World of Computer Science and Information Technology Journal (WCSIT)
ISSN: 2221-0741
Vol. 5, No. 10, 155-164, 2015

Data Quality Based Data Integration Approach

Mohamed Samir Abdel-Moneim
College of Computing and
Information Technology
Arab Academy for Science
Technology & Maritime Transport
Cairo, Egypt

Ali Hamed El-Bastawissy
Faculty of Computer Science
MSA University
Giza, Egypt

Mohamed Hamed Kholief
College of Computing and
Information Technology
Arab Academy for Science
Technology & Maritime Transport
Alexandria, Egypt

Abstract—Data integration systems (DIS) are systems where query answers are collected from a set of heterogeneous and autonomous data sources. Data integration systems can improve results by detecting the quality of the data sources and retrieve answers from the significant ones only. The quality measures of the data in the data sources not only help in determining the significant data sources for a given query but also help data integration systems produce results in a reasonable amount of time and with less errors. In this paper, we perform an experiment that shows a mechanism used to calculate and store a set of quality measures on data sources. The quality measures are, then, interactively used in selecting the most significant candidates of data sources to answer users' queries. The justification and evaluations are done using amalgam and THALIA benchmarks. We show that our approach dramatically improves query's answers.

Keywords-component data integration; quality measures; data sources; query answers; user preferences.

I. INTRODUCTION

Data Integration (DI) is the process of combining the data located at multiple locations, and allowing the user to view these data through a single unified view called global or mediated schema [1, 2]. The global schema is the interface where users submit their queries to a data integration system. The user no longer needs to know how to access the data sources, nor does he need to consider how to combine the results from different sources. The data requested by the user may be found at a single source, at many sources, or scattered across many sources.

Different architectures for data integration systems have been proposed, but broadly speaking, most systems fall between warehousing and virtual integration [3].

The quality of the data sources can dramatically change as data may be incomplete, inaccurate or out of date. In fact, the quality of the result depends mainly on two factors: the quality of the data at the data sources and the manipulation process that builds the resulting data from the data sources. Because the quality of the data sources can dramatically change, it is important to store some quality-related measures about the data sources to take it into consideration during query planning.

In our previous work [4], we presented an approach that is based on utilizing data quality (DQ) aspects in data integration systems in order to get satisfied query plans. Our approach is based on adding quality system components to be parts of any data integration system. Attribute values can be integrated from different data sources based on quality measures and user's preferences. We use quality measures to deliver query answers with satisfied quality. In this paper we perform an experiment that is based on that work [4]. The experiments were conducted using two publicly available benchmarks for data integration systems: Amalgam Integration Test Suite [5] and Test Harness for the Assessment of Legacy information Integration Approaches (THALIA) [6]. The work performed is not a complete data integration system. Rather, it's an extension to any data integration system.

The rest of this paper is organized as follows. In Section II, we briefly discuss the data quality dimensions used in our previous work. Section III illustrates the architecture and functions of our data integration quality system components. Section V describes our quality driven query processing algorithm. The experiments are described in section VI. The conclusion and future work are presented in Section VII.

This work is part of a complete research group composed of researches from Cairo University and Arab Academy for Science Technology & Maritime Transport (AAST) focusing on data integration topics [4, 7, 8, 9, 10].

II. Data quality dimensions usage in data integration

In general, data quality means "fitness for use" [11, 12]. So, the interpretation of the quality of data item depends on the user's needs. Wang and Strong [13] have empirically defined fifteen data quality dimensions considered by end users as the most significant. They classify these dimensions into contextual, intrinsic, representational and accessibility quality as shown in "Figure 1".

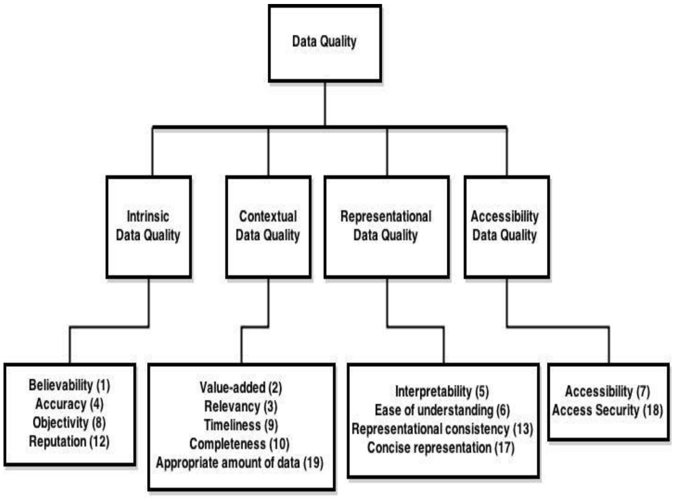


Figure 1. A conceptual framework of data quality

In our previous work, we selected data quality dimensions that could affect the data integration process and could be considered important from user's prospective. The data quality dimensions chosen were:

A. Accuracy

Wang and Strong [13] defines accuracy as "The extent to which data are correct, reliable, and certified free of error". Increasing accuracy of the query answer is important from user's prospective as data sources might contain incorrect or misspelling data.

B. Completeness

Completeness defined as "the extent to which data are of sufficient breadth, depth, and scope for the task at hand" [13]. Querying one data source gives a set of results. As the number of data sources queried increase, the result will be more complete.

