Improving Mediator-based Information Integration to Resolve Syntactic Heterogeneity

Muhammad Ali Abid  
COMSATS University Islamabad

Farhan Amin  
Yeungnam University

Muhammad Zeeshan (zeeshan.msc08@gmail.com)  
COMSATS University Islamabad

Research Article

Keywords: Executive Support System, Heterogeneity of Database, Recommender System, Web Services, Accuracy, Trust

Posted Date: November 16th, 2022

DOI: https://doi.org/10.21203/rs.3.rs-2253569/v1

License: This work is licensed under a Creative Commons Attribution 4.0 International License. Read Full License
Improving Mediator-based Information Integration to Resolve Syntactic Heterogeneity

Muhammad Ali Abid, Farhan Amin and Muhammad Zeeshan*

Assistant Professor Computer Science, University of Agriculture, Dera Ismail Khan

Department of Information and Communication Engineering, Yeungnam University, Gyeongsan 38541, Korea

Department of Mathematics, University of Agriculture, Dera Ismail Khan

ali.abid@uad.edu.pk (draliabidawan@gmail.com, farhanamin10@hotmail.com, zeeshan.msc08@gmail.com)

ABSTRACT

An emerging approach that addresses data heterogeneity challenges is the mediator-based architecture allowing transparent access to the data stored in many sources. Due to the growing diversity of data sources, the data integration process becomes a performance and administrative bottleneck. When dealing with decentralized heterogeneous data sources, the mediator-based technique is typically used to integrate the data. It describes a collection of applications that follow a number of data sources. The importance of analyzing and synthesizing the collected data has lately increased for academics researching autonomous and heterogeneous software systems. This study uses a mediator-based information integration model to improve pattern-based reasoning and overcome grammatical problems with integrating diverse information from web sources (IATs). The main goal of this study is to improve the mediator design for usage in the mediator-based information integration paradigm in order to address the problems caused by syntactic heterogeneity. Using our recommended methodology and enhancement strategy, the proposed technique would choose the pertinent domain from a variety of vendor-related data sources and antiquated file systems and deliver the necessary information set from heterogeneous data sources. Our suggested extended design functions well in the online bookstore as well, where there are several data sources and antiquated file systems. Future applications of this research include providing a thorough syntactic method that must be used to integrate data from various sources within the same organization into any Executive Support System (ESS).

Keywords: Executive Support System, Heterogeneity of Database, Recommender System, Web Services, Accuracy, Trust

1. Introduction

Although search engines use it to acquire important information depending on what the user wants, the relevance of the heterogeneous data on the web has quickly expanded in the field of data integration and exchange [1]. Information integration describes the challenge of providing an accurate and consistent picture of a set of data that is dispersed across several autonomous and heterogeneous data sources [2]. One of an IT recruiter's greatest concerns is tackling the issue of information integration into two key contexts [3].

i. EIS (Enhanced Information Solutions)

ii. Information Integration System between Organizations (Web)

Due to the fact that different online sources use various database management systems (DBMS) for the administration of their data and that each DBMS has its own syntax and semantics, we are only concentrating on the interior generational information integration (web) in our study [4]. Physical transparency, heterogeneity, extensibility, autonomy, and performance are only a few of the characteristics of information integration.

1.1 Problem Background

Various issues with information integration from diverse web sources need to be addressed by researchers [5][6]:

- Automated source wrapping
- Construction of global schemas
- Discovery of such mappings, transmission of source-to-global schema mappings
- Management of source access mechanism limitations
- Data reconciliation, cleansing, and extraction; How to transmit data using mappings
- The modelling issue: describing the global schema, sources, and mappings
- Querying problem
- The Data Exchange Paradox: How to Transfer Data Using Mappings
- To improve query answers, process updates from the sources, express glob, or he glob.
1.2 Problem Statement

The integration of data from many web sources presents hurdles in terms of syntax and semantics [7]. The grammatical problems are brought on by the definitions, structures, and restrictions put on the data that will be stored in the database. Contrarily, semantic difficulties relate to the semantics of database objects, i.e., metadata, which is essential in database systems.

The pattern basis technique might be used to improve the mediator based information integration model and resolve the grammatical problems associated with integrating heterogeneous information from web sources, according to the research hypothesis.

1.3 Research Aim

The goal of the study is to identify, assess, and pinpoint the flaws in the current approaches to the syntactic problems that the models created for integrating heterogeneous information confront.

The purpose of the work is to address issues related to the syntactic heterogeneity of information integration by enhancing the mediator architecture that should be applied in the mediator-based information integration model [8].

1.4 Research Objectives

The first of the research's two main goals is to enhance the mediator-based information model for diverse web sources. The second goal is to create and evaluate a pattern-based approach for improving the models now in use for integrating data from numerous sources of heterogeneous information [9].

1.5 Research Scope

The study is only focused on these issues:
1) This study makes use of the model that has been improved.
2) This work only addresses the syntactic issues related to the integration of diverse information.
3) This research was done to develop a novel method called "pattern basis," which makes use of the idea of a two-dimensional matrix[10].

1.6 Significance of the study

Both theoretically and practically, the research is significant. This study area's justification and drive are

I. To improve information integration, pattern-based techniques should be used by model-based mediators to prevent the syntactic problems brought on by heterogeneity.
II. This study offers a syntactic method for using any ESS to apply data from many sources inside the same company.

1.7 Proposed Methodology

The research approach used for this study is briefly described in this section. Section 3 of the document provides the details. The improved model is the one that is suggested in picture 1.

In the suggested model, we must examine the syntactic strategies currently in use when integrating data from various web sources, followed by the application of the pattern base technique [11] to assess the accuracy and clients' responses.

Fig: 1.1 Proposed Model for the Heterogeneous Information Integration
1.8 Research Contributions

This work aims to address problems with the syntax of data sources that are frequently represented in data base defining languages (DTDs), while data-related actions are expressed in XML schemas. For this research/study, we must apply a pattern-based approach that is based on the manipulation of the two-dimensional matrix that will be used to characterize the data. We will also consider the constraints placed on the data to be processed while responding to client-generated queries that request data from numerous sources with a variety of structural properties.

1.9 Proposal Organization

In chapter 1, we gave a general review of information integration concepts. The subsequent conversation centered on integrating disparate information.

In our discussion of heterogeneous information integration models, we covered many research contributions. In chapter 2, we provided a thorough literature assessment of recent developments in the field of integrating heterogeneous information using various models. We talked about various approaches to the heterogeneity of syntactic information. The approach for the suggested mechanism will be covered in Chapter 3. The Chapter 4 shows comprehensive architecture and the enhancements to be implemented for desired results. Finally, chapter 5 provides the conclusion.

2. Related Work

2.1 Information

According to the most technical definition, information is, by definition, a group of symbols combined to make a message. Signals and signs can be used to record information or to communicate. Information is anything that happens and changes the status of a dynamic system. Information (or speech) is conceptually presented [11].

Science research has received more attention recently than in the past due to the rapid change brought on by the proliferation of new technology. In the current example, we talk about inexpensive sensors that can provide measurements and quantities of data regarding storing biological data. Large amounts of data can be saved, managed, and used for analysis in these modern storage capacity devices to deal with difficult scenarios. Major concerns with collecting, integrating, managing, and making the data accessible to various applications for diverse purposes arise when dealing with data integration from multiple sources.

When a sample of data from one source is processed and linked with data from other sources to produce desired results by experimenting with interpolation techniques, [12] assert that some crucial data may be lost due to omission while using the same models for representing data for the purposes of sharing and data integration (e.g., noon vs. midnight).

Recording the time between collecting the sample and analyzing it is crucial for highlighting important points, as is making sure that the samples are distinct from the data sources. If errors are made, measurements can be performed to fix them. To store and evaluate the data contained in various databases, several data models, formats, and platforms need to be used. The same issues could arise while analyzing text to extract certain desired traits in the realm of data mining [13].

When dealing with semi-structured data, the XML language is becoming more and more popular, and many writers recommend it for overcoming the problems in data integration. This is due to its customizable structure during code processing and the exposure of it as metadata for constructing schemas using DDL.

There are other suitable semantics that may be utilized in data integration so that it can be presented in various ways even if dealing with the syntax of a class of documents is vital. Data comparisons can be conducted using simulation tools that incorporate experimental datasets when working with data sets and texts that can be read by computers. Assumptions can be utilized in many formats for this purpose. Additionally, using a complex workflow can be a simple way to do modern scientific research.

2.2 Information Integration

The XML language is gaining popularity for dealing with semi-structured data, and several authors suggest it as a solution to problems with data integration. Due to its adaptable structure during code processing and exposure as metadata for DDL-based schema building, this is the case [13]. Although dealing with a specific class of the document's syntax is essential, other suitable semantics may be used in data integration to offer other presentation options. When comparing data sets and texts that can be read by computers, simulation techniques that take experimental datasets into account can be used.

A range of programming interfaces and data structures are also available. You can access a variety of data types thanks to information integration technology (structured, unstructured, and semi-structured). The data can be changed into a format that makes it simple for employees across the entire organization to obtain information. There are the ensuing features available:

- Data transformation for business analysis and data exchange.
- Real-time read and write access and data placement
• Performance-based, up-to-date, and accessible management.

2.3 Information Integration Issues

Different problems that information integration must deal with can often be divided into two categories.

i). Syntactic Issues

These problems involve the DDL, or data definition language. Describing the design of the databases from which the incoming data will be sourced. These must be described.

a. Database organization
b. Restrictions Description
c. Definition of an entity
d. Classification of Attributes
e. Attributes Detailed
f. Attributes data formats
g. Constraint on the primary key
h. Reference Integrity Restrictions

ii). Semantic Issues

The DML, which is used to access and edit data in relational databases, is at the root of these issues. DML is used in numerous databases other than those that use IMS/DLI and CODASYL, such as IDMS [14]. Although it is not possible to alter the schema or database objects, the SQL programming language is a form of data manipulation language. The tables and stored procedures are examples of persistent database objects, but the data they contain cannot be changed. Alternatively, these objects can be modified using SQL schema statements, which are seen as components of a unique language for expressing data. Although these two varieties of SQL are comparable in terms of their technical syntax, data types, expressions, etc., they are different in terms of their fundamental objectives [15]. The initial word of a sentence, which is nearly always a verb, is how languages used for data manipulation are ordered in terms of their functional capabilities. These verbs pertain to SQL and are:

a). Retrieving information
b). Adding information
c). Current Information
d). Information transformation
e). Information removal

Internet has evolved into a crucial tool for achieving the goal by establishing an informational organizational system for business development. It has been created on a small-to-large scale and is colloquially referred to as “Information Island.” It must be employed in the development and planning of all different types of information systems. Heterogeneous Information Systems (HISs) are unable to process the reengineered yet crucial information when dealing with business applications. They are also unable to keep up with the quick changes in operating systems and high level programming languages.

The E-Hub platform, based on cutting-edge internet technologies and innovative open-ended business applications, is used to connect all the information for sharing purposes in order to overcome these problems [16].

To improve system speed, the E-Hub is also connected to SOA and other corporate plug-ins for processing business applications [17].

In order to integrate information from many data sources into their heterogeneous systems, businesses utilize an integrated framework that allows for key SOA and SCI components to be plugged in. Users make use of the services that are necessary and advantageous for integrating various sources.

These services are accessible to HUB-SOA, ERP, PDM, and other model applications, as well as CRM, DAS SQL, and OEADB for top management decision-making. These integrated services were initially used to promote various platforms among consumers, but over time, they came under a variety of limitations due to financial and performance requirements in order to achieve widespread adoption and utilization and address HIS-related concerns.

P-2-P integration model has been established in order to address the problems associated with linking various communication protocols by utilizing heterogeneous resources. The aforementioned model was created using the SPOKE integration paradigm for ESbusIM to connect many nodes located at various physical centers. Additionally, SOA was paired with a few distributed SPOKE integration Model components to sustain challenging resource integration activities and prevent failures. As a result, research on integration models for gathering, processing, and evaluating data from many sources has attracted researchers’ interest and been identified as a crucial area to be covered when discussing SOA and other service components [18].

In their conclusions, the authors make it abundantly obvious that the improved architecture, previously known as HUB-SOA, is based on autonomous SOA services that are open to the virtually infinite possibility of having full procedure. Every time a hub is set up on a platform to use the communication bus to connect to other services in the architecture, an ESB is created. The platform is referred to as HUB-SOA due to its adaptability. A middle layer...
capable of intelligent communication with other services adhering to the same architecture, or SOA, is required in order to make use of this upgraded service, or HUB-SOA. Common apps and services are used throughout the HUB-SOA structure to bring it all together, as has been mentioned. [19].

2.4 Issues in Integration of Information

There are several technical problems that develop when utilizing programmers to access heterogeneous data sources, such as VSAM, DB2, etc., and these problems cause outputs to be produced in a non-relational format [19].

- In contrast to a typical relational DBMS, COBOL applications offer a flexible file definition feature that enables the creation of files with several record types and record formats in a single dataset.
- As a result, it is tricky for users to map since the problem of looping the data structure with the linked list is difficult to address.
- Legacy programs were provided with this built-in feature to streamline data management and address issues with data and referential integrity in the Oracle DB engine.
- When organizing legacy data to boost the performance of the relational DBMS, an Oracle schema is created, consisting of a mapping of Oracle keys (primary and secondary)[20].

2.5 Common Methodologies

It is required to introduce many technical common techniques that are crucial for carrying out any form of data activities related to either integration or migration before explaining the methodology to deal with data integration.

2.5.1 Data integration

In order to do this, we must briefly discuss the following integration strategies within the limitations of the stated requirements:

**Consolidated:** This method involves moving all data to a single database before implementing data management principles through a central location. Transaction Processing (TP) is used as an illustration to show the differences between Oracle and non-Oracle approaches of indirectly utilizing commits and other character sets to affect the order of the data [21].

**Federated:** This type of DI solution involves moving data around in a data source for upkeep and updating before combining it with the data wherever it is required. This is accomplished by creating a virtual database that combines multiple data sources, masking each source before combining it into a single logical view. These kinds of solutions are typically launched by both sides [21].

**Shared:** This technique or approach involves moving events and data to one or more data sources, processing the resource in a queue before creating it, and then serving it to various applications. Prior to exchanging data with other sources or applications in these types of solutions, data is typically maintained using technologies like replication or message queuing?

Due to the multiple advantages of combined or consolidated data, Oracle provides strong assistance for data integration on a broad scale. Large-scale legacy system management can be challenging for enterprises, especially when they need to use data from outside sources and don't own the data themselves. Therefore, rather than going into detail, we’ll concentrate on federated or shared solutions [22].

When a result, there is a need for heterogeneity solutions to be created and implemented as systems are integrated on various models. There are numerous examples of structural conflicts in this regard [22], as well as conflicts involving names, databases, schemas, and other elements. Similar interoperability between several sources may be offered to address syntactic heterogeneity [23].

The heterogeneity challenges, which must be overcome in order to integrate the numerous data sources, are solved by looking for equivalence between distinct data sources. To demonstrate structural variability in the process of integrating diverse data sources, this work aims to provide a mediator-based integration of the pertinent literature.

3. RESEARCH METHODOLOGY

3.1 Overview

This section outlines the ideas that form the foundation of the study's methodology and overall operational structure. The study's main methodological tool has been identified as the relationship between key theoretical concepts in information integration and how these concepts are combined to provide a framework for conceptualizing sustainability. The criteria that were utilized to choose the study area and carry out this research are presented in this chapter. Additionally, it describes the study's methodology, the kinds of data that are needed, their prospective sources, analytic procedures, and techniques. The methodological concept utilized in this study has a strong base thanks to information from literature on the subject's study.
3.2 Conceptual Framework

Using a meta-database management system (DBMS) called a federated database system, various separate database systems are smoothly merged into a single FDS (FDS). The constituent databases are connected by a computer network even though they are spread out geographically [24]. A federated database system is superior to the (often difficult) job of combining many databases since the constituent database systems still preserve their independence. A federated database, also known as a virtual database, is created by combining all of the various databases that make up the federated database system. Data federation prevents the constituent databases from effectively integrating their data.

The conceptual framework shown in figure 2 explains how the data warehouse architecture operates. This contains data that is saved in several data sources with various data formats. When an organization requires information, it must first obtain it from various data sources, after which all obtained data must be reorganized in order to be saved and later provided to clients.

In the third stage, the data is cleaned, or verified for consistency and redundancy. Data is subsequently saved at a central data store, which then keeps data marts up to date in accordance with user requirements in the form of various queries. Additionally, the data warehouse must create a data log to record all transactions carried out in response to user queries. All organizational management levels are to receive information from the data warehouse in order to make better judgments. In other words, it aids the decision-support system. Data can be accessed and changed during online analytical processing using an XML structure. Data mining, which is the process of gaining more knowledge from a data repository for use in decision-making, is another endeavor that the data warehouse supports.

3.3 Research Strategy

The topic on which we have concentrated for our study is presented in this section. In order to combine data from many sources, a global schema must be built. The definition of the global schema is the unified perspective of the data's structure as it emerges from many sources with various file formats. Two key elements of the global schema in Figure 3 are Mediator and Wrapper. It is the mediator's duty to interpret user queries so that they can be delivered to wrappers for additional processing in the direction of data sources. The mediator sends the user a reply after receiving the response from the data sources, which was received by the wrappers. The wrappers translate the mediators' input into database queries, which are then executed by the data sources.

3.3.1 Implementation of Wrapper

Figure 3.2 above shows the functionality of wrappers, which must be registered before processing client requests. The system administrator can directly control the utilization of various data sources. System administrators must first develop a subclass to replace the default wrapper in order to modify it to implement from scratch while utilizing abstract classes before employing these wrappers. The layout of default wrapper processing is shown below figure 3.3.
For wrappers in the chosen relational database, SQL Wrapper offers a defaulting behavior in the instant display. System administrators set SQL queries and schema mappings to ensure that these wrappers function in accordance with user expectations and input queries.

![Diagram of Syntactic issues](image)

**3.4 Proposed Research Study**

We will focus our investigation on the syntactical problems associated with the merging of disparate types of information, as shown in figure 4. The query language's structural variations and restrictions lead to syntactic heterogeneity. Heterogeneous solutions must be created and used for system integration across multiple models. Examples of this include conflicts between names, databases, schemas, etc., as well as structural conflicts.

- Differences in structure: When two data models offer primitives that are incompatible with one another, such as when relational models do not support specialization and inheritance whereas object-oriented models (OOM) do, there are structural inconsistencies [24].
- There are discrepancies caused by the constraints when two models accommodate two different constraints.
- The query languages that a DBMS offers are another element that can have an impact on the heterogeneity of other components of a DBMS.

**3.5 Data Collection**

The information is gathered through various research publications with an emphasis on highlighting and addressing heterogeneity, taking into account the goals of the study and its scope.

The following libraries are examined in order to get the data required for this study, and are further displayed in the accompanying figure, despite having limited digital access.

- IEEE publications – IEEExplore
- Science Direct
- ACM Digital Library
- Google Scholar
- Scopus
3.6 Query Processing and Mapping

The global schema is represented by rows in the mapping table, whereas the local schema is represented by columns [25]. Our suggested approach provides an example of a general structure for handling a query to find author information.

The mapping table that follows pertains to (http://alignapi.gforge.inria.fr/edoal.html) uses the EDOAL language.

```xml
<Map>
  <Cell>
    <edoal: Property rdf: about = "src # Name "/>
    <entity1>
    <entity2>
    <edoal: Property >
    <edoal: Property: parseType = "Collection"> 
    <edoal: Property: parseType = "Collection"> 
    <edoal: Property: about = "trg # Author Name "/> 
    <edoal: Property:DomainRestriction >
    <edoal:Class rdf: about = "trg # book "/>
    </edoal:Class>
    </edoal: PropertyDomainRestriction >
    </edoal: Class>
    </edoal: Property>
    <edoal: or >
    <edoal: Property>
    </edoal: or>
    <edoal: Property>
    <entity1>
    <entity2>
    <relation> = </relation>
    <measure rdf: datatype = "http://www.w3.org/2001/XMLSchema # float"> >0.4 </measure>
  </Cell>
</map>
```

The search process includes three steps (Algorithm 1). The query's predicate is originally altered to be a triad with a predetermined meaning. The next stage treats these three elements as objects, and the third stage reflects them in the aim of mapping variables, predicates, and constants with certain qualities. As a result, the query is updated and output.

**Algorithms**: Create IAT for verdict writer statistics

1. Query Qin with mapping table MT as input
2. Query Qout as output
3: Using 1:1 mappings and GOut G Ain in place of IRIs in FILTER MT
4: Modifying the GOut Tripartite pattern (GOut, MT, predicate)
5: GOut Tripartite pattern revision (number 4) (GOut, MT, object)
6: New query containing GOut in Qout
3.7 Implementation of Wrapper

The system administrator has three alternatives available to him: either utilize the default wrapper, which supports different types; build or override the required subclass components; or attempt to implement using the abstract class from scratch. The code that is provided below demonstrates the wrapper's default behavior.

```java
1. Collection<DataContainer> dataContainers = new HashSet<>();
2. While (results. Next () ) {
3. DataContainer dataContainer = new DataContainer();
4. for (int i = 1; i <= resultsMetaData.getColumnCount(); i++) {
5. String column = resultsMetaData.getColumnLabel(i);
6. Object value = results.getObject(i);
7. dataContainer.addData(findMappedAttribute(column), value);
8. }
9. dataContainers.add(dataContainer);
10. }
11. Return dataContainers;
```

The system administrator in the current example uses queries and schema mappings to complete a specific task in compliance with user requirements.

4. Results and Discussions

4.1. Overview

For this research, it is essential to pick the best information integration technique from the options accessible, as well as to create a schema using XML and DTD. All of these techniques deliver insightful and convincing data. This shows how information integration techniques have helped us comprehend this study better. In summary, this chapter has set the stage for the in-depth debate in the next chapters. The Findings and Discussion

4.2 Syntactic heterogeneity

When different values are utilized to represent the same kind of information using the same data variables, this kind of heterogeneity can be seen at the data value level. We look at the data levels' heterogeneity in more detail and find a wide range of misalignments, including those affecting the data values, data values' language, and data kinds. The example in Figure 4.1 below exemplifies the aforementioned idea:

![Fig 4.1 Types of Syntactic Misalignments founds in heterogeneity](image)

4.3 Mediator and Heterogeneity

A mediator typically offers an integrated view over many views, as seen in figure 4.2 below.

![Fig: 4.2 Mediator-Wrapper Architecture](image)
In the aforementioned architecture, three elements client requests, wrappers, and data sources are coupled to one another to create data transactions. These data sources must be registered if heterogeneous data is to be retrieved from many sources. In the event that the requester and the service provider utilize different databases for the conceptualization of the domain, syntactic heterogeneity must be resolved after the issue at the data level has been identified. The mediator depends on the mapping in order to acquire integrated, read-only views from several data sources. [24]

Integrator Arbitration Tools (IATs) are required for this. Suppliers offered DBS and GFS; IATs are only concerned with reducing data heterogeneity from a number of data sources. The names of fields and the types of data are two instances of syntactic heterogeneity. The system receives user queries, which are then filtered based on the requirements of each individual user. An integrated view is provided or made available to the user as a logical perspective of the data sources, integrating the whole data at the source level and accessing it in line with user requirements for the purpose of retrieval.

The system that represents the functions of the applications is divided from the DBMS functions by this intermediary layer. So that the data may be merged, simplified, abstracted, and interpreted in order to fulfill the purpose of this intermediary layer [25]. The client's request for data kicks off the architecture process (Getting author names and IDs from publishing records is one example). The mediator, in this case IAT, receives the request from the requester and creates a query to get the needed data at the next level. Once the query and data set are connected, all fragmented data is merged into a single reactive data set and sent to the client.

The first component of the arbitration tool, known as the client, is where all requests are initialized, sent to the arbitration tool (IAT), and then it is determined where the data will be retrieved from in order to respond to the query.

![Fig 4.3 General View of Mediator System](image)

To provide the information, IAT pulls the data sources. A homogenous interface is suggested as a solution to the issue and to prevent any kind of conflict. The three-layer mediator architecture offered by [26] for the IMK mediator runtime environment is improved by this design [26]. The first layer of stored data, an interface through which they are connected, integration, which is handled by mediators, and finally, the data being delivered to the local data server in the application layer, are all described as the four functional layers that make up the runtime mediator environment.

### 4.4 Experimental Performance

Our proposed improved architecture retrieves the required data from several heterogeneous data sources and returns it. The vendors' data sources of choice were the domain and the conventional file system. The heterogeneity of the data rises when additional data sources are utilized. All registered data sources are reviewed and utilized in order to get a complete result or results based on a defined field. This extended architecture also functions efficiently in the online bookstore, which has numerous data sources and outdated file systems. Heterogeneity within the various data sources is a common occurrence when there are multiple data sources. The book title and the author's name are compared to return all of the author's published works.
First, as shown in Figure 4.4 above, in order to obtain the necessary data pertinent to author information, wrapper caching is switched off, and then lazy loading is applied to the results. To minimize duplications and replications, however, we have started the extraction of data from any online portal because web services only let four simultaneous requests. The performance of the author information integration in this example is related to running 10 queries to extract pertinent information.

The second example involved the system being integrated with a query that only permitted data to be taken from two sources, DATA1 and DATA2, with all other data sources being forbidden (see Figure 4.5 below). The query was executed ten times in order to achieve the desired performance.

To demonstrate the overall effectiveness of our suggested methodology while integrating data from various sources, we used the data that is shown in the table below.

<table>
<thead>
<tr>
<th>Cache</th>
<th>Lazy Loading</th>
<th>Type</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
</tr>
</thead>
<tbody>
<tr>
<td>No</td>
<td>No</td>
<td>S</td>
<td>517</td>
<td>630</td>
<td>655</td>
<td>732</td>
<td>255</td>
<td>229</td>
<td>294</td>
<td>329</td>
<td>374</td>
<td>403</td>
</tr>
<tr>
<td>No</td>
<td>No</td>
<td>P</td>
<td>570</td>
<td>1117</td>
<td>1150</td>
<td>1248</td>
<td>1280</td>
<td>1277</td>
<td>1359</td>
<td>1386</td>
<td>1390</td>
<td>1398</td>
</tr>
<tr>
<td>No</td>
<td>YES</td>
<td>S</td>
<td>329</td>
<td>316</td>
<td>292</td>
<td>302</td>
<td>329</td>
<td>319</td>
<td>301</td>
<td>302</td>
<td>294</td>
<td>345</td>
</tr>
<tr>
<td>No</td>
<td>YES</td>
<td>P</td>
<td>728</td>
<td>796</td>
<td>874</td>
<td>948</td>
<td>540</td>
<td>571</td>
<td>616</td>
<td>574</td>
<td>527</td>
<td>622</td>
</tr>
<tr>
<td>YES</td>
<td>YES</td>
<td>S</td>
<td>555</td>
<td>191</td>
<td>230</td>
<td>133</td>
<td>141</td>
<td>129</td>
<td>144</td>
<td>327</td>
<td>126</td>
<td>160</td>
</tr>
<tr>
<td>YES</td>
<td>YES</td>
<td>P</td>
<td>415</td>
<td>313</td>
<td>403</td>
<td>374</td>
<td>329</td>
<td>294</td>
<td>228</td>
<td>255</td>
<td>396</td>
<td>264</td>
</tr>
</tbody>
</table>

Table 4.1: Performance results
While conducting the suggested research, we established specific performance goals to demonstrate how lazy loading and caching influence the selection of different data sources and are crucial for system performance in general. Both were used ten times in different circumstances to obtain the findings. We must devote more effort to data extraction from various sources if we wish to have a greater impact on the information. According to the outcomes of the data collection and performance testing, we draw the conclusion that a particular set of data can be acquired from various sources utilizing a variety of techniques.

Conclusion
The purpose of this study was to analyze the issue of structural heterogeneity during the integration of various data sources and to offer solutions based on the body of literature acquired between 2011 and 2021.

We sought to address the following research question:

By utilizing the pattern basis technique, the mediator based information integration model might be enhanced in order to address the syntactic problems associated with the integration of heterogeneous information from web sources. The sources used for this study were IEEE, Science Direct, and the EndNote Desktop program. The papers that the search engines had retrieved were modified using the reference manager. Finding duplicate entries and retrieving whole manuscripts were both made possible by the EndNote Desktop software.

In order to carry out this study, sources were chosen, assessed based on the inclusion and exclusion criteria, and then studied and analyzed. There might be two different database servers that would each be used in a different way, as an example. The author ID and author name are both stored in a single database. Additional information on the author (including the author's journal name, publication name, and publication date) was simultaneously recorded in another database. The IMK makes it possible to register with a system that employs mediation without being attached to a database server connection that has already been created in order to obtain detailed information about the author's data. We must pay attention to some Big Data challenges that still need to be resolved in order to formalize with future work. Even with the reduction in the amount of human work required, data sources can still be insufficient and incompatible. However, as this study's findings show, a number of researchers are succeeding in overcoming some of these difficulties by utilizing parallel computing, Web Semantics, and AI ideas. It has also been noted that semantics is employed to raise the standard and dependability of various data objects. According to this study, algorithms are being used more frequently to analyze queries between multiple data sources and logs from heterogeneous data sources. Additionally, it was seen that there was increased interest in the capabilities for privacy, dependability, and OLAP in massive amounts of data.

Future research may focus on the aforementioned issues as well as the requirement to develop frameworks and semantic cubes in order to combine and manage data in a unified form.

Funding
This work is financially supported by the Higher Education Commission of Pakistan (Grant No: 7750/Federal/NRPU/R&D/HEC/2017).

Compliance with ethical standards
Conflict of Interests
The authors declare that there is no conflict of interest regarding the publication of this article.

Ethical Approval
This article does not contain any studies with human participants or animals performed by any of the authors.

Authorship Contributions
All authors contributed equally.

Data Availability Statements
Our manuscript has associated data.
References:


