iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 3 **Issue:** IV **Month of publication:** April 2015

DOI:

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)

Heterogeneous Database Integration Using XML

Prof. Y.R. Rochlani[#1], Neha M. Solio[#2], Kalyani R. Rawate[#3]
H.V.P.M. COET, SGBAU, Amravati (MH), India

Abstract---In this paper, based on the research of the existing heterogeneous database integration systems, according to the data exchange and sharing needs of enterprise heterogeneous databases, a framework for heterogeneous database integration system is proposed and designed, and the key technologies of the system implementation process are also described in detail. The system provides a heterogeneous data sharing and integration middle platform to achieve transparent operation and seamless integration of the heterogeneous data.
Keywords---Heterogeneous databases, XML, data integration

I. INTRODUCTION

Data integration is defined as the technique to integrate or collect data from different sources and merge them at one place and finally gives a virtual view to the users. Integration of multiple information systems aims at combining selected systems so that they form a unified new whole and give users the illusion of interacting with one single information system. Integration of information systems seems to be necessary in today's world to meet business and consumers needs. There are two reasons for integration: primarily, in a given set of information systems, an integrated view could be created to enable information access and reuse through a single access point. Secondarily, towards a particular information need, data from different information systems is accumulated to gain a more comprehensive basis towards the required need.

There are so many applications that are benefited from integration. In the area of Business Intelligence integrated information can be used for querying and reporting on business activities. In customer relationship management (CRM), integrated information on individual customers, business environment trends, and current sales can be used to improve customer. Enterprise Information Portals (EIP) present integrated company information as personalized web sites and represent single information access points primarily for employees, but also for customers, business partners, and the public. Lastly, in the area of E-Commerce and E-Business, an integrated information system acts as a facilitator as well as an enabler towards business transactions and services over computer networks. It is the process of combining data that is residing in different sources and providing users with a unified view of these data. The process involves standardization of data definition by using a common conceptual schema across a collection of data sources. Integrated data will be consistent and logically compatible in different systems or databases, and can use across time and users.

II. LITERATURE REVIEW

Data integration is relevant to a number of applications including enterprise information integration, medical information management, geographical information systems, and e-Commerce applications. Based on the architecture, there are two different kinds of systems: central data integration systems and peer-to-peer data integration systems . A central data integration system usually has a global schema, which provides the user with a uniform interface to access information stored in the data sources. In contrast, in a peer-to-peer data integration system, there are no global points of control on the data sources (or peers). Instead, any peer can accept user queries for the information distributed in the whole system. The two most important approaches for building a data integration system are Global-as-View and local-as-View. In the Global-as-View approach, every entity in the global schema is associated with a view over the source local schema. Therefore querying strategies are simple, but the evolution of the local source schemas is not easily supported. On the contrary, the local-as-View approach permits changes to source schemas without affecting the global schema, since the local schemas are defined as views over the global schema, but query processing can be complex. semantic approach integration is obtained by sharing a common ontology among the data sources. Semantic approach addresses not the structuring of the architecture of the integration, but how to resolve semantic conflicts between heterogeneous data sources.

III. PROBLEM DEFINITION

Integrating multiple information systems creates a unified virtual view to the user's imperative of the number of system or location of the actual stored data. The actual users are provided with a homogeneous logical view which is physically distributed over different heterogeneous data sources. For this, all data has to be represented using the same abstraction principles (unified global data model and unified semantics) [2]. Data integration is hard. The evidence is overwhelming. Every company we've talked to about their data has data integration problem. It's not just the IT people that moan about it either, it's IT users too and the company executives. Everywhere data is almost in a constant mess throughout. Today we have a dedicated sector of the industry devoted towards data integration solution; it generates about $3 billion in revenue and its growing space. Aside from that there are probably billions more spent on in house data integration efforts whether they employ the whiz data integration tools or not [3]. This task includes detection and resolution of schema and data conflicts regarding structure and semantics. In general, information systems are not designed for integration. Thus, whenever integrated access to different source systems is desired, the sources and their data that do not fit together have to be coalesced by additional adaptation and reconciliation functionality[1]. Note that there is not the one single integration problem. While the goal is always to provide a homogeneous, unified view on data from different sources.

IV. PROPOSED SYSTEM

Data integration is to provide a unified representation, storage and data management for various heterogeneous data environment, which is the basic function the heterogeneous data integration system must implement. Data integration shields the heterogeneity of the various heterogeneous data sources, and carries out unified operation to different data sources through heterogeneous data integration system. Therefore, the integrated heterogeneous data is unified for users.

