Integration of Big Data
Misconceptions, Problems, and Needed Capabilities

J. Johnson, EOIR Technologies
J. Folger, EOIR Technologies
T. Stevens, EOIR Technologies
T. Malyuta, New York City College of Technology

Abstract. The authors have been working on the problems of data integration for a number of years, in particular, integration of what is called Big Data – data defined by Gartner's famous three V's: volume, velocity and variety. Based on our experience and interactions with both customers and vendors, we discuss here the common misconceptions about Big Data integration and cloud technologies, analyze problems that need to be addressed,[1] and suggest capabilities of the solutions. The solution to which the authors contributed is described in a number of publications [1, 2, 5, 6].

Cloud Technologies and Integration Capabilities

Assuming that cloud storage technologies by themselves provide integration solutions is probably the main misconception. In general, no technology alone provides a solution to a problem; a technology can only be a part of the solution – a platform for the solution's implementation.

Yes, many cloud stores can accommodate data of different structures (they are semi-structured stores), and we can push into such a store data diverse in structure and semantics from different systems. But co-location of data is not data integration. Figure 1 schematically illustrates data about persons from different sources in the Big Table of Google – the storage solution that supports many Google applications and that gave birth to a number of similar cloud stores. Depending on the data integration requirements that are discussed below, representing data and metadata or data provenance will definitely require specific data structures and support of data management policies that go beyond what is offered by a specific storage technology.

Row Key	Column Data		
1000001	First Name: John	Last Name: Jones	DOB: 01/01/81
XYZ123	Name: John Jones	Age: 35	
1000002	Full Nmae: J. Jones	Country of Birth: Canada	

Figure 1. Data in the Google Big Table.

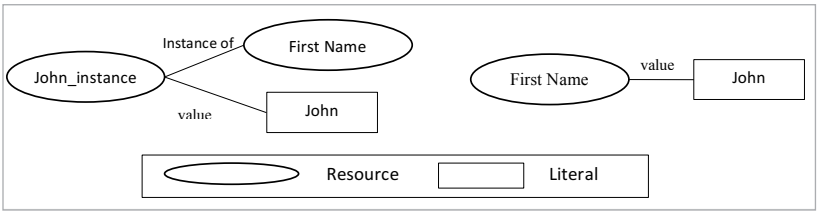

Figure 2. RDF data representation.

Most confusion is caused by RDF and OWL technologies – RDF-based languages support linkage of resources, and it is assumed that once we represent data in RDF/OWL, we immediately get an integrated store. Often, when building an integrated store, we are asked why data shouldn't just be represented in RDF/OWL. However, particular data (and metadata or provenance) can be represented in RDF/OWL in a number of ways, and in order to correctly interpret data from different sources, we need an approach to help us represent and manage all data in a particular way. In other words, we need an integration solution. Figure 2 schematically illustrates different ways of representing a fragment of data from Figure 1 in RDF; note that structural and semantic diversity of data in Figure 1 will result in structural and semantic diversity in RDF representations.

As the preceeding figures show, these technologies do not provide structural homogeneity, nor do they provide semantic alignment of diverse source data. In addition to having a unified structural representation, we also need a solution to resolve semantic heterogeneity.

Moreover, an integrated solution should allow users and applications to perform traditional data management tasks. Depending on requirements, these data management tasks can be simple (e.g., retrieve specific data from diverse data collections) or more complex (e.g., perform different data manipulations with tracking provenance of data and data changes).

What is Data Integration?

In [3] we defined integration as the alignment or harmonization of multiple heterogeneous data resources in such a way that search and analysis procedures can be applied to their combined content as if they formed a single resource. For Big Data, it is also important that diverse structures and semantics of sources are not changed unless desired, because: 1) considering the volume and velocity factors, the integration process must be agile and require minimal human involvement and 2) because of the variety factor, in many cases we do not know the content of a source to be integrated or do not have time to analyze it.

What Can Co-Location Provide?

Clearly, as simple illustrations in the figures above show, co-location does not provide the capabilities we require from an integrated store. Therefore, we do not consider a store that just accommodates a number of sources without providing data and semantics management capabilities across these sources to be an integrated store. As we stated earlier, co-location is not integration. Consider the simplest possible scenario when you only need to query co-located data:

- We cannot expect it to be more than a Google-type search.
- Without special support (which, again, is provided by the integrated solution), we do not know how to formulate the request for nor how to handle the results of searches on structurally and semantically diverse data. As an example, executing a cross-source search quickly breaks down without some level of semantic alignment of sources, resulting in missed or incorrect interpretation of the results.
- For anything more than that, we will need additional capabilities not provided by the cloud storage technology (e.g., to track

data provenance, to align diverse semantics of source data, or to support required data management policies).

Depending on requirements, we need to build an integration solution where a particular chosen cloud store will be one of the technological components.

Data Integration Problems

Data integration requirements can be very different in terms of the following:

- Variety of integrated data domains and, respectively, variety of data semantics: from one or few to an unlimited number. In some cases, there is a need to integrate data from different sources about one domain (for example, data about potential customers); in other cases, data to be integrated can be from different domains (for example, intelligence data).
- Types of data manipulations on the integrated contents. In some cases, users need to analyze integrated data – data that is not modified and is not necessarily kept in the store (which requires some data management). For example, a company might need to analyze the latest competitors' sales data. In other cases, data is collected and managed; this is often the case in the intelligence community. In the case of the latter, data management requirements can be quite different as well.
- Traditional issues of the integrated store: performance, scalability, security, consistency, concurrent access, the need to normalize data (e.g., country codes, date formats), and so on.
- Issues specific to Big Data: a priori unknown structure and semantics of data, the need to quickly and properly accommodate data with minimal human effort to support necessary use cases and workflows.

Dimensions of the Solution

The solution addressing these requirements in their totality needs to be based upon the following:

- An appropriate data representation approach that allows diverse source data structures to be adequately represented in the integrated store.
- A storage platform suitable for implementing this data representation approach and meeting the performance, scalability and other requirements.
- A method of dealing with diverse semantics of sources.
- Data and semantics management policies defined by and leveraging the previous three components.

Representation Approach and Implementation Platform

The most important (and often ignored) issue is that we need to start the discussion of a solution not with the storage structures of a particular technology (e.g., triples of RDF/OWL technologies, documents/attributes of MongoDB or relations of relational databases), but with the way we want to use them for our chosen data representation. In other words, we need a conceptual approach to data (metadata and provenance) representation – an approach that meets the integration requirements and is agnostic of a particular implementation. Only with such an approach can we consider a technology that will allow for its efficient implementation to meet such requirements as performance, scalability and security. Unfortunately, this well-established methodology to building data stores (usually based on relational databases) is neglected when using cloud storage technologies.

Semantic Alignment

We support the school of thought that a proper semantic alignment of different data models should be based on a collection of scientifically sound ontologies [3, 4, 5]. These ontologies can be applied at "arm's length" (without changes to the source data and semantics). This approach, while it does not allow for full semantic alignment, is applicable to most situations and is relatively easy and quick. To illustrate, we revisit the sources described in Figure 1 and introduce the term "PersonName" in the alignment ontology. The subsequent assertion that 'Name' and 'Full Name' are the same as "PersonName" allows for cross model searches within the integrated store using the aligned term. Additionally, if desired, we can build a fully aligned ontology-based representation of the source data that requires transformations of data structures, semantics and values.

Main Features of Our Solution

In this paper we focused on common misconceptions and factors that must be considered when building an integration solution leveraging cloud-based technologies. Our intention was to deliberately avoid suggesting a particular data integration approach; however, we want to mention in the context of this discussion some important requirements and features that our solution is targeting (various aspects of the approach are described in detail in referenced materials). The integration approach is required to do the following:

- Accommodate data from any number of systems, each storing data about different domains in a variety of representations without changes to the semantics and structures unless desired.
- Support data accumulation, alignment of diverse semantics, and data cultivation (users and applications should be able to maintain and enrich accumulated data) in a concurrent and distributed manner.
- Maintain comprehensive metadata and provenance through the full life cycle of data.
- Support Google-style and semantic searches on data and provenance across represented data sources.
- Be scalable and performant given the volume and velocity of data.

The solution addressing all these requirements must:

- Be based on a conceptual (i.e., platform-agnostic) data and metadata representation model that has been implemented on different technologies: relational (MySQL) and Cloud (Rya).
- Support rapid data representation – the ability to represent structurally diverse data/metadata without loss or distortion with minimal human effort.
- Enable fast data utilization – the ability to query across sources in a Google-like manner immediately after data ingestion and get structured data as a result.

– Leverage arm's-length semantic alignment of data with the help of ontologies3 and, with it, the ability to meaningfully query across sources. Our representation and storage also accommodate the transformed ontological representation (when native semantics, structures, formats and values are transformed to those of the ontology) of the source data as well. Such store, however, will require time and human effort to build.
– Provide efficient and consistent comprehensive data management in a distributed and concurrent environment.

Conclusion

The paper does not describe a particular integrated solution but rather attempts to draw attention to some misconceptions about applying cloud technologies for data integration. We also tried to define with broad strokes a set of problems each integrated solution needs to address. Ignoring even one of these problems can make a solution unusable, so customers should analyze any suggested solution by considering the problems in their totality.

REFERENCES

1. Johnson, J., Folger, J., Stevens, T., Malyuta, T., Parent, K. & Brutus, J.R. (June 2016.) "Data Integration and Sharing Infrastructure on the Cloud." MSS Sensor and Information Fusion Symposium, Virginia.
2. Johnson, J., Folger, J., Stevens, T., Malyuta, T., Parent, K. (November 2014.) "The Intelligence Cloud and Distributed Management of Heterogeneous Entities." MSS Sensor and Information Fusion Symposium, Virginia.
3. Smith, B., Malyuta, T., Mandrick, W., Fu, C., Parent, K. & Patel, M. (2012.) "Horizontal Integration of Warfighter Intelligence Data: A Shared Semantic Resource for the Intelligence Community." STIDS Conference. http://stids.c4i.gmu.edu/papers/STIDSPapers/STIDS2012_T14_SmithEtAl_HorizontalIntegrationOfWarfighterIntel.pdf.
4. Smith, B., Malyuta T., Salmen, D., Mandrick, W., Parent, K., Bardhan, S. & Johnson, J. (2012.) "Ontology for the Intelligence Analyst." Crosstalk: The Journal of Defense Software Engineering. http://www.academia.edu/2824160/Ontology_for_the_intelligence_analyst.
5. Salmen, D., Malyuta, T., Hansen, A., Cronen, S. & Smith, B. (2011.) "Integration of Intelligence Data through Semantic Enhancement." STIDS Conference. http://stids.c4i.gmu.edu/STIDS2011/papers/STIDS2011_CR_T1_SalmenEtAl.pdf.
6. Hansen, A., Salmen, D., Malyuta, T. & Antunes, N. (July 26–29, 2010.) "An Evolving Integrated DataSpace on the Cloud." MSS Sensor and Information Fusion Symposium, Las Vegas, Nevada.

NOTES

1. As we focus here on integration of data with a variety of structures and semantics that are not known a priori, we exclude from this discussion the integration solutions like traditional data warehouses where a priori known source data are transformed into specific structures, semantics, and formats that are beneficial for particular types of analytics.
2. We rely on a collection of so-called "Common Core Ontologies," based on the Basic Formal Ontology (BFO) as the upper-level ontology.

ABOUT THE AUTHORS

Mr. Jamie Johnson (corresponding author) is a software developer at EOIR Technologies. He has worked with the intelligence community for the past 13 years as a Department of Defense civilian employee and as a civilian contractor. Most recently he has worked on cloud-scale search and indexing technologies. He received a master's degree in computer engineering from Stevens Institute of Technology and a bachelor's degree in computer engineering from Rutgers University.
jjohnson@eoir.com
732-223-7413

Mr. Jacob Folger is a software developer at EOIR Technologies. He has worked as a civilian contractor for the Mission Command and intelligence communities for a combined five years. He is currently participating in the development of the Army's next-generation mission planning software, and has received his master's and bachelor's degrees in computer science from Monmouth University.
jfolger@eoir.com
732-223-7413

Mr. Trevor Stevens is a software developer at EOIR Technologies. He has spent the last five years supporting various intelligence community projects. His most recent work has focused around cloud scale search and indexing. He received his bachelor's in computer science from Rowan University.
tstevens@eoir.com
732-223-7413

Dr. Tatiana Malyuta is an associate professor of the New York College of Technology of CUNY. She is a subject matter expert in data design and data integration and has been working on data integration solutions for the Intelligence Community for the last 10 years. She received a master's degree in applied mathematics and a Ph.D. in computer science from the State Polytechnic University in Lviv, Ukraine.
tmalyuta@thedatascience.org
201-248-4332