Chapter 2
Dataspace Management for Large Data Sets

Marko Niinimaki and Peter Thanisch

2.1 Introduction

Ever since researchers started talking about "Big Data," they have mentioned troubles related to it, alongside of its benefits [1]. These troubles often arise from the sheer volume of data but can also involve difficulties of processing, integrating, and analyzing the data. Indeed, an insightful definition of "Big Data" states that a data set can be called Big Data if it is formidable to perform capture, curation, analysis, and visualization on it using the current technologies [2].

However, these troubles sound familiar to computer scientists who have been exposed to statistical databases. Despite progress in data integration [3], use of an integrated business intelligence platform often requires a nontrivial process of identifying data sources, deciphering the meaning of their data, harmonizing the data, and then loading it into a system that can be used to analyze it. This process is known as ETL (Extract-Load-Transform) [4], and in business intelligence the resulting data often ends up in data warehouse servers and, when needed by an analyst, in online analytical processing (OLAP) cubes [5].

But in many organizations, ETL skills are in short supply, and incorporating a new ETL process into a production environment is a lengthy and arduous process. So what happens to the data sources that we find potentially interesting, yet we are unable to include in our current ETL processes? In the mid-2000s, the concept of dataspaces was introduced to describe a situation where there is "some identifiable scope and control across the data and underlying systems, and hence one can

M. Niinimaki (✉)
Department of Business and Technology, Webster University Thailand, Bangkok, Thailand
e-mail: niinimakim@webster.ac.th

P. Thanisch
School of Information Sciences, University of Tampere, Tampere, Finland
e-mail: peter.thanisch@sis.uta.fi

P. Vasant et al. (eds.), *Innovative Computing Trends and Applications*,
EAI/Springer Innovations in Communication and Computing,
https://doi.org/10.1007/978-3-030-03898-4_2

identify a space of data, which, if managed in a principled way, will offer significant benefits to the organization" [6]. Moreover, a dataspace support platform (DSSP) provides services for managing such collections of data. Specifically, a DSSP should provide services helping to identify sources in a dataspace and interrelate them, offering basic query mechanisms over them, including the ability to introspect about their contents [3].

In this paper, we present a DSSP specifically designed for data integration using Microsoft Excel data models [7]. The design of the DSSP is workflow-oriented: the user uploads the data sets in the DSSP and describes the semantics of the data by filling a form provided by the DSSP. This metadata can be utilized when the data is imported into the user's data model. We specifically address the problem known as *summarizability* [8, 9], by including in the DSSP metadata information needed to determine summarizability in OLAP. Moreover, we describe a process for utilizing the metadata when designing Excel data models using the DAX (Data Analysis Expression) language [10] and Excel's graphical data model designer.

Our aim is to build and support a dataspace where the data is used within an organization (like a university). This approach corresponds to Halevy et al.'s use case of a dataspace for scientific data management [3]. In other related research, Dittrich [11] presents iMeMex, a comprehensive personal data search and management system that unfortunately is no more functional. Mirza et al. discuss the practicality of a dataspace system from a user's point of view, mentioning specifically the challenge of combining online and local data sources [12]. Moilanen et al. [13] demonstrate a harmonization platform for XML, but their system is not very suitable for large data sets. Niinimaki and Niemi [14] demonstrate an ETL process with RDF/XML data, but do not discuss how the RDF ontologies and instances are built and populated. Contrary to these approaches, we emphasize a dataspace as a repository of large data sets, the design of data models, and verify the data model using the dataspace. The design of our platform supports the workflow described above.

In our use case, we use real data with magnitude of tens of millions of observations, demonstrate real analysis cases using the data model, and evaluate the performance of our system.

The rest of the paper is organized as follows. In Sect. 2.2, we describe the data that we plan to analyze. This serves as a generalizable basis for Sect. 2.3 that describes the design of our DSSP, and in Sect. 2.4 we study the system's performance. Finally, Sect. 2.5 contains a summary, research questions for future work, and conclusions.

2.2 Data and Summarizability

Our sample analysis concerns trade of electronic goods among various countries, and the data for the analysis comes from many sources. The most important data sets in the analysis are import-export data from UN COMTRADE (international

Export-of-commodities-2000s: Year, exporting country (reporter), importing country (partner), type of commodity (prod), value of trade in USD. Field prodgroupcode is derived from prod.
Prodgroup: HS 2-digit product group, product group's English name.
Cia_factbook_country_gdp: Country (ISO3 code), GDP per capita purchasing parity adjusted in 2004 USD.
Countrycodes: ISO3 codes and English names for countries and autonomous areas.