The data forms involved in heterogeneous database are mainly structured data, semi-structured data and unstructured data three types. Structured data widely exists in a variety of information system database, the most common relational database. Semi-structured data commonly has Web pages as the chief representative, and XML can effectively manage and process such data. Unstructured data has common files, email and various documents. A practical information integration system should have intelligence, openness and initiative. Intelligence is to carry out unified processing, filtering, reduction, abstraction, integration and induction works for the structured, semi-structured and unstructured data from different databases [4].Openness is a heterogeneous and distributed database, which must solve the mismatching problem of the information expression with the structure. Initiative is to regulate the existing Internet data representation, exchange and service mechanism to provide proactive service mechanism.

XML is developed and established by the Internet organization W3C, the purpose is not only to meet the ever-growing network application needs, but also to ensure it have good reliability and interoperability in the alternation through the network. XML is a structured and semi-structured data markup language, to define a set of common format for the structured documents and data on Web pages and provide a way for the structured data to write a text file[5]. XML is a markup language independent of the system to express data information, has become a common data exchange format in network system. XML has been widely used in computer and network-related aspects.

V. IMPLEMENTATION

System aims user friendly mediation platform for the integration and provides user querying disparate heterogeneous information system. To implement both of the design modules we need two backend relational database servers and one frontend software application that can be connected to two or more backend database servers independently. For two backend database servers we selected two most popular and featured relational database servers

A. My-SQL Server and

B. MariaDB Server.

MySQL: In MySQL Server we created a database named "Drive" with two tables "Personal" and "DRIVING_LICENSE". Personal table contains personal information of the person with attributes FULLNAME (primary key),Age, Address, Mobile_No, Birth_Date and EmailID. LicenseDetails table contains Driving license information of the person with ID(primary key), NAME (foreign key), Village, District, and State, DATE_OF_ISSUE, EXPIRY_DATE, CITY, STATE.

MariaDB: The MariaDB RDBMS stores data logically in the form of tablespaces and physically in the form of data files ("datafiles"). Tablespaces can contain various types of memory segments, such as Data Segments, Index Segments, etc. Segments in turn comprise one or more extents. Extents comprise groups of contiguous data blocks. Data blocks forms the basic units of data storage. In MariaDB Server we created a database named “Mdb” with two tables “PERSONALDETAILS” and “ADHAR_DETAILS”.PERSONALDETAILS table contains personal information of the person with attributes NAME (primary key), CITY, PHONE. “ADHAR_DETAILS” table contains information of the person with attributes ADHAR_NO (primary key), NAME (foreign key), DATE_OF_ISSUE, GENDER. For implementation of Schema Integration module and Query Engine module, the frontend software application we selected is Eclipse Luna.