C. Cost

Cost is the amount of money required for a query. Considering cost is important so that users can choose between free and commercial data sources.

D. Response Time

It is the amount of time when the mediator submit a query and receive the complete response from the data source. Response time is important as users waiting a long time for a response are more willing to abandon the query.

E. Timeliness

Timeliness is how old the data are in a data source [14]. Timeliness is important as some data sources might be outdated and the user might be interested in getting up-to-date data.

III. Quality System Components

The quality system component consists of (1) Data quality acquisition and (2) user input. The quality system components are integrated in the mediator-wrapper architecture. See green boxes in "Figure 2."

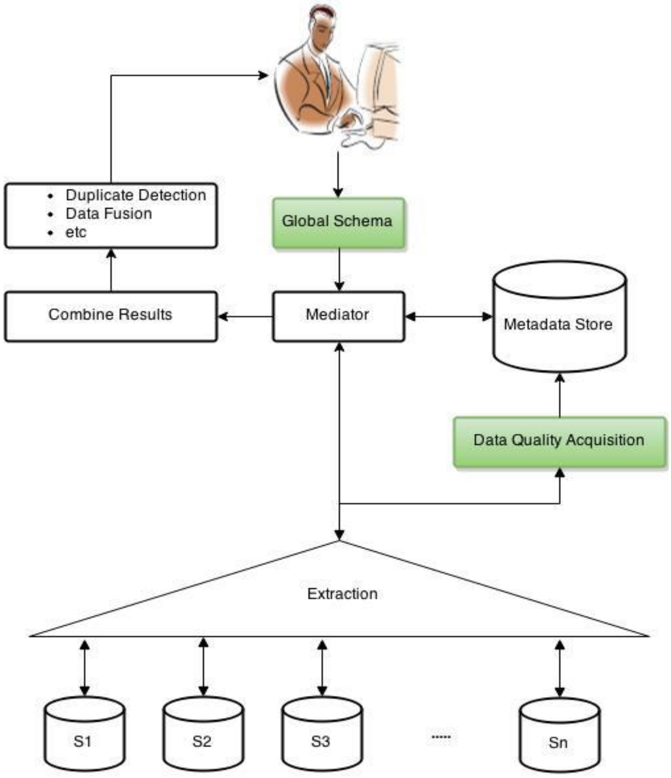


Figure 2. Data integration system quality system components

In the following sub-sections, we present the structure and the functionality of each component.

A. Data Quality Acquisition

The data quality acquisition (DQA) component is responsible for extracting attributes and relations from the data sources and store them in the metadata store. It is also responsible for executing data quality queries against the data sources, receiving the results and store them in the metadata store. The metadata store used by the DQA is shown in "Figure 3":

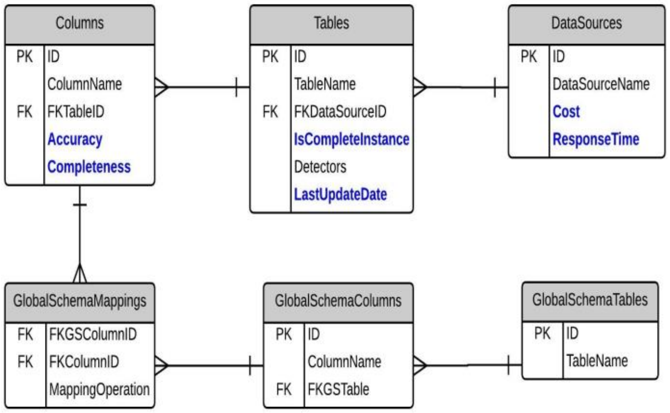


Figure 3. Metadata structure

Table I illustrates the data quality dimensions selected in our previous work and the granularity for each dimension.

TABLE I. DATA QUALITY DIMENSIONS AND GRANULARITIES LEVELS

DQ Dimension	**Measures granularities**		
	Data source level	***Relation level***	***Attribute level***
Accuracy			✓
Completeness			✓
Cost	✓		
Response time	✓		
Timeliness		✓	

In the following sub-sections, we describe how we measure each dimension presented in table I. These quality measures may enhance the quality of the data fusion process. Data quality dimensions chosen are highlighted in blue in “Figure 3”.

1) Accuracy

Tomas C. Redman [15] present the data accuracy measurement framework (“Figure 4”) for understanding the various measurement techniques based on choices made regarding four factors: where to measure the data, which part of the data will be measured, how to measure the data and the granularity of the measures.

We applied Redman’s data accuracy measurement framework in our case by selecting from the choices for each of the four factors.