Fig. 2.1 Data set descriptions

trade statistics database),[1] HS (Harmonized System) trade group descriptions from the same source, and per-country GDP per capita figures from *CIA World Factbook*.[2] Moreover, we have written a program that translates country names used in *CIA World Factbook* and Wikipedia in ISO3 county codes and a program to download tables from Wikipedia in the CSV (comma-separated values) format.

All the data sets deal explicitly with countries on a yearly basis; thus a potential integration of the data is not especially hard. Yet, we need to ensure that roles like exporter and importer are sufficiently well expressed. The main data sets, as extracted, are shown in Fig. 2.1. The export-of-commodities set covers years 2000–2015 and has about 94 million lines. As a demonstration of data analysis after data integration, we have selected the following scenario: A company interested in producing electronics goods is seeking to expand outside high-income countries. An analyst is tasked to find out which low- and middle-income countries had largest increases in electronics exports in recent years (2000–2015).

In a multidimensional data model, there is a set of numeric measures that are the objects of analysis. Each of the numeric measures depends on a set of dimensions, which provide the context for the measure [5]. An implementation of this model is called a dimensional database or a cube. Figure 2.2 shows the data and relations prepared for a cube. Dimensions in a cube can be hierarchical, as product—product group. The entire hierarchy is called a dimension and product, and product groups are called its levels.

Within Excel, columns of data tables can be connected with a relation if they are compatible. Here we use simple technical criteria for compatibility of two columns: at least one of the columns must have unique values and both columns must share some values. The connected columns in Fig. 2.2 in export-of-commodities-2000s are reporter and partner to countrycode's ISO3 code and to cia_factbook's country, productgroupcode to prodgroup's prodgroupcode.

Summarizability can be informally defined as "correctness of aggregate data with respect to individual observations." In practice, this means that when a user queries a cube, the query should be inspected in such a way that the query's result cannot violate the conditions of summarizability. Niemi et al. [9] formulate the conditions as follows: (1) the aggregation operation (usually sum, mean, or count) is

[1] http://comtrade.un.org.

[2] https://www.cia.gov/library/publications/the-world-factbook/.

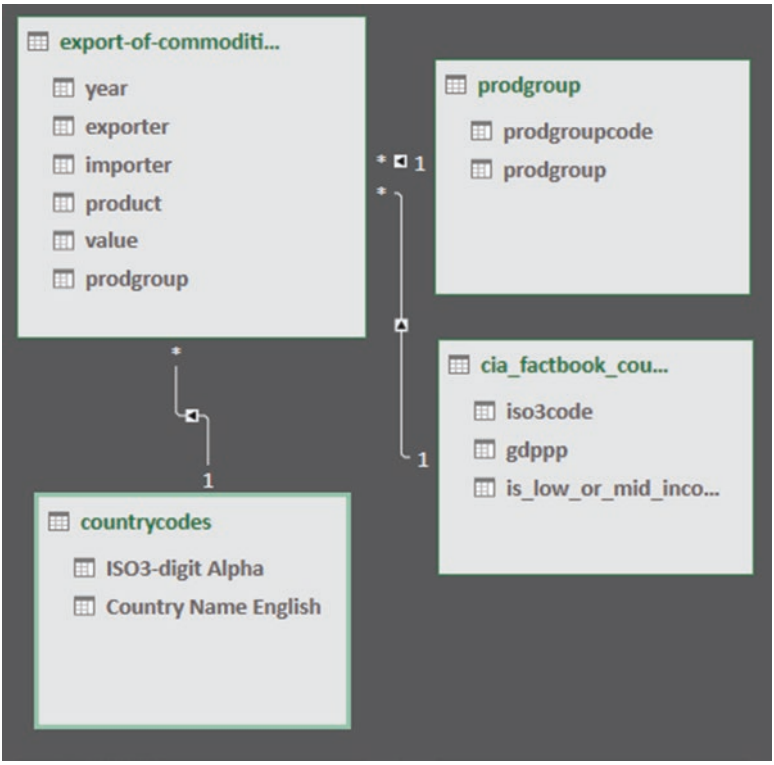

Fig. 2.2 Relations in the model

appropriate for the measure, and (2) the measure is appropriate for the aggregation levels in the cube's dimensions. We are specifically interested in:

- The statistical scales of measure variables (see [15]), nominal, ordinal, interval, and ratio scales, since they affect which aggregation operation can be applied to the data.
- The "eventness" type of measure variable. These we classify [9] as (1) tally measures that are intrinsic information about a specific event like quantity sold in sales data, (2) reckoning measures like inventory levels, and (3) snapshot measures that are indirect measurement based on data at hand like currency exchange rates.

With our dataspace system, we help the analyst detect problems with summarizability by collecting characteristics of data like the statistical scales and eventness of measures. Our ultimate goal is to construct a system that automatically detects summarizability problems in pivot tables created by data analysts.[3] The design of

[3] Niemi et al. [9] present an algorithm that discovers the correctness of additivity in OLAP queries expressed in an MDX-like query language. However, in addition to the statistical scale and "eventness," the algorithm uses "measure depends on dimension" information: for instance, a currency measure depends on country dimension. This kind of information is harder to provide in a dataspace application.

system at hand is less ambitious: the dataspace system lets us identify potential dimensions and measures and dimensions that are compatible. The compatibility of data within Excel is then checked by a macro that gets the information from the dataspace system. Details of the design of the dataspace system and its integration with Excel are given in the next section.

2.3 System Design

The goal of the design is to provide a catalogue of data sets, combined with their metadata. The users will upload data sets of their interest into the catalogue. After each upload the catalogue software lets the user specify a description of the data set. The data set is expected to have recognizable fields that represent potential measures or (levels of) dimensions. For each measure the user will record its statistical scale, "eventness," and unit of measurement. For each dimension, the user will record a description and a set of fields in other data sets such that the dimension is compatible with them. The compatibility of fields is stored in the XML format to a file (compatibility.xml) accessible by the HTTP protocol. An interface for entering the metadata (and field compatibility data) after a file upload is shown in Fig. 2.3.

The uploaded data sets with their metadata are presented using a web interface shown in Fig. 2.4.

The dataspace management application is a Python program that utilizes the Flask web programming framework [16].

In our case study, a company is interested in expanding their business as described in the introduction: among low- and medium-income countries, which ones have grown their exports in electronics most during 2000–2015. The following data sets are thus selected using the interface shown in Fig. 2.4: export-of-commodities-2000s, prodgroup, cia_factbook_countries_by_gdppp, and countrynames_iso3. Assisted by the user interface (Fig. 2.4) or by using the DAX language, the data analyst imports the data in Microsoft Excel's data model [17] and creates the relations within the data, as was shown in Fig. 2.2.

We can now verify some basic aspects of the data models using our verifier macro. We assume that the user has imported the data sets from the dataspace system as Excel tables, either as tables within Excel sheets or in the data model. The

Fig. 2.3 Metadata editor

Set name	Lines	Description	Fields
countrynames-iso3.csv	287	Country names with ISO3 counterparts	Country Name English:English name ISO3-digit Alpha:ISO3 code
cia_factbook_countries_by_gdppp.csv	231	Countries by GDP purch parity adjusted in 2004 USD	gdppp:GDP purch parity adjusted in 2004 USD -measure:USD -scale:ratio iso3code:ISO 3 letter code of the country
export-of-commodities-2000s.csv	94904739	Export of commodities COMTRADE 2000-1015	partner:importer prodcode:product code reporter:exporter value:value of trade -eventness:tally -measure:USD -scale:ratio year:year of trade

Fig. 2.4 Some uploaded data sets with metadata

Fig. 2.5 Excel macro compatibility warning

relations within the data can be expressed either using DAX-"related" expressions or using a graphical tool.[4] The verifier macro has two data sources:

- Compatibility of dimension fields as recorded in the compatibility.xml file (see above)
- The relations of tables accessible by the Excel VBA API as "primary table name/primary column name" and "foreign table name/foreign column name"

By comparing the data set and column names in each of the sources, the macro prints if the fields are compatible.[5] If the macro discovers incompatibilities, it displays a message as in Fig. 2.5.

In our analysis case, a few adjustments to the data need to be done before the findings can be presented. First, a dummy field "is_low_or_mid_income" is introduced to the cia_factbook_countries_by_gdppp data, since it will be used as a

[4] In its simple form, DAX-related expressions state that a value in $table_1/column_a$ has a counterpart in $table_2/column_b$ like a country's ISO3 code has a country name counterpart. The graphical tool lets the users connect tables by their columns. For instance, trade data's exporter is given as an ISO3 code that we connect with country data's ISO3 column.

[5] In more detail: the macro iterates over data sources d and each data source's each field df. The names of the data sources and the fields are assumed to be the same as in the dataspace system. If a data source field d_if_j is connected to another d_kf_l, the macro tries to find the following entry in the compatible. xml file: <compatible><file>d_i</file><field>f_j</field><file>d_k</file><field>f_l</field></compatible>.

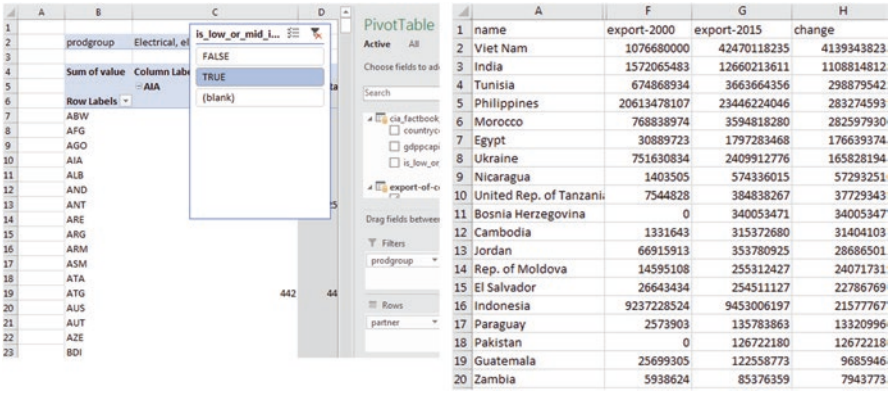

Fig. 2.6 Pivot table (left) and analysis based on the data model (right)

filter in the pivot table. Second, a product group code is added in the export-of-commodities-2000s data. The product group is simply the 2 first digits of the trade product HS code, 85 for electrical, electronic equipment, whereas the "prod" field in the export-of-commodities-2000s data is 6-digit, for instance, 850,890 for "parts, hand tools with self-contained electric motor."[6] Figure 2.6 shows the design of the pivot table such that we apply filters "exporter only low- and middle-income countries" and "product group only electrical, electronic equipment." Moreover, the figure demonstrates the result of the analysis.

2.4 Performance Measurements

In this section we discuss the time consumed in loading the data and the overhead imposed by the dataspace management system. The tests were performed on a computer with Intel Core i5 dual-core 2.4 GHz CPU, 4 GB RAM, and a normal 500 GB 7200 rpm disk. The operating system was 64-bit Windows 7 Enterprise with Microsoft Excel 2016 MSO 64-bit.

Performance of the Dataspace Application

We have tested the dataspace application with multiple random generated files in addition to real files discussed in Sect. 2.3. To eliminate network delays, the tests were performed on the same computer where the application was running. The upload speed is ca 5 MB/s and the download speed ca 38 MB/s. The size of the largest file, export-of-commodities-2000s, was 2.3 GB. Its upload time was 7 min 37 s and download time 1 min 3 s.

[6] https://comtrade.un.org/db/mr/rfCommoditiesList.aspx.

Data Loading Performance

Since the export of commodities 2000–2015 data set is very large, it had a dominant role when loading data into Excel's data model. We have used both this data set and artificially constructed sets to determine the performance. Overall, with our hardware configuration, Excel reads data from CVS files into its data model at ca. 1 million lines per minute, and the performance remains linear. Loading the export-of-commodities-2000s data set took 8 min 30 s, and generating the pivot table of Fig. 2.5 from the data took about 1 min.

2.5 Summary, Conclusions, and Future Work

In this paper we have presented a web-based dataspace management system for large data sets. Our principal aim is to help data analysts discover problems with data integration. This is done by storing the data sets together with their metadata. The metadata includes scale, "eventness," and unit for measures and a list of compatible fields for dimensions.

The web software was build using the Python Flask framework, available for Microsoft Windows and other operating systems. The software is freely available at https://sourceforge.net/projects/simple-dataspace-management.

Our future research will have three focus areas: summarizability, platform, and cloud integration.

We aim at providing full integrated support to detecting summarizability when the analyst creates an OLAP cube based on the data model. For this purpose we shall improve the "verifier" Excel macro so that it will be able to inspect all DAX language formulas from a pivot table. Since aggregation can be done using the DAX language, the macro will be able to compare the DAX aggregation formula with the metadata from the data management system and find out if the aggregation is correct in terms of summarizability.

The second focus is the dataspace platform itself. We shall expand the dataspace software to make it more user-friendly and capable of storing/inspecting data in various formats. Moreover, we are working on a feature that allows the user to test designs for OLAP cubes using the platform. Figure 2.7 demonstrates the feature: the user has selected the data sets and their fields, and the software analyzes if the combination will produce a good joined data set.

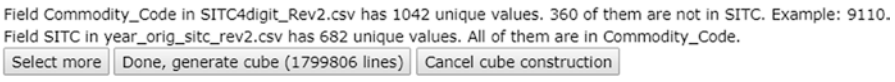

Fig. 2.7 Tools for analysis joining data

The third focus is obvious: when dealing with Big Data, cloud systems are its natural habitat (see [18]). We have started regrafting the dataspace manager as a Google Cloud [19] application, and the results have been promising.

Acknowledgments The authors wish to thank COMTRADE for the access to their export data and Dr. Leslie Klieb for comments.

References

1. McFedries, P.: The coming of data deluge. IEEE Spectrum. **48**, 19 (2011)
2. Chen, P., Zhang, C.-Y.: Data-intensive applications, challenges, techniques and technologies: a survey on big data. Inf. Sci. **275**, 314–347 (2014)
3. Halevy, A., Franklin, M., Maier, D.: Principles of dataspace systems. In: Proceedings of the Twenty-Fifth ACM SIGMOD-SIGACT-SIGART Symposium on Principles of Database Systems, 2006
4. Rahm, E., Do, H.H.: Data cleaning: problems and current approaches. IEEE Data Eng. Bull. **23**, 3–13 (2000)
5. Chaudhuri, S., Dayal, U., Vivek, N.: An overview of business intelligence technology. Comm. ACM. **54**(8), 88–98 (2011)
6. Franklin, M., Halevy, A., Maier, D.: From databases to dataspaces: a new abstraction for information management. ACM Sigmod Rec. **34**(4), 27–33 (2005)
7. Winston, W.: Microsoft Excel Data Analysis and Business Modeling, 5th edn, p. 864. Microsoft Press, Redmond (2016)
8. Lenz, H.-J., Shoshani, A.: Summarizability in OLAP and statistical data bases. In: Proceedings of the Ninth International Conference on Scientific and Statistical Database Management, 1997
9. Niemi, T., Niinimäki, M., Thanisch, P., Nummenmaa, J.: Detecting summarizability in OLAP. Data Knowl. Eng. **89**, 1–20 (2014)
10. Harinath, S., Pihlgren, R., Guang-Yeu Lee, D., Sirmon, J., Bruckner, R.R.: Professional Microsoft SQL Server 2012 Analysis Services with MDX and DAX. Wiley, Hoboken (2012)
11. Dittrich, J.-P.: iMeMex: a platform for personal dataspace management. In: Proceedings of Workshops of International ACM SIGIR Conference on Research and Development in Information Retrieval, 2006
12. Mirza, H.T., Chen, L., Chen, G.: Practicability of dataspace systems. Int. J. Digital Content Technol. Appl. **4**, 3 (2010)
13. Moilanen, K., Niemi, T., Näppilä, T., Kuru, M.: A visual XML dataspace approach for satisfying ad hoc information needs. J. Assoc. Inf. Sci. Technol. **66**(11), 2304–2320 (2015)
14. Niinimaki, M., Niemi, T.: An ETL process for OLAP using RDF/OWL ontologies. J. Data Semantics. **XIII**, 97–119 (2009)
15. Stevens, S.: On the theory of scales of measurement. Science. **103**(2684), 677–680 (1947)
16. Grinberg, M.: Flask Web Development: Developing Web Applications with Python. O'Reilly Media, Sebastopol (2014)
17. Winston, W.: Microsoft Excel Data Analysis and Business Modeling. Microsoft Press, Redmond (2016)
18. Hashem, I.A.T., Yaqoob, I., Anuar, N.B., Mokhtar, S., Gani, A., Khan, S.: The rise of "big data" on cloud computing: review and open research issues. Inf. Syst. **47**, 98–115 (2015)
19. Cusumano, M.: Cloud computing and SaaS as new computing platforms. Commun. ACM. **53**(4), 27–29 (2010)