ISSN No. 0976-5697

Volume 5, No. 7, September-October 2014

International Journal of Advanced Research in Computer Science

RESEARCH PAPER

Available Online at www.ijarcs.info

Development of database for Interleukins: A New approach in Database Management System

N.Deepak Kumar[1], Dr.A.Ramamohan Reddy[2]
[1,2]Dept.Of CSE,
SVUniversity, Tirupati,India,

Abstract: Advancing our understanding of mechanisms of immune regulation in allergy, asthma, autoimmune diseases, tumor development, organ transplantation, and chronic infection are the some of the diseases where Interleukins play an important role. The immune and inflammatory cells interact through Interleukins and reciprocal regulation with counter balance among TH and regulatory T cells, as well as the subsets of B cells will offer opportunities for immune interventions.

A database manages information and allows organizing data, ensuring completeness and integrity, and transforming the data from one form to another. It make search through the data efficiently to find the desired information. In the present work a database for Interleukins (proteins) have been created as the data related to Interleukins is increasing day by day, it has become difficult for researchers to manage their structure, classification and functions. There are 37 Interleukins have been discovered by the researchers and many more may be added in the future. Interleukins study is mostly useful for diagnosing and treating all the diseases of a human body.

Keywords: database, interleukins, protein, immune cells, Data base management systems

I. INTRODUCTION

Interleukins are biologically active glycoproteins derived primarily from activated lymphocytes and macrophages. Tremendous insight into the biochemical and biological properties of interleukins has been gained with advances in recombinant DNA technology, protein purification, and cell-culture techniques. The biological properties of interleukins include induction of T-lymphocyte activation and proliferation, augmentation of neutrophil, macrophage, and T-lymphocyte cytooxicity, and promotion of B lymphocyte and multilineage bone marrow stem-cell precursor growth and differentiation. Interleukins may play a role in the pathogenesis of several important diseases. Interleukin therapy is likely to play an important role in the treatment of cancer, infectious diseases, and immunodeficiency syndromes. [12,14].

Specified design processes are standard in the software development industry, and there are many design processes described in the software engineering literature. The details of the design process are less crucial than the use of a process. However, there are some crucial steps, such as gathering requirements. Requirements document what the database is trying to accomplish. Most databases have to make data model compromises. Databases have been used to manage and integrate large volumes of complex data in other disciplines for decades [1].

Development of a data model is another crucial step because this helps to identify potential problems in the design early on in the project, when they are still easy to correct. The most common tool used for this purpose in relational database design is the entity relationship diagram. This type of diagram represents the real-world entities about which the database will store information, and the relationships between those entities. Use the database to enforce data integrity. A database should protect the integrity or consistency of the data that it stores. The strong theoretical basis of relational DBMS provides rules of normalization, which, if followed, will ensure basic data integrity. These rules ensure that all information is stored in the smallest meaningful pieces and is stored in only one place, preventing data duplication and the concomitant potential for internal inconsistencies. A database that obeys these rules is said to be normalized. Normalization splits related data across multiple tables, requiring queries to perform operations called joins to reassemble the data.

The recent bioinformatics literature includes numerous papers about databases, but these primarily focus on the need for integration across existing databases [2,3], report the design and use of specific databases [4–9], or argue for better large scientific databases and the systematic changes necessary to accomplish this goal [10,11]. All this information is valuable, but does not provide much help to novice database designers**.**

II. METHODOLOGY

Attempting to develop a data model for all biology is inviting disaster. Instead, focus on the subset of biological information relevant to the project. However, limiting the scope of the database should not be confused with designing the database with only the immediate requirements of a particular application in mind. The database design can ignore or simplify information outside of its scope, but should fully represent the information within the subset of biology that it covers, even if the current application does not use the full complexity of the data. This will make the database more robust, allowing it to persist through multiple iterations of the application. For instance, a database that stores information about protein stability can ignore or simplify the link between proteins and genes, but should be able to handle stability measurements produced by a variety of different techniques, even if the initial version of the application accessing the database deals only with thermal denaturation.

Some projects might require a data model that covers a large area of biology. In this case, the data model and application should be developed in several steps by breaking the subject area into portions, and delivering an application for one portion at a time . The first portion for which the database is developed can be chosen by identifying a minimum subset of the full project requirements that will lead to a useful application. A database and application to meet this subset of requirements is then implemented, and user feedback is gathered before moving on to the next subset of requirements. The redesign of the Protein Data Bank (PDB) by the Research Collabatory for Structural Bioinformatics (RCSB) demonstrates this principle. When the RCSB assumed control of the PDB, there were many additional features requested by the user community.

However, implementing all these features would have resulted in an unacceptably long time for the development of the first RCSB PDB website. The RCSB chose to implement only a portion of the full requirements first, migrating the core function of storing and organizing biomolecular structural data before extensive new functionality was introduced. The prevalence of ‘flat files’ (files in which information is stored in a structured text format) in bioinformatics also influences database scope. Many existing bioinformatics applications rely upon specific flat file formats, such as the PDB format used to store the biomolecular structure information mentioned below. The database designer must decide whether to leave a subset of the data in the flat files or to parse these files into database structures and recreate the flat file format as necessary. This decision can only be made in the context of how the data in the flat files will be used, and how ‘clean’ the data in these files is. If the data in the flat files has many inconsistencies, and/or there is a need to provide access to specific portions of a flat file, it is usually worth the effort to parse the data in the files into the database. If the flat files are free from major inconsistencies and are primarily accessed in their entirety, parsing them into the database might be a waste of time and project resources.

The database design should be able to accommodate realistic test data early in the design process. It is important to include ‘pathological examples’ (i.e. example data that represent the most complex relationships in the future dataset). For instance, consider the design of a set of tables to store the relationship between genes and proteins. Without any knowledge of biology, one might erroneously assume this is a one-to-one relationship (each gene relates to one and only one protein, and vice versa). Realistic test data quickly reveal the flaw in this design: because of splicing variations, one gene can produce multiple proteins, resulting in a one-to-many relationship. Further consideration reveals the pathological case: owing to genetic redundancy, one protein can also be produced by multiple genes. The gene–protein relationship is therefore most correctly modeled as a many-to-many relationship.

Some complicated relationships might be unimportant for the goal of a database and can be simplified. For instance, a database concerned primarily with protein function might need to identify only one gene for a protein. However, any simplifying assumptions made in the data model should always be documented. Also, it must be certain that the more complicated relationship is truly outside of the scope of the database, and not merely absent from the requirements of the current application accessing the database (see ‘Keep the database manageable’ above).

The magnitude of data is also an issue in biological databases. A good design for a database with one hundred rows can be a disaster for a database with one million rows. The design process must take the volume of data into account, particularly during the physical design of the database.

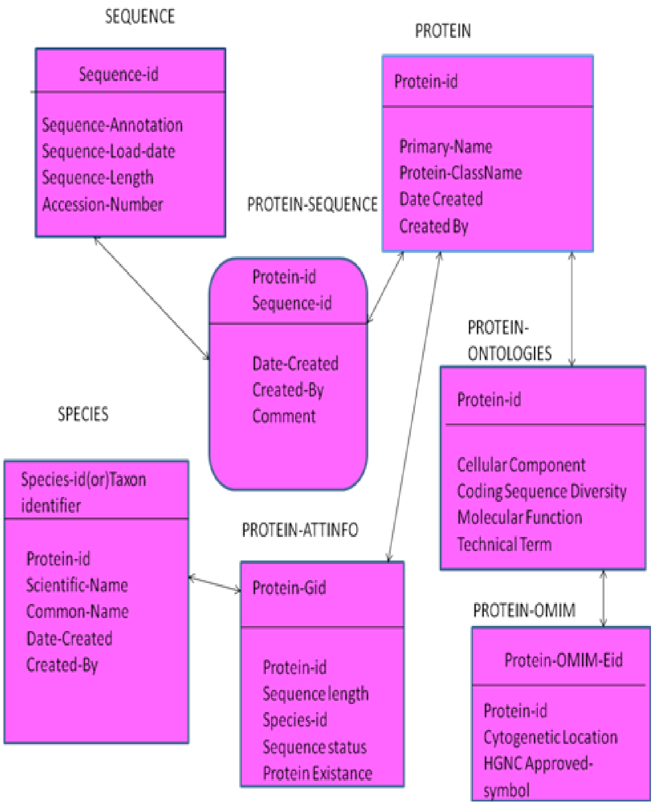


Figure I. ER Diagram for storing information about Proteins

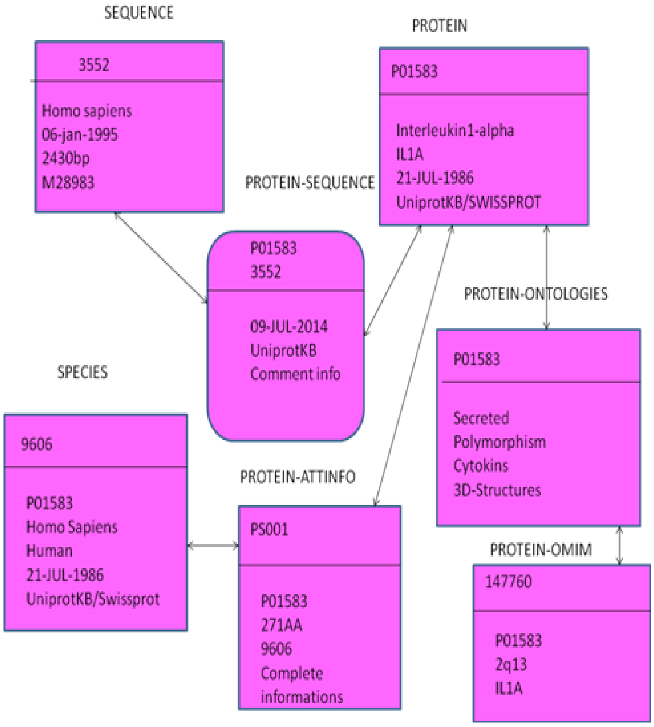


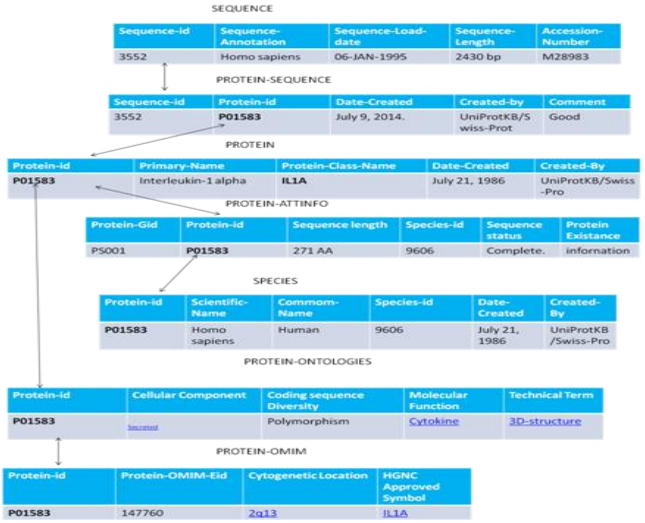
Figure II. Example ER Diagram for storing information about Interleukins. An hypothetical data was shown for Interleukin database.

III. RESULTS

The Interleukin database model consists of Protein, Protein Information, Protein OMIM information, Protein Sequence, Protein attribute informations , Sequence and Species. The entity relationship diagram is shown here for Interleukin database. In Protein, the information related to Interleukin ID, Protein primary name, Class name, date created and created by will be given. Sequence entity contain sequence id, sequence annotation, sequence load date, sequence length, accession number. Protein Sequence entity contain attributes are sequence id, protein id, date created, created By, Comment. Similarly Species entity contain the species id, protein id, scientific name, common name, Date created, created By attributes. Similarly Protein ontologies entity contain the protein id, cellular component, coding sequence diversity, Molecular function, technical term attributes. Similarly Protein Attribute entity contain Protein Gid, protein id, sequence length, species id, sequence status, protein existence attributes. Similarly Protein OMIM entity contain Protein Eid,protein id, cytogenetic location, HGNC approved symbol attributes. In each and every entity contain protein id attribute, through protein id attribute we will display the all related informations in the Database.

IV. CONCLUSION

In this work an attempt performed for creation of Interleukin database for the first time by considering hypothetical information on the details of the Interleukins. As the number of discoveries on the Interleukins is increasing day by day the creation of a separate protein biological database for Interleukins along with effective robustic DBMS method is used. There are many applications of Interleukins in Biology which make them unique to understand and the information exclusively on them is very much useful to all the researchers.



SEQUENCE

Sequence-id	Sequence-Annotation	Sequence-Load-date	Sequence-Length	Accession-Number
3552	Homo sapiens	06-JAN-1995	2430 bp	M28983

PROTEIN-SEQUENCE

Sequence-id	Protein-id	Date-Created	Created-by	Comment
3552	**P01583**	July 9, 2014.	UniProtKB/Swiss-Prot	Good

PROTEIN

Protein-id	Primary-Name	Protein-Class-Name	Date-Created	Created-By
P01583	Interleukin-1 alpha	**IL1A**	July 21, 1986	UniProtKB/Swiss-Pro

PROTEIN-ATTINFO

Protein-Gid	Protein-id	Sequence length	Species-id	Sequence status	Protein Existance
PS001	**P01583**	271 AA	9606	Complete.	infornation

SPECIES

Protein-id	Scientific-Name	Commom-Name	Species-id	Date-Created	Created-By
P01583	Homo sapiens	Human	9606	July 21, 1986	UniProtKB /Swiss-Pro

PROTEIN-ONTOLOGIES

Protein-id	Cellular Component	Coding sequence Diversity	Molecular Function	Technical Term
P01583	Secreted	Polymorphism	Cytokine	3D-structure

PROTEIN-OMIM

Protein-id	Protein-OMIM-Eid	Cytogenetic Location	HGNC Approved Symbol
P01583	147760	2q13	IL1A

Figure III.Design Database based on above ER-Diagram.

V. REFERENCES

[1]. Achuthsankar S Nair Computational Biology & Bioinformatics – A gentle Overview, Communications of Computer Society of India, January 2007

[2]. Aluru, Srinivas, ed. Handbook of Computational Molecular Biology. Chapman & Hall/Crc, 2006.ISBN 1-58488-406-1 (Chapman & Hall/Crc Computer and Information Science Series)

[3]. Baldi, P and Brunak, S, Bioinformatics: The Machine Learning Approach, 2nd edition. MIT Press, 2001. ISBN 0-262-02506-X

[4]. Barnes, M.R. and Gray, I.C., eds., Bioinformatics for Geneticists, first edition. Wiley, 2003. ISBN 0-470-84394-2

[5]. Peri S, et al. (2003). "Development of human protein reference database as an initial platform for approaching systems biology in humans". Genome Research 13: 2363–71.doi:10.1101/gr.1680803.

[6]. Gandhi, T.K.B. et al. Analysis of the human protein interactome and comparison with yeast, worm and fly interaction datasets. Nature Genetics. 2006. 3, 285–293

[7]. Mathivanan, S. et al. An evaluation of human protein–protein interaction data in the public domain. BMC Bioinformatics. 2006. 7, S19

[8]. Mishra, G. et al. Human protein reference database—2006 update. Nucleic Acids Research. 2006. 34, 411–414

[9]. Mathivanan, S. et al. Human Proteinpedia enables sharing of human protein data. Nature Biotechnology. 2008. 26, 164–167

[10]. Amanchy, R. et al. A compendium of curated phosphorylation-based substrate and binding motifs. Nature Biotechnology. 2007. 25, 285–286

[11]. Mathivanan S, Periaswamy B, Gandhi TK et al. (2006). "An evaluation of human protein-protein interaction data in the public domain". BMC Bioinformatics. 7 Suppl 5: S19.doi:10.1186/1471-2105-7-S5-S19. PMC 1764475. PMID 17254303.

[12]. Brocker, C; Thompson, D; Matsumoto, A; Nebert, DW; Vasiliou, V (Oct 2010). "Evolutionary divergence and functions of the human interleukin (IL) gene family.". Human Genomics 5 (1): 30–55. doi:10.1186/1479-7364-5-1-30. PMC 3390169. PMID 21106488.

[13]. Khadka, A (2014). "Interleukins in Therapeutics". PharmaTutor 2 (4): 67–72.

[14]. Priestle JP, Schär HP, Grütter MG (December 1989). "Crystallographic refinement of interleukin 1 beta at 2.0 A resolution". Proc. Natl. Acad. Sci. U.S.A. 86 (24): 9667–71. doi:10.1073/pnas.86.24.9667. PMC 298562. PMID 2602367.

[15]. Arai K, Yokota T, Arai N, Lee F, Rennick D, Mosmann T (1985). "Use of a cDNA expression vector for isolation of mouse interleukin 2 cDNA clones: expression of T-cell growth-factor activity after transfection of monkey cells". Proc. Natl. Acad. Sci. U.S.A. 82 (1): 68–72. doi:10.1073/pnas.82.1.68. PMC 396972. PMID 3918306