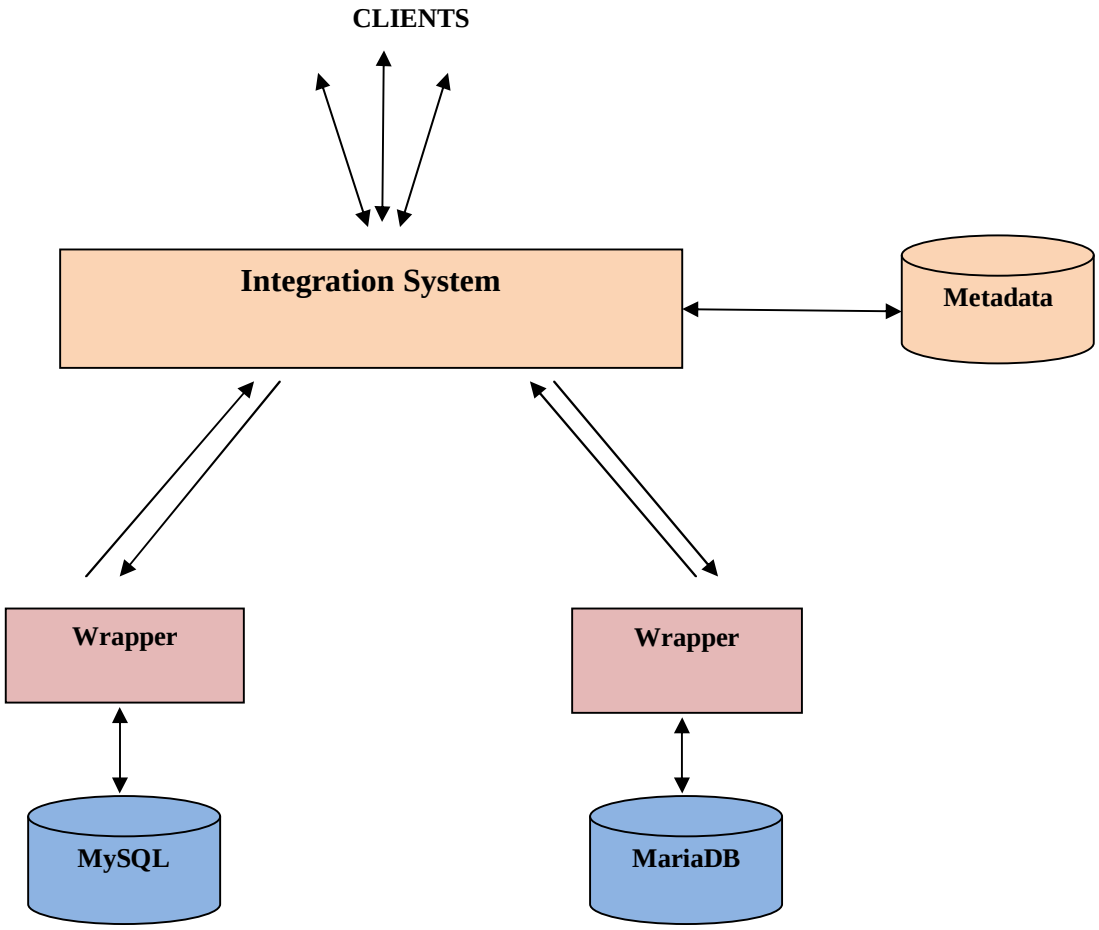


Fig. 1 Database Integration System Architecture

VI. KEY TECHNOLOGIES OF DATABASE INTEGRATION SYSTEM

A. JavaBean technology

JavaBean is a software component model to describe Java, somewhat similar to Microsoft COM component concept. In the Java model, the functions of the Java program can be infinitely expanded by JavaBean, and new applications can be rapidly generated through the JavaBean combination. JavaBean also can achieve code reuse, while has very great significance for the program maintenance. Through the Java virtual machine JavaBean can be run correctly. JavaBean provides for the Java component-based development system. And the query manager and data packager in this system are all the JavaBean components based on the Java

language.

B. Connection pool

Connection pool is a kind of entity which manages the connection as a resource, and a typical example of such resource is the database connection. The basic idea of the connection pool is to pre-establish some connections to store in the memory for use. To establish a database connection will consume considerable system resources, but once established, the query can be sent to obtain results through it. The number of queries a connection in its life cycle can process is not limit, so a database connection from a certain way is a resource. Using connection pool, when the program needs to establish a database connection, it only needs to take one from the memory to use instead of new.

Similarly, after use, simply to replace to the memory and the connection establishment and disconnection are both managed by the connection pool itself. At the same time, we can also through setting connection pool parameters to control the number of connections and the maximum use frequency of each connection. The use of connection pool will greatly enhance the process efficiency, and we can through its own management mechanism to monitor the quantity, use of the database connection. The connection pool technology allows the data packager efficiently, stably and reliably access to the database connection, to minimize the waste of data resources. Tomcat is the standard of the Java Servlet and JavaServer Pages technologies, is free software developed based on the Apache license. Tomcat application server itself comes with database connection pool features, so administrators can modify the appropriate values according to needs and the hardware configurations to achieve the best results. The more commonly used parameters such as maximum number of requests received, connection timeout, connection upload timeout, buffer, the maximum number of active connections, the minimum idle connection, and so on. Therefore, this paper directly uses the database connection pool functions of the Tomcat application server itself.

C. Data Extraction using XML

Typically, in schema-based systems (e.g., RDBMS), the description of data (or meta-data) is available, query-language syntax is known, and the type and format of results are well-defined and hence they can be retrieved programmatically (e.g., ODBC/JDBC connection to a database). However, in the case of web repositories, although a page can be retrieved based on a url (or filling forms in the case of hidden web), the output structure of data is neither pre-determined nor remains the same over extended periods of time. The extracted information needs to be parsed as HTML or XML data types (using the meta-data of the page) and interpreted. In the past, several systems such as Ariadne [7], TSIMMIS [6], InfoMaster [9],etc. had been designed for extraction of semi-structured and unstructured data within an associated domain.

However, the design of a comprehensive framework that provides a seamless extraction mechanism (for any type of data across any domain) in response to a user query continues to persist as a difficult challenge. Currently, wrappers [10] are typically employed for the extraction and integration of heterogeneous data. A wrapper is a program that is specific to every data source, and translates the source data to a form that the integration system's query processor can further process. Wrappers typically locate the web-pages that contain the desired information (based on appropriate parameters generated by the query plan) and extract the specific data from the page. Since the number of diverse data sources on the web continues to grow at a rapid rate, manual construction of wrappers proves to be an expensive task. There is a rapid need for developing automation tools that can design, develop and maintain wrappers effectively. Even though a number of integration systems have focussed on automated wrapper generation (Ariadne's Stalker [11], MetaQuerier [12], TSIMMIS [13], InfoMaster [8], and Tukwila [14]), the task of generating on-the-fly wrappers for extracting heterogeneous data from autonomous sources with minimum human intervention is complicated. The Information Manifold [15] prototype claimed that the problem of wrapping semi-structured sources would be irrelevant as XML will eliminate the need for wrapper construction tools. This is an optimistic assumption since there are some problems in querying semi-structured data that will not disappear, for several reasons:

1) Some data applications may not want to actively share their data with anyone whocan access their web-page,
2) Legacy web applications will continue to exist for many years to come, and
3) Within individual domains, XML will greatly simplify the access to sources; however, across diverse domains, it is highly unlikely that an agreement on the granularity for modeling the information will be established.

VII. CONCLUSION AND FUTURE WORK

The high complexity in integrating today's biomedical data has two root causes: the fact that the data are increasingly distributed and generated by collaborative environments, and the fact that they are processed, analyzed and visualized by increasingly more complex tools. Our system provides a heterogeneous data sharing and integration middle platform to achieve transparent operation and seamless integration of the heterogeneous data, allowing users to more easily publish data to the Internet/Intranet, to provide a technical basis for users' heterogeneous data sources at a higher level.

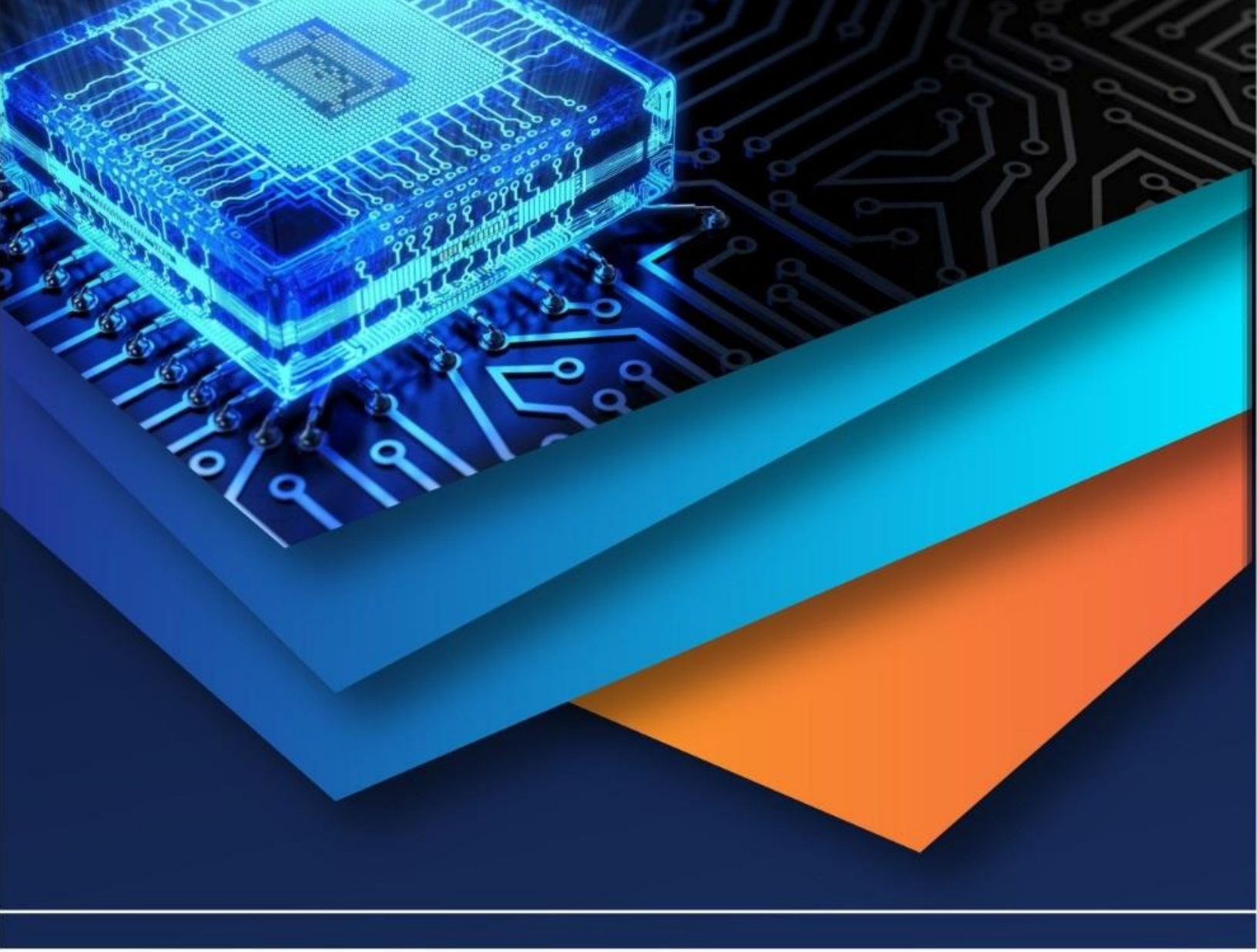
XML, an evolving Web technology, is poised to help the task of data integration and reduce the work of reconciling heterogeneous data sources. There are efforts going on in bringing out a language that can specify the semantics associated with data content. Also the development of suitable tools for XML-based integration of heterogeneous sources is also steadily going on.