- Where measurements are taken: We measured accuracy from the data sources. (i.e. from database).
- What attributes to include: We measured accuracy on the data sources’ attributes that correspond to global schema’s attributes.
- The measurement device: We will compared the value of each attribute to its domain of allowed values. Complaints and domain experts’ feedback were also used to identify erred data and a correction for them which help improve accuracy measure.
- The scale on which results are reported: Attribute level.

$$\text{Attribute Accuracy} = \frac{\text{Number of fields judget correctly}}{\text{Number of fields tested}} \qquad (1)$$

2) Completeness

The Literature classifies completeness into three types: column completeness, schema completeness, and population completeness [16]. At the most abstract level, schema completeness refers to the degree to which all required information are present in a particular data set. At the data level, column completeness can be defined as the measure of the missing values for a column in a table. Each of the three types can be measured by dividing the number of incomplete items by the total number of items and subtracting from 1 [16].

$$\text{Schema/Attribute completeness} = 1 - \frac{\text{Number of incomplete items}}{\text{Total number of items}} \quad (2)$$

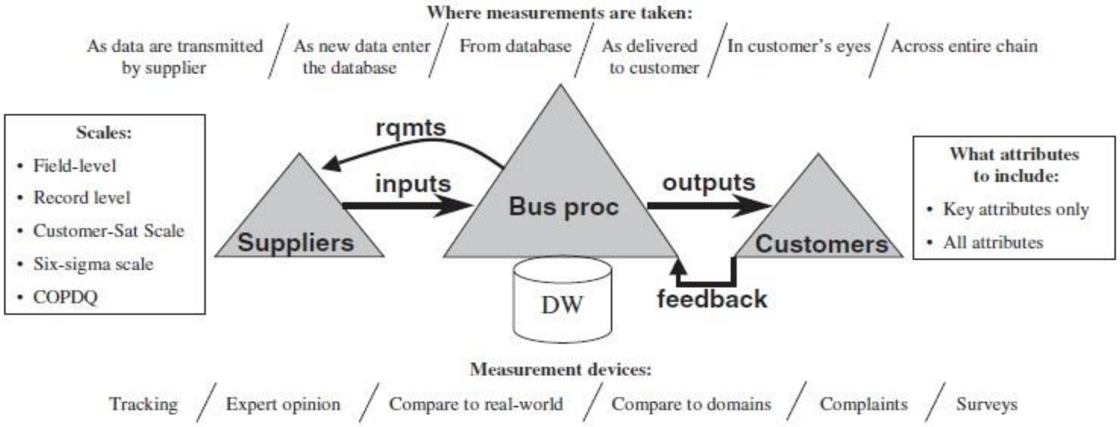


Figure 4. The data accuracy measurement framework

The range for completeness is 0 - 1, where 0 represents the lowest score and 1 represents the high score.

We add a custom data quality criteria called “Complete instance relation” that can be measured at schema level. A relation is marked as complete instance if its cardinality is complete. (i.e. all the tuples are represented in the relation). This information will be given directly to the data integration system by the end user through an input form.

3) Cost

It is the price for accessing specific data source. We assume that the user is charged on pay-by-query basis. The cost per query is measured in US dollar.

4) Response Time

We measured the response time of a data source by sending a bunch of queries to the data sources to judge their average response time for different types of queries at different times of day.

5) Timeliness

We measured timeliness by using the update information provided by the data source. We assumed that the data source updates its data at the relation level and the data at the data sources are not archived.

B. User Input

To give users the option to specify constraints on the retrieved result, we used the proposal of Gertz and Schmitt [17]. We added two options to the SQL dialect. The first one is cost which is the amount of money a user can pay and the second option called fusion that can be set to true or false and is used to give the user the option to retrieve data from all possible data sources.

A query Q with quality constraint expressed on the mediated schema expressed in an extended SQL syntax:

```
Select A1,…..,Ak
from G
where < selection condition >
with < data quality goal >
fusion < true | false >
Cost < x$ >
Where A1.A2,…, Ai are global attributes of G
```

Selection condition: conditions used to filter the data.

Data quality goal: quality dimensions defined on the selected attribute Ai and gets a value according to table II.

TABLE II. DATA QUALITY DIMENSIONS LEVELS

Level	***Start threshold***
High	70
Meduim	50
Low	0

Fusion: When set to true, this means that the user wants to fuse data from all possible data sources. When set to false, the mediator selects only one alternative that has the minimum number of data sources.

Cost: the amount in US dollar the user can pay.

IV. QUALITY DRIVEN QUERY PROCESSING

The data requested by the user is usually located on more than one data source. Every combination of data sources that meet the user's requirements (attributes and quality criteria) is an alternative. If a single data source can meet all user's requirement, this is an alternative. Given a query Q against the mediated schema asking for A1,…..,An attributes with or without quality requirements, We developed a quality-driven query algorithm presented in [4] to determine which combinations of sources can answer the query.

V. EXPERIMENTS AND RESULTS

In this section, we describe the implementation of the query planning algorithm and the quality system components. The goals of the experiments are to measure the response time, number of data sources needed to answer a given query and the cost of accessing the data sources to answer the queries. The experiments were done according to the following execution paths:

- When no quality measure were calculated. (i.e. the data integration system ignores the pre-saved quality measures as if they weren't exist)
- Default execution. This means if the user didn't specify quality constraints, the DIS retrieves the best result according to the pre-saved quality measures.
- When user specifies quality constraints. In this case, the user has selected some attributes and specified quality constraints on some of them.

Also during the experiments, all attributes from the global schema were selected.

We ran the experiments on a laptop shipped with an Intel Core i7-2760QM with 4 x 2.4 GHz CPU and 6 GB RAM. The laptop operates with Windows 7 ultimate edition. The tools used for the experiments were Microsoft SQL Server 2014® and Microsoft visual C# 4.5®.

