Virtual Data Integration of Heterogeneous Genomic Biological Knowledge Base

Muhammad Shahzad, Kamran Ahsan and Muhammad Azhar Hussain

***Abstract* – In the world of Human Genome Project, massive biological information has been exponentially growing and generating with the passage of time. To grasp and hold this huge genomic information, more than hundreds of special kinds of biological databases have been developed. Moreover these biological data sources that contain experimental data are in different forms and it is also growing in volume. These biological data sources become more valuable and meaningful for scientific analysis when it integrates with other related biological data sources. There is no any single database that provides complete picture of any datum. The biggest challenge and important issue in the science of integration is to establish a unified environment for bioinformatics databases. This paper has proposed new approach in distributed virtual data integration for heterogeneous genomic biological databases. The novelty in our model is the use of Mediator Server based approach that has some distinct advantages over the conventional distributed approach. These advantages have been also stated in this paper. Proposed integration methodology will serve as a driving force for scientists and researchers to investigate new biological standards and theories in the science of bioinformatics**.

***Index Terms* – Bioinformatics, data integration, biological databases, mediator-based integration, global schema approach, genomics.**

I. INTRODUCTION

In modern era, lot of milestones has been achieved in the world of human genome project. The effort in this project is not only to decode the human beings but also produce the biography which is another important interest in the field of biographical science. It is enormously needed matter for the learning of human genomics to understand and analyze this knowledge and it also helps the bioinformaticians for the development and production of the bioinformatics.

To accommodate large data of genomic experiments, many bioinformatics databases have been developed. Amongst these hundreds of databases, three main international sources of nucleic acid databases are: DDJB [1] (DNA Data Base of Japan), Genebank [2] and the EMBL [3] (European Molecular Biology Laboratory) database. All these databases are synchronized with each other on regular basis and accession number of each entry are also consistent. On a broad range of bioinformatics databases, there are 4 types of categories for the databases of bioinformatics: such as genome databases including many species genome and as well as single model organism genomes like ensemble genomes [4], MGI mouse genome [5], etc., primary structure database of protein and nucleic acid sequences, 3D protein structure databases and the databases of secondary class which contains the information and documentation of 3 above aforesaid categories of bioinformatic databases [6]. All above mentioned databases have their own targets by which they perform data collection and collation of biological experiments. There is also some sort of relevant data processing and data query services.

Currently, National Center for Biotechnology Information (NCBI) provides data analysis and data query services to their users. They developed their own private database query system such as Entrez for Genbank. SRS [39] system was developed by European Molecular Biology Laboratory (EMBL) for the same perspective. The central point is to establish common query environment for the heterogeneous databases users. Due to the varieties of data sources available, therefore data integration has been one of the challenging subjects for couple of decades. Many of these data sources are developed on the standards of database structure with well-defined query interface like relational database and some support object oriented database upto certain extent. Other data sources are some kind of exchangeable and interchangeable formats with some limitation and restriction like Entrez and ACeDB. And some data sources are in the form of flat files with specific formal structure (e.g. flat file of Genebank and ASN.1 exchange data format) from which data can be retrieved with parsing technique. To provide unified environment, numerous data integration techniques have been proposing from the community of bioinformatics. This paper presents virtual data integration approach to cope the integration issues of heterogeneous data sources.

This paper includes five sections to address the data integration in the heterogeneous bioinformatics sources: First, it will highlight the different data integration approaches with their pros and cons that are currently in used. Secondly, out proposed approach will be discussed with the comparison of existing approach and methods. At the third section, architecture of our proposed model will be elaborated. In fourth section is about discussion in nutshell. Finally, the paper will conclude however with a future work.

II. DATA INTEGRATION APPROACHES IN BIOINFORMATICS

There are different approaches and classification of data integration. Usually integration approaches depends on the

Muhammad Shahzad - National University of Computer and Emerging Sciences, Email: mshahzad@nu.edu.pk, Azhar Hussain ,Dr. Kamran Ahsan, Assistant Professor , Federal Urdu University of Arts, Science & Technology, University Email: kamran.ahsan@fuuast.edu.pk. Manuscript received July 27, 2015; revised on August 30 and November 16, 2015; accepted on December 31, 2015.

 ISSN – 1999-4974

data model as discussed in research work of Köhler [7], Stein [8], and Davidson [9]. General classification of integration data model is of three types like text based, structured data, and linked data. In case of a system's sources viewed as exporting text, integration system mush have capability of providing text and keyword searching amongst different data sources. For structured data model system, two further sub classification of integration are being incorporated, one of which is provide the warehoused approach of data from different sources and other one provides access of data sources on demand. The third approach helps in effective navigation services in the linked data environment like set of browsing records.

In Materializing and integration in data warehouse approach, it creates new local warehouse which contains data from difference sources. This type of system provide query interpreter at the warehouse level instead of individual actual sources. There is a data translation mechanism across different sources in warehousing which require some standard format into which data is translated from multiple sources [10]. As in warehousing, all the data is at one location, therefore less access of network resources eliminates the problems like low response time, unavailable of resources, bottleneck situation in network, and moreover warehousing also provide efficient query optimization [11,12]. The major disadvantages of materialized warehouse are the reliability of result and the cost for the maintenance of the updated or new result from the multiple sources [12]. Genomics Unified Schema (GUS) [13, 14] is the example system of materialized warehouse. This system allows their user to filter and annotate the retrieve data as per their own requirement.

Another approach is navigation integration, which refers to the linking capabilities of the data available on the web sources. This linked-based integration provides the knowledge to the users about the browsing paths which lead through the pages and data sources in order to reach the target for the finding of information. Browsing web page and data source makes a workflow in which the output of one source would be an input for another sources or tools [15]. Each query in this kind of integration, are transformed into path expressions which infect excludes relational data modeling [16, 17]. BioNavigator [33], Entrez [34] and many other systems falls in this class of integration. In BioNavigator, it allows the users to describe their own query execution paths and provide the provision to reuse these paths later on for the same set of query [35, 36]. Whereas Entrez is a search engine at NCBI (National Center for Biotechnology Information) website and it allows users to execute their query across multiple sources transparently.

Integration based on mediated approach is about creating virtual integration environment system for the multiple sources. The data stays at the sources, but it creates an illusion on user of being working on real database. In this kind of system it provides common query interface for multiple heterogeneous data sources. Mediator, a software component is responsible to receive query form the user in one language and translate or reformulate into source based schema. This mapping mechanism between users provided query and source-based query schema is categorized as GAV (global-as- view) and LAV (local-as-view) [11, 17]. The GAV based mediator is designed to entertain user query with respect to source type schema. Whereas LAV based mediator supports user request with respect to global schema representation.

There are many systems that have been developed on mediator-wrapper based approach. The TSIMMIS (The Stanford-IBM Manager of Multiple Information Sources) is one of them. In this system, it integrates information of heterogeneous type including structured and semi-structured data [31]. TSIMMIS architecture has shown in Fig.1. The wrapper in this Fig. 1 translates the data object into unified model of information. It also performs conversion of queries into some common model so that it can be executed. The wrapper component is also responsible to take care about the response from the data source and converts this result back into common model.

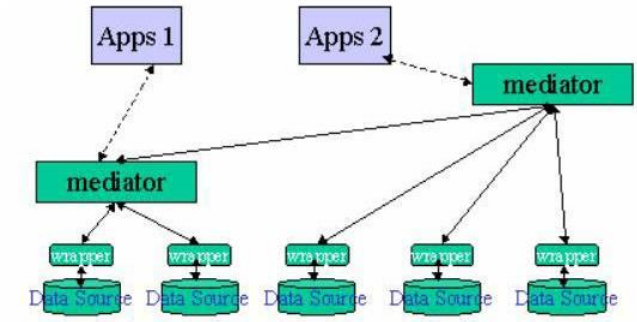

Fig.1 Architecture of TSIMMIS [31]

In another mediator based architecture system in the bioinformatics community is TAMBIS (Transparent access to Multiple Bioinformatics Information Sources) [32]. TAMBIS queries are designed like a graphical interface in which users can browse their needs through concepts in the global schema and choose it as per their own interest. User can express their requirement through graphical GRAIL system which is based on source independent logic description, and then it is further translated into source specific query execution plan's format using CPL. TAMBIS use CPL because it has huge available built-in libraries relevant to bioinformatics sources. Through these built-in capabilities of CPL, TAMBIS can handle complex data types related to bioinformatics concepts. To access the underlying data sources, TAMBIS use external wrapper of BioKleisli system. Fig. 2 shows the architecture of TAMBIS.

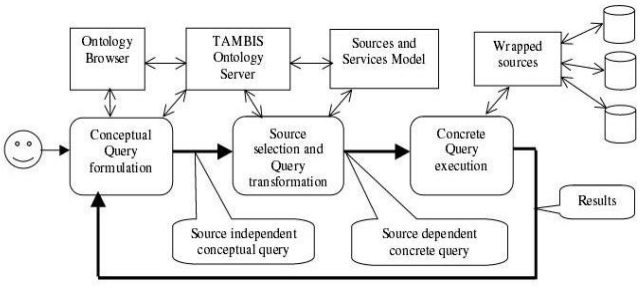

Fig.2 Architecture of TAMBIS [32]

ISSN – 1999-4974

The most important thing of the TAMBIS is its use of ontologies unlike the other system. TAMBIS ontology servers help in the biological concepts classification. Sources and services models provide the mapping between the ontology concept and CPL functions. This mapping definitely helps in query formulation task of the users. Moreover it provides simple interface for the user to pose the query without any trouble and as well as no worry about navigating sources.

III. VIRTUAL DATA INTEGRATION WITH MEDIATOR APPROACH

In VDI the main component is mediator, which is responsible to provide effective reception for back and forth movement of data from multiple sources to user query interface and vice versa. Effective mapping between source schema and data source schema has been a challenging problem for a long time for the data integration community. Inside the mediator, generally there are three sub-module i.e. reformulation engine, plan generator and execution engine. Due to different schemas at VDI (virtual database) and real database level, reformulation engine rewrite the user provided query into source specific environment. Three famous mechanisms for query reformulation are GAV (global-as-view), LAV (local- as-view) and GLAV (global-and-local-as-view) [18]. Whereas plan generator define the rules that how results from the multiple sources can be combine. And final execution engine execute the plans as per defined by the plan generator. In some literature, data source catalogue (DSC) has also recommended in VDI. DSC contains the meta-information about source capabilities, reliability of the sources, mirror sources, etc. Furthermore this mediator component is communicated with wrapper tool which is responsible for exporting data into source format. Some VDI implementer place wrapper program as the part of the mediator and some place it at the source. Figure 3 is showing general architecture of VDI.

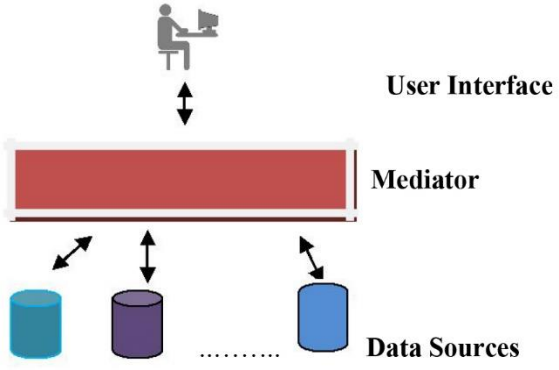

Fig.3 General Architecture of VDI

However the major goal of the architecture of mediator was distributed modules of the software which transparently translate user query into source specific format and share the abstraction of the data to the application [19]. Keeping this in view, our proposed model for VDI is based on distributed mediator system. In this system, several peer mediators have used with one or more wrappers to entertain user query to process data from numerous kind of data sources. The goal of choosing multiple peer mediators is load balancing while processing request. Each of these mediators has their own reformulation engine, plan generator and execution engine. This will definitely reduce the overheads on the mediator and moreover it overall process become speedy.

To accommodate all the requirements of the mediator smoothly, each mediator has to provide mapping services from global schema to source schema for the one type of source. E.g. one mediator is associated to access relational database using ODBC [20, 21], one is reserved to provide access to XML files [22], another mediator offers access to internet search system [23] and one can be attached with CAD system [24]. Consistency across the data in multiple sources can be achieved in this model with allowing inter-communication between peer mediators through some network protocol.

IV. ADVANTAGES OF MEDIATOR OVER OTHER APPROACHES

As discussed earlier that there are three basic classifications in the field of data integration such as link-based integration, warehouse (structured data) based integration and mediator based (virtual data) integration. Warehouse and link-based integration have been using for many purposes. Genomic Unified Schemas (GUS) [14, 37] is one of the most famous systems that based on the warehouse approach. In Warehouse integration, as it materializes or unifies the multiple sources' data into single local warehouse, therefore it comprises of the benefits like no problem in response time, data unavailability, and no network bottleneck. But still it has strong drawback in term of data reliability because of outdated results due to no synchronization between sources and warehouse database after loading data from sources [38]. Moreover it is not always possible or practical to accommodate multiple sources data into single data source due to the large volume set. The only solution to this problem is the mediator approach because in this approach whenever user posts the queries, result would always be fetched from the actual sources.

In navigational (link) based approach, the authors in [37] has pointed out that this approach is not as much appropriate as other integration approached due to the lack of querying functionality. Moreover it cannot ensure about the availability at every time of the data at the sources. BioNavigator [36], SRS [39] and Entrez [34] are some of the examples of the link based integration systems. Whereas in mediator approach, user writes the well generic query and then it translates into source specific format. So in mediator based integration, it always guaranteed the data availability at every time.

In summary mediator approach presents the virtual unified database environment of multiple data sources in which all transaction related to data transformation and accessing are entertained on the fly. Therefore it comprises of following benefits over other approaches:

ISSN – 1999-4974

- No overheads of storage and update operation
- Always retrieves the most current information from the actual data sources when query the system.
- Only small amount of relevant data is retrieved. Unwanted data is discarded.
- It provides the secure and limited access to actual sources data.

In addition to all above, integration performance will be increased by using mediator based approach due to its capabilities of transparently materialize the data sources.

V. DISTRIBUTED MEDIATOR APPROACH

The last section presented the mediator approach and depicts the behavior that it bridges the user application and database layer. Many mediator systems have been designed in a manner that all the mediation functionality has adopted into a centralized system. However this kind of single centralized mediator system does not acquire the functionalities like atomicity, decentralization, flexibility, scalable integration, etc. [40]. In addition to this single mediator, it is expected that to integrate large number of different types of sources and to understand multiple knowledge domains, is very complex and difficult to maintain. This reason derived the integration community to present distributed mediated system.

In [41], authors stated that in the mediator system with distributed mediators, all mediators are specialized in particular domain knowledge and associated with particular subset of all sources. This kind of system refers to Peer-to- Peer (P2P) mediator system. This peer architecture covers most of advantages that could be care by the single centralized mediator system. Amongst all, dynamic availability is one of typical benefits that ensures the in case network problem or if any of the mediator fails to respond the query then other mediator can join react the request without any disruption in the whole operations. Furthermore all these P2P mediator approach are autonomous, decentralized, and flexible and have provision for the scalable integration. But with all aforesaid features of the P2P system, still it needs to be more efficient in term performance in overall operations of integration system. In the coming up session, some of the P2P mediator systems have been discussed with their shortcoming so that it leads to our proposed model.

VI. SHORTCOMINGS OF DISTRIBUTED MEDIATOR APPROACH

There are many projects that lie in the category of distributed mediator architecture such as TSIMMIS [31] and Peagasus [42], in which all mediators communicated with each other but no remarkable results are appeared in the field of integration. The AURORA [43] project presents 2-tier model of mediation with distributed components of three types. Fig.4 shows the basic architecture of the AURORA projects, in which at the first tier of this model, heterogeneous sources are combined by homogenized through mediators. Each source is connected with wrapper module and then followed by homogenization mediators. At the second tier, all homogenized mediators are integrated by multiple integration mediators. In fact there was no single uniform integrated view of the system but integration has been performed by many mediators. In contrast to this AURORA project, our proposed system reduces the communication overheads between multiple mediators at different tier. In our system, all mediators only need to send and receive data to and from the mediator server.

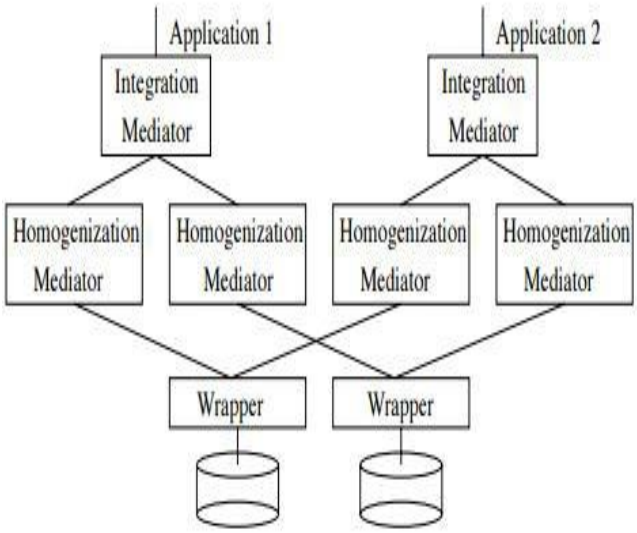

Fig.4 Architecture of AURORA [43]

Another project is DIOM (distributed interoperable object model) [43, 44, 45], which is based on the scalable integration of multiple sources with composability property. This project is one of that, which has implemented distributed mediator architecture in which each of the mediator access other mediators and/or as well as wrappers. One of most typical feature of DIOM is the automatically selection of the sources on the basis of user preferences. Query conflicts have been also automatically resolved with the help of preferences provided by the users. But all type of query processing including controlling and compilation has been performed centrally. Therefore in the DIOM system, no specific and clear differences between the Mediators and the data sources has presented in the framework of the optimization [46].

The Distributed Information Search Component (DISCO) is another mediator system [47], in which different distributed mediators can access distributed wrappers as shown below in Fig. 5. To find unavailable sources is one of most significant feature of this system (DISCO). This novel feature for the graceful handling of the unavailable data sources has achieved with the new semantics for the evaluation of query. In case of source unavailability, DISCO processes the query partially and then returns it to the application so that it can provide partial result to the posted user query. But still in distributed architecture of DISCO, there is a performance issue because of extensive network communication cost amongst different mediators and other

ISSN – 1999-4974

components during the query processing. In our proposed model, we resolve this issue, which is our one of the most prominent feature.

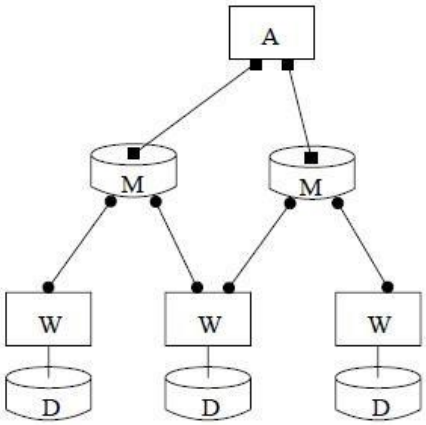

Fig.5 Architecture of DISCO [46]

Our conclusion to this is that many distributed mediator systems have been in used but none of these systems reports issues related to effective communication between mediators, which is our one of the primary objectives of our research work. Though, our proposed work will also be suitable for the scalability in data integration but this is our additional work.

VII. ARCHITECTURE OF PROPOSED VDI WITH SERVER BASED APPROACH

Fig. 6 shows a schematic description of our proposed architecture of VDI based on distributed multiple mediators' architecture. Users have an illusion of real databases created by mediator component. When user query a data in the form of global schema through user interface, then Mediator server receive the request and broadcast it to all mediators. Mediator sever is responsible for registering name of the sub-mediator when new mediator is added and it also contains the information about the location of the mediator and other relevant data of sub-mediator. Mediator server also provides the inter-communication between peer mediators for achieving consistency and sharing or exchanging of data. The schema mechanism of mediator server is GAV (global-as-view), in which it receives the query from the users. By reading user query, it then transfers it to the entire mediators peer. The main role of the mediator server is to provide single functional view of the data to the users. The most prominent feature of our model is the union operation, which is applied by the mediator server, when it receives the result back from the mediators peer. The mediator server receives the result in the form of global schema because it helps it to present the result as per user's format.

Each mediator peer has their own reformulation engine, plan generator and execution engine. Every peer mediator is linked with one wrapper module, which contains the knowledge of query translation from global schema format to source specific schema [20]. A wrapper has a capability of query processing and query translation from a specific type of data sources available at external region. It works like interfaces for the external data sources e.g. PDB [25], GO [26], MGI [27], etc. and contains information about local schema definition and data. Wrapper module has also knowledge about query rewrite rules through which it efficiently processes the query and translates it into particular type of external data sources [28].

The proposed architecture has also capability for extending more external sources. For integration with the new instance, the mediator server must define the view for all such kind of external sources. Once the view for new instance of mediator has been defined, then the system can smoothly process any user query. Moreover, data source catalogue is associated with each of the peer mediator and it provides all meta-information that are required for processing the request. More precisely, the proposed system has capability to effectively provide the virtual data integration for the bioinformatics data sources.

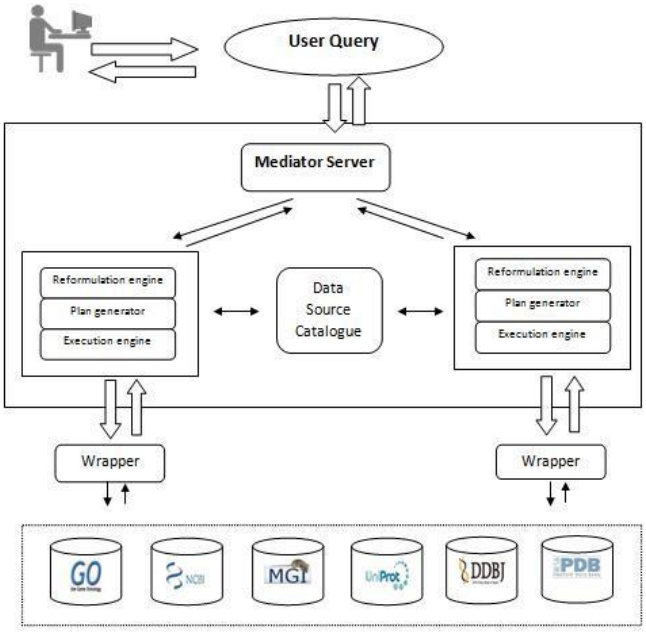

Fig.6 Proposed Architecture of VDI

VIII. INSIGHT OF THE PROPOSED VDI MODEL

The architecture which this research addresses can easily be understood by an example. Let suppose, we want to assemble data set from the three different types of sources i.e. S1, S2 and S4. These sources contain the information about different protein attributes based on sequences of proteins. Table 1, Table2 and Table 3 shows the three relations (tables) corresponding to the sources S1, S2 & S3 respectively. The first two sources i.e. S1 and S2 are complementary and the third one i.e. S3 is different but related to S1 & S2. At the mediator level, relations in the global schema are treated as views of the union of local source's relations. Because this paper only consider GAV (Global-as-view) schema for our mediators, therefore following example is

ISSN – 1999-4974

showing the global relation that contains the information about protein sequence.

ProtSeq (ID, Sequence) ← *S1 (ID, Name, Sequence, Motifs, ECNumber)*
ProtSeq(ID, Sequence) ← *S2 (ID, Name, Sequence, Motifs, ECNumber)*

The above two global relation are disjunctive view, in other words these relation are disjunctive operation on a conjunctive queries. Data source catalogue defines these queries in term of union operation of the source's projection S1 and S2, i.e.

$$ProtSeq := ID, Sequence(S1) \cup ID, Sequence(S2)$$

Another example for global relation if we want protein ID, Name and Species:

ProtSpec (ID, Name, Species) ← *S1 (ID, Name, Sequence, Motifs, ECNumber), S3 (ID, Name, Species, Functions)*

In the above view, we have defined firstly joint query on sources S1 & S3 over the attribute *ID* and then a projection on ID, Name & Species.

```
SELECT S1.ID, Name, Species
FROM S1, S3
WHERE S1.ID = S3.ID;
```

It can be written as in relation algebra:

$$ProtSpec(ID, Name, Species) := ID, Name, Species(S1 \infty ID\ S2)$$

TABLE 1. PROTIEN INFORMATION IN S1

Data Source	ID	Name	Sequence	Motifs	EC Number
S1	P07363	Chemotaxis protein CheA	MSMDISDFYQ TFFDEADELL ADMEQHLLVL	Contains # HPt domain	EC:2.7.13.3 Chemotaxis protein CheA
	P35626	Beta-adrenergic receptor kinase 2	MADLEAVLAD VSYLMAMEKS	RGS PROT_ KIN_ DOM PH_DOMAIN	2.7.1.126 Beta-adrenergic receptor kinase

TABLE 2. PROTIEN INFORMATION IN S2

Data Source	ID	Name	Sequence	Motifs	EC Number
S1	P07363	Chemotaxis protein CheA	MSMDISDFYQ TFFDEADELL ADMEQHLLVL	Contains # HPt domain	EC:2.7.13.3 Chemotaxis protein CheA
	P35626	Beta-adrenergic receptor kinase 2	MADLEAVLAD VSYLMAMEKS	RGS PROT_ KIN_ DOM PH_DOMAIN	2.7.1.126 Beta-adrenergic receptor kinase

TABLE 3. PROTIEN INFORMATION IN S3

Data Source	ID	Name	Species	Functions
S2	P07363	Protein kinase CheY/CheA complex (bacteria chemotaxis)	Escherichia coli (bacteria)	Kinase, Transferase
	3C7N	Heat shock protein Hsc70/Hsp110 complex	Bovine/Yeast	adenyl-nucleotide exchange factor activity

At this stage, when user poses the query and mediator server broadcast it to all mediators peer. The reformulation engine of each mediator translates the user query into global schema format and forwards it to plan generator. The most typical and complex part of the VDI system is the plan generator, in which it identifies the data sources which it has information. Moreover plan generator component of VDI is also responsible to determine the relevant sub-queries so that to send it to its associated wrapper. Each of the mediator peer only take care about the collection of answers from the particular source. But the plan generator just a plan, it needs to be executed. For this, execution engine takes the plan and forward it to the concern or relevant wrapper. After that execution engine waits for response from local sources and when it gets the results then it forward back it to Mediator server. As all mediator peers are working on partial query, therefore it is not possible to compose cumulative and complete results from one mediator. Mediator Server now takes in action and receives the answers from all mediators and finally performs union operation on the result sets. With this union operation of mediator server, it is possible to produce complete result.

The data source catalogue contains the rules which tells that how values of required relations would be compute. E.g. from the above relational sources like S1, S2, and S3 which contains the relations, if we wish to compute protein function, then its view definition is declared as:

ProtFunc(ID, Name, Function) ← *S1(ID, Name, Sequence, Motifs, ECNumber), S3(ID, Name, Species, Functions)*

The above global schema is defined in term of conjunctive query on the local relations. To access the attributes of base relations in the view i.e. *ProtFunc[D]*, we only need to create an instance of the base schema *D*.

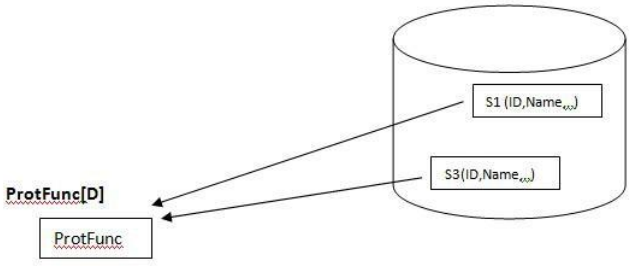

Fig. 7 shows the instance of the base relation

IX. DISCUSSION

In this proposed model, the generic architecture of virtual data integration for bioinformatics databases has defined. Earlier systems have been in use for couple of decades amongst integration community but still it needs to be effective and more efficient in case of heterogeneous data sources. The main goal of such kind of systems is to establish an environment for the people like biologist and bioinformaticians in which they can acquire knowledge from the bulk of heterogeneous data sources, create their hypothesis and then test it their reliabilities. In other words, make maximum tasks to be automated for the community of scientists and researchers. Furthermore, it will reduce the amount of time to process the data and such type

ISSN – 1999-4974

of system in fact helps to achieve goals in minimum interaction. This leads to express that computational system must be enough flexible and reliable to the users.

Additionally, due to varieties of biological data sources, the imperious effort is that, system must automatically adopt the capabilities of source format and its representation. As these days extracting of source description or source schema is doing manually by two different experts from the field of integration and biology. Therefore such kind of VDI solution will definitely ease the generation of the biological results and also it helps to reduce the overall processing cost and time. For successful and well-organized refinement of query plans and execution in the integration system, the source statistics of the source description must be gathered [29, 30]. More precisely, other than important statistics, the ideal system must consider the things like the response time on the average, response time of dependent query, intersection between multiple sources, and other numerous quality or essential statistics related to data density and data freshness.

X. CONCLUSION

This paper has explored different solution like warehousing technique, linked based integration and mediated based integration with their pros and cons. Finally this proposed approach has presented new model in the mediated based integration. The new proposed architecture take cares the entire shortcoming in the previous integration mechanism. With this proposed integration methodology, it can be driving force for scientists and researchers to investigate new biological standards and theories in the science of bioinformatics.

To conclude, this paper proposed new mediator based integration model. There are still so many issues to be explored. Opportunities are enough in this arena for some groundbreaking contribution and bring significant development in the industry. Future challenge is to implement this model which definitely helps in comparative study of the different integration model.

ACKNOWLEDGMENT

The authors would like to thank all the faculty members of the Department of Computer Science, – Federal Urdu University and Dr. Kamran Azim (International Center for Chemical and Biological Sciences, HEJ University of Karachi), for their technical guidelines for this research.

REFERENCES

[1] Miyazaki S., Sugawara, H., Gojobori, T., Tateno, Y (2003). DNA Data Bank of Japan (DDBJ) in XML, Nucleic Acids Res, 31, p.13–1656

[2] Dennis A. Benson, Ilene Karsch-Mizrachi, David J. Lipman, James Ostell, David L. (2005,). Wheeler, "GenBank," Nucleic Acids Res (Database issue), p. 34-38.

[3] Kulikova T., Aldebert, P., Althorpe, N., Baker, W., Bates, K., Browne, P., van den Broek, A., Cochrane, G., Duggan, K., Eberhardt, R. et al, (2004,). The EMBL Nucleotide Sequence Database, Nucleic Acids Res, 32, p.27–30.2

[4] Kersey, P. J.; Staines, D. M.; Lawson, D.; Kulesha, E.; Derwent, P.; Humphrey, J. C.; Hughes, D. S. T.; Keenan, S.; Kerhornou, A.; Koscielny, G.; Langridge, N.; McDowall, M. D.; Megy, K.; Maheswari, U.; Nuhn, M.; Paulini, M.; Pedro, H.; Toneva, I.; Wilson, D.; Yates, A.; Birney, E. (2011). "Ensembl Genomes: An integrative resource for genome-scale data from non-vertebrate species". Nucleic Acids Research 40 (Database issue): D91– D97. doi:10.1093/nar/gkr895. PMC 3245118. PMID 2206744 7

[5] Bult CJ, Kadin JA, Richardson JE, Blake JA, Eppig JT; Mouse Genome Database Group. The Mouse Genome Database: enhancements and updates. Nucl. Acids Res. 2010 Jan;38(Database issue):D586-92

[6] Yuelan Liu, Jian hua Wang, "Data Integration of Bioinformatics Database Based on Web Services", International Journal of Web Applications Volume 1 Number 3 September 2009.

[7] Köhler J, Philippi S, Lange M. SEMEDA: ontology based semantic integration of biological databases. Bioinformatics 2003;19:2420–7.

[8] Stein LD. Integrating biological databases. Nat Rev Genet 2003; 4:337–45.

[9] Davidson SB, Overton C, Buneman P. Challenges in integrating biological data sources. J Comput Biol 1995;2:557–72.

[10] W. Sujansky. Heterogeneous Database Integration in Biomedecine. Methodological Review, Journal of Biomedical Informatics, 34, 285-298, 2001.

[11] D. Florescu, A.Y. Levy, and A.O. Mendelzon. Database Techniques for the World-WideWeb: A Survey. ACMSIGMOD Record, 27(3), 59-74, 1998.

[12] S. Davidson, C. Overton and P. Buneman. Challenges in Integrating Biological Data Sources. Journal of Computational Biology. Vol 2, No 4, 1995.

[13] S. Davidson, J. Crabtree, B. Brunk, J. Schug, V. Tannen, C. Overton and C. Stoeckert. K2/Kleisli and GUS: Experiments in Integrated Access to Genomic Data Sources. IBM Systems Journal, 40(2), 512-531, 2001.

[14] The GUS Platform for Functional Genomics. http://www.gusdb.org, 2003.

[15] David Buttler, Matthew Coleman1, Terence Critchlow, Renato Fileto, Wei Han, Ling Liu, Calton Pu, Daniel Rocco, Li Xiong. Querying Multiple Bioinformatics Data Sources: Can Semantic Web Research Help? ACM SIGMOD Record, 31(4), 2002.

[16] P. Mork, A. Halevy, and P. Tarczy-Hornoch. A Model for Data Integration Systems of Biomedical Data Applied to Online Genetic Databases. In Proceedings of the Symposium of the American Medical Informatics Association, 2001.

[17] M. Lenzerini. Data Integration: A Theoretical Perspective. ACM Symposium on Principles of Database Systems (PODS), 2002.

[18] Leopoldo Bertossi, Virtual Data Integration Tutorial, School of Computer Science - Carleton University, Ottawa, Canada

[19] G. Wiederhold: Mediators in the Architecture of Future Information Systems. IEEE Computer, 25(3), 38-49, 1992.

[20] G. Fahl, T. Risch: Query Processing over Object Views of Relational Data. The VLDB Journal, 6(4), 261-281, 1997.

[21] S. Brandani: Multi-database Access from Amos II using ODBC. Linkoping Electronic Press, 3(19), Dec., 1998, http://www.ep.liu.se/ea/cis/1998/019/.

[22] H. Lin, T. Risch, and T. Katchaounov: Adaptive data mediation over XML data. In special issue on Web Information Systems Applications of Journal of Applied System Studies, 3(2), 2002.

[23] T. Katchaounov, T. Risch, and S. Z• urcher: Object- Oriented Mediator Queries to Internet Search Engines, Int. Workshop on

 ISSN – 1999-4974

E_cient Web-based Information Systems (EWIS), Montpellier, France, 2002.

[24] M.Koparanova and T.Risch: Completing CAD Data Queries for Visualization, Int. Database Engineering and Applications Symp. (IDEAS 2002) Edmonton, Canada, 2002.

[25] Berman HM, Westbrook J, Feng Z, Gilliland G, Bhat TN, Weissig H, et al. The protein data bank. Nucleic Acids Res 2000;28: 235–42.

[26] Ashburner M, Ball CA, Blake JA, Botstein D, Butler H, Cherry JM, et al. Gene ontology: tool for the unification of biology. The Gene Ontol Consortium Nat Genet 2000;25:25–9.

[27] Eppig JT, Bult CJ, Kadin JA, Richardson JE, Blake JA, Anagnostopoulos A, et al. Mouse genome database group. The mouse genome database (MGD): from genes to mice—a community resource for mouse biology. Nucleic Acids Res 2005;33:D471–5.

[28] Tore Risch, Vanja Josifovski, and Timour Katchaounov, Functional Data Integration in a Distributed Mediator System, The Functional Approach to Data Management, 2004, pp 211-238, DOI 10.1007/978-3-662-05372-0_9

[29] Z. Nie and S. Kambhampati. Joint Optimization of Cost and Coverage of Query Plans in Data Integration. In Proceedings of the 10th Intl. Conference on Information and Knowledge Management (CIKM), 2001.

[30] Z. Nie and S. Kambhampati. A Frequency-based Approach for Mining Coverage Statistics in Data Integration. 20th Intl. Conference on Data Engineering (ICDE), 2004.

[31] George Shi Data Integration using Agent based Mediator-Wrapper Architecture, Dept. of Electrical and Computer Engineering, The University of Calgary, 2002

[32] Robert Stevens, Patricia Baker, Sean Bechhofer, Gary Ng, Alex Jacoby, Norman W. Paton, Carole A. Goble and Andy Brass. TAMBIS: Transparent Access to Multiple Bioinformatics Information Sources, Oxford University Press 2000, Vol. 16 no. 2 2000, Pages 184–185

[33] BioSift Radia. http://www.biosift.com/products/radia/-radia.asp, 2003.

[34] Entrez - Search and Retrieval System. http://www.ncbi.nlm.nih.gov/Entrez, 2003.

[35] BioNavigator Solutions. http://www.bionavigator.com, 2003.

[36] Entigen. BioNavigator - BioNode & BioNodeSA: Overview. http://www.entigen.com/library, 2001.

[37] S. Davidson, J. Crabtree, B. Brunk, J. Schug, V. Tannen, C. Overton and C. Stoeckert. K2/Kleisli and GUS: Experiments in Integrated Access to Genomic Data Sources. IBM Systems Journal, 40(2), 512-531, 2001.

[38] Thomas Hernandez & Subbarao Kambhampati, "Integration of Biological Sources:Current Systems and Challenges Ahead"

[39] Katchaounov, T. 2003. Query Processing for Peer Mediator Databases. Acta Universitatis Upsaliensis. Comprehensive Summaries of Uppsala Dissertations from the Faculty of Science and Technology 901. 73 pp. Uppsala. ISBN 91-554-5770-3.

[40] Gio Wiederhold. Mediators in the Architecture of Future Information Systems. IEEE Computer, 25(3):38–49, March 1992.

[41] W. Du and M. Shan. Query Processing in Pegasus. In *Object-Oriented Multidatabase Systems: A Solution for Advanced Applications*. Pretince Hall, Englewood Cliffs, 1996.

[42] Ling-Ling Yan, M. Tamer O¨ zsu, and Ling Liu. Accessing Heterogeneous Data through Homogenization and Integration Mediators. In *Proceedings of the Second IFCIS International Conference on Cooperative Information Systems*, pages 130–139. IEEE Computer Society, 1997.

[43] Kirill Richine. Distributed Query Scheduling in The Context of DIOM: An Experiment. Tech. report TR97-03, Department of Computing Science, University of Alberta, 1997.

[44] Ling Liu and Calton Pu. An Adaptive Object-Oriented Approach to Integration and Access of Heterogeneous Information Sources. *Distributed and Parallel Databases*, 5(2):167–205, April 1997.

[45] Tore Risch and Vanja Josifovski, "Distributed data integration by object-oriented mediator servers", CONCURRENCY AND COMPUTATION: PRACTICE AND EXPERIENCE, Concurrency Computat.: Pract. Exper. 2001; 14:1–21 (DOI: 10.1002/cpe.607)

[46] Anthony Tomasic, Louiqa Raschid, and Patrick Valduriez. Scaling Access to Heterogeneous Data Sources with DISCO. *IEEE Transactions on Knowledge and Data Engineering*, 10(5):808–823, 1998.

 ISSN – 1999-4974