VIII. ACKNOWLEDGMENT

We would like to thank Prof Y.R. Rochlani, for his extended support and encouragement for carrying out this work. We wish special thanks to our department's faculty members who helped us to carry this work.

REFERENCES

[1]. Patrick Ziegler, Klaus R.Dittrich, *Data Integration - Problems, Approaches, and Perspectives*, Database Technology Research Group, Department of Informatics, University of Zurich.
[2]. NesimeTatbul, Streaming Data Integration: Challenges and Opportunities, ETH Zurich, Switzerland.
[3]. Cohen, W.W.1998a. Integration of heterogeneous databases without common domains,Using queries based on textual similarit. ProceedingsofACMSIGMOD-98 (Seattle,WA,1998).
[4]. Fernandez M F, Simeon J, Wadler P.A Semi-monad forSemi-structured Data[C], In Proc. of the 8th International Conference on Database Theory, London, UK,2001:263-300.
[5]. Jiang H F, Lu H J, Wang W, et al. XR-Tree: Indexing XML Data for Efficient Structural Joins[C].The19th International Conference on Data Engineering, 2003:56-60.
[6]. S. S. Chawathe, H. Garcia-Molina, J. Hammer, K. Ireland, Y. Papakonstantinou, J. D. Ullman, and J. Widom, "The TSIMMIS Project: Integration of Heterogeneous Information Sources." in IPSJ, 1994, pp. 7–18.
[7]. C. A. Knoblock, S. Minton, J. L. Ambite, N. Ashish, I. Muslea, A. Philpot, and S. Tejada, "The Ariadne Approach to Web-Based Information Integration."Int. J. Cooperative Inf. Syst., vol. 10, no. 1-2, pp. 145–169, 2001.
[8]. O. M. Duschka and M. R. Genesereth, "Query Planning in Infomaster." in Selected Areas in Cryptography, 1997, pp. 109–111.
[9]. M. R. Genesereth, A. M. Keller, and O. M. Duschka, "Infomaster: An Information Integration System." inSIGMOD Conference, 1997, pp. 539–542.
[10] T. Kabisch and M. Neiling, "Wrapping of Web Sources with restricted Query Interfaces by Query Tunneling."Electronic Notes on Theoretical Computer Science, vol. 150, no. 2, pp. 55–70, 2006.22
[11] I. Muslea, S. Minton, and C. A. Knoblock, "Hierarchical Wrapper Induction for Semistructured Information sources." in Autonomous Agents and Multi-Agent Systems, 2001.
[12] K. C.-C. Chang, B. He, and Z. Zhang, "Toward Large Scale Integration: Building aMetaQuerier over Databases on the Web." in CIDR, 2005, pp. 44–55.
[13] J. Hammer, H. Garcia-Molina, S. Nestorov, R. Yerneni, M. Breunig, and V. Vassalos, "Templatebased wrappers in the TSIMMIS system." in SIGMOD Conference, 1997, pp. 532–535.
[14] Z. G. Ives, D. Florescu, M. Friedman, A. Levy, and D. S. Weld, "Adaptive Query Processing for Internet Applications." in IEEE Computer Society Technical Committee on Data Engineering, 1999, pp. 19–26.
[15] A. Y. Levy, "Information Manifold Approach to Data Integration." IEEE Intelligent Systems, pp. 1312–1316, 1998.



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY