactics, *International Security* **26**(3) (2002), 39–55.
[18] J. Spohrer and P.P. Maglio, Service science: Toward a smarter planet, in: *Introduction to Service Engineering*, G. Salvendy and W. Karwowski, eds, Wiley, Hoboken, NJ, USA, 2010.