DESIGN AND DEVELOPMENT OF THE AFRICAN *PLASMODIUM FALCIPARUM* DATABASE – (afriPFdb)

BY

EWEJOBI ITUNUOLUWA MARIAN
B.Sc. (Computer Science)

A project submitted to the Department of Computer and Information Sciences, College of Science & Technology, Covenant University, Ota, Nigeria.

In partial fulfillment of the requirements for the award of the degree of Master of Science (M.Sc.) in Computer Science.

2009

CHAPTER ONE

INTRODUCTION

1.1 BACKGROUND INFORMATION

Malaria is one of the planet's deadliest diseases and one of the leading causes of sickness and death in the developing world. Africa has suffered and is still suffering from the adverse socio-economic effects of malaria; It is intimately connected with poverty. Judged as both a root cause and a consequence of poverty, it is most intractable for the poorest countries in the world. According to GSK’s (GlaxoSmithKline) Corporate Responsibility Report 2007, it affects the health and economic growth of nations and individuals alike and is costing Africa about $12 billion a year in economic output. It has a greater impact on Africa's human resources than simple lost earnings. Another indirect cost of malaria is the human pain and suffering caused by the disease. It also hampers children's schooling and social development through both absenteeism and permanent neurological and other damage associated with severe episodes of the disease (© 2008 Millennium Promise). According to WHO’s (World Health Organization) World Malaria Report 2008, there were an estimated 247 million malaria cases worldwide in 2006, of which 91% or 230 million were due to *Plasmodium falciparum*. The vast majority of cases (86%) were in the African Region, followed by the South-East Asia (9%) and Eastern Mediterranean regions (3%) as shown in figure 1.1.

There are four major human malaria strains (*Plasmodium falciparum, Plasmodium vivax, Plasmodium ovale, Plasmodium malariae*). *Plasmodium falciparum* is the most common and deadly form. The percentage of malaria cases due to *Plasmodium falciparum* exceeded 75% in most African countries but only in a few countries outside Africa. In Africa, Nineteen of the most populous countries accounted for 90% of estimated cases in 2006 with Nigeria having the highest percent of the 90% estimated cases **(**WHO's World Malaria Report 2008) as shown in figure 1.2.

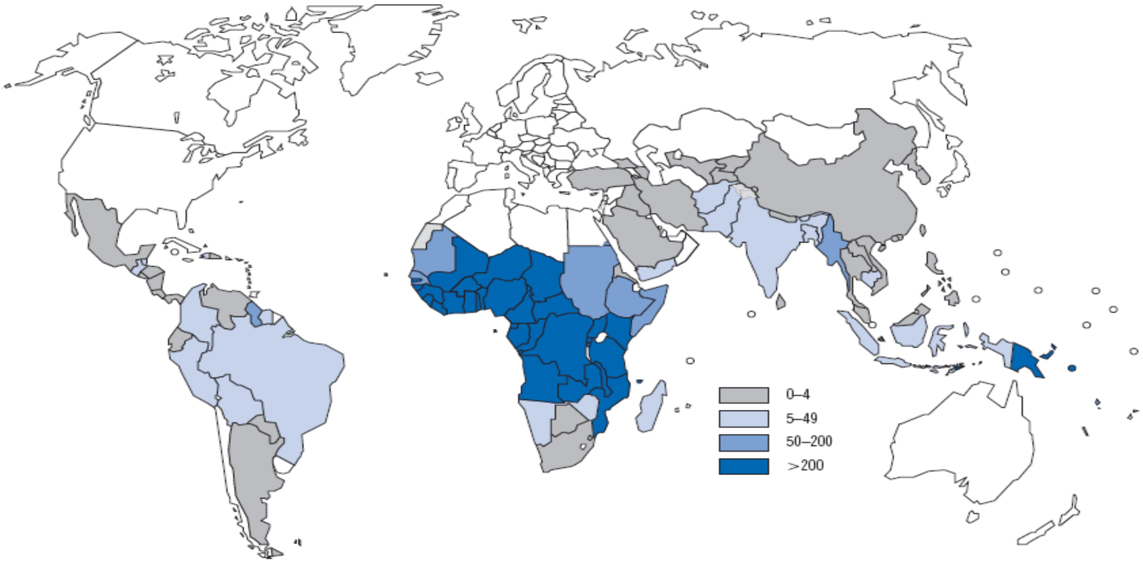


Fig 1.1 Estimated incidence of malaria per 1000, 2006 (WHO's World Malaria Report 2008)

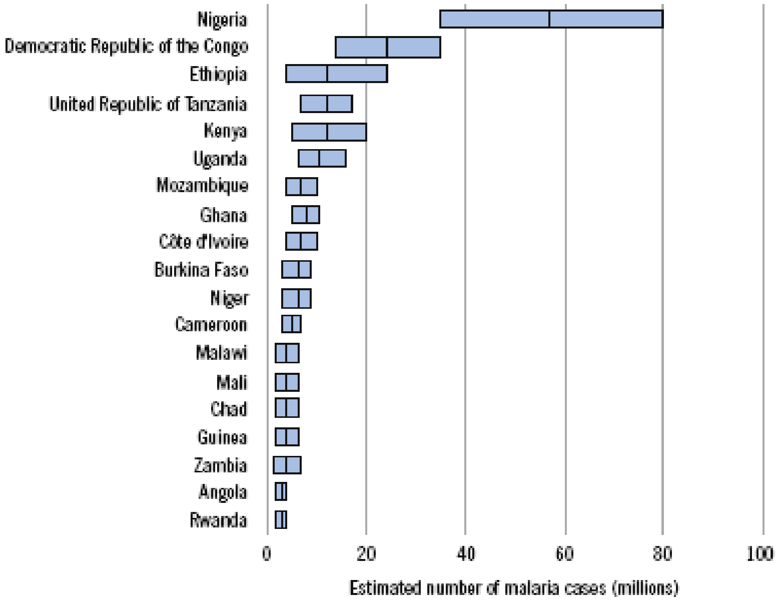


Fig 1.2 Nineteen countries estimated to have 90% of cases in African Region, 2006 (WHO's World Malaria Report 2008)

This calls for increased speed in the search for more potent/effective cure that will preserve the lives of many. Chloroquine and most anti-malaria drugs are fast becoming in-effective as the parasite has grown resistance to them. Therefore, there is a huge need to discover and validate new drug or vaccine targets to enable the development of new treatments for malaria (Bulashevska, S. et al; 2007).

This project work is aimed at building a publicly accessible database through the use of a relational database architecture that will house *Plasmodium falciparum* genome sequence data and the results that we obtained by applying *in-silico* tools for quick access such that researchers can integrate this resource with other relevant data sets, and exploit the resulting information for functional studies, including identification of novel drug targets and candidate vaccine antigens.

We believe that the experimental results that will be obtained from our data will drive work in malaria research that will quicken the discovery pipeline of drugs and vaccines.

1.2 STATEMENT OF THE PROBLEM

KEGG database is mainly developed using *in-silico* analysis, while BioCyc is experimentally curated but strongly in many cases confirmed the *in-silico* results displayed on the metabolic network of KEGG. Therefore, the database, we implemented (afriPFdb), in this project work is to display our findings developed using *in-silico* tools for experimental validation.

One other important lacking utility from other databases is informative detail on *Plasmodium falciparum's* tRNAs' (transfer RNA) important sites for drug targets. Transfer RNA carries amino acids to the ribosomes for incorporation into a protein. We also incorporated this into our database 'The African Plasmodium falciparum Database' (afriPFdb).

1.3 AIM AND OBJECTIVES

The aim of this research work is to build a publicly accessible database that will house in-depth and up-to-date information on *Plasmodium falciparum* from built *in-silico* tools for easy and quick access by malaria researchers.

This aim will be achieved through the following objectives;

i. To design and build a well structured database that will enhance an 'easy to find' presentation of available information.

ii. To design a simple and easy to navigate website with adequate and correct biological data on *Plasmodium falciparum*.

iii. To build a database to accommodate constant updates as is the nature of biological data.

iv. To design a website to house and publish the findings on malaria research which can then be validated experimentally.

1.4 METHODOLOGY

In developing afriPFdb, the following methods were adopted; the web pages (frontend) were designed with 'Web Page Maker' and then exported to HTML (Hyper Text Markup Language).

The middle ware (i.e. web server) used was Apache server with PHP (Hypertext Preprocessor) as the server side scripting language to accept and send information to and fro the front end and the backend.

The back end (i.e. database) used was a relational database model which was designed using PostgreSQL due to its platform independence and free licensing cost.

The whole application presently runs on 3-tier architecture.

1.5 SIGNIFICANCE OF STUDY

This work finds relevance in genomics, proteomics, databases, drug design, and also in providing a publicly accessible repository on *Plasmodium falciparum* for computational results from built in-silico tools.

1.6 SCOPE OF STUDY

The database is solely for *Plasmodium falciparum* and does not include information about other species of *Plasmodium*.

1.7 LIMITATIONS OF STUDY

i. Due to constantly changing nature of biological data and the fact that malaria research is an on-going work, there would be need for constant updating of the database.

ii. The website does not feature data analysis resources like ‘BLAST’ and so on.

1.8 EXPECTED CONTRIBUTION

A new web based database dedicated to providing relevant and up to date information on *Plasmodium falciparum* for researchers which will drive the work in malaria research that will hasten the discovery pipeline of drugs and vaccines.

1.9 ARRANGEMENT OF OUTLINE

This project would evolve through the following stages; Chapter two is a review of relevant literature and includes exposition on basic concepts of databases, malaria and Plasmodium. It includes a review of existing biological databases. Chapter three presents the design of afriPFdb and its development process using various design tools to give a proper description of the work. It also includes the descriptions of the architecture, modules and the database tables.Chapter four discusses the implementation of the system. It presents screenshots of the results obtained when the system is implemented. Chapter five presents the summary and conclusion from the results of the study and makes some recommendations for further work.

CHAPTER TWO

LITERATURE REVIEW

2.1 WHAT IS A DATABASE?

A Database is a structured collection of records or data that is stored in a computer system. The structure is achieved by organizing the data according to a database model. The model commonly used is the relational model. It is a collection of information organized in such a way that a computer program can quickly select desired pieces of data (Wikipedia, the free encyclopedia). It can also be described as an organized body of related information.

The information in a database should be

- Structured
- Searchable
- Updatable
- Cross-linked
- Available
- Capable of being shared among multiple applications

Therefore, a database is a computerized record keeping system. In order for a database to be functional, it must not only store large amounts of records well, but be easily accessible. In addition, new information and changes should also be fairly easy to input. Besides these features, all databases that are created should be built with high data integrity and the ability to recover data if hardware fails.

Obviously, many databases store confidential and important information that can not be easily accessible by just anyone. Many databases require passwords and other security features in order to access the information. While some databases are accessible via the internet through a network, others are closed systems and can only be accessed on site.

They can be very small (less than 1 MB) or extremely large and complicated (terabytes as in many government databases) however, all databases are usually stored and located on hard disk or other types of storage devices and are accessed via computer. Large databases may require separate servers and locations, however many small databases can fit easily as files located on the computer's hard drive.

There are many different types of database. the description of the records contained in a database will often determine its type. Therefore, the most common types of databases are:

- Bibliographic databases
- Full-text databases
- Numeric databases
- Biological databases
- Image databases
- Audio/Video databases

- Mixed databases (a combination of any or all types of information)

2.1.1 FUNCTIONS OF A DATABASE

The functions of a database among many others include;

i. Store the data
ii. Provide a standardized method for retrieving or changing the data
iii. Create, read, update and delete views from the database. A view is a particular way of looking at a data in a database. The data is always stored in tables and views present the data to the user by displaying only certain columns from one or more tables.
iv. Enforce constraints. Constraints are also built into the database based upon business requirements.

2.1.2 ADVANTAGES OF A DATABASE

Databases, were created to solve the problems with file-oriented systems. The "flat file" system was a start and thus, it was seriously inefficient. In order to find a record, someone would have to read through the entire file and hope it was not the last record. With a hundred thousand records, One can imagine the dilemma.

What was needed, computer scientists thought (using existing metaphors again) was a card catalog, a means to achieve random access processing, that is the ability to efficiently access a single record without searching the entire file to find it.

The result was the indexed file-oriented system in which a single index file stored "key" words and pointers to records that were stored elsewhere. This made retrieval much more efficient. It worked just like a card catalog in a library. To find a record, one needed only to search for keys rather than reading entire records.

However, even with the benefits of indexing, the file-oriented system still suffered from problems including:

i. **Data Redundancy** - the same data might be stored in different places

ii. **Poor Data Control** - redundant data might be slightly different for example when Ms. Jones changes her name to Mrs. Johnson and the change is only reflected in some of the files containing her record.

iii. **Inability to Easily Manipulate Data** - it was a tedious and error prone activity to modify files by hand

iv. **Cryptic work flows** - accessing the data could take excessive programming effort and was too difficult for real-users (as opposed to programmers).

The advantages of using a database include;

1. It provides a greater level of data security and confidentiality than a flat file system. Specifically, when accessing a logical record in a flat file, the application can see all data elements--including any confidential or privileged data. To minimize this, many customers have resorted to putting sensitive data into a separately managed file, and linking the two as necessary. This may cause data consistency issues.
2. It reduces the application programming effort.
3. It manages more efficiently the creation and modification of, and access to, data. As you know, if new data elements need to be added to a file, then all applications that use that file must be rewritten, even those that do not use the new data element. This is not the case when using a Database Management System (DBMS).
4. Improve interoperability
5. Reduce inconsistency
6. Improve efficiency

Others include;

i. Compactness

ii. Speed of retrievals and searches

iii. Easy to use

iv. Current

v. Accurate

vi. Allows easy sharing of data between multiple users (Selena Sol, *1998*)

2.1.3 TYPES OF DATABASE MODELS

There are four main types of databases:

i. Hierarchical
ii. Network
iii. Relational
iv. Object-Oriented

The Hierarchical Model

A hierarchical data model is a data model in which the data is organized into a tree-like structure. The structure allows repeat information using parent/child relationships: each parent can have many children but each child only has one parent. Data can be lost if a child has no parent. All attributes of a specific record are listed under an entity type. In a database, an entity type is the equivalent of a table; each individual record is represented as a row and an attribute as a column. Entity types are related to each other using *1: N* mapping, this is also known as one-to-many relationships. (Wikipedia, the free encyclopedia; 2008)

The most recognized example of hierarchical model database is an Information Management System IMS designed by IBM Introduced in 1968.XML can be viewed as a hierarchical database, information is organized into tree, and collections of XML files can be used as a database.

The Network Model

The **network model** is a database model conceived as a flexible way of representing objects and their relationships. The network model original inventor was Charles Bachman, and it was developed into a standard specification published in 1969 by the CODASYL Consortium.

Where the hierarchical model structures data as a tree of records, with each record having one parent record and many children, the network model allows each record to have multiple parent and child records, forming a lattice structure. It extends hierarchical model to allow children to have multiple parents, careful design can avoid data duplication.

The Network Model has:

i. Records
ii. Links between records

The main argument in favor of the network model, in comparison to the hierarchic model, was that it allowed a more natural modeling of relationships between entities. Although the model was widely implemented and used, it failed to become dominant for two main reasons. Firstly, IBM chose to stick to the hierarchical model with semi-network extensions in their established products such as IMS (Information Management system) and DL/I (Data Language/1, the language system used to access IBM's IMS databases, and its data communication system.). Secondly, it was eventually displaced by the relational model, which offered a higher-level, more declarative interface. Until the early 1980s the performance benefits of the low-level navigational interfaces offered by hierarchical and network databases were persuasive for many large-scale applications, but as hardware became faster, the extra productivity and flexibility of the relational model led to the gradual obsolescence of the network model in corporate enterprise usage. Listed below are some well-known Network Databases;

i. TurboIMAGE
ii. IDMS
iii. RDM Embedded
iv. RDM Server (Wikipedia, the free encyclopedia; accessed 2008)

The Relational Model

The relational model was invented by E.F. Codd as a general model of data, and subsequently maintained and developed by Chris Date and Hugh Darwen among others. In The Third Manifesto (first published in 1995) Date and Darwen show how the relational model can accommodate certain desired object-oriented features. The relational model for database management is a database model based on first-order predicate logic. Information is modeled as tables (relations) with links between tables.

It has a rigorous mathematical basis which allows prevention of data duplication, other data integrity problems and simplifies data access. It is the dominant model in use today. (Wikipedia, the free encyclopedia; accessed 2008)

Examples of DBMS that use the relational model amongst others include;

i. Oracle
ii. IBM DB2
iii. MS SQL Server
iv. PostgreSQL
v. MySQL

Relational model attempts to correctly represent data, without regard to how it will be used. In other models, how the data will be used can greatly influence the design. Relational Databases were intended to free users from needing a programmer to write new code to answer each new question. This is particularly useful in science: scientists will always think of a new question!

The operators at the user's disposal…are operators that generate new tables from old, and those operators include SELECT, PROJECT, PRODUCT, UNION, INTERSECT, DIFFERENCE, JOIN, and DIVIDE.

Relational Terms

Candidate Keys: a candidate key can uniquely identify each row, and cannot be reduced: i.e., there is no subset of the attributes in the key that also uniquely identify each row.

Primary key: a primary key is the candidate key chosen to be used.

Alternate keys = candidate keys not chosen to be primary key.

Properties of Relational Tables

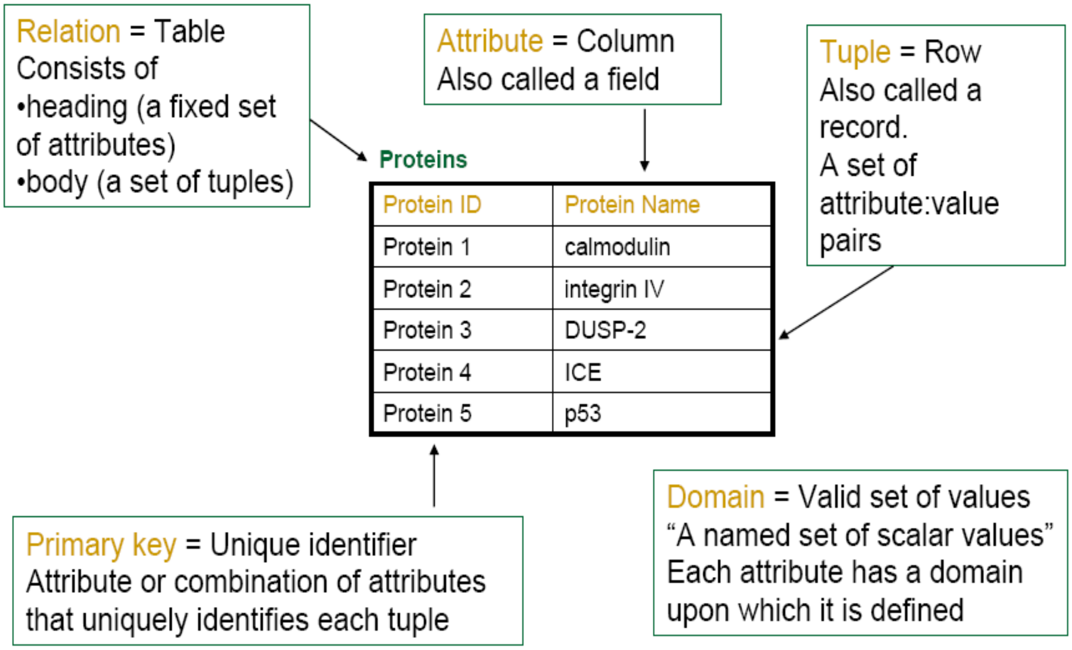
The following properties are a consequence of the definition of relations, attributes, and domains:

- Each column has a unique name (The heading = a fixed set of attributes)
- All entries in a given column are of the same kind (Attributes are defined on a domain)
- There are no duplicate tuples. "Each row is unique", the body of a relation is a mathematical set: sets do not have duplicate elements. Primary key ensures this rule is upheld

- The sequence of tuples is unimportant. Sets are unordered. Database Administration may change the way in which rows are partitioned in storage to improve performance of certain queries
- Attribute values are atomic, i.e. entries in columns are single-valued.

Types of Relations

- Base relation - an autonomous relation (i.e., not defined in terms of another relation). What we typically mean when we talk about database tables
- Derived relation - a relation defined in terms of other relations. Query results, for instance.
- View – a named derived relation. "View" is an abstract table containing selected data from other tables, a virtual report.
- Materialized view= a view in which data is actually copied.



Relation = Table
Consists of
•heading (a fixed set of attributes)
•body (a set of tuples)

Attribute = Column
Also called a field

Tuple = Row
Also called a record.
A set of attribute:value pairs

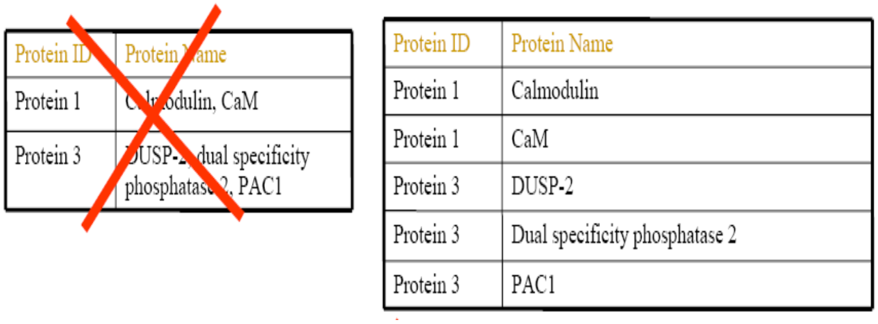
Proteins

Protein ID	Protein Name
Protein 1	calmodulin
Protein 2	integrin IV
Protein 3	DUSP-2
Protein 4	ICE
Protein 5	p53

Primary key = Unique identifier
Attribute or combination of attributes that uniquely identifies each tuple

Domain = Valid set of values
“A named set of scalar values”
Each attribute has a domain upon which it is defined

Fig 2.1 Some relational terms explained



~~Protein ID~~	~~Protein Name~~
~~Protein 1~~	~~Calmodulin, CaM~~
~~Protein 3~~	~~DUSP-2, dual specificity phosphatase 2, PAC1~~

Protein ID	Protein Name
Protein 1	Calmodulin
Protein 1	CaM
Protein 3	DUSP-2
Protein 3	Dual specificity phosphatase 2
Protein 3	PAC1

Fig 2.2 Attribute values must be atomic

Data Integrity

There are three types of data integrity, they include;

Entity Integrity

Referential Integrity

Domain Integrity

Entity Integrity

- No part of the primary key may be NULL

  NULL = absence of value or value doesn't exist. The primary key uniquely identifies a row. If part is NULL, it means that we do not know the value. Therefore, we can't uniquely identify the row

Referential Integrity

A foreign key value must either

- Match a primary key value in the referenced table or be NULL

Domain Integrity

Attribute integrity: values of an attribute are taken from the specified domain

The Object Oriented Model

The Object Oriented databases were introduced in conjunction with rise in object-oriented programming techniques. There are difficulties integrating object oriented programming and relational databases. object oriented databases often have the same problems network databases had. They Lack easy data access of relational databases. Major relational databases have introduced “object extensions”. There are ongoing debate about how best to integrate databases and object oriented programming

OODBs store data in classes, with associations between classes, Integrates data storage with data manipulation and have methods that are part of the object.Care must be exercised to avoid data duplication and “orphan” data. (Wikipedia, the free encyclopedia; accessed 2008).

2.2 DATABASE MANAGEMENT SYSTEMS

In order to have a highly efficient database system, you need to incorporate a program that manages the queries and information stored on the system. A database relies upon software to organize the storage of data. This software is known as a **database management system (DBMS).**

Database management systems are categorized according to the database model that they support. The model tends to determine the query languages that are available to access the database. A great deal of the internal engineering of a DBMS, however, is independent of the data model, and is concerned with managing factors such as performance, concurrency, integrity, and recovery from hardware failures. In these areas there are large differences between products.

2.2.1 WHAT IS A DATABASE MANAGEMENT SYSTEM (DBMS)?

A database management system is a suite of software applications that together make it possible for people or businesses to store, modify, and extract information from a database. It can also be described as a software program designed to allow the creation of specially organized files, as well as data entry, manipulation, removal, and reporting for those files (The_Techref_Glossary)

A "Database Management System" is software that defines a database, stores the data, supports a query language, produces reports, and create data entry screens. (Wikipedia, the free encyclopedia accessed 2008).

A Database Management System (DBMS) is software used to manage a database, *i.e.* allowing insertion of new data, update or deletion of old data, and retrieval of stored data by imposing constraints. It is a a suite of interrelated computer programs designed to manage message processing and database update in a tightly controlled manner.

Database Management System: A collection of programs that enables you to enter, organize, and select data in a database. The typical Database Management System includes the capability of storing data in tables, creating relationships between the data tables, querying data, creating reports, and managing the database itself. A DBMS includes of four main parts: Modeling language, data structure, database query language, and transaction mechanism:

DBMS are categorized according to their data structures or types. It is a set of prewritten programs that are used to store, update and retrieve a Database. The DBMS accepts requests for data from the application program and instructs the operating system to transfer the appropriate data. When a DBMS is used, information systems can be changed much more easily as the organization's information requirements change. New categories of data can be added to the database without disruption to the existing system.

Examples of DBMSs include:

i. Oracle database
ii. IBM DB2
iii. Adaptive Server Enterprise
iv. FileMaker
v. Firebird
vi. Ingres
vii. Informix
viii. Microsoft Access

ix. Microsoft SQL Server

x. Microsoft Visual FoxPro

xi. MySQL

xii. PostgreSQL

xiii. Progress

xiv. SQLite

xv. Teradata

xvi. CSQL

xvii. OpenLink Virtuoso

xviii. Daffodil DB

2.2.2 DBMS FEATURES AND CAPABILITIES

Alternatively, and especially in connection with the relational model of database management, the relation between attributes drawn from a specified set of domains can be seen as being primary. For instance, the database might indicate that a car that was originally "red" might fade to "pink" in time, provided it was of some particular "make" with an inferior paint job. Such higher arity relationships provide information on all of the underlying domains at the same time, with none of them being privileged above the others.

Throughout recent history, specialized databases have existed for scientific, geospatial, imaging, document storage and like uses, the functionality drawn from such applications has lately begun appearing in mainstream DBMSs as well. However, the main focus there, at least when aimed at the commercial data processing market, is still on descriptive attributes on repetitive record structures.

Therefore, the DBMSs of today roll together frequently-needed services or features of attribute management, by externalizing such functionality to the DBMS, applications effectively share code with each other and are relieved of much internal complexity.

Features commonly offered by database management systems include:

Query ability

Querying is the process of requesting attribute information from various perspectives and combinations of factors. For example: "How many 2-door cars in Texas are green?" A database query language and report writer allow users to interactively interrogate the database, analyze its data and update it according to the users privileges on data.

Backup and replication

Copies of attributes need to be made regularly in case primary disks or other equipment fails. A periodic copy of attributes may also be created for a distant organization that cannot readily access the original. DBMS usually provide utilities to facilitate the process of extracting and disseminating attribute sets. When data is replicated between database servers, so that the information remains consistent throughout the database system and users cannot tell or even know which server in the DBMS they are using, the system is said to exhibit replication transparency.

Rule enforcement

Often one wants to apply rules to attributes so that the attributes are clean and reliable. For example, we may have a rule that says each car can have only one engine associated with it (identified by Engine Number). If somebody tries to associate a second engine with a given car, we want the DBMS to deny such a request and display an error message. However, with changes in the model specification such as, in this example, hybrid gas-electric cars, rules may need to change. Ideally such rules should be able to be added and removed as needed without significant data layout redesign.

Security

Often it is desirable to limit who can see or change which attributes or groups of attributes. This may be managed directly by individual, or by the assignment of individuals and privileges to groups, or (in the most elaborate models) through the assignment of individuals and groups to roles which are then granted entitlements.

Computation

There are common computations requested on attributes such as counting, summing, averaging, sorting, grouping, cross-referencing, etc., rather than have each computer application implement these from scratch, they can rely on the DBMS to supply such calculations.

Change and access logging

We may want to know who accessed what attributes, what was changed, and when it was changed. Logging services allow this by keeping a record of access occurrences and changes.

Automated optimization

If there are frequently occurring usage patterns or requests, some DBMS can adjust themselves to improve the speed of those interactions. In some cases the DBMS will merely provide tools to monitor performance, allowing a human expert to make the necessary adjustments after reviewing the statistics collected.

2.2.3 ADVANTAGES OF DATABASE MANAGEMENT SYSTEMS

The use of a database management system such as IMS/DB or DB2 to implement the database also provides additional advantages. The DBMS:

i. Allows multiple tasks to access and update the data simultaneously, while preserving database integrity. This is particularly important where large numbers of users are accessing the data through an online application.

ii. Provides facilities for the application to update multiple database records and ensures that the application data in the various records remains consistent even if an application failure occurs.

iii. Is able to put confidential or sensitive data in a separate segment (in IMS) or table (in DB2). In contrast, in a PDS or VSAM flat file, the application program gets access to every data element in the logical record. Some of these elements might contain data that should be restricted.

iv. Provides utilities that control and implement backup and recovery of the data, preventing loss of vital business data.

v. Provides utilities to monitor and tune access to the data.

vi. Is able to change the structure of the logical record (by adding or moving data fields). Such changes usually require that every application that accesses the

VSAM or PDS file must be reassembled or recompiled, even if it does not need the added or changed fields. A properly designed data base insulates the application programmer from such changes.

Keep in mind, however, that the use of a database and database management system will not, in itself, produce the advantages detailed here. It also requires the proper design and administration of the databases, and development of the applications.

2.2.4 WHY USE POSTGRESQL FOR WEB APPLICATION DEVELOPMENT?

This section answers the question that some clients ask: "Why use PostgreSQL (as opposed to MySQL, Oracle, MS-Access, FoxPro, etc.) for web application development?"

PostgreSQL is a Relational Database Management System (RDBMS) used as the back-end data management component for the database driven websites. As There are other popular choices available, but the choice of RDBMS is often highly dependent on the parameters and requirements of an individual project.

Which Databases to Consider?

We will be comparing PostgreSQL to other common database server packages that are choices for web application development: MySQL, Oracle, and Microsoft SQL Server.

There are also a number of desktop database packages that some use for web application development; these include Access, FoxPro, FileMaker Pro, and others. While these applications are often inexpensive and user-friendly for simple desktop or workgroup database management, they are rarely suited for server-based application deployment.Desktop database applications usually only support one or a small number of simultaneous users. This makes them unusable for web applications where multiple simultaneous connections are needed. Desktop database applications usually have weak security. Most web applications need much more robust security systems than these desktop-based packages provide. They are usually not designed for web application deployment or other networked services. Their architecture, scalability, and overall performance are rarely optimized for use in an Internet-based application, making them a liability for serious database driven site development.

Choosing PostgreSQL

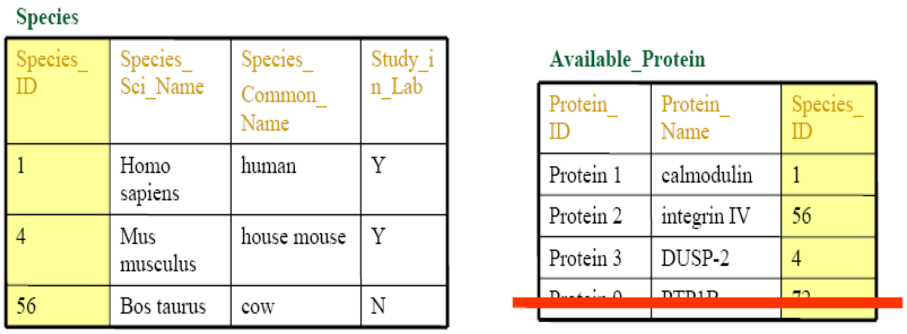
General Advantages

There are several key advantages of using PostgreSQL:

1. PostgreSQL is free, Open Source software
2. PostgreSQL has excellent commercial and community support options
3. PostgreSQL has legendary reliability and stability
4. PostgreSQL is very scalable and extensible
5. PostgreSQL is cross platform
6. PostgreSQL is designed for high volume environments
7. PostgreSQL is easy to administer

Features

Here is a more detailed list of features that are offered by these top RDBMS packages:



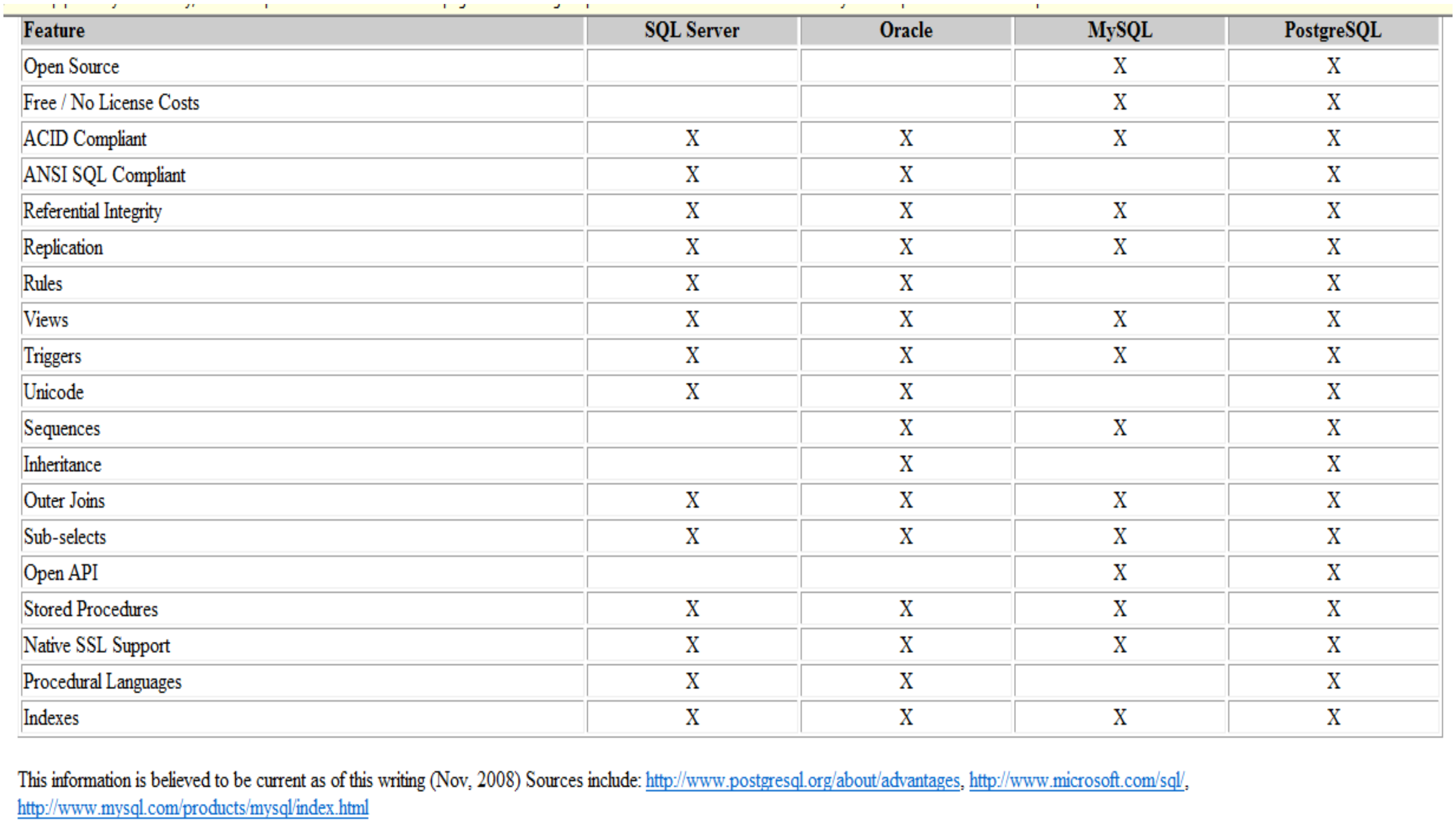
Species

Species_ID	Species_Sci_Name	Species_Common_Name	Study_in_Lab
1	Homo sapiens	human	Y
4	Mus musculus	house mouse	Y
56	Bos taurus	cow	N

Available_Protein

Protein_ID	Protein_Name	Species_ID
Protein 1	calmodulin	1
Protein 2	integrin IV	56
Protein 3	DUSP-2	4
~~Protein 9~~	~~PTP1B~~	~~73~~

Fig 2.3 Referential Integrity



Feature	SQL Server	Oracle	MySQL	PostgreSQL
Open Source			X	X
Free / No License Costs			X	X
ACID Compliant	X	X	X	X
ANSI SQL Compliant	X	X		X
Referential Integrity	X	X	X	X
Replication	X	X	X	X
Rules	X	X		X
Views	X	X	X	X
Triggers	X	X	X	X
Unicode	X	X		X
Sequences		X	X	X
Inheritance		X		X
Outer Joins	X	X	X	X
Sub-selects	X	X	X	X
Open API			X	X
Stored Procedures	X	X	X	X
Native SSL Support	X	X	X	X
Procedural Languages	X	X		X
Indexes	X	X	X	X

This information is believed to be current as of this writing (Nov, 2008) Sources include: http://www.postgresql.org/about/advantages, http://www.microsoft.com/sql/, http://www.mysql.com/products/mysql/index.html

Fig 2.4 Features of top RDBMS packages

Most all of the above features are key for developing robust, scalable web-based applications, and PostgreSQL clearly provides excellent value in this regard.

Performance

PostgreSQL, like its companions, will always require tuning and optimizing based on the particular application. And so each of the systems such as SQL Server, Oracle, MySQL, and PostgreSQL have particular areas of performance where they excel.

PostgreSQL is generally optimized and faster than others for scenarios involving high transactional loads, high numbers of users (especially non-read-only applications), and complex queries. Other related features like views, ACID compliance, etc. also contribute to PostgreSQL's overall performance.

Database benchmarking appears to be an area where little industry-wide data is available, again mostly due to the degree to which each database system can be tuned to meet certain performance expectations.

Commercial Support

There exists an extensive network of companies and individuals providing commercial consulting and support for PostgreSQL. One listing of those entities is available at http://techdocs.postgresql.org/companies.php. This is in contrast to proprietary support contracts needed for systems like SQL Server and Oracle.

In addition to the formal support channels, there are a wide variety of mailing lists, support groups, and other online collaborations where one looking for technical support and consultation can turn. Our experience is that any advanced PostgreSQL questions can usually be resolved very quickly at little or no material cost.

Industry Acceptance / Wide-spread Usage

With any mission critical application, it is important to ensure that the technologies in use are established within their industry and accepted by others as a trusted name. In addition to PostgreSQL's large network of commercial support, it enjoys significant usage by many notable organizations employing it for mission critical or large scale applications.

A partial list of companies that use PostgreSQL for their applications include;

- **Affymetrix** - A market leader in the creation of state-of-the-art tools for the genetic research industry, uses PostgreSQL in their Transcriptome Project to store data about large-scale RNA expression experiments derived from high-density GeneChip® microarrays.
- **Afilias** - A global provider of domain name registry services. This Ireland-based company manages over 900,000 domain names and over 10 million records in its PostgreSQL-backed database.

- **The American Chemical Society** - The largest professional organization of Chemists in the world, with over 165,000 members, and a website that receives more than 12 million visits every day. Their Journal Archive stores 125 years of full publications (2.5 million pages, more than 1 terabyte of data) using PostgreSQL.
- **Cognitivity** - The online e-Learning provider uses PostgreSQL as the preferred database for their presentation and management software.
- Big Enterprises like TIM (Celular Service Company), Mesbla , Toyota, HP , Fujitsu...... are using PostgreSQL

2.3 BIOLOGICAL DATABASES

Over the last decades, a huge amount of biological data has been accumulated as the rapid development of biotechnology. In order to understand and explain biological phenomena from the data, people are now focusing more on data analysis originating from their former work and using those results to direct their experiments. Thus, we need a tool to organize all the data, biological databases having been considered as such a tool to assist scientists in data management.

Up to now, huge amounts of biological data have been collected from different biological sources. Biological data is produced in a digital form which needs to be stored in a database, which is supposed to satisfy different user groups in their own researches. A well-designed biological database is a powerful tool which can contribute a lot to biological researches.

Currently, a lot of bioinformatics work is concerned with the technology of databases. These databases include both "public" repositories of gene data like GenBank or the Protein DataBank (PDB), and private databases like those used by research groups involved in gene mapping projects or those held by biotech companies. Making such databases accessible via open standards like the Web is very important since consumers of bioinformatics data use a range of computer platforms: from the more powerful and forbidding UNIX boxes favoured by the developers and curators to the far friendlier Macs often found populating the labs of computer-wary biologists. A few popular biological databases are GenBank from NCBI (National Center for Biotechnology Information), SwissProt from the Swiss Institute of Bioinformatics and PIR from the Protein Information Resource.

Building a biological database is, in theory, no different from building a database for an investment bank, government agency, business or another scientific endeavor. One needs to understand the information the database is going to store and present, translate that understanding into a rigid framework.

There are two problems often encountered with biological databases which are rare in other database implementations. The first is that the ‘true’ biological interpretation of data stored in a database not only can change over time, but discovering new relationships between aspects of the data is part of the motivation for storing information in a database.

The second is the inability to rarely have a full understanding of both biology and programming affects all levels of building databases, from programmers to funders and from users to reviewers. Thankfully there are now far more people with dual skill sets graduating, and there are a number of specific educational courses that blend computer science and biology. This problem dominates many aspects of working biological databases, and the only solution is to find and hire good people who have at least an appreciation of the other field.

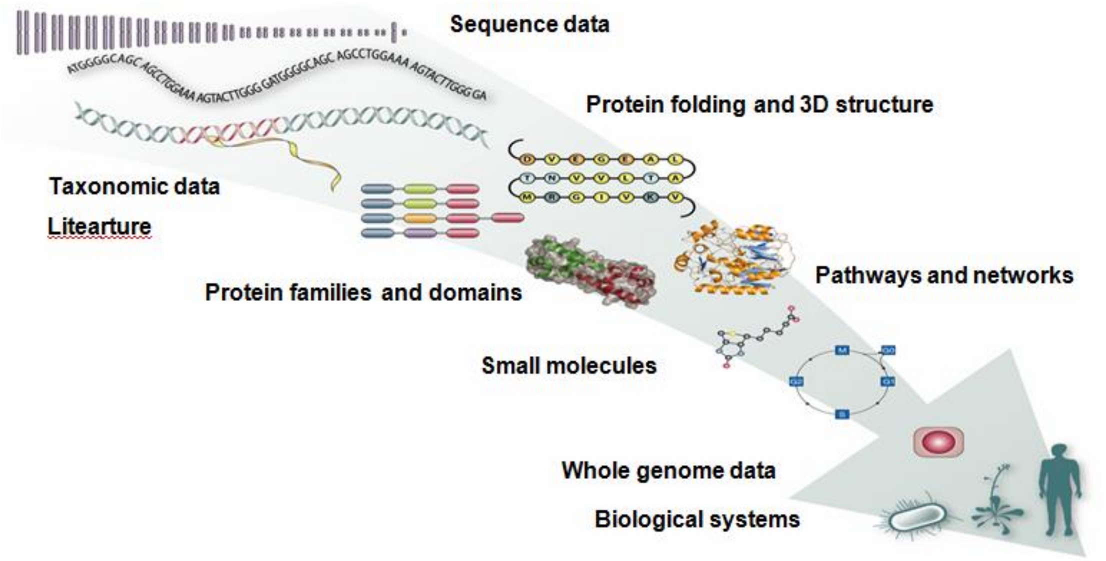

Sequence data
ATGGGGCAGC AGCCTGGAAA AGTACTTGGG GATGGGGCAGC AGCCTGGAAA AGTACTTGGG GA
Protein folding and 3D structure
Taxonomic data
Litearture
Pathways and networks
Protein families and domains
Small molecules
Whole genome data
Biological systems

Fig 2.5 Biological systems

Annually, the journal Nucleic Acids Research (NAR) dedicates an entire issue (first issue in January) to all available biological databases, which are recorded in tabular form with the respective URLs. Furthermore, for a number of databases, original articles describe their functions. This database issue, is a good starting point for working with biological databases.

The Nucleic Acids Research online Molecular Biology Database Collection is a public repository that lists more than 1000 databases. The Database Issue is freely available, and categorizes all the publically available online databases related to computational biology (or bioinformatics). The 2008 update includes 1078 databases, 110 more than the previous one. The complete database list and summaries are available online at the Nucleic Acids Research web site, http://nar.oxfordjournals.org/.

2.3.1 CLASSIFICATION OF BIOLOGICAL DATABASES

Depending on the kind of data included, different categories of biological databases can be distinguished. They are

- Primary databases
- Secondary databases
- Specialized/composite databases

Primary databases contain primary sequence information (nucleotide or protein) and accompanying annotation information regarding function, bibliographies, cross-references to other databases, etc. The growth of the primary databases gave rise to serious and valid questions on the format of the sequences, reliability and the comprehensiveness of the databases. Examples of these include Swiss-Prot & PIR (Protein Information Resource) for protein sequences, GenBank, EMBL Nucleotide Sequence Database & DDBJ (DNA DataBase of Japan) for Genome sequences and the PDB (Protein Databank) for protein structures.

A secondary database contains derived information from the primary database. A secondary sequence database contains information like the conserved sequence, signature sequence and active site residues of the protein families arrived by multiple sequence alignment of a set of related proteins. A secondary structure database contains entries of the PDB in an organized way. These contain entries that are classified according to their structure like all alpha proteins, all beta proteins, etc. These also contain information on conserved secondary structure motifs of a particular protein. Some of the secondary databases created and hosted by various researchers at their individual laboratories include;

- **Gene** http://www.ncbi.nlm.nih.gov/sites/entrez?db=gene

  Gene is a searchable database of genes, including information about their chromosomal location and their function.

- **PROSITE:** database of protein domains, families and functional sites http://us.expasy.org/prosite/
  PROSITE is a secondary database, which analyses sequence data from primary databases.
- **TIGR Genome Projects (J. Craig Venter Institute)** http://www.tigr.org/db.shtml
  Suite of databases of DNA and protein sequences, gene expression, and other information, for humans and other organisms.
- **UniProt (**http://www.ebi.ac.uk/uniprot/**)** combines Swiss-Prot, PIR, and TrEMBL.
- **FlyBase** http://flybase.bio.indiana.edu
  A database of the genome of Drosophila – a fruit fly used in genetics research.
- **HIV Databases** http://www.hiv.lanl.gov/content/
- Information on genetic sequences, immunological epitopes, drug-resistance, associated mutations, and vaccine trials. Databases hosted by the Los Alamos National Laboratory, funded by the NIH.
- **TAIR** http://www.arabidopsis.org
- Genetic and molecular biology information for *Arabidopsis thaliana*, a member of the mustard family used in genetics research.

Composite/specialized database amalgamates a variety of different primary database sources, which obviates the need to search multiple resources. Different composite database use different primary database and different criteria in their search algorithm. Various options for search has also been incorporated in the composite database. Examples include;

- The National Center for Biotechnology Information (NCBI)
- OMIM (Online Mendelian Inheritance in Man) which contains information about the proteins involved in genetic diseases.
- **Ensembl** http://www.ensembl.org/
  Software system for searching information on the genomes of various organisms including humans, rats and mosquitoes.
- Literature Database **- PubMed**

The Nucleic Acids Research NAR Database Categories List

Nucleotide Sequence Databases

RNA sequence databases

Protein sequence databases

Structure Databases

Genomics Databases (non-vertebrate)

Metabolic and Signaling Pathways

Human and other Vertebrate Genomes

Human Genes and Diseases

Microarray Data and other Gene Expression Databases

Proteomics Resources

Other Molecular Biology Databases

Organelle databases

Plant databases

Immunological databases

2.3.2 FLAT FILE FORMATS FOR SEQUENCES

Biological data is data or measurements collected from biological sources, which is stored or exchanged in a digital form. Biological data are commonly stored in flat files or databases. Examples of biological data are DNA base-pair sequences, and population data used in ecology.

Several formats currently exist for storing biological data. They include;

- Fasta
- GenBank/GB
- EMBL
- NBRF
- GCG
- IG/Stanford
- Phylip
- DNAStrider
- Plain/Raw and many others.

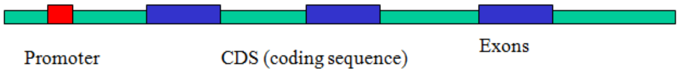
Fasta Format

A sequence in FASTA format begins with a single-line description, followed by lines of sequence data. The description line is distinguished from the sequence data by a greater-than (">") symbol in the first column. The word following the ">" symbol is the identifier of the sequence, and the rest of the line is the description (both are optional). There should be no space between the ">" and the first letter of the identifier. It is recommended that all lines of text be shorter than 80 characters. The sequence ends if another line starting with a ">" appears; this indicates the start of another sequence. A simple example of one sequence in FASTA format:

```
>MAL9|PFI0515w| prenylated protein, putative
ATGAATTTATTAGCAATTTTTTTGACAAAATATGTAGAAGATGTTATTTTCCTTTGT
AATGCCACAGATTTAAATTCTTATTCTTTTATAAAAAAAAAGGCCTTTAAAGAAGC
CGCCTTTTTTGTGGCTAGAACTATTCCATCACGGATAGAATATAATACCAAAGAAA
TAATAACTCATGAAAATAATACAGTTTTCGCTTTTAAATATGAAGACAATATTTGTC
CTATTGTAATAGCTACAGACGATTATCCGGAAAGGGTTGCTTTTTATATGATAAATG
AGATTTATAGAGATTTCATAAGTACCATTCCCAAGGAAGAATGGTCGAGTGTAAAA
CAAGACAATAAAATATCGTTTAATTTGTATACATATTTAACAAAATATAAGGACCC
TTTAAATTGCGACGCCATAACCCAAACAAATATAAAAATTAATGAAAATATTGAAA
AAGTTAGAGTAACCATGGATGCTCTTATAAGGAATAGAGAAAATTTGGACGTCCTT
GTTGATAAGAGCAAGGATCTTTCATCAACGACAAAACAGTTATTCAAACAAAGTAA
AAAACTGAAGAAGAAGCAATGTTGCAGCATTATGTGA
```

2.4 REVIEW OF EXISTING BIOLOGICAL DATABASES

2.4.1 Nucleotide Sequence Database



- EMBL, DDBJ, GenBank
- Data submitted by sequence owner
- Must provide certain information and CDS if applicable

2.4.1.1 EMBL Nucleotide Sequence Database

The **European Molecular Biology Laboratory** (**EMBL**) is a molecular biology research institution supported by 20 European countries and Australia as associate member state. The EMBL was created in 1974 and is a non-profit organisation funded by public research money from its member states. Research at EMBL is conducted by approximately 85 independent groups covering the spectrum of molecular biology. The Laboratory operates from five sites: the main Laboratory in Heidelberg, and Outstations in Hinxton (the European Bioinformatics Institute), Grenoble, Hamburg, and Monterotondo near Rome.

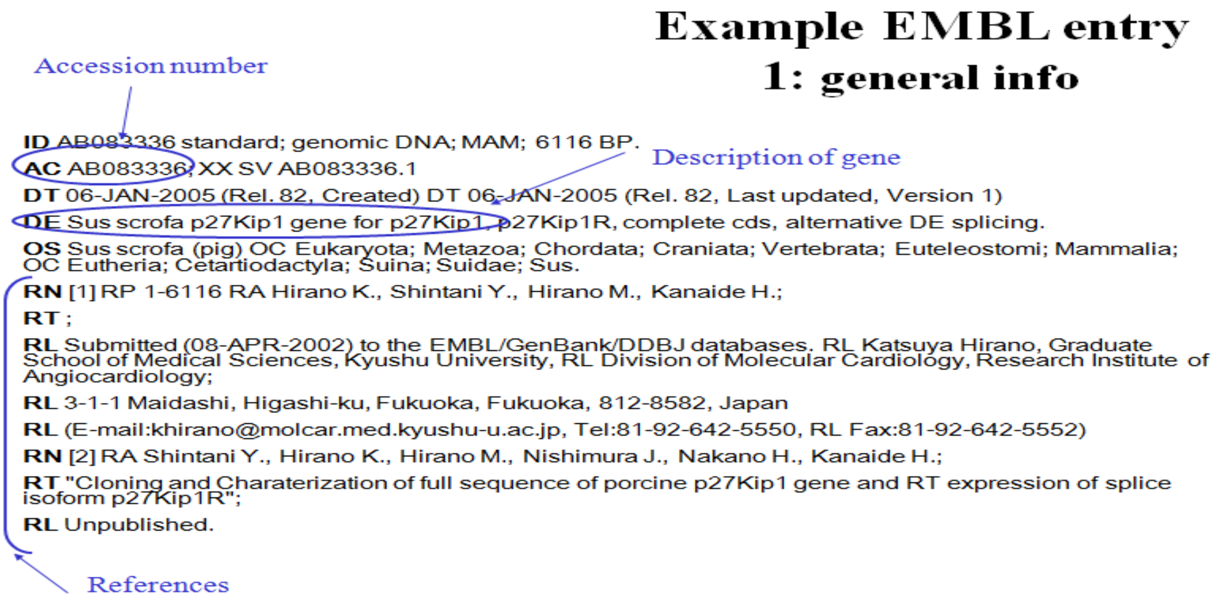
The EMBL Nucleotide Sequence Database is a comprehensive database of DNA and RNA sequences collected from the scientific literature and patent applications and directly submitted from researchers and sequencing groups. Data collection is done in collaboration with GenBank (USA) and the DNA Database of Japan (DDBJ).

It is a comprehensive collection of nucleotide sequences maintained at the European Bioinformatics Institute (EBI). Data are received from genome sequencing centres, individual scientists and patent offices.

New data are released daily into the EMBL database and are immediately available. The EMBL database is stored and maintained in an ORACLE data management system

Established in 1980, the database was historically tightly coupled to the publication of sequences in the scientific literature, but quickly electronic submissions became usual practice. Today, the volume of data submitted by direct transfer of data from major sequencing centres, such as the Sanger Centre, overshadows all other input. In recent years the EMBL database has doubled in size nearly every year and on the October 1, 1999 contained 4.7 million entries representing over 3.6 Gigabases of nucleotide sequence.

In 2006, the volume of data has continued to grow exponentially. Access to the data is provided via SRS, ftp and variety of other methods. Extensive external and internal cross-references enable users to search for related information across other databases and within the database. All available resources can be accessed via the EBI home page at http://www.ebi.ac.uk/. Changes over the past year include changes to the file format, further development of the EMBLCDS dataset and developments to the XML format.



Example EMBL entry
1: general info

ID AB083336 standard; genomic DNA; MAM; 6116 BP.

AC AB083336; XX SV AB083336.1

DT 06-JAN-2005 (Rel. 82, Created) DT 06-JAN-2005 (Rel. 82, Last updated, Version 1)

DE Sus scrofa p27Kip1 gene for p27Kip1, p27Kip1R, complete cds, alternative DE splicing.

OS Sus scrofa (pig) OC Eukaryota; Metazoa; Chordata; Craniata; Vertebrata; Euteleostomi; Mammalia; OC Eutheria; Cetartiodactyla; Suina; Suidae; Sus.

RN [1] RP 1-6116 RA Hirano K., Shintani Y., Hirano M., Kanaide H.;

RT ;

RL Submitted (08-APR-2002) to the EMBL/GenBank/DDBJ databases. RL Katsuya Hirano, Graduate School of Medical Sciences, Kyushu University, RL Division of Molecular Cardiology, Research Institute of Angiocardiology;

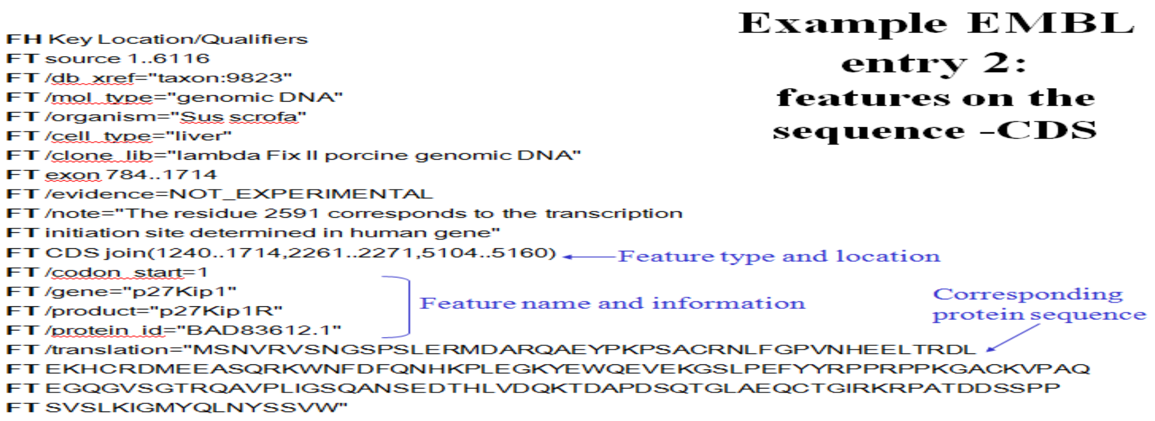
RL 3-1-1 Maidashi, Higashi-ku, Fukuoka, Fukuoka, 812-8582, Japan

RL (E-mail:khirano@molcar.med.kyushu-u.ac.jp, Tel:81-92-642-5550, RL Fax:81-92-642-5552)

RN [2] RA Shintani Y., Hirano K., Hirano M., Nishimura J., Nakano H., Kanaide H.;

RT "Cloning and Charaterization of full sequence of porcine p27Kip1 gene and RT expression of splice isoform p27Kip1R";

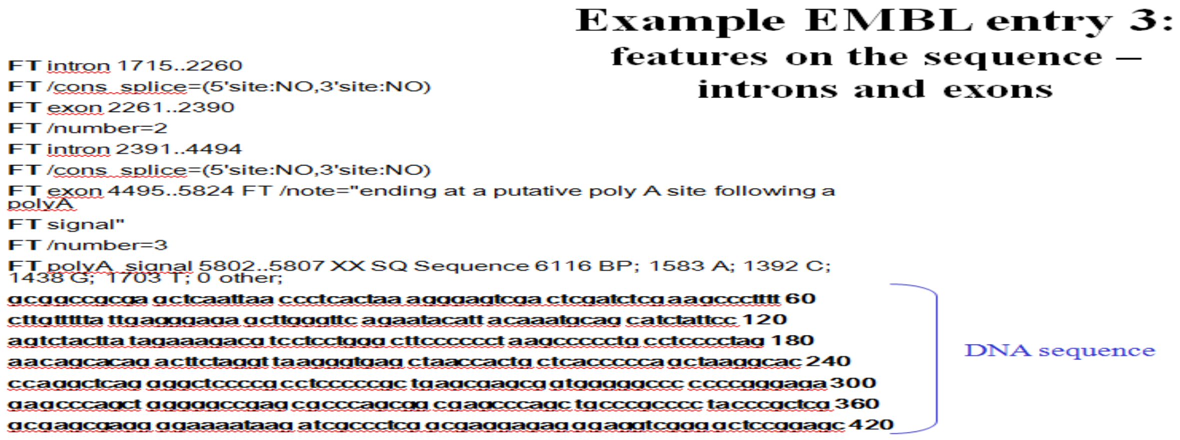
RL Unpublished.



Example EMBL entry 2: features on the sequence -CDS

```
FH Key Location/Qualifiers
FT source 1..6116
FT /db_xref="taxon:9823"
FT /mol_type="genomic DNA"
FT /organism="Sus scrofa"
FT /cell_type="liver"
FT /clone_lib="lambda Fix II porcine genomic DNA"
FT exon 784..1714
FT /evidence=NOT_EXPERIMENTAL
FT /note="The residue 2591 corresponds to the transcription
FT initiation site determined in human gene"
FT CDS join(1240..1714,2261..2271,5104..5160)
FT /codon_start=1
FT /gene="p27Kip1"
FT /product="p27Kip1R"
FT /protein_id="BAD83612.1"
FT /translation="MSNVRVSNGSPSLERMDARQAEYPKPSACRNLFGPVNHEELTRDL
FT EKHCRDMEEASQRKWNFDFQNHKPLEGKYEWQEVEKGSLPEFYYRPPRPPKGACKVPAQ
FT EGQGVSGTRQAVPLIGSQANSEDTHLVDQKTDAPDSQTGLAEQCTGIRKRPATDDSSPP
FT SVSLKIGMYQLNYSSVW"
```

Feature type and location
Feature name and information
Corresponding protein sequence



Example EMBL entry 3: features on the sequence – introns and exons

```
FT intron 1715..2260
FT /cons_splice=(5'site:NO,3'site:NO)
FT exon 2261..2390
FT /number=2
FT intron 2391..4494
FT /cons_splice=(5'site:NO,3'site:NO)
FT exon 4495..5824 FT /note="ending at a putative poly A site following a
polyA
FT signal"
FT /number=3
FT polyA_signal 5802..5807 XX SQ Sequence 6116 BP; 1583 A; 1392 C;
1438 G; 1703 T; 0 other;
gcggccgcga gctcaattaa ccctcactaa agggagtcga ctcgatctcg aagccctttt 60
cttgttttta ttgagggaga gcttgggttc agaatacatt acaaatgcag catctattcc 120
agtctactta tagaaagacg tcctcctggg cttcccccct aagccccctg cctcccctag 180
aacagcacag acttctaggt taagggtgag ctaaccactg ctcaccccca gctaaggcac 240
ccaggctcag gggctccccg cctcccccgc tgagcgagcg gtgggggccc ccccgggaga 300
gagcccagct gggggccgag cgcccagcgg cgagcccagc tgcccgcccc tacccgctcg 360
gcgagcgagg ggaaaataag atcgccctcg gcgaggagag ggaggtcggg gctccggagc 420
```

Fig 2.6 Examples of EMBL entry

2.4.1.2 GENBANK

GenBank is a comprehensive public database of nucleotide sequences and supporting bibliographic and biological annotation, built and distributed by the National Center for Biotechnology Information (NCBI), a division of the National Library of Medicine (NLM), located on the campus of the US National Institutes of Health (NIH) in Bethesda, MD.

NCBI builds GenBank primarily from the submission of sequence data from authors and from the bulk submission of expressed sequence tag (EST), genome survey sequence (GSS) and other high-throughput data from sequencing centers. The US Office of Patents and Trademarks (USPTO) also contributes sequences from issued patents. GenBank incorporates sequences submitted to the EMBL Data Library in the United Kingdom and the DNA Databank of Japan

(DDBJ) as part of a long-standing international collaboration between the three databases in which data are exchanged daily to ensure a uniform and comprehensive collection of sequence information. NCBI makes the GenBank data available at no cost over the Internet, via FTP and a wide range of web-based retrieval and analysis services, which operate on the GenBank data

GenBank continues to grow at an exponential rate with 7.9 million new sequences added over the past 12 months. As of Release 143 in August 2004, GenBank contained over 41.8 billion nucleotide bases from 37.3 million individual sequences. Complete genomes (http://www.ncbi.nlm.nih.gov/Genomes/index.html) represent a growing portion of the database, with over 50 of more than 180 complete microbial genomes in GenBank deposited over the past year. The number of eukaryote genomes for which coverage and assembly are good continues to increase as well, with over 20 such assemblies now available, including that of the reference human genome..

Database sequences are classified and can be queried using a comprehensive sequence-based taxonomy (http://www.ncbi.nlm.nih.gov/Taxonomy/taxonomyhome.html) developed by NCBI in collaboration with EMBL and DDBJ and with the valuable assistance of external advisers and curators. Over 165000 named species are represented in GenBank and new species are being added at the rate of over 2000 per month. About 19% of the sequences in GenBank are of human origin and 13% of all sequences are human ESTs. After *Homo sapiens*, the top species in GenBank in terms of number of bases are *Mus musculus*, *Rattus norvegicus*, *Danio rerio*, *Zea mays*, *Oryza sativa*, *Drosophila melanogaster*, *Gallus gallus* and *Canis familiaris*.

2.4.2 Protein Sequence Databases

- UniProt:
  - Swiss-Prot –manually curated, distinguishes between experimental and computationally derived annotation
  - TrEMBL - Automatic translation of EMBL, no manual curation, some automatic annotation.
- GenPept -GenBank translations.
- RefSeq - Non-redundant sequences for certain organisms.

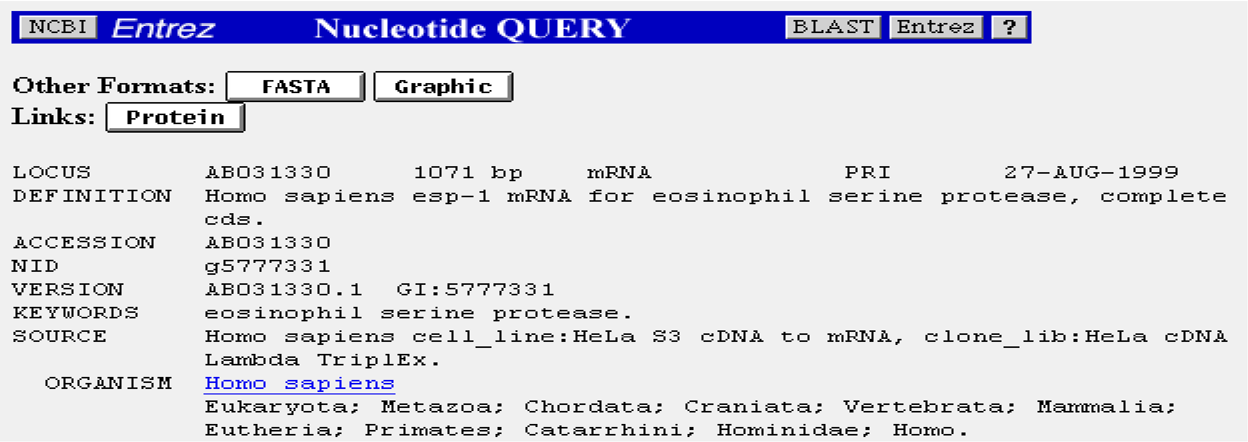
- IPI – International protein Index –combination of many protein sequence databases.
-

2.4.2.1 SWISS-PROT Database

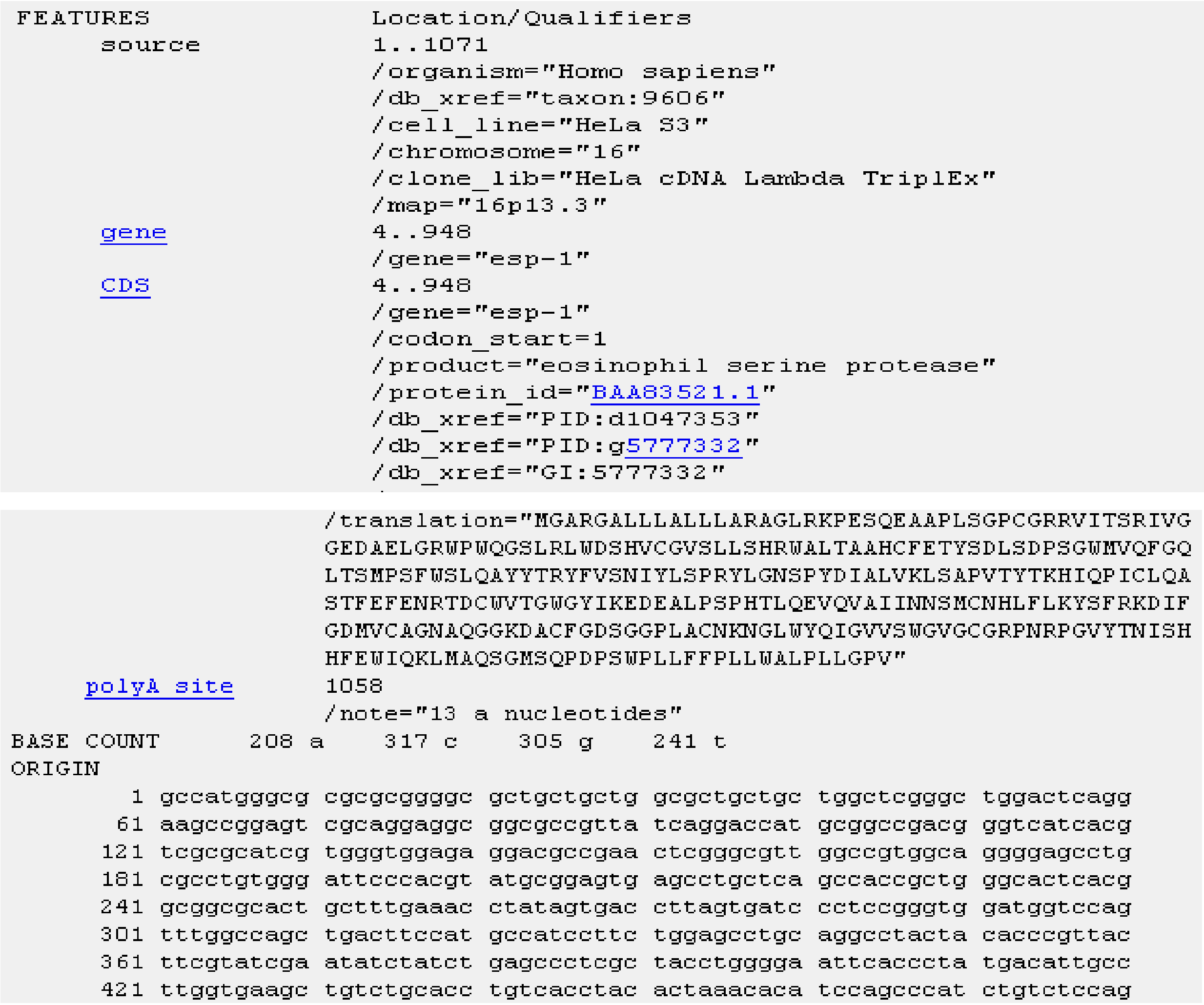
Swiss-Prot is a manually curated biological database of protein sequences. Swiss-Prot was created in 1986 by Amos Bairoch during his PhD and developed by the Swiss Institute of Bioinformatics and the European Bioinformatics Institute.[Swiss-Prot strives to provide reliable protein sequences associated with a high level of annotation (such as the description of the function of a protein, its domains structure, post-translational modifications, variants, etc.), a minimal level of redundancy and high level of integration with other databases.

In 2002, the UniProt consortium was created: it is a collaboration between the Swiss Institute of Bioinformatics, the European Bioinfomatics Institute and the Protein Information Resource (PIR), funded by the National Institutes of Health. Swiss-Prot and its automatically curated supplement TrEMBL, have joined with the Protein Information Resource protein database to produce the UniProt Knowledgebase, the world's most comprehensive catalogue of information on proteins. As of 3 April 2007, UniProtKB/Swiss-Prot release 52.2 contains 263,525 entries. As of 3 April 2007, the UniProtKB/TrEMBL release 35.2 contains 4,232,122 entries. The current Swiss-Prot Release is version 56.4 as of 04-Nov-2008, and contains 400771 entries.

GenBank file format



```
LOCUS       AB031330      1071 bp    mRNA            PRI       27-AUG-1999
DEFINITION  Homo sapiens esp-1 mRNA for eosinophil serine protease, complete
            cds.
ACCESSION   AB031330
NID         g5777331
VERSION     AB031330.1  GI:5777331
KEYWORDS    eosinophil serine protease.
SOURCE      Homo sapiens cell_line:HeLa S3 cDNA to mRNA, clone_lib:HeLa cDNA
            Lambda TriplEx.
  ORGANISM  Homo sapiens
            Eukaryota; Metazoa; Chordata; Craniata; Vertebrata; Mammalia;
            Eutheria; Primates; Catarrhini; Hominidae; Homo.
```



```
FEATURES             Location/Qualifiers
     source          1..1071
                     /organism="Homo sapiens"
                     /db_xref="taxon:9606"
                     /cell_line="HeLa S3"
                     /chromosome="16"
                     /clone_lib="HeLa cDNA Lambda TriplEx"
                     /map="16p13.3"
     gene            4..948
                     /gene="esp-1"
     CDS             4..948
                     /gene="esp-1"
                     /codon_start=1
                     /product="eosinophil serine protease"
                     /protein_id="BAA83521.1"
                     /db_xref="PID:d1047353"
                     /db_xref="PID:g5777332"
                     /db_xref="GI:5777332"

                     /translation="MGARGALLLALLLARAGLRKPESQEAAPLSGPCGRRVITSRIVG
                     GEDAELGRWPWQGSLRLWDSHVCGVSLLSHRWALTAAHCFETYSDLSDPSGWMVQFGQ
                     LTSMPSFWSLQAYYTRYFVSNIYLSPRYLGNSPYDIALVKLSAPVTYTKHIQPICLQA
                     STFEFENRTDCWVTGWGYIKEDEALPSPHTLQEVQVAIINNSMCNHLFLKYSFRKDIF
                     GDMVCAGNAQGGKDACFGDSGGPLACNKNGLWYQIGVVSWGVGCGRPNRPGVYTNISH
                     HFEWIQKLMAQSGMSQPDPSWPLLFFPLLWALPLLGPV"
     polyA_site      1058
                     /note="13 a nucleotides"
BASE COUNT      208 a    317 c    305 g    241 t
ORIGIN
        1 gccatgggcg cgcgcggggc gctgctgctg gcgctgctgc tggctcgggc tggactcagg
       61 aagccggagt cgcaggaggc ggcgccgtta tcaggaccat gcggccgacg ggtcatcacg
      121 tcgcgcatcg tgggtggaga ggacgccgaa ctcgggcgtt ggccgtggca ggggagcctg
      181 cgcctgtggg attcccacgt atgcggagtg agcctgctca gccaccgctg ggcactcacg
      241 gcggcgcact gctttgaaac ctatagtgac cttagtgatc cctccgggtg gatggtccag
      301 tttggccagc tgacttccat gccatccttc tggagcctgc aggcctacta cacccgttac
      361 ttcgtatcga atatctatct gagccctcgc tacctgggga attcacccta tgacattgcc
      421 ttggtgaagc tgtctgcacc tgtcacctac actaaacaca tccagcccat ctgtctccag
```

Fig 2.7 Examples of GENBank file format

The UniProt consortium produced 3 database components, each optimised for different uses. The UniProt Knowledgebase (UniProtKB (Swiss-Prot + TrEMBL)), the UniProt Non-redundant Reference (UniRef) databases, which combine closely related sequences into a single record to speed similarity searches and the UniProt Archive (UniParc), which is a comprehensive repository of protein sequences, reflecting the history of all protein sequences.

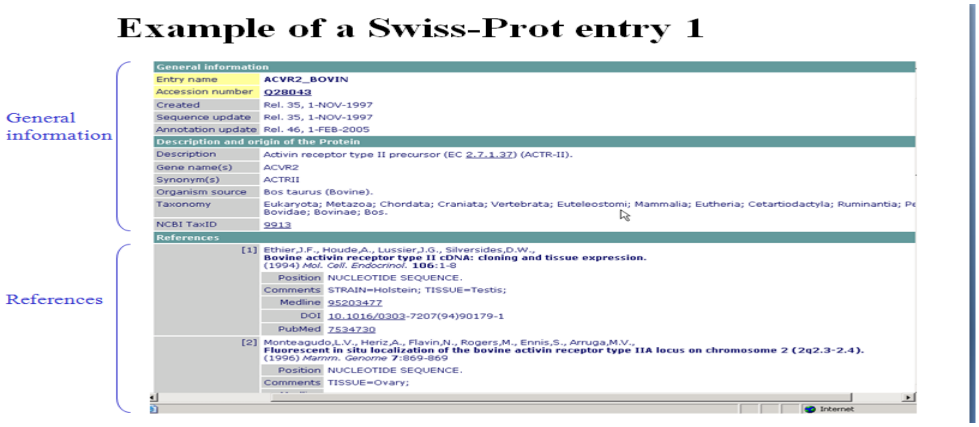
2.4.3 General Genomic Databases

2.4.3.1 **KEGG: Kyoto Encyclopedia of Genes and Genomes**

The KEGG, the Kyoto Encyclopedia of Genes and Genomes, was initiated by the Japanese human genome programme in 1995. According to the developers they consider KEGG to be a "computer representation" of the biological system. The KEGG database can be utilized for modeling and simulation, browsing and retrieval of data. It is a part of the systems biology approach. KEGG maintains five main databases;

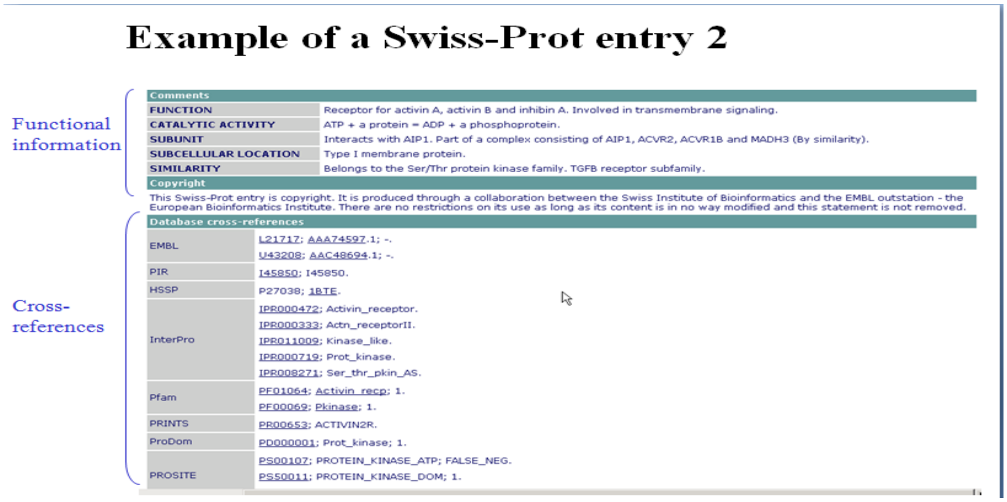
- KEGG Atlas
- KEGG Pathway
- KEGG Genes
- KEGG Ligand
- KEGG BRITE

The KEGG database is designed to facilitate understanding of higher-order protein and cellular functions (e.g., metabolic networks) using genomic and molecular information. Kyoto Encyclopedia of Genes and Genomes (KEGG) is a knowledge base for systematic analysis of gene functions in terms of the networks of genes and molecules. The major component of KEGG is the PATHWAY database that consists of graphical diagrams of biochemical pathways including most of the known metabolic pathways and some of the known regulatory pathways. The pathway information is also represented by the ortholog group tables summarizing orthologous and paralogous gene groups among different organisms.



Example of a Swiss-Prot entry 1

General information	
Entry name	**ACVR2_BOVIN**
Accession number	**Q28043**
Created	Rel. 35, 1-NOV-1997
Sequence update	Rel. 35, 1-NOV-1997
Annotation update	Rel. 46, 1-FEB-2005
Description and origin of the Protein	
Description	Activin receptor type II precursor (EC 2.7.1.37) (ACTR-II).
Gene name(s)	ACVR2
Synonym(s)	ACTRII
Organism source	Bos taurus (Bovine).
Taxonomy	Eukaryota; Metazoa; Chordata; Craniata; Vertebrata; Euteleostomi; Mammalia; Eutheria; Cetartiodactyla; Ruminantia; Pe Bovidae; Bovinae; Bos.
NCBI TaxID	9913
References	
[1]	Ethier,J.F., Houde,A., Lussier,J.G., Silversides,D.W., **Bovine activin receptor type II cDNA: cloning and tissue expression.** (1994) *Mol. Cell. Endocrinol.* **106**:1-8 Position NUCLEOTIDE SEQUENCE. Comments STRAIN=Holstein; TISSUE=Testis; Medline 95203477 DOI 10.1016/0303-7207(94)90179-1 PubMed 7534730
[2]	Monteagudo,L.V., Heriz,A., Flavin,N., Rogers,M., Ennis,S., Arruga,M.V., **Fluorescent in situ localization of the bovine activin receptor type IIA locus on chromosome 2 (2q2.3-2.4).** (1996) *Mamm. Genome* **7**:869-869 Position NUCLEOTIDE SEQUENCE. Comments TISSUE=Ovary;



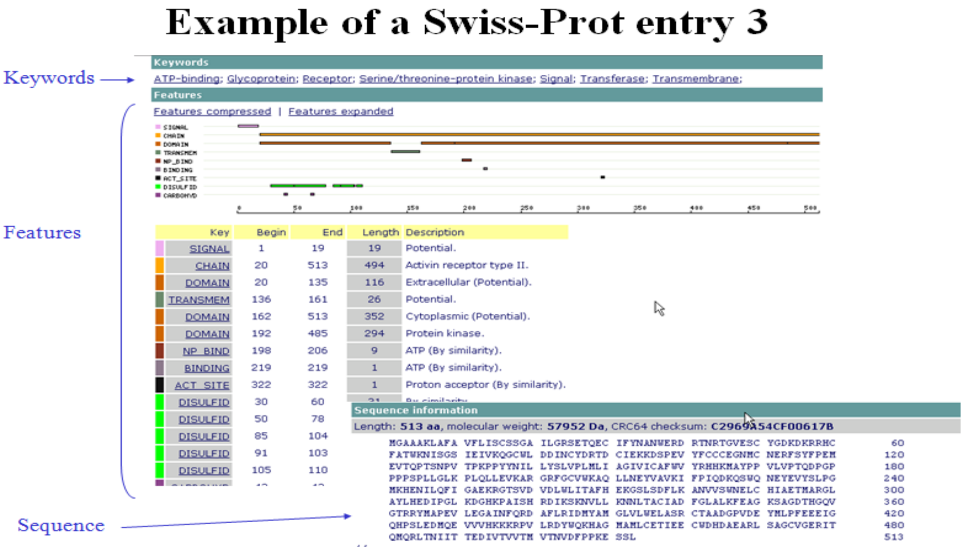
Example of a Swiss-Prot entry 2

Comments	
FUNCTION	Receptor for activin A, activin B and inhibin A. Involved in transmembrane signaling.
CATALYTIC ACTIVITY	ATP + a protein = ADP + a phosphoprotein.
SUBUNIT	Interacts with AIP1. Part of a complex consisting of AIP1, ACVR2, ACVR1B and MADH3 (By similarity).
SUBCELLULAR LOCATION	Type I membrane protein.
SIMILARITY	Belongs to the Ser/Thr protein kinase family. TGFB receptor subfamily.

Copyright

This Swiss-Prot entry is copyright. It is produced through a collaboration between the Swiss Institute of Bioinformatics and the EMBL outstation - the European Bioinformatics Institute. There are no restrictions on its use as long as its content is in no way modified and this statement is not removed.

Database cross-references	
EMBL	L21717; AAA74597.1; -. U43208; AAC48694.1; -.
PIR	I45850; I45850.
HSSP	P27038; 1BTE.
InterPro	IPR000472; Activin_receptor. IPR000333; Actn_receptorII. IPR011009; Kinase_like. IPR000719; Prot_kinase. IPR008271; Ser_thr_pkin_AS.
Pfam	PF01064; Activin_recp; 1. PF00069; Pkinase; 1.
PRINTS	PR00653; ACTIVIN2R.
ProDom	PD000001; Prot_kinase; 1.
PROSITE	PS00107; PROTEIN_KINASE_ATP; FALSE_NEG. PS50011; PROTEIN_KINASE_DOM; 1.



Example of a Swiss-Prot entry 3

Keywords

ATP-binding; Glycoprotein; Receptor; Serine/threonine-protein kinase; Signal; Transferase; Transmembrane;

Features

Features compressed | Features expanded

Key	Begin	End	Length	Description
SIGNAL	1	19	19	Potential.
CHAIN	20	513	494	Activin receptor type II.
DOMAIN	20	135	116	Extracellular (Potential).
TRANSMEM	136	161	26	Potential.
DOMAIN	162	513	352	Cytoplasmic (Potential).
DOMAIN	192	485	294	Protein kinase.
NP_BIND	198	206	9	ATP (By similarity).
BINDING	219	219	1	ATP (By similarity).
ACT_SITE	322	322	1	Proton acceptor (By similarity).
DISULFID	30	60	31	By similarity.
DISULFID	50	78		
DISULFID	85	104		
DISULFID	91	103		
DISULFID	105	110		
CARBOHYD	43	43		

Sequence information

Length: **513 aa**, molecular weight: **57952 Da**, CRC64 checksum: **C2969A54CF00617B**

```
MGAAAKLAFA VFLISCSSGA ILGRSETQEC IFYNANWERD RTNRTGVESC YGDKDKRRHC      60
FATWKNISGS IEIVKQGCWL DDINCYDRTD CIEKKDSPEV YFCCCEGNMC NERFSYFPEM     120
EVTQPTSNPV TPKPPYYNIL LYSLVPLMLI AGIVICAFWV YRHHKMAYPP VLVPTQDPGP     180
PPPSPLLGLK PLQLLEVKAR GRFGCVWKAQ LLNEYVAVKI FPIQDKQSWQ NEYEVYSLPG     240
MKHENILQFI GAEKRGTSVD VDLWLITAFH EKGSLSDFLK ANVVSWNELC HIAETMARGL     300
AYLHEDIPGL KDGHKPAISH RDIKSKNVLL KNNLTACIAD FGLALKFEAG KSAGDTHGQV     360
GTRRYMAPEV LEGAINFQRD AFLRIDMYAM GLVLWELASR CTAADGPVDE YMLPFEEEIG     420
QHPSLEDMQE VVVHKKKRPV LRDYWQKHAG MAMLCETIEE CWDHDAEARL SAGCVGERIT     480
QMQRLTNIIT TEDIVTVVTM VTNVDFPPKE SSL                                  513
```

Fig 2.8 Examples of Swiss Prot Entry

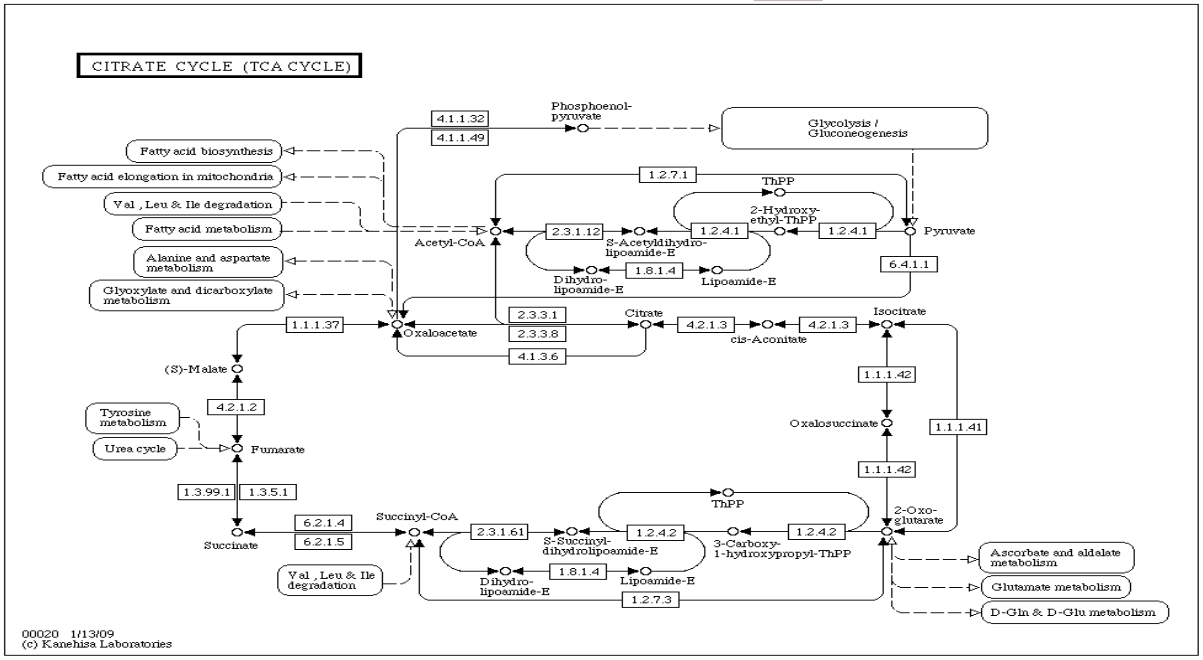


Fig 2.9 An example of a metabolic network of TCA cycle in KEGG Pathways (http://www.genome.ad.jp/kegg/pathway/map/map00020.html)

2.4.4 Unicellular Eukaryotes Genome Databases

2.4.4.1 PlasmoDB: The Plasmodium genome resource

PlasmoDB (http://PlasmoDB.org) is the official database of the *Plasmodium falciparum* genome sequencing consortium. A database integrating experimental and computational data. This resource incorporates the recently completed P. falciparum genome sequence and annotation, as well as draft sequence and annotation emerging from other Plasmodium sequencing projects. The urgent need to identify potential drug and vaccine targets has driven the *P. falciparum* genome sequencing project from its inception. In addition to DNA sequence data and analyses of predicted genes and proteins, Gene Ontology (GO) assignments have been provided by the sequencing centers and others in the malaria research community, and this curated annotation greatly facilitates drug target discovery. To assist in the identification of potential vaccine targets, annotated genes have been scored for potential T-cell epitopes using the SYFPEITHI method .

PlasmoDB currently houses information from five parasite species and provides tools for intra- and inter-species comparisons. Sequence information is integrated with other genomic-scale data emerging from the Plasmodium research community, including gene expression analysis from EST, SAGE and microarray projects and proteomics studies. The relational schema used to build PlasmoDB, GUS (Genomics Unified Schema) employs a highly structured format to accommodate the diverse data types generated by sequence and expression projects. A variety of tools allow researchers to formulate complex, biologically-based, queries of the database. A stand-alone version of the database is also available on CD-ROM (P. falciparum GenePlot), facilitating access to the data in situations where internet access is difficult (e.g. by malaria researchers working in the field). The goal of PlasmoDB is to facilitate utilization of the vast quantities of genomic-scale data produced by the global malaria research community. The software used to develop PlasmoDB has been used to create a second Apicomplexan parasite genome database, ToxoDB (http://ToxoDB.org).

PlasmoDB is a functional genomic database for Plasmodium spp. that provides a resource for data analysis and visualization in a gene-by-gene or genome-wide scale. PlasmoDB belongs to a family of genomic resources that are housed under the EuPathDB (http://EuPathDB.org) Bioinformatics Resource Center (BRC) umbrella. The latest release, PlasmoDB 5.5, contains numerous new data types from several broad categories-annotated genomes, evidence of transcription, proteomics evidence, protein function evidence, population biology and evolution. Data in PlasmoDB can be queried by selecting the data of interest from a query grid or drop down menus. Various results can then be combined with each other on the query history page.

Search results can be downloaded with associated functional data and registered users can store their query history for future retrieval or analysis.

2.5 WHAT IS *PLASMODIUM*?

Plasmodium is a genus of parasitic protozoa. Infection with this genus is known as malaria. The genus *Plasmodium* was created in 1885 by Marchiafava and Celli. Currently over 200 species in this genus are recognized and new species continue to be described.

Plasmodium belongs to the family *Plasmodiidae* (Levine, 1988), order *Haemosporidia* and phylum *Apicomplexa*. There are currently 450 recognised species in this order. Many species of this order are undergoing reexamination of their taxonomy with DNA analysis.

Plasmodium, the parasite responsible for human malaria, is among the most researched genera of parasites in the world. Despite extensive studies on possible control methods, infection in humans continues to grow in tropic and sub-tropic areas.

There are four types of *Plasmodium* which cause human malaria:

- *Plasmodium falciparum*
- *Plasmodium vivax*
- *Plasmodium ovale*
- *Plasmodium malariae*

All of these are transmitted to human hosts solely by way of Anopheles mosquito vectors. *Plasmodium* is one of the oldest known parasites; its long history suggests a long, adaptive relationship with the human host. Today cases of the disease are increasing in non-malarious countries as more people travel to Africa, India, Brazil, and some Asian nations, where the mosquito vectors are most prevalent. Symptoms of the disease may go unnoticed or misdiagnosed; clinical signs include fever, chills, weakness, headache, vomiting, diarrhea, anemia, pulmonary and renal dysfunction, neurologic changes. Untreated malaria may result in death.

Plasmodium is a parasite which is widely distributed all over the world. Because it requires warm, humid environments for replication in the insect vector, malaria-generating species of *Plasmodium* are generally limited to tropical and sub-tropical locations. Global warming and population migrations do have a bearing on *Plasmodium*'s distribution. *Plasmodium falciparum* is the most widespread in tropical and sub-tropical areas. *Plasmodium ovale* is most prevalent in the west coast region of Africa. *Plasmodium malariae* has a widespread distribution area but is fairly scattered within this area. *Plasmodium vivax*, like *falciparum*, ranges over a wide area, but in relatively rare in African countries. A number of methods of control have been tested and some, including use of DDT, have proved worthwhile, but drug resistance and other health concerns make some of these methods undesirable.

Malaria parasites are transmitted from one person to another by the female anopheline mosquito. The males do not transmit the disease as they feed only on plant juices. There are about 380 species of anopheline mosquito, but only 60 or so are able to transmit the parasite. Like all other mosquitos, the anophelines breed in water, each species having its preferred breeding grounds, feeding patterns and resting place. Their sensitivity to insecticides is also highly variable.

Cell Structure and Metabolism

While the four major species of *Plasmodium* differ in some ways from each other, they all share the same complex life cycle involving the insect (mosquito) vector and the human host. When an infected Anophele mosquito bites a human, sporozoites are injected with the saliva. The sporozoites are 10 -15 µm in length and about 1 µm in diameter. They have a thin outer membrane, a double inner membrane below which lies the subpelicular microtubules. They have 3 polar rings and the rhoptries are long, extending half the length of the body. The micronemes, convoluted elongate bodies, run forward to the anterior of the sporozoite entering a common duct with the rhoptries. Mitochondria are located at the posterior end. After entering the circulatory system, the sporozoites make quick work of invading liver cells using the apical organelles (characteristic of all apicomplexans;).

Inside the host's liver cell the *Plasmodium* cell undergoes asexual replication. The products of this replication, called merozoites, are released into the circulatory system. The merozoites invade erythrocites and become enlarged ring-shaped trophozoites. In this stage the cells ingest the host cytoplasm and proteolyze hemoglobin into amino acids. Several rounds of nuclear divison yield a schizont. From these schizonts merozoites bud, which are released after rupturing the erythrocites. More erythrocites are invaded, and the cycle is reinitiated.

Sometimes instead of schizogony (as the schizont cycle is known) the parasites will reproduce sexually into micro- or macrogametocytes. Through gametogenesis these micro- or macrogametocytes morph into micro- or macrogametes. This may only occur after the gametocytes have been ingested by a mosquito. After said ingestion, the microgametocyte undergoes three nuclear divisions; the resulting eight nuclei become associated with thrashing flagella (this process is called exflagellation). The highly motile microgametes fuse with macrogametes and produce a zygote, which then develops into an ookinete. Once reaching the space between the epithelial cells and the basal lamina of the host, the ookinete develops into an oocyst. Asexual replication results in the production of a large number of sporozoites, which are released into the body cavity of the mosquito vector upon the maturation of the oocyst. The sporozoites are able to recognize the salivary gland of the vector, and are injected into the vertebrate host during the mosquito's blood meal, thus beginning the process over again. The four species of *Plasmodium* that affect humans are different morphologically, slightly in terms of their life cycles, in terms of their host erythrocite preferences, and varying clinical symptoms.

The genome of the most common form of *Plasmodium* which causes human malaria, *Plasmodium falciparum*, has been sequenced completely, yielding 14 chromosomes and 5,300 genes--a large number of which are responsible for dodging the host's immunities. The average gene density is approximately 1 gene/4,338 base pairs. The mapping of this genome sequence provides new avenues for research on possible vaccines.

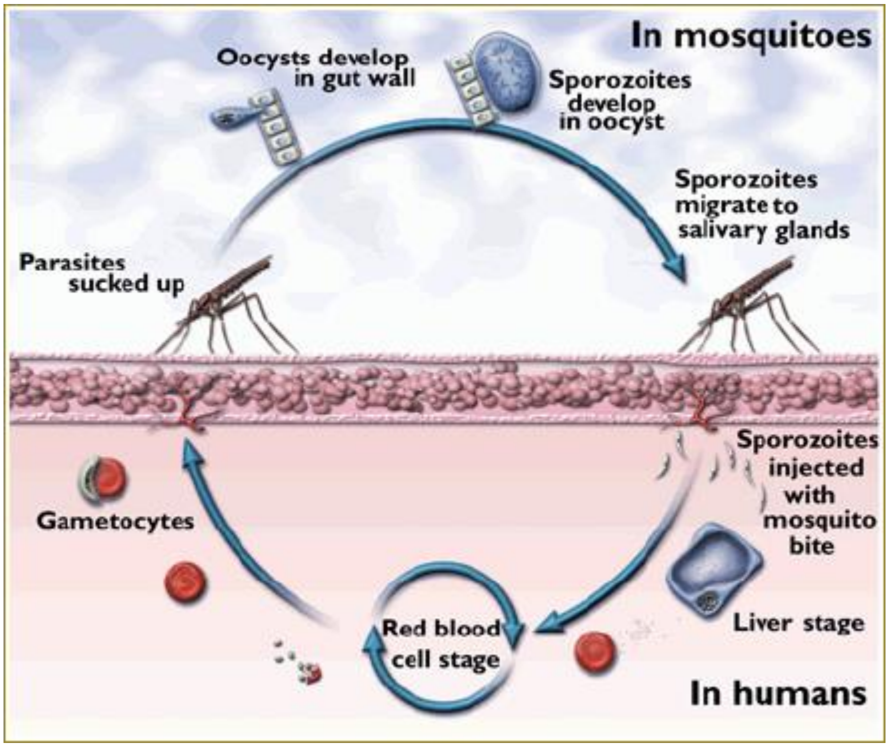

In mosquitoes
Oocysts develop in gut wall
Sporozoites develop in oocyst
Sporozoites migrate to salivary glands
Parasites sucked up
Sporozoites injected with mosquito bite
Gametocytes
Liver stage
Red blood cell stage
In humans

Fig 2.10 Life cycle of *Plasmodium*

CHAPTER THREE

ANALYSIS AND DESIGN

3.1 SYSTEM ARCHITECTURE

Due to the fact that **afriPFdb** is to be made publicly accessible, it is deployed on a web-based architecture. The web works based on the client/server architecture. That is both the server and the client application are responsible for some sort of processing. A Web application is commonly structured as a 3-tier application. The web browser constitutes the first tier, a middleware engine using some dynamic web content technology such as: common gateway interface (CGI), hypertext preprocessor (PHP), java servlets or java server page (JSP), active server pages (ASP) constitute the middle-tier and the database is the third tier.

The middle-tier may be multi-tiered. That is, it can be composed of several other servers with designated responsibilities, hence the over-all architecture is said to be N-tier. A fundamental rule in a 3-tier architecture is that the client has no direct line of communication with the data tier. That is, all communications are routed through the middleware tier.

The client/server technology evolved as a result of downsizing of mainframe applications and upsizing of microcomputer applications. With the popularity of computer networking, the client/server network is particularly ideal for larger number of users/workstations, with the server serving as the repository of all applications or databases, while the clients store their files on or request for services from the server. Transaction processing may be done on both the server and the client, hence the term, distributed processing. There is separation of functionality in client/server technology. The client (front–end) does data presentation and or processing, while the server (back-end) does storage, security and major data processing. The client/server inter-relationship is given in terms of layers and tiers.

Three-Tier Architecture

The three code layers exist on three (3) servers (Presentation, Application and Database server.

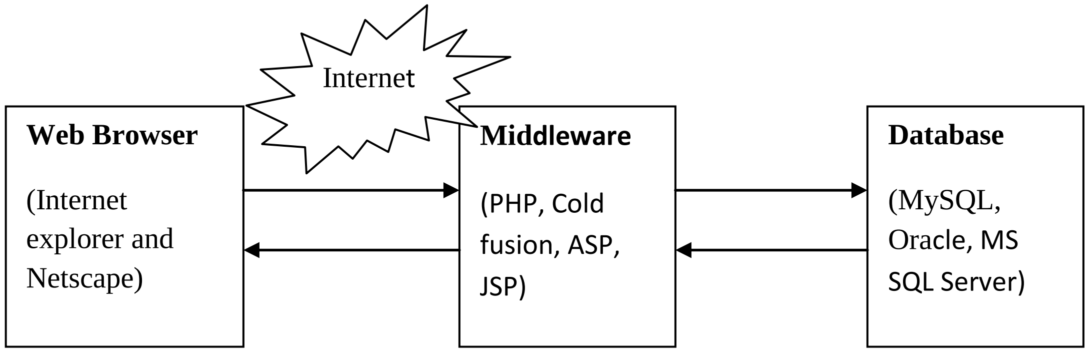

Internet
Web Browser
(Internet explorer and Netscape)
Middleware
(PHP, Cold fusion, ASP, JSP)
Database
(MySQL, Oracle, MS SQL Server)

Figure 3.1 Structure of Web Application

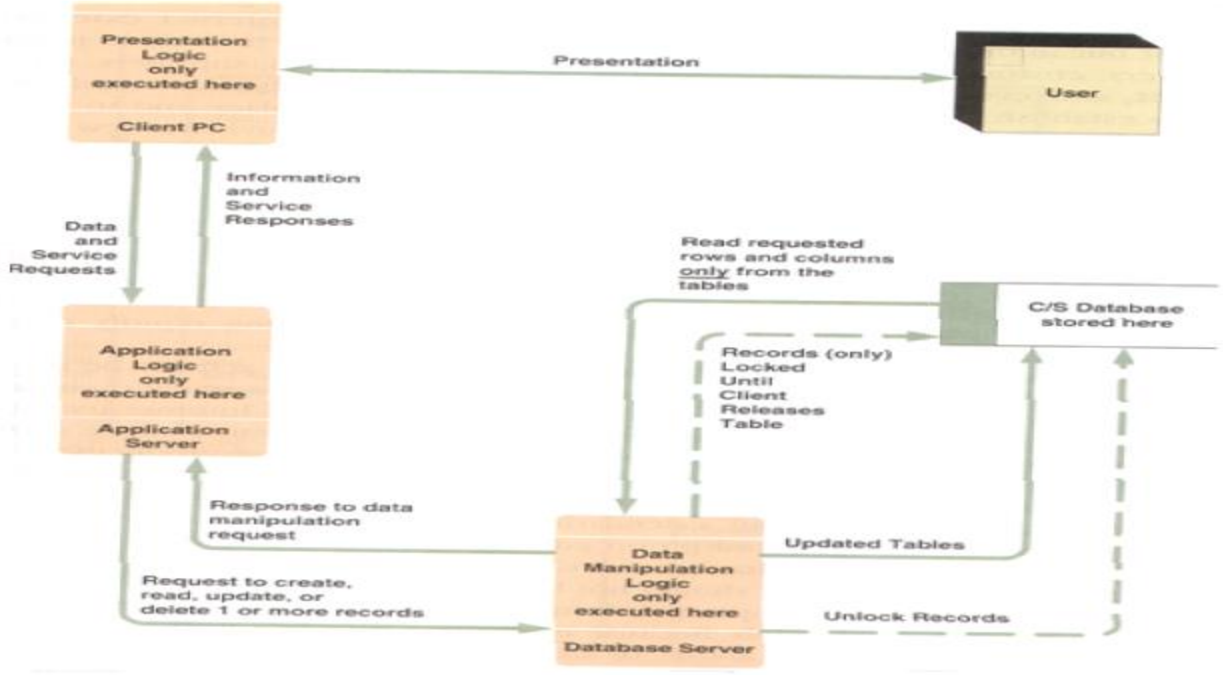

Presentation Logic only executed here
Client PC
Presentation
User
Data and Service Requests
Information and Service Responses
Read requested rows and columns only from the tables
C/S Database stored here
Records (only) Locked Until Client Releases Table
Application Logic only executed here
Application Server
Response to data manipulation request
Data Manipulation Logic only executed here
Database Server
Updated Tables
Request to create, read, update, or delete 1 or more records
Unlock Records

Fig 3.2 Three tier client/server architecture (adapted from system adapted from systems analysis & design methods Whitten et al; 2004)

In afriPFdb, the presentation logic was designed using the HyperText Markup Language (HTML), User requests and response were handled using PHP scripts through Apache web server and PostgreSQL was used as the database server.

3.2 THE MODULES

The current version of the work afriPFdb currently has four main modules;

- Genomics
- Enzymes
- Biological Metabolic Pathways
- Anti-malarial lead compounds

Genomics is the study of an organism's entire genome. The genomics module consists of information on *Plasmodium falciparum*'s genes: gene ids, exon location, encoding enzymes, metabolic pathways, tRNA genes, chromosome list with some 3D structures and the distribution of tRNA genes on the chromosome.

Enzymes : the Enzymes module contains information about *Plasmodium falciparum*'s enzymes: enzyme name, a link to its sequence on PDB (Protein Data Bank), enzyme commission (E.C) number, encoding genes and its structure id if available on PDB. This module also includes a comprehensive list of some predicted drug targets and more information on them.

Metabolic Pathways: this module comprises of list of all the metabolic pathways in *Plasmodium falciparum*'s metabolism and the reactions and genes that make up each pathway. This module also provides information on the gene regulation modalities of two (for now) important pathways of *Plasmodium falciparum* which are the glycolysis and the apicoplast/plastid metabolism.

A **lead compound** in drug discovery is a chemical compound that has pharmacological or biological activity and whose chemical structure is used as a starting point for chemical modifications in order to improve potency, selectivity, absorption, safety and pharmacological properties (Wikipedia, free encyclopedia; accessed 2008). The Anti-malaria lead compounds modules gives a list of the anti malaria lead compound, its structure and a link to its sequence on the NCBI website.

Below is the functional decomposition diagram which shows the decomposition of the system into sub systems.

3.2.1 DECOMPOSITION DIAGRAM

A decomposition diagram also called a hierarchy chart, shows the top-down functional decomposition and structure of a system. A decomposition diagram is essentially a planning tool for more detailed process models namely data flow diagram. See fig 3.3 for the decomposition diagram of this work.

3.2.2 DATA FLOW DIAGRAM (DFD)

A data flow is data in motion. It represents an input of data to a process or the output of data (or information) from a process. A data flow is also used to represent the creation, reading, deletion, or updating of data in a file or a database called a data store.A **data flow diagram** (DFD) is a tool that depicts the flow of data through a system and the work or processing performed by that system. See fig 3.4-6 for the DFD's for the different modules for the system.

3.2.3 WEBSITE DESIGN

See fig 3.7 for the website design.

3.2.4 USE CASE DIAGRAM

Use cases describe the system functions from the perspective of external uses and in a manner and technology they understand. Use cases are result of decomposing the scope of system functionality into smaller statements of system functionality. A Use case represents a single goal of the system and describes a sequence of activities and user interactions in trying to accomplishing the goal. See fig 3.8 for the use case diagram of this work.

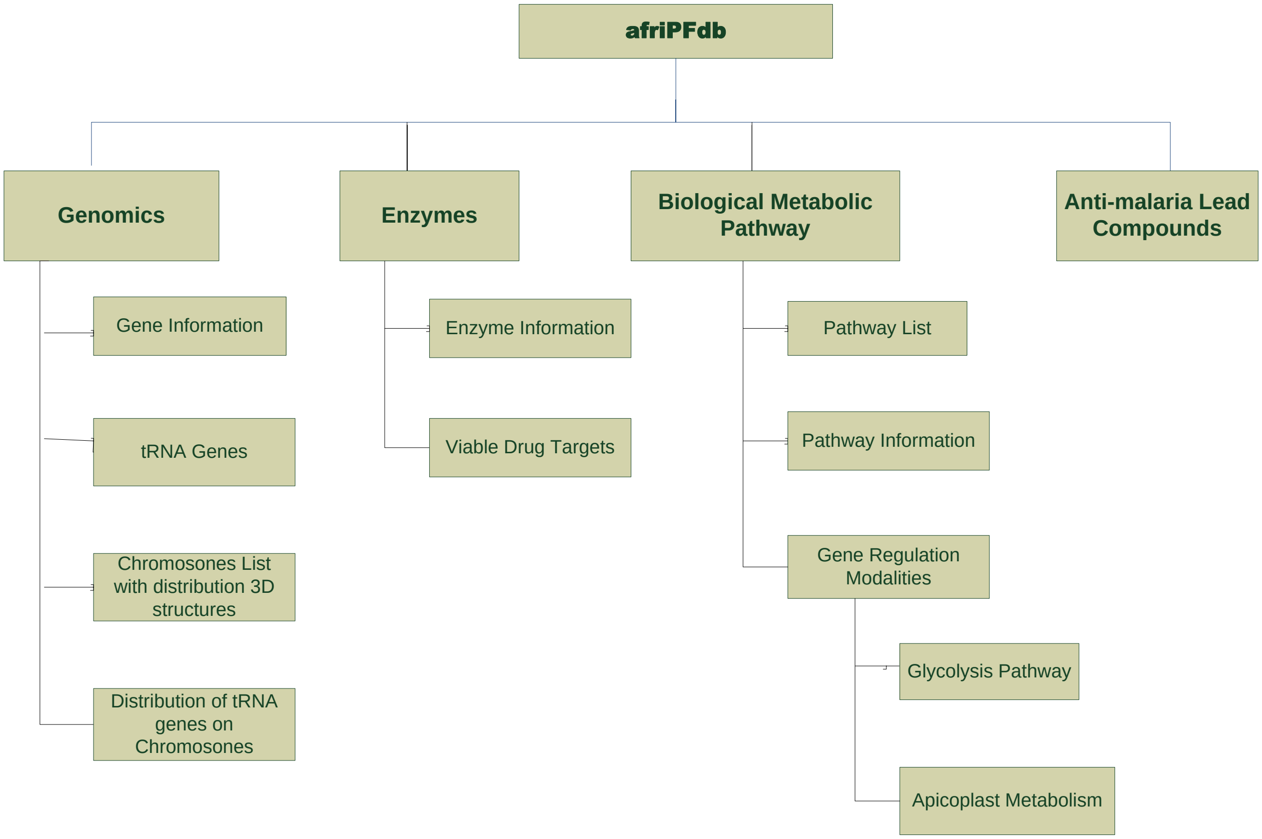

afriPFdb
Genomics
Enzymes
Biological Metabolic Pathway
Anti-malaria Lead Compounds
Gene Information
tRNA Genes
Chromosones List with distribution 3D structures
Distribution of tRNA genes on Chromosones
Enzyme Information
Viable Drug Targets
Pathway List
Pathway Information
Gene Regulation Modalities
Glycolysis Pathway
Apicoplast Metabolism

Fig 3.3 Functional Decomposition Diagram of the System

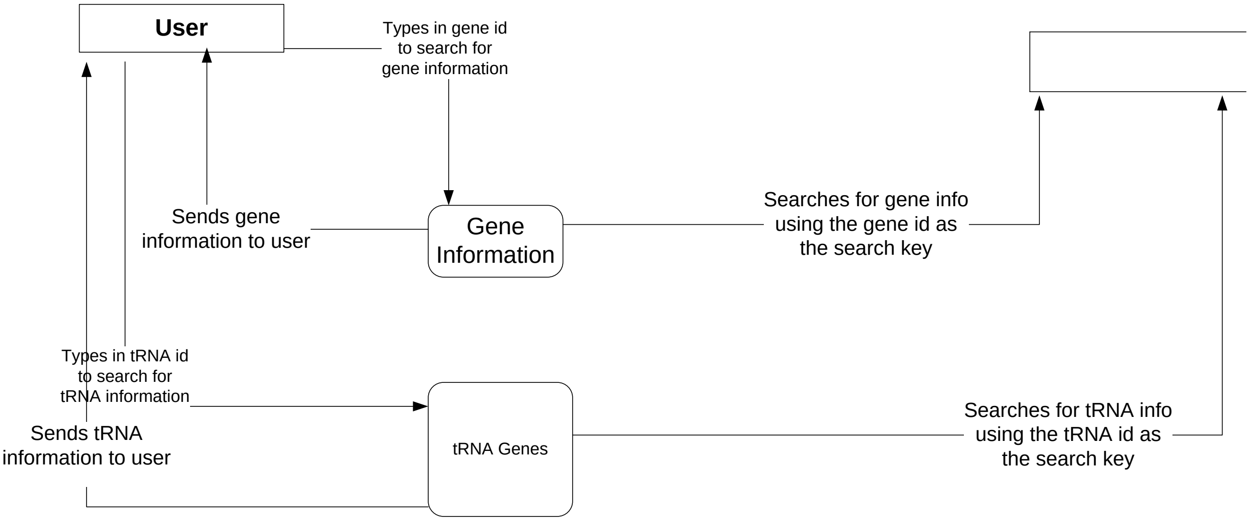

User
Types in gene id to search for gene information
Sends gene information to user
Gene Information
Searches for gene info using the gene id as the search key
Types in tRNA id to search for tRNA information
Sends tRNA information to user
tRNA Genes
Searches for tRNA info using the tRNA id as the search key

Fig 3.4 Data Flow diagram for the Genomics module

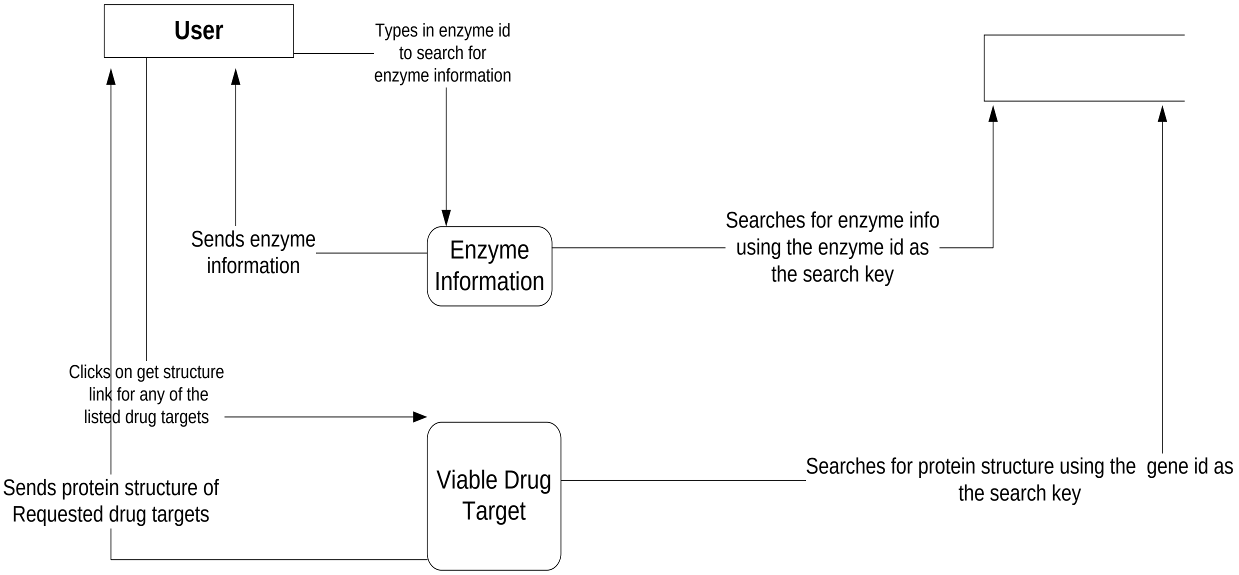

User
Types in enzyme id to search for enzyme information
Sends enzyme information
Enzyme Information
Searches for enzyme info using the enzyme id as the search key
Clicks on get structure link for any of the listed drug targets
Sends protein structure of Requested drug targets
Viable Drug Target
Searches for protein structure using the gene id as the search key

Fig 3.5 Data Flow Diagram for the Enzyme module

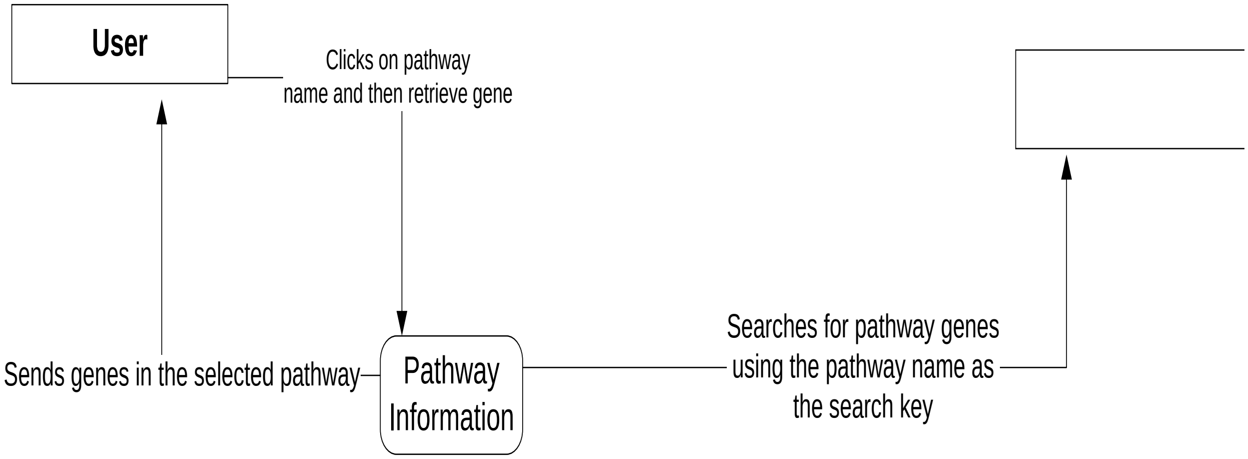
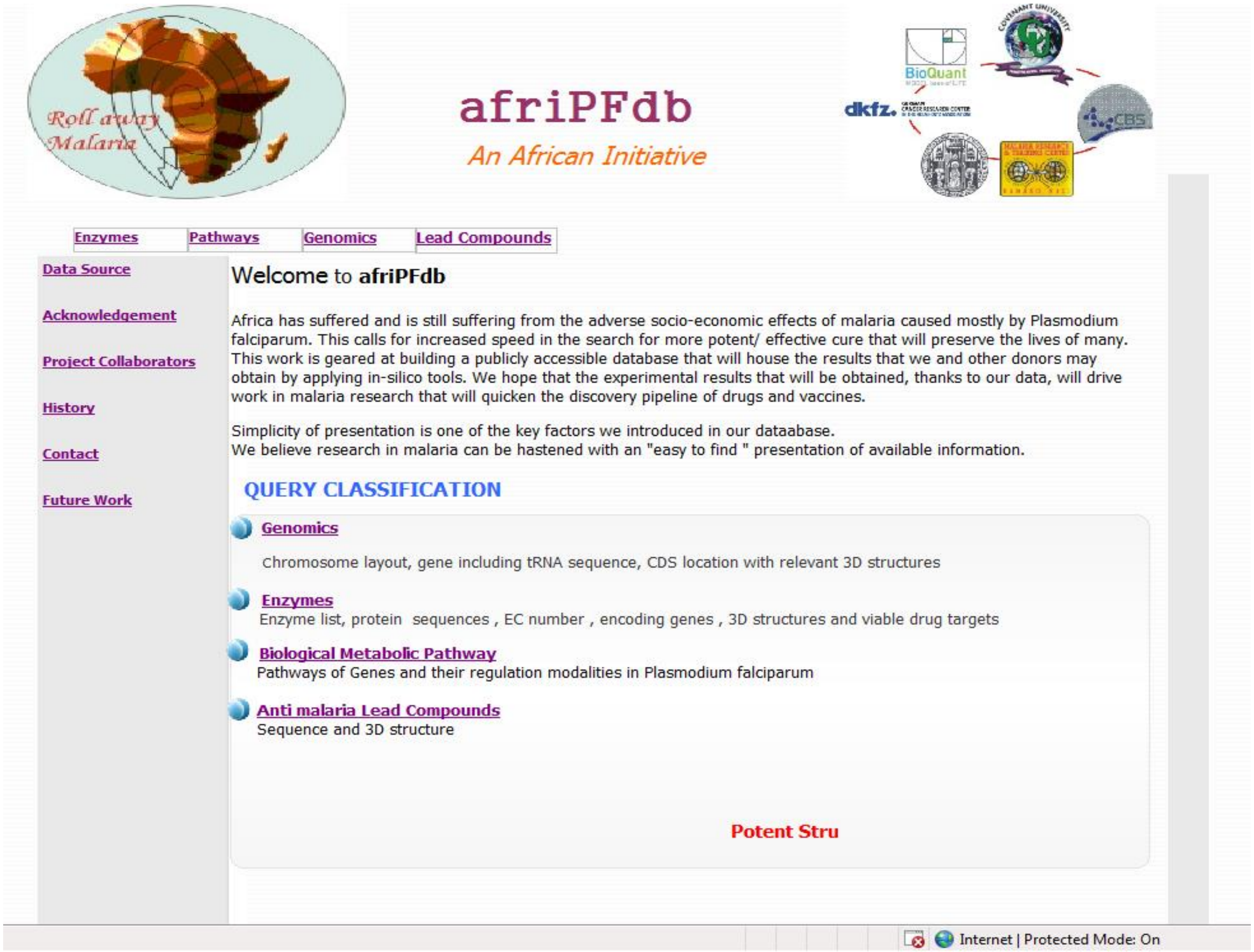

User
Clicks on pathway name and then retrieve gene
Sends genes in the selected pathway
Pathway Information
Searches for pathway genes using the pathway name as the search key

Fig 3.6 Data Flow Diagram for the Pathway module



Welcome to **afriPFdb**

Africa has suffered and is still suffering from the adverse socio-economic effects of malaria caused mostly by Plasmodium falciparum. This calls for increased speed in the search for more potent/ effective cure that will preserve the lives of many. This work is geared at building a publicly accessible database that will house the results that we and other donors may obtain by applying in-silico tools. We hope that the experimental results that will be obtained, thanks to our data, will drive work in malaria research that will quicken the discovery pipeline of drugs and vaccines.

Simplicity of presentation is one of the key factors we introduced in our dataabase.
We believe research in malaria can be hastened with an "easy to find " presentation of available information.

QUERY CLASSIFICATION

- Genomics
  Chromosome layout, gene including tRNA sequence, CDS location with relevant 3D structures
- Enzymes
  Enzyme list, protein sequences , EC number , encoding genes , 3D structures and viable drug targets
- Biological Metabolic Pathway
  Pathways of Genes and their regulation modalities in Plasmodium falciparum
- Anti malaria Lead Compounds
  Sequence and 3D structure

Potent Stru

Fig 3.7 afriPFdb Home Page

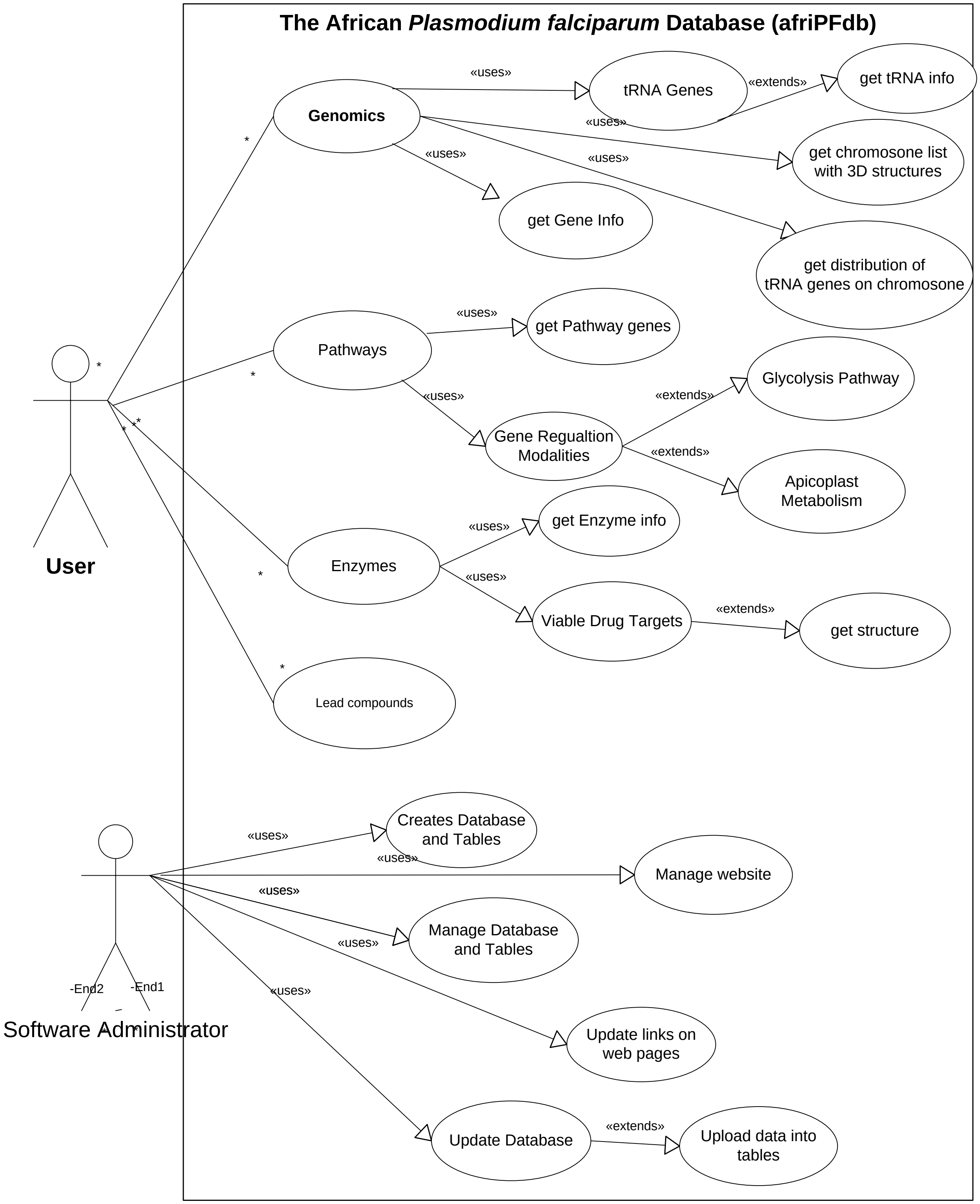


Fig 3.8 Use Case Diagram for the System

The functions of the Administrator include managing the database and the website as well. He functions both as the website manager and the database administrator. As the database administrator, he is responsible for uploading data into the database and ensuring that the database is running and up-to date.

In order for the database administrator to upload genome data into the database there will be a need for the file to be uploaded to be in the right format. The genome data used in this work was got from PlasmoCyc and EMBL flat files. PostgreSQL can only upload files in the following format; Tab delimited, CSV (comma separated values, XML). For this work the CSV file format was used and each of the flat files were converted to CSV files and then uploaded into their respective tables.

3.3 DATABASE DESIGN

PostgreSQL was used as the Relational Database Management System (RDBMS) because it is readily available (open source), cross platform, highly scalable and extensible. As such it can easily accommodate the large amount of biological data that is to be stored. It also allows and the constant changes that will be made to it subsequently. PostgreSQL was also selected because of its easy administration. PostgreSQL makes work easier for the database administrator whose job is to constantly manage and maintain the database. The database design is shown below using the ER (Enity-Relationship) diagram and the description of the database tables. See fig 3.10 for the ER diagram of this work.

3.3.1 THE DESCRIPTION OF TABLES

The database afriPFdb consists of six tables and one view. They are explained below;

1. **Genes:** this table describes each gene of *Plasmodium falciparum* with respect to its gene id, type, its chromosome component, and the enzyme it encodes. The table contains 5800 records (genes). The primary key for this table is gene_id. See table 3.1
2. **Pathways:** this table contains information on the metabolic pathways of *Plasmodium falciparum* such as the Pathway id, common name, the list of reactions in the pathway and its sub pathways (where applicable). This table has 184 records (pathways). The primary key for this table is pathway_id. See table 3.2

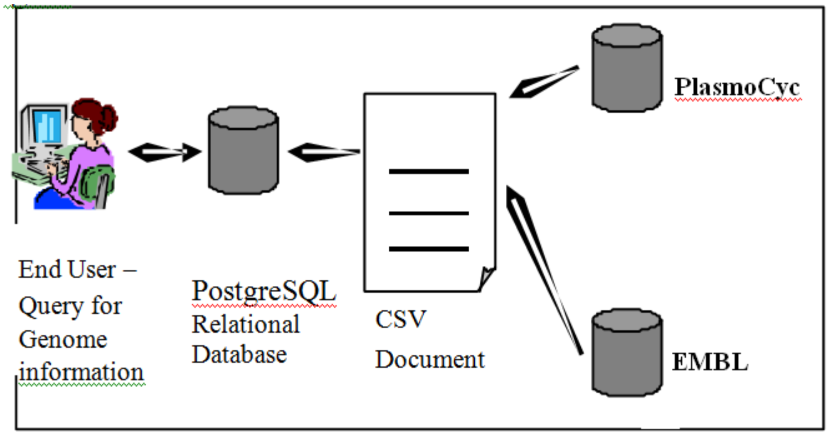


Fig 3.9 Uploading flat files into Database

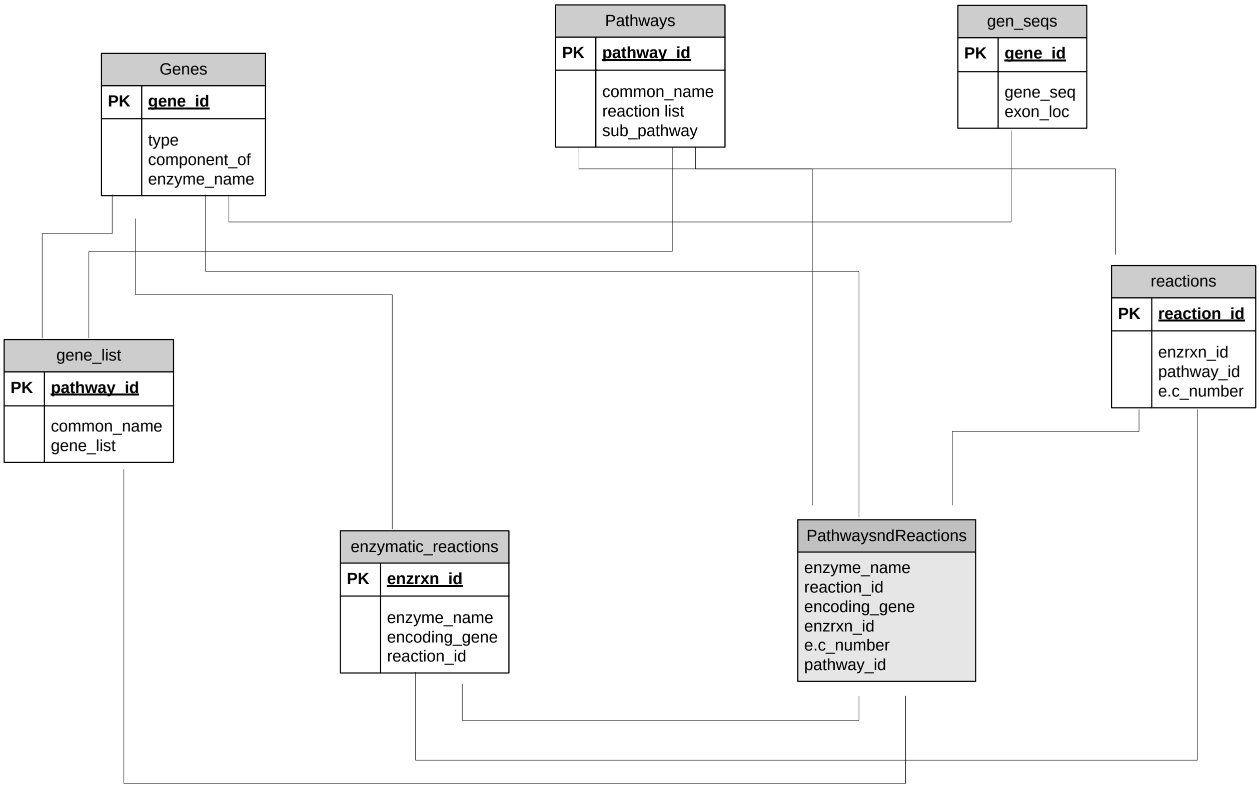
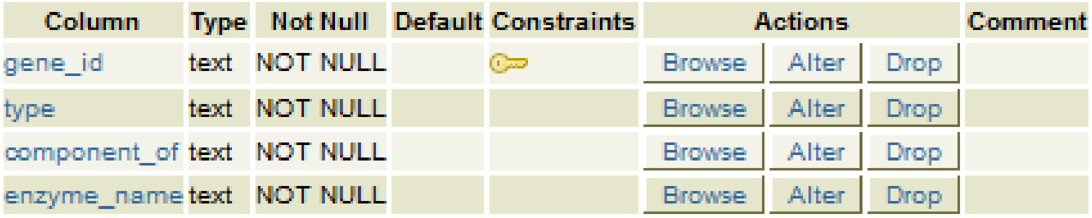


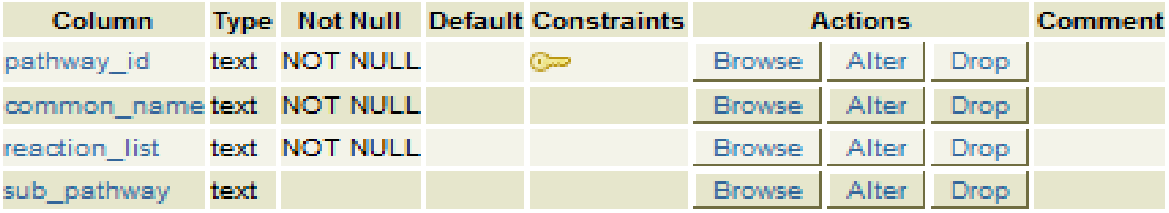
Fig 3.10 Entity Relationship Diagram

Table 3.1 Genes



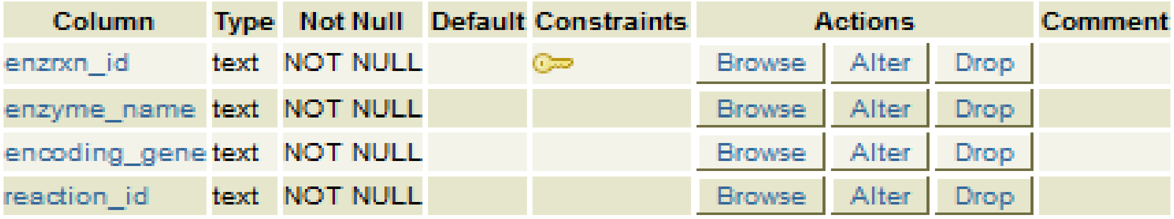
Column	Type	Not Null	Default	Constraints	Actions			Comment
gene_id	text	NOT NULL		🔑	Browse	Alter	Drop	
type	text	NOT NULL			Browse	Alter	Drop	
component_of	text	NOT NULL			Browse	Alter	Drop	
enzyme_name	text	NOT NULL			Browse	Alter	Drop	

Table 3.2 Pathways



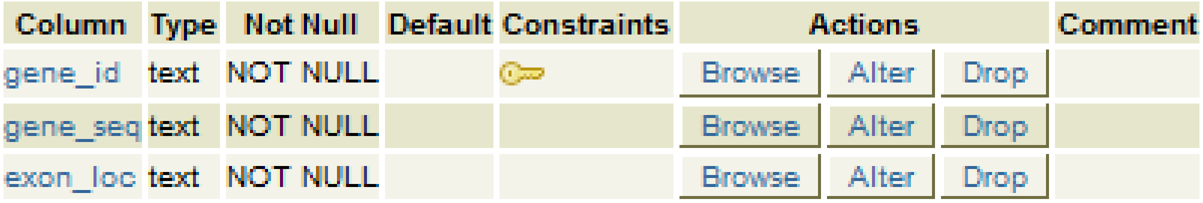
Column	Type	Not Null	Default	Constraints	Actions			Comment
pathway_id	text	NOT NULL		🔑	Browse	Alter	Drop	
common_name	text	NOT NULL			Browse	Alter	Drop	
reaction_list	text	NOT NULL			Browse	Alter	Drop	
sub_pathway	text				Browse	Alter	Drop	

Table 3.3 Enzymatic Reactions



Column	Type	Not Null	Default	Constraints	Actions			Comment
enzrxn_id	text	NOT NULL		🔑	Browse	Alter	Drop	
enzyme_name	text	NOT NULL			Browse	Alter	Drop	
encoding_gene	text	NOT NULL			Browse	Alter	Drop	
reaction_id	text	NOT NULL			Browse	Alter	Drop	

