Interoperability and Navigation Between Medical Databases Using Context Graph

Nawaz Khan
School of Computing Science, Middlesex University
White Hart Lane, London N17 8HR, UK
n.x.khan@mdx.ac.uk

Shahedur Rahman
School of Computing Science, Middlesex University
Trent Park, Bramley Road, London N14 4YZ, UK
s.rahman@mdx.ac.uk

Abstract— **This paper proposes a framework for integrating the medical web data. The framework has proposed to develop a cooperative environment to share information on the basis of common purposes for the medical databases. This framework is based on a context of the web data for interaction with different biological data resources. A formal description of context of resources and their relationships have been described here using context graph. The context graph is implemented in Resource Document Framework. A higher level construct, i.e. *has, provide and access*, have been described for each graph which provides an integration domain for accessing the web based databases. The integration domain unifies the contexts of integration and provides navigational plan to access the resources.**

I. INTRODUCTION

Web based gene variance information are emerging in huge volume. For effective management and consolidation of these valuable gene data, we need to develop a framework. Although, a number of discrete analysis tools and software already exist for analysing these variations, however, an integrated and cooperative environment is necessary to convert the laboratory raw data to more meaningful biological information. This implies that initially it is necessary to develop an interoperable generic data model which will then summarise the information available at public domain by creating integration domain, web based query and automated navigation to the resources. Understanding the underlying molecular mechanism of gene requires to access all the relevant data that are available on the web and to derive a complete data sets based on the context of the integration for further analysis. A typical example of this context would be to understand the diseases associated with any particular gene, or to understand phenotypic variation of genome products. In attempt to achieve this researchers have concentrated on database integration based on query optimisation or query script writing, for example, [2][9] [12][14]. Other researchers have suggested metadata and knowledge based integration of molecular biology databases, for example, [8][11]. These approaches require powerful programming languages such as CPL which allows users to submit very complex queries. However, it is difficult to imagine a biologist will be able to write a complex queries using such a complex language.

To integrate web data, a mediator is introduced in many researches, for example [13]. In this approach a wrapper communicate with the resources based on extraction rules. However, these rules need to be rewritten and additional semantics need to be added in the wrapper for transactions. This technique also needs to be extended to tackle biological data resource complexities [10]. Therefore, this paper has proposed to use Resource Document Framework (RDF) for resource mapping and to develop a context graph on the basis of cooperation assumptions within the participating resources. However, the RDF model needs to be further extended to higher-level constructs so that it can tackle the biological web complexities. For example, the same data may exist at different web under different context or it may not follow the traditional approach for relational algebraic operation. In this regard if the higher level constructs can include the context of the web resources then it will increase the semantics of the webs. This approach is unique for the following features:

(i) Description of contexts of the resources - implementation of these contexts into RDF will increase the interoperability by providing the description of the resources and the navigation plan for accessing the web based databases.

(ii) Greater flexibility to design its own navigational plan – this includes choices for the participating resources and query.

This approach for integrating heterogeneous multiple biological resources is quite novel because the research has attempted to synchronise core attributes and views of a component database with medical web resources and it has provided web data correlation independently and dynamically [6]. This approach provides an alternative to the use of generic schema for database integration. The following sections explore this approach and describe the features of component database, the relationship context among the participating resources and navigational plan to traverse among the participating web resources. The paper also presents an example to demonstrate this approach.

II. DEDICATED COMPONENT DATABASE

At present the interoperability between databases is achieved by designing generic class and attributes applicable for general community. This leads to overlapping attributes and objects represented by different classes in different databases. As a result it causes data redundancy.

The research focuses on designing a schema for genetic mutation database which will have unique classes, attributes and objects. The data models are unique in the sense that they will have no overlapping components with other genetic databases. This implies that if attributes a_{mbd1}, a_{mbd2}, a_{mbd3}, ...a_{mbdn} exist in classes C_{mbd1}, C_{mbd2}, ...C_{mbn} of public domain molecular biology databases in schema S_{mbd1}, S_{mbd2}, ...S_{mbdn}, then the same attributes (a_{mbd1}, ...a_{mbdn}) and classes (C_{mbd1}, ...C_{mbdn}) do not exist in

the schema of genetic mutation database S_{gd}. The intersection of C_{mbd1}, C_{mbd2}, ...C_{mbn} can exist but the intersection of C_{mbd1},C_{mbdn} with any classes of S_{gd} does not exist. The classes, $C_{gd1,.....}C_{gdn}$ which belong to S_{gd} are unique for any particular element within all the heterogeneous databases of scientific interest. The elements of our schema $C_{gd1},...,C_{gdn}$ are not a subset of these $C_{mbd1},...,C_{mbdn}$ classes. Furthermore, the elements C_{mbd1}, C_{mbd2}, ..,C_{mbdn} do not have any union of attributes belonging to $C_{mbd1},...,C_{mbdn}$ classes. This will ensure that C_{gd} can never be a part of any derived subclass or superclass. The attributes of classes $C_{gd1},...,C_{gdn}$ will not be a composite attributes by taking the values from the attributes of $C_{mbd1},...,C_{mbdn}$ classes. This will ensure no data redundancy and data dependency during constructing data integration domain for interoperability between heterogeneous databases. These rules are summarised in Table 1.

TABLE 1
NON-OVERLAP SCHEMA INTEGRATION RULES

(i) Attribute $a_{gd} \not\subset \{b_1,...,b_n\}$ *of* C_{mbdi} *classes where* $1 \le i \le n$ *[since* b_n *is attribute of classes which belongs to* S_{mbdi}*]*
(ii) The attributes a_{gd} *of* C_{gd} *will not have the union of attributes* $\{b_1,...,b_n\}$ *of* C_{mbdi} *classes where* $1 \le i \le n$ *[since* b_n *is attribute of classes which belongs to* S_{mbdi}*]*
(iii) The attributes a_{gd} *of* C_{gd} *will not have the intersection of attributes* $\{b_1,...,b_n\}$ *of* C_{mbdi} *classes where* $1 \le i \le n$ *[since* b_n *is attribute of classes which belongs to* S_{mbdi}*]*
(iv) The attribute a_{gd} *of class* C_{gdi} *will not be a composition of other attributes like* $b_1[C_{mbd1}]$, $b_2[C_{mbd2}]$,....., $b_n[C_{mbdn}]$ *where each* C_{mbdn} $(n \ge 1)$ *denotes a class and each* b_n *denotes an attribute associated with* C_{mbdk} *(where k=1,2,....,n).*

The component database stores inheritance patterns, potential recombinants, *lod score*, *relative dinucleotide mutabilities (rdm)* and *mutation rate* for analysis of gene mutation data. A complete schema for mutation data classification, laboratory protocol, pathological lesions and genetic traits have been developed using XML. A description of these schema is provided in [7].

The following section describes the overall approach that is used to achieve correlation between the dedicated component laboratory databases and web data resources.

III. INTERPRETATION AND IMPLEMENTATION OF CONTEXTS

The following sections describe the context graph of resources to demonstrate the contents and links that exist among the webs. The webs relationships are based on the semantics of the webs and on the context of the purpose of integration. The research has proposed how context of data item for a particular web can be implemented in the RDF for resource description and mapping.

A. Context Graph Model

A context graph model represents the meaning, contents and the properties of data. It involves a group of databases and their relationship between the objects of different entities in a particular subject domain. A conceptual context captures the domain knowledge and forces to represent a conceptual semantic view of the underlying data. For example, in medical and biology domain a fixed set of descriptions of relationships between objects does not necessarily mean the semantic similarity between them, rather, a conceptual semantics are necessary to establish a link between the data items.

A context graph G is described using the following parameters: node name and unique ID, operators: a set of values, entry point in pages, resource relation, edges: links between the nodes and labelled image object contents.

A graph G is defined with three interrelated subsets as: $G=(S, E, L)$, where S denotes the object in resource, i.e., page or any particular content, E defines the edges and L defines the element of an image object. S can be expressed as $\{u_1(v_1)...u_n(v_n)\}$ where u denotes the resource name or ID and v denotes the operator. $u(v)$ denotes an object v of a particular resource u. If any node is linked with another node, then it can be expressed with edge E as $E \subseteq u \times u$; where each $e \in E$ is represented as $e = u1.u2$ if e is an edge linking resource u_1 and u_2. L denotes a labelled image object element with a list of values. The context graph also describes the entry point to nodes. In order for an entry point node to be accessed directly the operator needs to have a constant value. Figure 1 shows the proposed context graph model for PDB, GDB and OMIM. This context graph model describes how the PDB, GDB and OMIM objects are related to each other and how they provide access to other target pages either by means of node name (direct linking) or by using search forms. The diagram shows only those portions of the schema which are related to the integration part or which are related to share a common view for integration.

Assuming any medical web resource page $p(x)$ has relationship with other page $q(y)$ which can be accessed as:
- outgoing edge nodes initiated by the $p(x)$ exist in different server
- search forms on the page for an access to the target page in different node
- page leads to search form in different node for accessing the target page

These scenarios are depicted in context graph as follows:
1. link from page $p(x)$ to $q(y)$ is labelled with an operator i

$$u_i \rightarrow u_j$$
$$i(x,y) \rightarrow q(y)$$

2. i must be provided before accessing the target page

$$form\ u_i \rightarrow u_j$$
$$i(x,y) \rightarrow q(y)$$

3. a page leads to the target page form to provide an operator i before accessing the target page

$$u_i \rightarrow u_j\ form\ u_j \rightarrow u_k$$
$$i(x,y) \rightarrow f(y) \rightarrow q(y)$$

Other parameters to describe a context graph for the resource webs are Page contents and Element relationship, Entry point relationship, Outgoing edges and Node type. The parameters with their meaning are described here and their representations are shown in Figure 1.

Page contents and element relationship: The node contents in the graphs are represented with thin one headed arrow (regular arrow) pointing from an object node *u1* to node *u2*. It indicates value relationship and properties

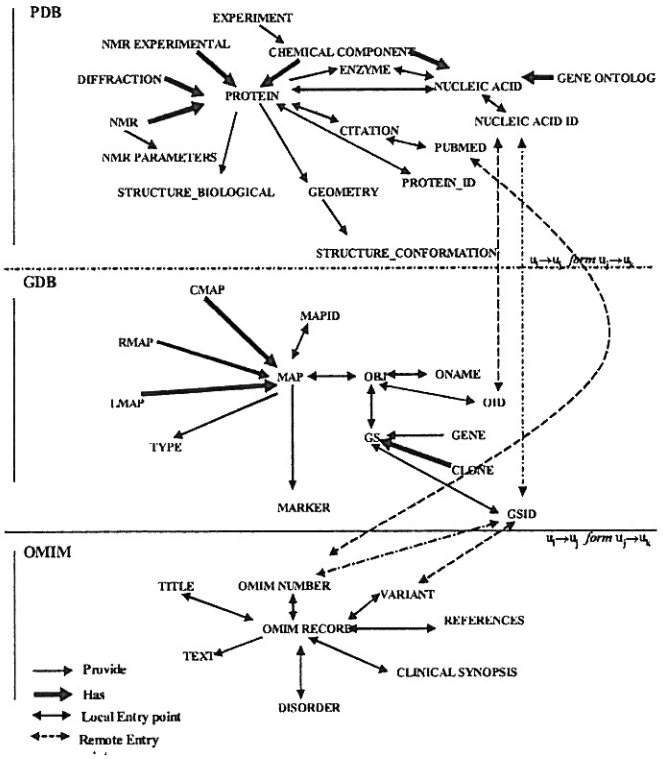


Fig 1. Context graph to demonstrate web data links

departing from a node and ending at a value node. For example, *Structure_biology* and *Geometry* of protein in PDB are described in the structure page and the *Geometry* information then leads to the *sturucture_conformation* page.

Entry point relationship: Entry point relationship is represented by an edge linking nodes u_i and u_j where u_i, $u_j \in S$. Double headed arrow in the graph describes the entry point of a page. The arrow linking for example, *nucleicacid_ID, protein_ID* provides the link to the structure of the proteins. In GDB, *GSID*, *OID*, *ONAME* and object *(OBJ)* provide the link for the gene map. The elements disorder, clinical synopsis, OMIM number in OMIM provides the link to the OMIM record.

Outgoing edges: outgoing edges from any page to the target page are described with dashed arrow head. One value node of one page will provide the target value node of another page. In this case, a search form, like links, maps relationship to other pages. The value of the target page parameter, *Y,* must be provided before accessing the link. This is expressed as follows:

$$u_i \rightarrow u_j \; form \; u_j \rightarrow u_k$$
$$i(x,y) \rightarrow f(y) \rightarrow q(y)$$

Node *i(x,y)* takes to the search form which leads to the target page with the given parameter *y*. It is seen from Figure 1 that *Nucleic_acid_id* from page Nucleic Acid will provide a search form for GDB. The parameter *GDB_ID* will lead to the target page of GS (gene sequence). Consecutively, *GSID* of GDB web schema will provide a search form for OMIM entry.

Node type: Node types are used to describe the node elements such as Structure in PDB. For example, the contents of PDB are *Diffraction, NMR and Chemical components.* These contents make an element of a node which is *Protein* in our example. In GDB, CMAP (contig map), LMAP(linkage map) and RMAP (radiation hybrid map) make an element which is MAP. These node types are represented with thick arrow pointing towards an element of a page departing from the content.

B. Context Graph Interpretation for Resource Mapping

Each interpretation in context graph *I* defines a mapping *M.* A set of values for particular mapping *M* is assumed to have a set of values called *V.* The map requires to contain triple: *has h, provide p* and *access a.* An interpretation of *I* for mapping *M* is defined as follows:

1. A nonempty set *R* of resources, called the domain of *I* and superset of value *V.*
2. Resource access *a* points to the set of resources, *R1, R2..Rn,* if any value *x* are in *R1, R2..Rn* where *I(x)* identifies arguments for which the resources are true.
3. A resource R composed of a set of elements *h* where $h=\{e_1, e_2, \ldots e_n\}$.
4. A resource provides a set of values *p* where *I(p)= true.*

The mapping is for the *resource {r1:h, r1:p, r1:a}* where *<{r1:h, r1:p, r1:a}>= true* if any value *x* in *r1* for which interpretation *I(x)* is in resource *{r2:h, r2:p, r2:a}* and *<{r2:h, r2:p, r2a}>=true* if and only if any value *y* in *r2* for which interpretation *I(y)* is in resource *{rn:h, rn:p, rn:a}* and *<{rn:h, rn:p, rn:a }>=true.* In such case the map denotes all the objects in the resources.

Resource R_1 maps to other resources if any of the context among *has h, provide p* and *access a* satisfy other resources R_2 to R_n. R_2 and R_n also maps each other if the context value satisfies the resources. The resource from where the searching has started can point to one or multiple resources but it must point to at least one resource. The mandatory and optional resource mapping are denoted with solid and dashed arrow respectively in Figure 2. If resource presents in any particular integration domain exceeds more than two, then the third resource needs to be mapped either by the starting resource which maps other resources or by the second resource which is accessed through the first resource.

C. Context Representation in RDF

The main feature of RDF is to provide interoperability by adding semantics to the web resources. RDF describes the whole web page or part of the web page as resource and these resources are named by Uniform Resource Identifier (URI). URI with their properties and values are used to define a RDF statement. These URIs consist of protocols which are used to connect with other nodes and DNS name of the host which leads to the target page to execute a query. These are used as prefix of URI and then a parameter list is attached to specify a particular component of a web resource. These parameters refer to the remote element or a set of elements. It also defines global operation, i.e. *search* and *field* operation for OMIM and *explore* operation for PDB.

The integration domain *D* in RDF has a set of triple which are *<Pi, {SDi}, {Ok}>. Pi* denotes the attribute tag to access the URI prefix, *SD* denotes a set of URI prefixes and *O* denotes a set of key values for target page. For example, the PDB entries can be described in terms of attribute tag, source description and values using RDF modelling. If an individual protein is identified by their

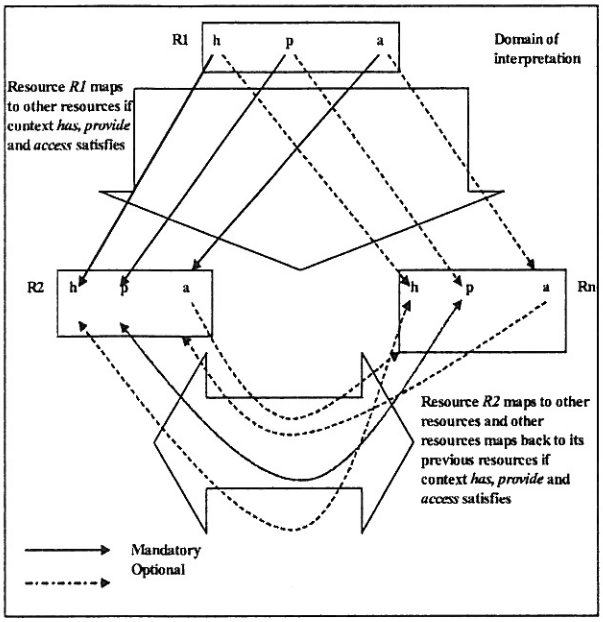


Fig. 2. Resource mapping

unique_ID 1AAP, then the protein and structural details of that particular protein can be accessed from resource http://www.rcsb.org/pdb/cgi/explore.cgi.

To represent a collection of resources, RDF uses an additional resource type, i.e. container, which identifies the specific collection. A membership relation between this container resource and the resources that belongs to the collection is defined by a set of properties. Container resources have other properties in addition to the membership properties and the *type* property, *e.g.*, additional statements. For example, a protein data bank resource contains individual protein descriptions which are accessed using their respective *PDBID* values.

The implementation of context in RDF is not direct as it is for resource description. The context description in RDF requires considering the following characteristics:

- Contexts may be related to other contexts
- All the contexts have their own values, properties and resources
- The values of the contexts determine the contents of the resources
- The properties of the contexts determine the reasons of unifying the resources
- The resource access provides the target for any particular elements.

These characteristics are expressed by describing a context as a statement set with additional structural and logical properties. These are explained as follows:

rdfc:context: is a subclass of *rdfc:StatementSet*, and it represents a context. By inheritance this consists of a set of statements. Context implements a set of statements which provide values, properties and resources. A context can be a set of contexts for any particular purpose. For example, any context for resource PDB can be described as:

[ProteinDataBank]--------rdf:type----→ [rdfc:Context]
{statement of PDB values
statement of PDB properties
statement of other resources}

ProteinDataBank is a context for PDB resource. This ProteinDataBank context has its own values and properties. It also provides the URI for other resources.

rdftype:has: indicates values that is a member of a context, and which is also asserted to be true for a particular resource. This corresponds to the values that the resource contains. For resource PDB it is described as follows:

[ProteinDataBank]-rdf:type→ [rdfc:Context]
{[www.rcnbi.pdb.org]-- has → NMR
[www.rcnbi.pdb.org]--has→DIFFRACTION
[www.rcnbi.pdb.org]--has→CHEMICAL COMPONENTS}

PDB resource has NMR, Diffraction and Chemical components in the context of ProteinDataBank.

rdftype:provide: indicates properties to show the particular reasons for accessing the resource. This is true for one particular resource in one context, but this can also be true for any other resources in another context.

[ProteinDataBank] --------rdf:type --→ [rdfc:Context]
{[www.rcnbi.pdb.org]------provide→ [BiologicalStructure]
[www.rcnbi.pdb.org] ------ provide → [ProteinID]
[www.rcnbi.pdb.org] ------ provide → [NucleicAcidID]}

The resource is described with a set of values in the context of *ProteinDataBank.* In the context of *ProteinDataBank* the resource has a set of values, *NMR, Diffraction* and *Chemical components.* This resource provides Geometry, Enzyme and BiologyStructure of a protein. This BiologyStructure, Geometry and Enzyme have its own values which are implemented by *has.*

rdfctype:access: are the remote or local URI targets to have access to any particular element. For any resource PDB it can be described as follows:

[ProteinDataBank] --------rdf:type --→ [rdfc:Context]
{PID_1]---------- access → www.omim.org.cgi.bin?=value
[NID_1] --------- access → www.gdb.org.cgi.bin?=value}

In the above example, *access* indicates the target page of the web within the context of *ProteinDataBank* which is true for a particular *PID (protein ID) and NID (nucleic acid ID).* Any given values for *PID_1, PID_2* and *NID_1* will lead to the target page of resources. The context of GDB and OMIM resources are described using similar approaches.

D. Unifying the Contexts for Navigation

A context is the collection of attributes of any resource where a resource is described in terms of its values, objects it is providing and any connection to other resources. The set of expressed contexts for each resource is integrated by creating a unifying context for all the contexts. This unifying context is used as the domain or range of integration. It leads to all the resources which are associated with each other and which represent a collection of statements describing the objects present within it. This also allows any context to hold another context without knowing the detail physical structure. This provides a modular approach for describing any high-level relationships among the components. The following example shows how to integrate PDB, GDB and OMIM.

[IntegrationDomain] ---rdf:type → [rdfc:type]
{[ProteinDataBank] --- provide → [PID]
[ProteinDataBank] --- provide → [NID]
{[PID] ----------has → "value"
[NID] ----------has → "value"}
[GenomeDataBank] ---provide → [GID]

{[GID] ---------has → "value"}
[OmimRecord] ----provide→ [OmimNumber]
{[OmimNumber] ---- has → "value"}
[ProteinDataBank] --- access → [GenomeDataBank]
[GenomeDataBank] --- access → [OmimRecord]
[ProteinDataBank] --- access → [OmimRecord]}
The contexts, ProteinDataBank, GenomeDataBank and OmimRecord, are unified to a new context *[IntegrationDomain]*. All the resources are unified to *[IntegrationDomain]* in this example. This initiates the query for the given data, such as *protein ID (PID),* gene *sequence ID (GID) and Omim record ID (OMIMNumber),* to look for the matching data value within their individual contexts. The integration domain provides a navigational plan to explore and execute the logical plans for a set of resource webs. In the above example, a map is described using link, *e.g.* ProteinDataBank provides link for GenomeDataBank, GenomeDataBank provides link for OmimRecord and ProteinDataBank also provides link for OmimRecord. The values for PID, GID and OMIMNumber provide the operators for accessing the web nodes individually. Thus, integration domain is providing a navigational plan to explore and to execute a query on a set of resource webs. The integration domain selects the location of the resources and a set of values, such as, PID, GID and OMIM number. These values act as relation atom to provide the link with the resources. For example, the web which has protein information needs to link with web which has gene information. The relation atom for these two web resources are Protein ID and Nucleic Acid ID. For instance, we can reach the GDB node with particular *GSID* if the PDB node with particular *PID* and *NID* are provided and if *PID* and NID act as relational atom. So, it can be expressed as

$$\text{PDB}(\text{PID}) \wedge \text{PDB}(\text{NID}) \xrightarrow{\text{PID, NID}} \text{GDB}(\text{GSID})$$

E. Search Strategy

This type of complex search needs to combine genetic, developmental, image, and other textual format of data. A context graph model is used to combine these multi-resources into the integration domain. The context graph represents the correlation among the public domain databases.

The correlation operation is carried out between the laboratory based component database and the web resources. The approach allows the researchers to retrieve genotypic information correlated to local gene data. The protein product corresponding to the genotypic information is then searched from the global protein database. The relevant gene product disease information from OMIM is then retrieved on the basis of this correlation.

Context graph model combines these multi-resources into the integration domain. To establish link with this multi-resource, the context graph is implemented in the conceptual level of an independent database which stores all the relevant laboratory results. It also describes the integration domain for multi-resources. A number of agents, namely mapping linker, meta data extractor, converter, dispatcher and result integrator are employed as an integral part of the component database.

The search initiation for target pages residing in multiple resources starts by taking the input stream from the integration domain. The integration domain provides the maps for the resources and supplies a set of values to reach the target page. The integration domain act as virtual table and the *Dispatcher* receives the input data from this table. It passes the data to the hyperlinks in order to reach the target. All specific search operators and maps of the hyperlinks for any individual integration plan are transmitted to the *Dispatcher*. The *'Dispatcher'* is an agent which applies a set of search operators O to the respective data resources R as described in the 'Mapping Linker'. The 'Mapping Linker' is a resource navigator which provides a set of links related to the overall search result.

The search initiation is a bottom up approach. It receives the elements from each node and then pass through to the next node to receive more elements. Finally, when all the elements are collected from the target nodes, the integrator combines these into one object. A constructor module based on this concept is implemented to extract the element contents from the target HTML pages.

A DOM interface is used to provide a set of API calls for accessing the content of the documents. A wrapper designed for DOM-compliant data sources export this information to the dynamic buffer (a virtual table) for each such API call. The input parameters for the DOM calls are *accession numbers* of the biological resources which are obtained by scanning through the RDF data. To illustrate the process, an example to find information on Alzheimer disease is shown here. The protein (*APP1*) for Alzheimer's disease is dispatched individually as searching operator to the respective data resources for unifying them into a single page. For example, the following individual resources along with their operators are collected from RDF (Table 2). These operators are then sent to the respective databases for information on Alzheimer's disease and to correlate the *geno* and *phenotypic* information. Figure 3 shows the final combined result in HTML, e.g., mutation rate, likelihood in male and female, phenotypic details, diagnosis environment, pathological lesion details and coding region. The combined result has collected these different information elements from different biological resources. Table 3 shows the sources of these information elements. All the elements which are collected from different resources are embedded in a single page, called the 'Trait Analysis' (Figure 3).

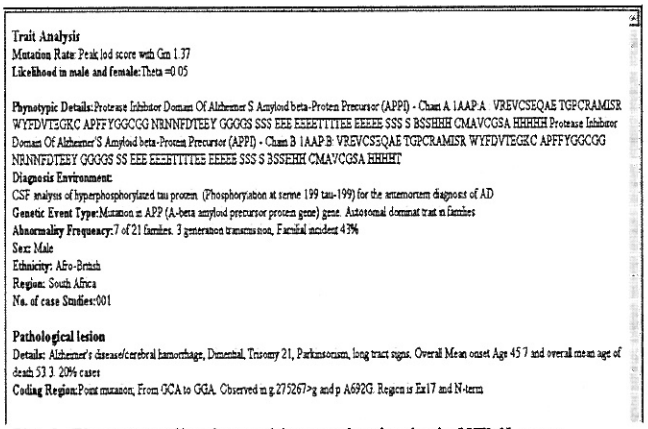
IV. DISCUSSION

A mediator based approach to integrate genome databases has been carried out by many researchers, for example, [3][5]. These techniques are heavily dependent on 'global schema integration' approach. In this technique metadata is used to describe the integration schema and the external schemas of each of the federation's data resources. Another approach is TAMBIS [1] where a classification hierarchy of protein is developed with many pre-defined subclasses. This technique works well when the query follows a fixed framework of the hierarchy and returns the values of corresponding member to the specific subclasses. However, unlike these approaches, this research has described a non-redundant schema integration concept. In this approach the local schema will not hold any

subcomponent, union or intersection of other objects of resource databases. This concept will not allow any redundancy of attributes and thus it will be less prone to loose data integrity. It also reduces the risk to loose information capacity. The proposed approach is not based on present concepts like storing structural meta data, instead, it has attempted to describe the resources and a set of operators to access to the resources for automated navigation.

TABLE 2
ACCESSING THE TARGET NODES USING SPECIFIC OPERATORS

Resource r1← http://us.expasy.org/cgi-bin/niceprot?.PI=P05067
Resource r2← http://us.expasy.org/cgi-bin/blast.PI? sequence=P05067
Resource r3← http://www.rcsb.org/pdb/cgi/explore.cgi?Operator pdbId=1AAP
Resource r4← http://www.gdb.org/gdb-bin/genera/accno? accessionNum=GDB:119692
Resource r5← http://www.ncbi.nlm.nih.gov/htbin-post/Omim? dispmim=104300


Trait Analysis
Mutation Rate: Peak lod score with Gm 1.37
Likelihood in male and female: Theta =0 05

Phynotypic Details: Protease Inhibitor Domain Of Alzheimer S Amyloid beta-Protein Precursor (APPI) - Chain A 1AAP:A VREVCSEQAE TGPCRAMISR WYFDVTEGKC APFFYGGCGG NRNNFDTEEY GGGGS SSS EEE EEEETTTTEE EEEEE SSS S BSSEHH CMAVCGSA HHHHH Protease Inhibitor Domain Of Alzheimer'S Amyloid beta-Protein Precursor (APPI) - Chain B 1AAP:B VREVCSEQAE TGPCRAMISR WYFDVTEGKC APFFYGGCGG NRNNFDTEEY GGGGS SS EEE EEEETTTTEE EEEEE SSS S BSSEHH CMAVCGSA HHHHT
Diagnosis Environment:
CSF analysis of hyperphosphorylated tau protein (Phosphorylation at serine 199 tau-199) for the antemortem diagnosis of AD
Genetic Event Type: Mutation in APP (A-beta amyloid precursor protein gene) gene. Autosomal dominat trait in families
Abnormality Frequency: 7 of 21 families. 3 generation transmission, Familial incident 43%
Sex: Male
Ethnicity: Afro-British
Region: South Africa
No. of case Studies: 001

Pathological lesion
Details: Alzheimer's disease/cerebral hamorrhage, Dementia, Trisomy 21, Parkinsonism, long tract signs. Overall Mean onset Age 45 7 and overall mean age of death 53 3. 20% cases
Coding Region: Point mutation, From GCA to GGA. Observed in g.275267>g and p A692G. Region is Ex17 and N-term

Fig. 3. Elements collection and integration in single HTML page

TABLE 3
ELEMENTS COLLECTED FROM THE RESOURCES

Elements	Resources
Mutation rate, likelihood, diagnosis environment and pathological lesion	**OMIM**
Phenotytypic details	**PDB**
Coding Region, Genetic Event Type	**GDB**
Sex, Ethnicity, No. of cases	**Local database**

An approach suggested by [4][13] attempted to integrate web data from multi-resources, however, it used declarative language to represent web data. This approach is static and the technique is replaced by describing a context, not content, for integration. The reason for utilising context for integration is that it represents the roles of the participating objects instead of the meaning of the objects. A model of subject domain is developed in this research on the basis of 'purpose of integration', and not on the basis of ontology. It is evident from research that biological data has overlapping meaning and a multi-axial ontology is required to build for biological data consistency. Thus, the research has attempted to develop an integration domain based on the contexts of the integration where object relationships are prominent and it ignores the object value matching.

The research has emphasised on a framework to ensure a cooperative environment for database integration by describing individual contexts and then unifying them within a parent context. The strength of the approach lies in the fact that it is applied on the loose binding of databases. Also the approach has shown how a discrete component database can be developed according to the need of a specific laboratory and which can be synchronised with the resource databases that can serve the medical community to a much greater extent. This establishes an alternative approach to the conventional consolidation of schemas into one standard global view.

V. REFERENCES

[1] Baker, P., Brass, A., Bechhofer, S., Goble, C., Paton, N. and Stevens, R.; TAMBIS-transparent access to multiple biological information sources; In Proceedings of the International conference on Intelligent Systems for Molecular Biology, 1998, pp. 25-34.

[2] Chen, I.A., Markowitz, V.M.: An Overview of the Object-Protocol Model (OPM) and OPM Data Management Tools. Information Systems, vol. 20, 1995, pp 393—418

[3] Freidman, M., Levy, A and Millstein, T.: Navigational Plans for Data Integration. In proceedings of the Sixteenth National Conference on Artificial Intelligence and Eleventh Conference on Innovative Application of Artificial Intelligence, 1999, pp. 67-73

[4] Halevy, A. Y. Ives, Z. G., Mork, P. and Tatrinov, I. PIAZZA: data management infrastructure for semantic web applications, 12th International WWW conference, 2003, pp556-567

[5] Kemp, G.J.L., Angelopulos, N. and Gray P.M.D., A Schema-Based Approach to Building a Bioinformatics Database Federation. Proceedings IEEE International Symposium on Bioinformatics and Biomedical Engineering, 2000, p13-20.

[6] Kemper, A and Wiesner, C.: Hyper Queries: Dynamic Distributed Query Processing on the internet, Proceedings of Very Large Database Conference. 2001, pp.551-560.

[7] Khan, N and Rahman, S., Object modeling of gene mutation data for variance analysis. 6th World Conference on Systemics, Informatics and Cybernatics, 2002, pp 301-305

[8] Kohler, J, Lange, M, Hofestadt, R and Schulze-Kremer, S.: Logical and Semantic Database Integration. Proc. 2nd IEEE Symposium on Bioinformatics and Bioengineering Conference. 2000, pp. 77-80.

[9] Markowitz, V.M.: Heterogeneous Molecular Biology Database, Vol 2, no: 4, 1995, Journal of Computational Biology, 1995, pp537-538.

[10] Martin, C.R. A.: Can we integrate bioinformatics data on the internet? Meeting report, Trends in Biotechnology, vol. 19, No. 9, 2001, pp 327-328.

[11] Paton, N.W., Stevens, R., Baker, P., Goble, C.A., Bechhofer, S., and Brass, A, Query processing in the TAMBIS bioinformatics source integration system; Proc. 11th International conference on Scientific and Statistical Database Management, 1999, pp 138-147.

[12] Paton, N.W., Khan, S.A., Hayes, A;, Moussouni F., Brass, A., Elilbeck, K., Goble, A.C., Hubbard, S.J. and Oliver, S.G., Conceptual modelling of genomic information, Bioinformatics, vol. 16, no. 6, 2000, pp 548-557.

[13] Sahuguet, A. and Azavant, F. (2001); Building intelligent web applications using lightweight wrappers; Data and Knowledge Engineering, vol. 36, no. 3; pp 283-316.

[14] Schoobach, c., Kowalski-Saunders. P. and Brusic, V.: Data warehousing in molecular biology, Briefings in Bioinformatics, vol: 1, no: 2, 2000, pp190-198.