A. Amalgam

Amalgam is a benchmark which consists of several schemas and datasets storing bibliographic information. It consists of four schemas. Each schema represents a data source. The authors of the benchmark require anyone who needs the data to request it from them. So, we requested the data from the authors and gratefully received it. We created the schemas in a SQL server database called "Amalgam" and loaded the data into it.

The first component of our quality system components is the data quality acquisition component. As we mentioned in section III, the data quality acquisition component is responsible for extracting attributes and relations from the data sources and store them in the metadata store. It is also responsible for executing data quality queries against the data sources, receiving the results and store them in the metadata store. So, we created the metadata store described in "Fig 3" which consists of six tables in the same database "Amalgam". We created a tool to map the global schema columns with the local schema columns. Table III shows the global schema tables and global schema columns used:

TABLE III. GLOBAL SCHEMA TABLES AND COLUMNS

Global Schema Table	**Global Schema Columns**				
Article	ArticleID	Title	Author	Journal	Year
	Month	Pages	Volume	Location	Abstract
Book	Title	Publisher	Year	Month	Pages

The data quality queries executed by the data quality acquisition component were implemented as stored procedures. Those stored procedures contain the equations used to calculate the completeness and accuracy of the attributes in the data sources. The stored procedures were ran according to a SQL server scheduled job. The job can be customized by the system administrator according to the data sources change frequency. Also we can change the queries used by the data quality acquisition anytime. Whenever data quality acquisition completes a run, the quality measures in the metadata store will be updated with the new values.

The second component is the user input. The purpose of the user input component is to give the users the option to specify quality constraints on the retrieved result. The user selects the required attributes and optionally specify a data quality constraint on each selected attribute. The user can choose between accuracy and completeness. The user also has to select the level of the DQ constraint which can be: high, medium or low. These levels get values according to table II. Also the user can check the fusion option and specify a cost of accessing a data source in case there are data sources that require a cost.

We considered two different scenarios w.r.t fusion option. In the first scenario, the fusion option is set to false while in the second is set to true. Regardless of the scenarios, table IV shows the quality measures of the data sources:

TABLE IV. THE QUALITY MEASURES OF THE DATA SOURCES

Data Source	Complete instance tables	Cost	Response time
S1	Article, Author	3$	500 sec
S2	authors, citForm, journal, abstracts, months, numbers, pages, titles, volumes, years	2$	500 sec
S3	author, article	4$	500 sec
S4	author , publication	5$	500 sec

The cost criteria selected were 7$. Hence, all data sources will be used to answer the query.

a) Response time

Table V shows the response time of our approach in both scenarios (when fusion is false and true) and according to the different execution paths:

TABLE V. RESPONSE TIME

Execution path	Attributes with DQ Constraints	Response time (sec)	
		Fusion = false	Fusion = true
No quality measure	-	1.445 sec	1.455 sec
User specified quality	Title, Journal year	0.772 sec	1.328 sec
Default	-	0.458 sec	1.408 sec

The results in table V show that response time is reduced after adding the quality measures even if the user did specify quality criteria regardless of the fusion option.

b) Number of accessed data sources

Table IV shows the number of accessed data sources needed to answer the query.

TABLE VI. NUMBER OF ACCESSED DATA SOURCES

Execution path	Attributes with DQ constraints	Number of accessed data sources	
		Fusion = false	Fusion = true
No quality measure	-	4	4
User specified quality	Title, Journal year	2	4
Default	-	2	4

The results in table VI show that if no quality measures were added, the DIS needs to query the whole data sources. While after adding quality measures, the number of data sources reduced to 2 instead of 4. The number of data sources remain 4 when fusion was set to true, because the query planning algorithm merged the data sources in all alternatives and queried each data source only once. The alternatives generated were consisted of the 4 data sources.

c) Cost of answering the query

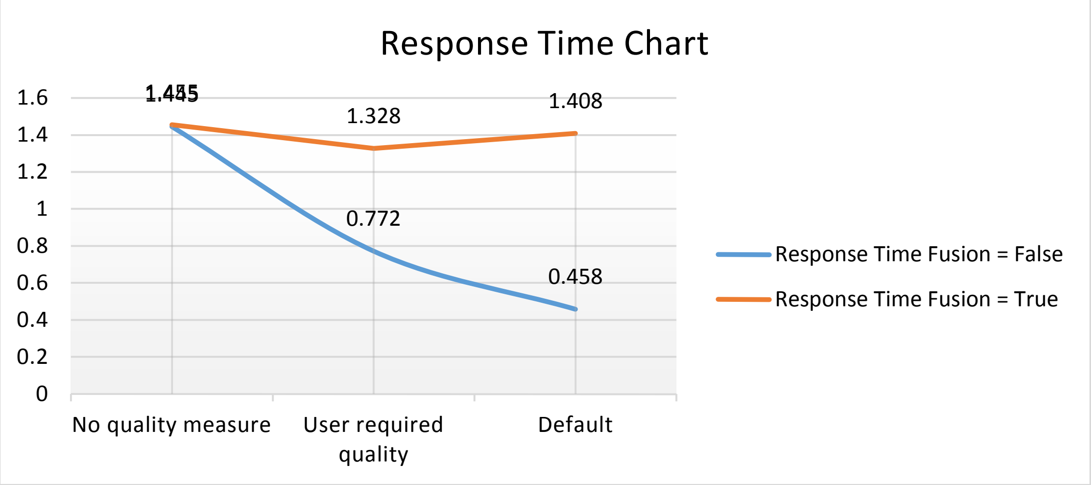
Table VII shows the amount of money needed to answer the query.

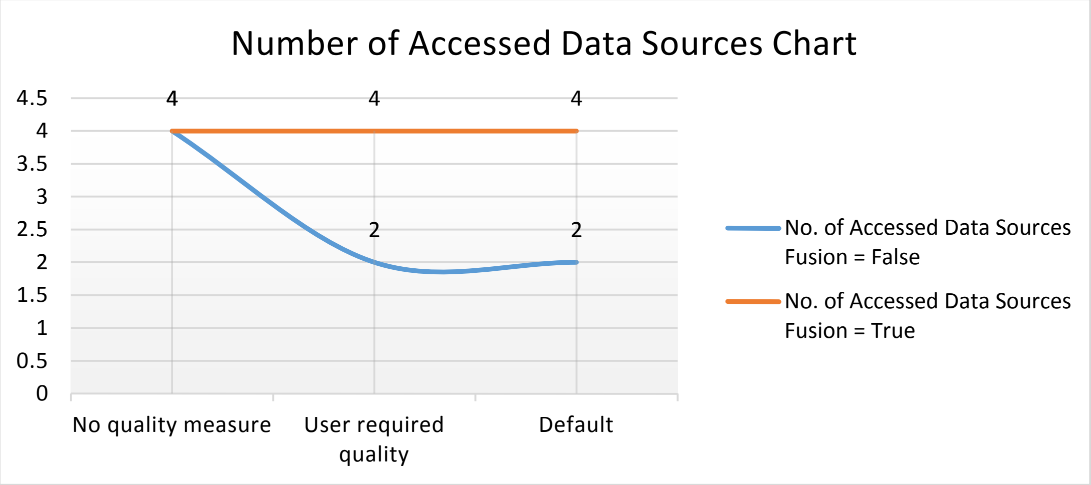
TABLE VII. COST NEEDED TO ANSWER THE QUERY

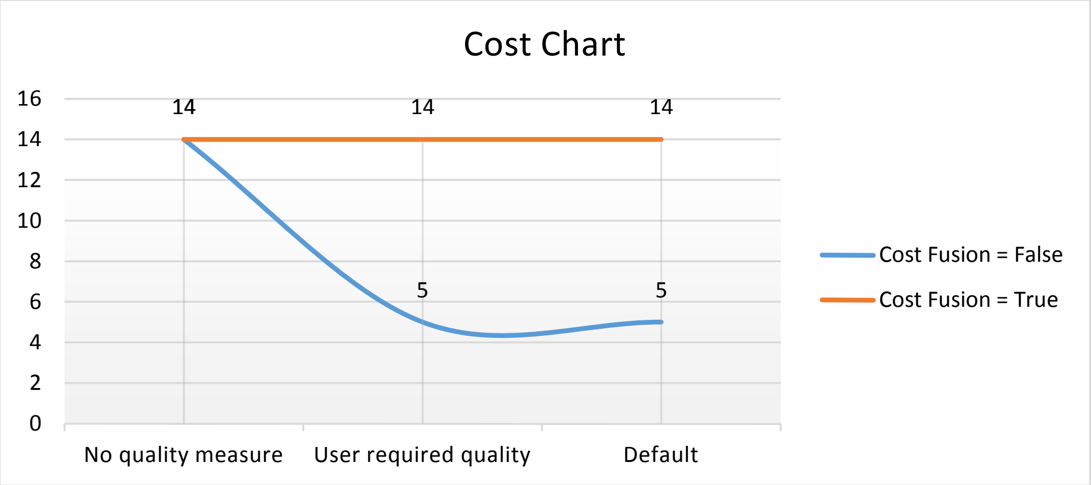
Execution path	Attributes with DQ constraints	Cost($)	
		Fusion = false	Fusion = true
No quality measure	-	14	14
User specified quality	Title, Journal year	5	14
Default	-	5	14

The results in table VII show that if the data sources do require cost and according to the cost assumptions in table IV, the amount of money is reduced after adding the quality measures. However, it remains the same when fusion was set to true, because when fusion set to true, the query planning may access the whole data sources according to the alternatives generated.

The following charts represent the results of amalgam benchmark:







B. THALIA

THALIA (Test Harness for the Assessment of Legacy information Integration Approaches) is a public available testbed and benchmark for information integration systems. It provides 42 downloadable sources representing university course catalog from computer science around the world. The goal of the benchmark is a systematic classification of the different types of syntactic and semantic heterogeneities that are described by the twelve queries provided.

As we did in amalgam benchmark, all schemas and data provided by THALIA benchmark were loaded into a relational database. The database called “Thalia”. For the data quality acquisition component, we created the metadata store described in “Fig 3” which consists of six tables in the same database “Thalia”. We used the same mapping tool we used in amalgam benchmark to map the global schema columns with the local schema columns. Table VIII shows the global schema table and global schema columns used:

TABLE VIII. GLOBAL SCHEMA TABLES AND COLUMNS

Global Schema Table	**Global Schema Columns**			
Course	Code	CourseName	Instructor	Credits
	Prerequisite	Days	Building	Room
	HomePage	Description		

The first component of our quality system components is the data quality acquisition component. As with amalgam benchmark, the data quality queries executed by the data quality acquisition component were implemented as stored procedures.

The second component is the user input. As with amalgam benchmark, the user can choose between accuracy and completeness and the level of the DQ constraint which can be: high, medium or low. Also the user can check the fusion option and specify a cost of accessing a data source in case there are data sources that require a cost.

We considered two different scenarios w.r.t fusion option. In the first scenario, the fusion option is set to false while in the second is set to true. Regardless of the scenarios, table IX shows the quality measures of the data sources:

TABLE IX. THE QUALITY MEEASURES OF THE DATA SOURCES

Data Source	**Is complete instance**	**Cost**	**Response time**
Arizona State University	True	3	500
Bilkent University	False	3	500
Bosphorus University	True	3	500
Boston University	False	3	500
Brown University	False	3	500
California Institute of Technology	False	3	500
Carnegie Mellon University	False	3	500
Columbia University	False	3	500
Cornell University	False	3	500
Eidgenössische Technische Hochschule Zürich	True	3	500
Florida International University	True	3	500
Florida State University	True	3	500
Furman University	False	3	500
Georgia Tech	False	3	500
Harvard University	False	3	500
Hong Kong University	False	3	500
Istanbul Technical University	False	3	500
Kansas State University	False	3	500
Michigan State University	False	3	500
MiddleEast Technical University	False	3	500
NewYork University	False	3	500
Northwestern University	False	3	500
Pennsylvania State University	False	3	500
Rochester Institute of Technology	True	3	500
Stanford University	False	3	500
UniversidaddePuertoRico Bayamon	False	3	500
University of Arizona	False	3	500
University of Berkeley	False	3	500
University of California Los Angeles	True	3	500
University of California SanDiego	True	3	500
University of Florida	False	3	500
University of Illinoisat UrbanaChampaign	False	3	500
University of Iowa	False	3	500
University of Maryland	False	3	500
University of MassachusettsBoston	False	3	500
University of Michigan	True	3	500
University of NewSouth Wales Sydney Australia	False	3	500
University of Toronto	False	3	500
University of Virginia	True	3	500
Washington University	True	3	500
Worcester Polytechnic Institute	False	3	500
Yale University	False	3	500

Since each data source has one table, the complete instance table measure is attached to the data sources. The cost criteria selected were 7$. Hence, all data sources will be used to answer the query.

a) Response time

Table X shows the response time of our approach when fusion is false and true and according to the different execution paths:

TABLE X. RESPONSE TIME

Execution path	Attributes with DQ constraints	Response time (sec)	
		Fusion = false	Fusion = true
No quality measure	-	1.723 sec	1.716 sec
User specified quality	CourseName, Code, Instructor	0.831 sec	1.801 sec
Default	-	0.774 sec	1.706 sec

The results in table X show that response time is reduced after adding the quality measures when fusion was false even if the user did specify quality criteria. When fusion option was true, it required a little time because the quality constraints checks. However, the default execution requires time less than when no quality measures were calculated.

b) Number of accessed data sources

Table XI shows the number of accessed data sources needed to answer the query:

TABLE XI. NUMBER OF ACCESSED DATA SOURCES

Execution path	Attributes with DQ constraints	Number of accessed data sources	
		Fusion = false	Fusion = true
No quality measure	-	42	42
User specified quality	Title, Journal year	5	11
Default	-	5	11

The results in table XI show that if no quality measures were added, the DIS needs to query the whole data sources. While after adding quality measures, the number of data sources reduced to 5 instead of 42 when fusion was false and to 11 when fusion was true.

c) Cost of answering the query

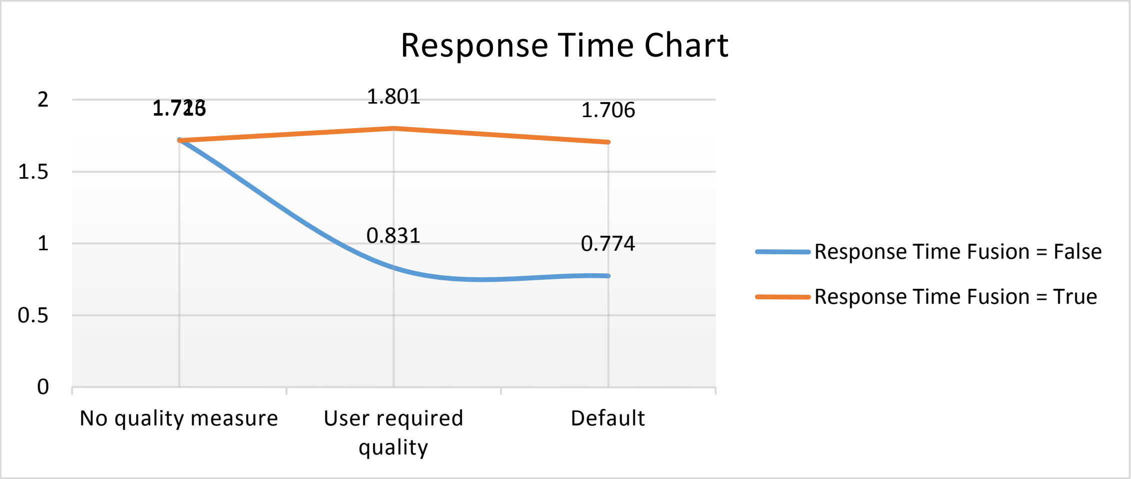
Table XII shows the amount of money needed to answer the query.

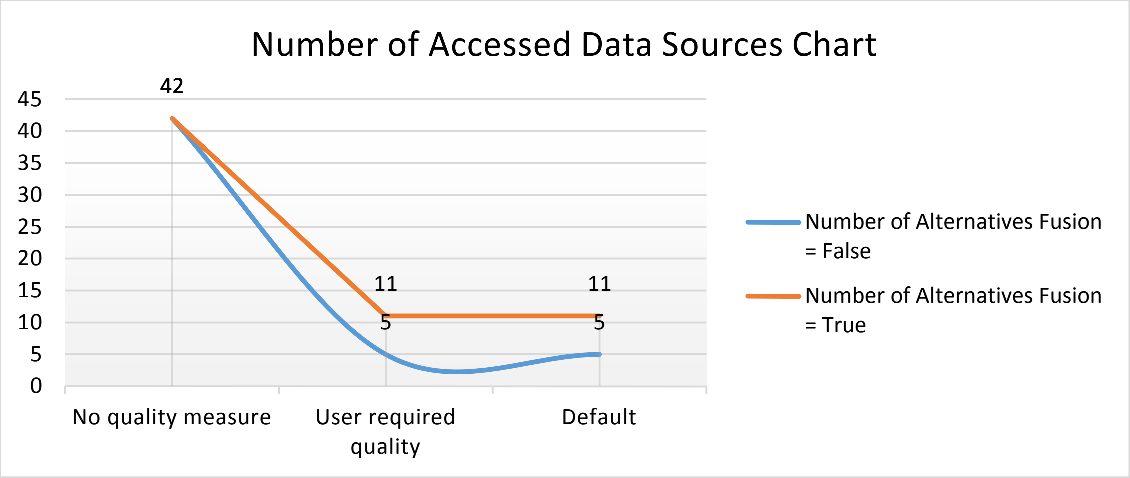
TABLE XII. COST NEEDED TO ANSWER THE QUERY

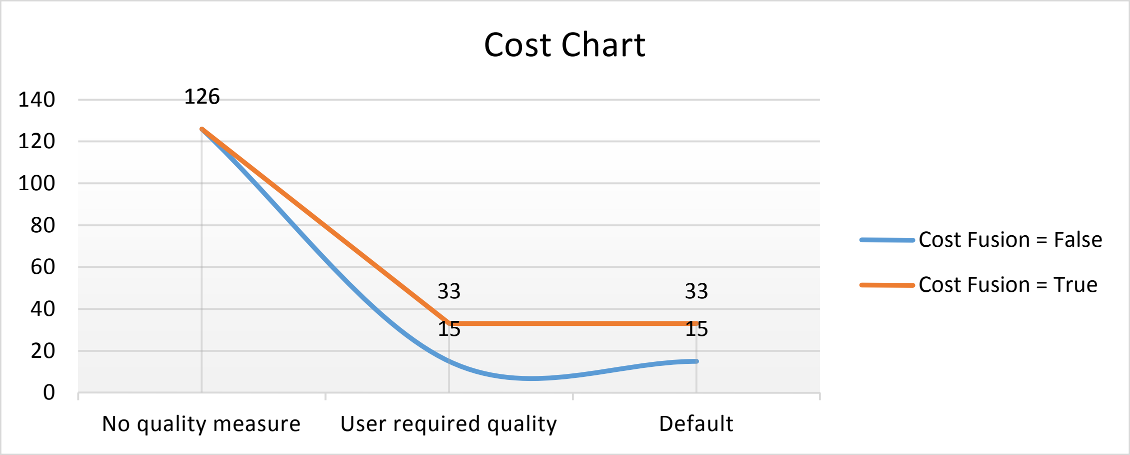
Execution path	Attributes with DQ constraints	Cost($)	
		Fusion = false	Fusion = true
No quality measure	-	126	126
User specified quality	Title, Journal year	15	33
Default	-	15	33

The results in table XII show that if the data sources do require cost and according to the cost assumptions in table IX, the amount of money is reduced after adding the quality measures regardless if fusion option. When fusion is true, the query planning accesses all data sources in all alternatives generated. That's why the cost is high when fusion is true.

The following charts represent the results of THALIA benchmark:







VI. CONCLUSION AND FUTURE WORK

Data integration systems may suffer from producing results that not only lack the quality but also take a long time to arrive.

In this paper, we have pointed out the importance of data quality in integrating autonomous data sources. The main contribution of this paper is an efficient method aimed at selecting a few possible data sources to provide more quality oriented result to the user. We added quality system components to integrate data quality dimensions in a data integration environment for structured data sources only. With the help of these criteria, we developed a quality driven query execution algorithm to generate high quality plan that meets user's requirements. Our experiments show that our approach delivers result in a reasonable amount of time and using the minimum number of data sources possible. Further research will extend the approach to be applied on different types of data sources such as semi-structured and unstructured data sources.

REFERENCES

[1] M. Lenzerini, "Data integration: a theoretical perspective," in Proceedings of the twenty-first ACM SIGMOD-SIGACT-SIGART symposium on Principles of database system, Madison, Wisconsin, June 03-05, 2002.

[2] A. Y. Halevy, "Answering queries using views: A survey," The VLDB Journal — The International Journal on Very Large Data Bases, vol. 10, no. 4, pp. 270-294, December 2001.

[3] A. Doan, A. Halevy and Z. Ives, Principles of Data Integration, San Francisco, CA: Morgan Kaufmann Publishers Inc., 2012.

[4] M. Samir, A. H. El-Bastawissy and M. Kholeif, "Quality Driven Approach for Data Integration Systems (http://icit.zuj.edu.jo/icit15/DOI/Database_and_Data_Mining/0078.pdf)," in The 7th International Conference on Information Technology, Amman, Jordan, 2015.

[5] R. J. Miller, D. Fisla, M. Huang, D. Kalmuk, F. Ku and V. Lee, "The Amalgam Schema and Data Integration Test Suite http://dblab.cs.toronto.edu/~miller/amalgam/," 2001.

[6] J. Hammer and M. Stonebraker, "Thalia: Test harness for the assessment of legacy information integration approaches," in In Proceedings of the International Conference on Data Engineering (ICDE) Pages 485-486, 2005.

[7] A. Al-Qadri, A. H. El-Bastawisssy and O. M. Hegazi, "Approximating Source Accuracy Using Dublicate Records in Data Integration," IOSR

Journal of Computer Engineering (IOSR-JCE), vol. 13, no. 3, pp. 68-72, 2013.

[8] M. M. Hafez, A. H. El-Bastawissy and O. M. Hegazy, "A statistical data fusion technique in virtal data integration environment," International Journal of Data Mining & Knowledge Management Process (IJDKP), vol. 3, no. 5, pp. 25-38, September 2013.

[9] M. M. Hafez, A. H. El-Bastawissy and O. M. Hegazy, "Using Information Gain in Data Fusion and Ranking," in Recent Advances in Computer Engineering, Communications and Information Technology, pp. 157–165, 2014.

[10] A. Z. El Qutaany, A. H. El Bastawissy and O. Hegazy, "A Technique for Mutual Inconsistencies Detection and Resolution in Virtual Data Integration Environment," in Informatics and Systems (INFOS), 2010 The 7th International Conference on, 2010.

[11] J. M. Juran, The Quality Control Handbook, 3rd ed., New York: McGraw-Hill, 1974.

[12] G. Kumar Tayi and D. P. Ballou, Examining data quality, Communications of the ACM,, v.41 n.2, p.54-57, Feb. 1998.

[13] R. Y. Wang and D. M. Strong, "Beyond accuracy: what data quality means to data consumers," Journal of Management Information Systems, vol. 12, no. 4, pp. 5-33, March 1996.

[14] F. Naumann, Quality-Driven Query Answering for Integrated Information Systems, Springer-Verlag Berlin, Heidelberg, 2002.

[15] T. C. Redman, "Measuring Data Accuracy A Framework and Review," in Information Quality, R. Y. Wang, E. M. Pierce, S. E. Madnick and C. W. Fisher, Eds., Armonk, NY, M.E. Sharpe, 2005, pp. 21-36.

[16] L. L. Pipino, Y. W. Lee and R. Y. Wang, "Data quality assessment," Communications of the ACM, vol. 45, no. 4, pp. 211-218, April 2002.

[17] M. Gertz and I. Schmitt, "Data integration techniques based on data quality aspects," in 3rd National Workshop on Federal Databases, Magdeburg, Germany, 1998.

[18] G. Wiederhold, "Mediators in the Architecture of Future Information Systems," IEEE Computer, vol. 25, no. 3, pp. 38-49, March 1992.

[19] Y. Wand and R. Y. Wang, "Anchoring data quality dimensions in ontological foundations," Communications of the ACM, vol. 39, no. 11, pp. 86-95, November 1996.

[20] M. Spiliopoulou, "A calibration mechanism identifying the optimization technique of a multidatabase participant," in Proc. of the Conf. on Parallel and Distributed Computing Systems (PDCS), Dijon, France, September 1996.

[21] M. Scannapieco, A. Virgillito, C. Marchetti, M. Mecella and R. Baldoni, "The daquincis architecture: a platform for exchanging and improving data quality in cooperative information systems," Information Systems, vol. 29, no. 7, pp. 551 - 582, October 2004.

[22] F. Naumann, U. Leser and J. C. Freyta, "Quality-driven integration of heterogenous information systems," in 25th proceeding of the International Conference on Very Large Databases (VLDB) , p.447-458, Edinburgh, Scotland, September 07-10, 1999.

[23] M. Ge and M. Helfert, "A Review of Information Quality Research - Develop a Research Agenda," in Proceedings of the 12th International Conference on Information Quality (ICIQ 07), MIT, Massachusetts, USA, November 9-11, 2007.

[24] A. Charnes, W. W. Cooper and L. Rhodes, "Measuring the efficiency of decision making units," European Journal of Operational Research, vol. 2, no. 6, pp. 429-444, November 1978.

[25] C. Batini and M. Scannapieco, Data Quality: Concepts, Methodologies and Techniques (Data-Centric Systems and Applications), Secaucus, NJ: Springer-Verlag New York, Inc, 2006.