Table 3.4 Gen Seqs



Column	Type	Not Null	Default	Constraints	Actions			Comment
gene_id	text	NOT NULL		🔑	Browse	Alter	Drop	
gene_seq	text	NOT NULL			Browse	Alter	Drop	
exon_loc	text	NOT NULL			Browse	Alter	Drop	

Table 3.5 Gene List

[redacted]

3. **Enzymatic_Reactions:** this table consists of enzymatic reactions in *Plasmodium falciparum*. It includes enzrxn id, enzyme name, encoding gene and reaction id. The enzrxn_id is used as the primary key. It has a total 821 records. See table 3.3

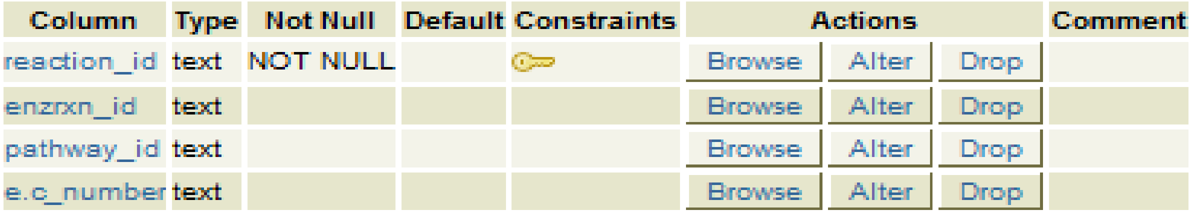
4. **Gen_Seqs:** this table contains the nucleotide sequences of the genes in the genes table together with the exon location for each of the genes. The gene id is used as the primary key and the other attributes should not be null. See table 3.4

5. **Gene_List:** this table contains a list of genes for each of the pathways listed in the Pathways table. Pathway_id is the primary key. See table 3.5

6. **Reactions:** this table contains a list of the reactions with the pathway each of them belong, the enzymes in those reactions and the enzyme commission number. It has a total of 809 records and the reaction_id is the primary key. See table 3.6

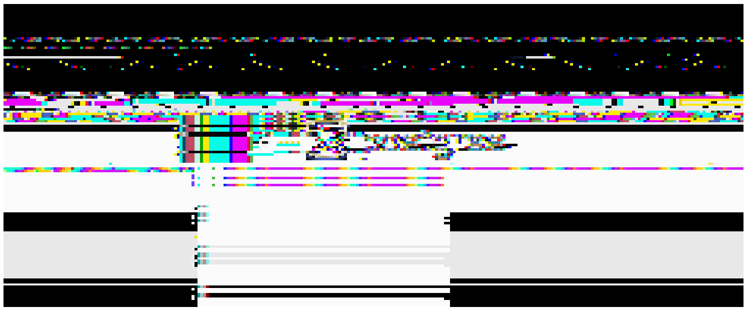
7. **PathwaysndReactions:** this is a view that was created from the Enzymatic Reactions table and the Pathways table. It contains enzyme name, reaction id, encoding gene, enzrxn id, e.c number and pathway id. See table 3.7

Table 3.6 Reactions



Column	Type	Not Null	Default	Constraints	Actions			Comment
reaction_id	text	NOT NULL		🔑	Browse	Alter	Drop	
enzrxn_id	text				Browse	Alter	Drop	
pathway_id	text				Browse	Alter	Drop	
e.c_number	text				Browse	Alter	Drop	

Table 3.7 PathwaysndReactions



CHAPTER FOUR

IMPLEMENTATION

4.1 THE SYSTEM REQUIREMENTS

4.1.1 THE SOFTWARE REQUIREMENTS

Table 4.1 The Software Requirements

Requirement	Software
Operating System	Microsoft Windows Vista
Database Management Database	PostgreSQL
Programming Language	PHP
Development Tool	Adobe Dreamweaver CS3 / Web Page Maker
Web Server	Apache HTTP Server

Table 4.2 The Web client software requirement

Requirement	Software
Operating System	Microsoft Windows Vista / XP / NT / 2000. Linux, Macintosh, etc
Internet Browser	Internet Explorer 6+

4.1.2 THE HARDWARE REQUIREMENTS

Table 4.3 Hardware Requirement

Minimum Requirements
Pentium III, 959MHz, CPU
256MB RAM
14” Color monitor
16 Bit Video Graphic Adapter (VGA)
Modem or Ethernet Card
Keyboard and Mouse
Uninterruptible Power Supply

4.2 THE IMPLEMENTATION TOOLS USED

The tools used include PostgreSQL Relational Database Management System, PHP, Apache HTTP Server all run on Microsoft Windows. WAPPStack form the Bitnami Stacks was used to implement this system. WAPPStack is an easy to install software platform that greatly simplifies the deployment of Open Source web stacks on Windows. It includes ready-to-run versions of Apache, PostgreSQL, PHP, phpPgAdmin and required dependencies and installs in minutes. Dreamwaver and Web Page Maker was also used for the website design and the PHP codes.

4.3 DESCRIPTION OF THE SYSTEM

There are four main modules for the system. They are Genomics, Enzymes, Metabolic Pathways and Lead compounds.

Genomics: to access this module, the user clicks on the genomics link on the Home page or any other page that has the Genomics link. See fig 4.1

Genomics.php: when the user inputs the gene id of a particular gene, the genomics.php is called. It uses the gene id as the search key and query the Gen_Seqs and the PathwaysndReactions tables to produce the exon location of the sequence, the enzyme that the gene encodes, the pathway(s) the gene belongs and the gene sequence. See fig 4.2

tRNA Genes: this page gives information on tRNA genes i.e tRNA id, sequence and its location. See fig 4.3

tRNA.php: this is triggered when the user types in any tRNA id and clicks on the search button. See fig 4.4

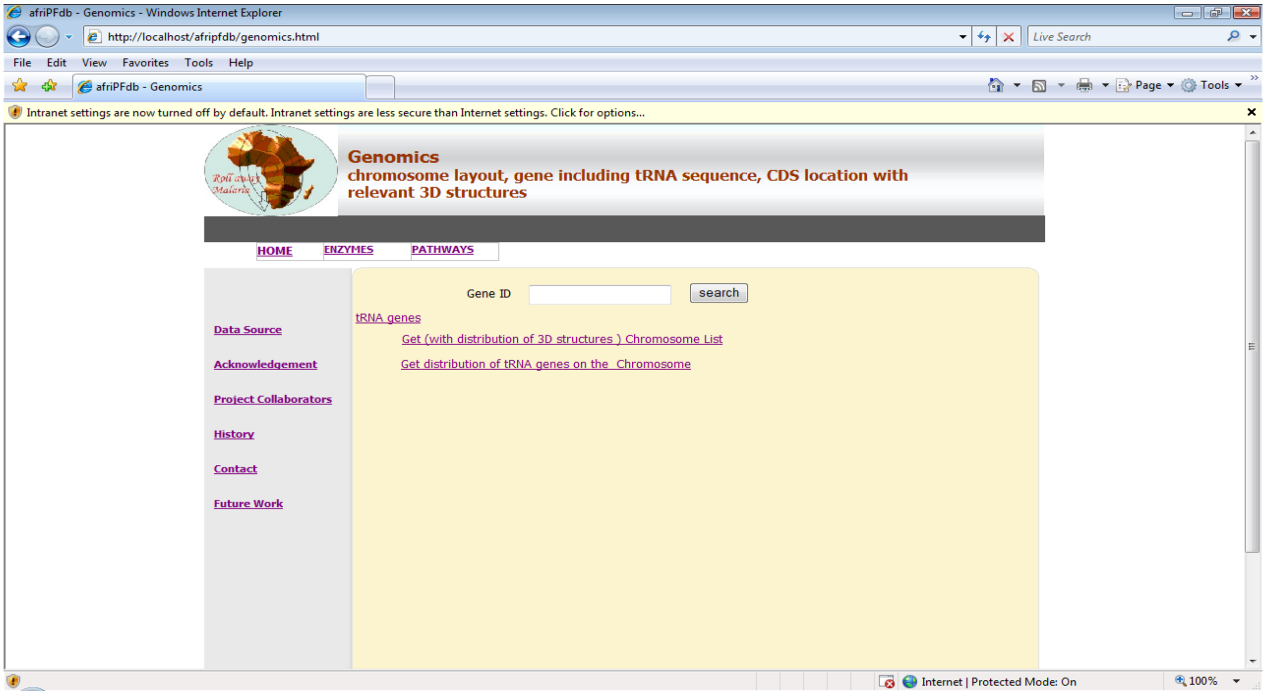
Distribution of 3D structures on chromosones: this page shows the distribution of the genes on the each of *Plasmodium falciparum*'s chromosones. There are 14 chromosones in *Plasmodium falciparum*. The page also provides a link to each of the structures of these genes on Protein Data Bank (PDB). See fig 4.5

Distribution of tRNA genes on chromosones: See fig 4.6

Enzymes

This module can be accessed when the user clicks on the Enzymes link on the Home page or any other page that has the Enzymes link. See fig 4.7

Enzyme.php: this is called **w**hen the user inputs an enzyme name on the enzyme page and clicks on retrieve enzyme. See fig 4.8


afriPFdb - Genomics - Windows Internet Explorer
http://localhost/afripfdb/genomics.html
Live Search
File Edit View Favorites Tools Help
afriPFdb - Genomics
Page
Tools
Intranet settings are now turned off by default. Intranet settings are less secure than Internet settings. Click for options...
Roll away Malaria

Genomics
chromosome layout, gene including tRNA sequence, CDS location with relevant 3D structures

HOME ENZYMES PATHWAYS

Data Source
Acknowledgement
Project Collaborators
History
Contact
Future Work

Gene ID search

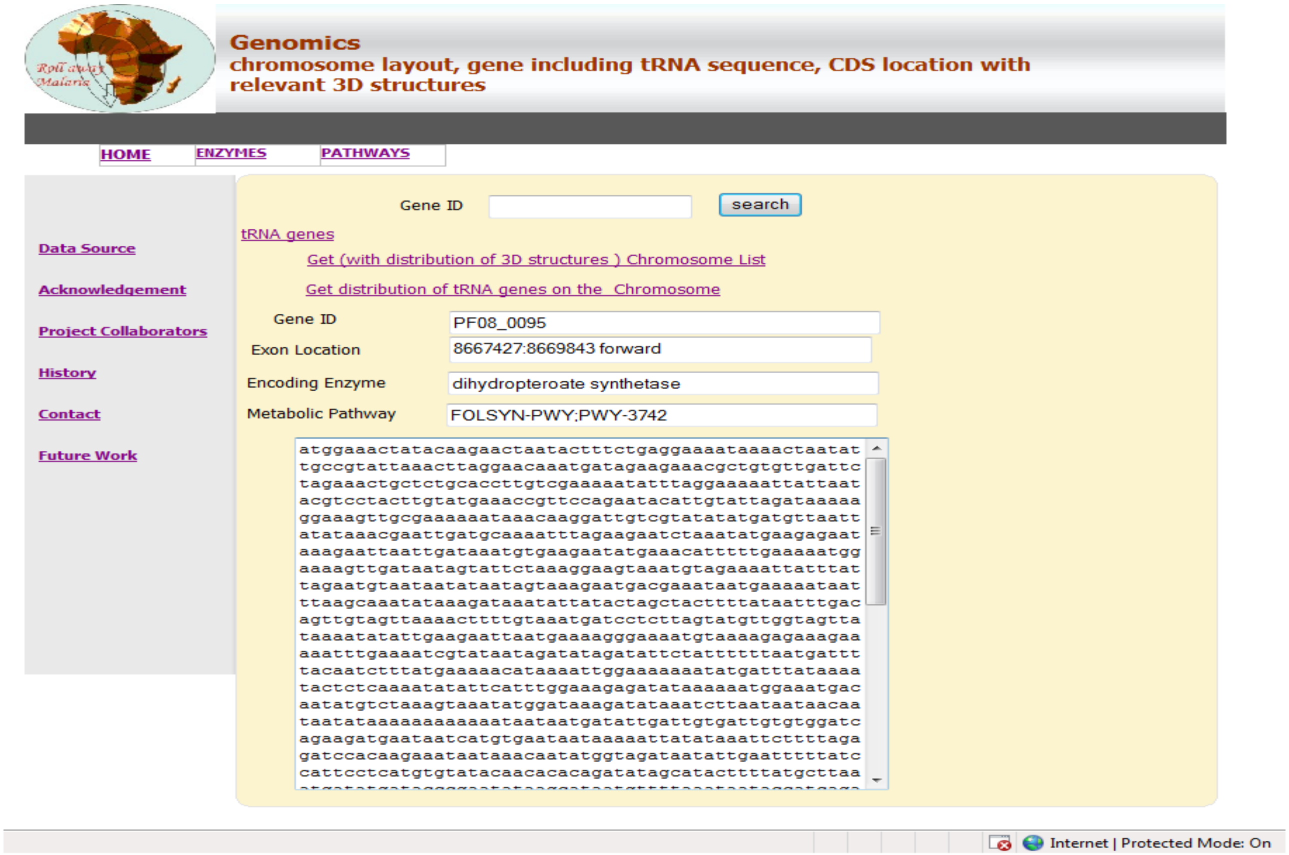
tRNA genes

Get (with distribution of 3D structures) Chromosome List

Get distribution of tRNA genes on the Chromosome

Internet | Protected Mode: On
100%

Fig 4.1 Genomics Page



Genomics
chromosome layout, gene including tRNA sequence, CDS location with relevant 3D structures

HOME ENZYMES PATHWAYS

Data Source
Acknowledgement
Project Collaborators
History
Contact
Future Work

Gene ID search

tRNA genes

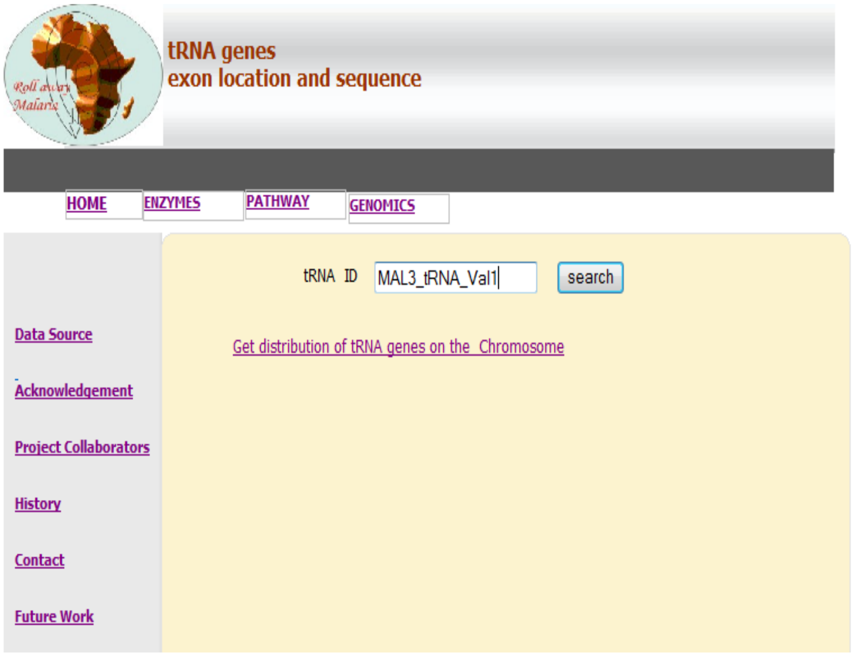
Get (with distribution of 3D structures) Chromosome List

Get distribution of tRNA genes on the Chromosome

Gene ID	PF08_0095
Exon Location	8667427:8669843 forward
Encoding Enzyme	dihydropteroate synthetase
Metabolic Pathway	FOLSYN-PWY;PWY-3742

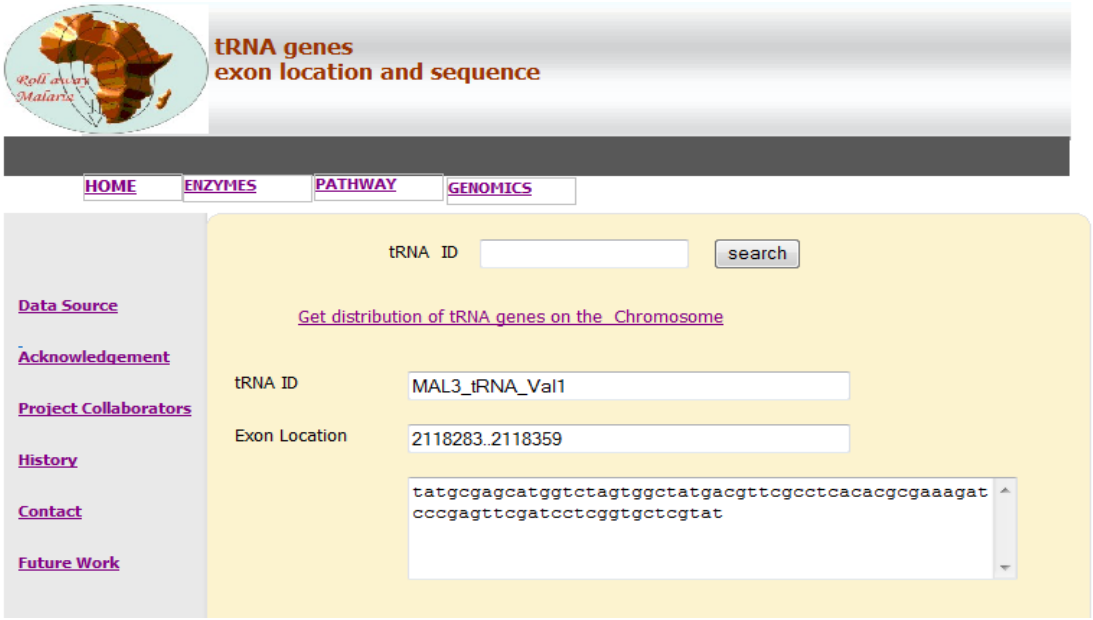
```
atggaaactatacaagaactaatactttctgaggaaaataaaactaatat
tgccgtattaaacttaggaacaaatgatagaagaaacgctgtgttgattc
tagaaactgctctgcaccttgtcgaaaaatatttaggaaaaattattaat
acgtcctacttgtatgaaaccgttccagaatacattgtattagataaaaa
ggaaagttgcgaaaaaataaacaaggattgtcgtatatatgatgttaatt
atataaacgaattgatgcaaaatttagaagaatctaaatatgaagagaat
aaagaattaattgataaatgtgaagaatatgaaacatttttgaaaaatgg
aaaagttgataatagtattctaaaggaagtaaatgtagaaaattatttat
tagaatgtaataatataatagtaaagaatgacgaaataatgaaaaataat
ttaagcaaatataaagataaatattatactagctacttttataatttgac
agttgtagttaaaacttttgtaaatgatcctcttagtatgttggtagtta
taaaatatattgaagaattaatgaaaagggaaaatgtaaaagagaaagaa
aaatttgaaaatcgtataatagatatagatattctatttttaatgattt
tacaatctttatgaaaaacataaaattggaaaaaaatatgatttataaaa
tactctcaaaatatattcatttggaaagagatataaaaaatggaaatgac
aatatgtctaaagtaaatatggataaagatataaatcttaataataacaa
taatataaaaaaaaaaaataataatgatattgattgtgattgtgtggatc
agaagatgaataatcatgtgaataataaaaattatataaattcttttaga
gatccacaagaaataataaacaatatggtagataatattgaattttttatc
cattcctcatgtgtatacaacacacagatatagcatacttttatgcttaa
```

Fig 4.2 Genomics.php



tRNA ID MAL3_tRNA_Val1

Fig 4.3 tRNA genes



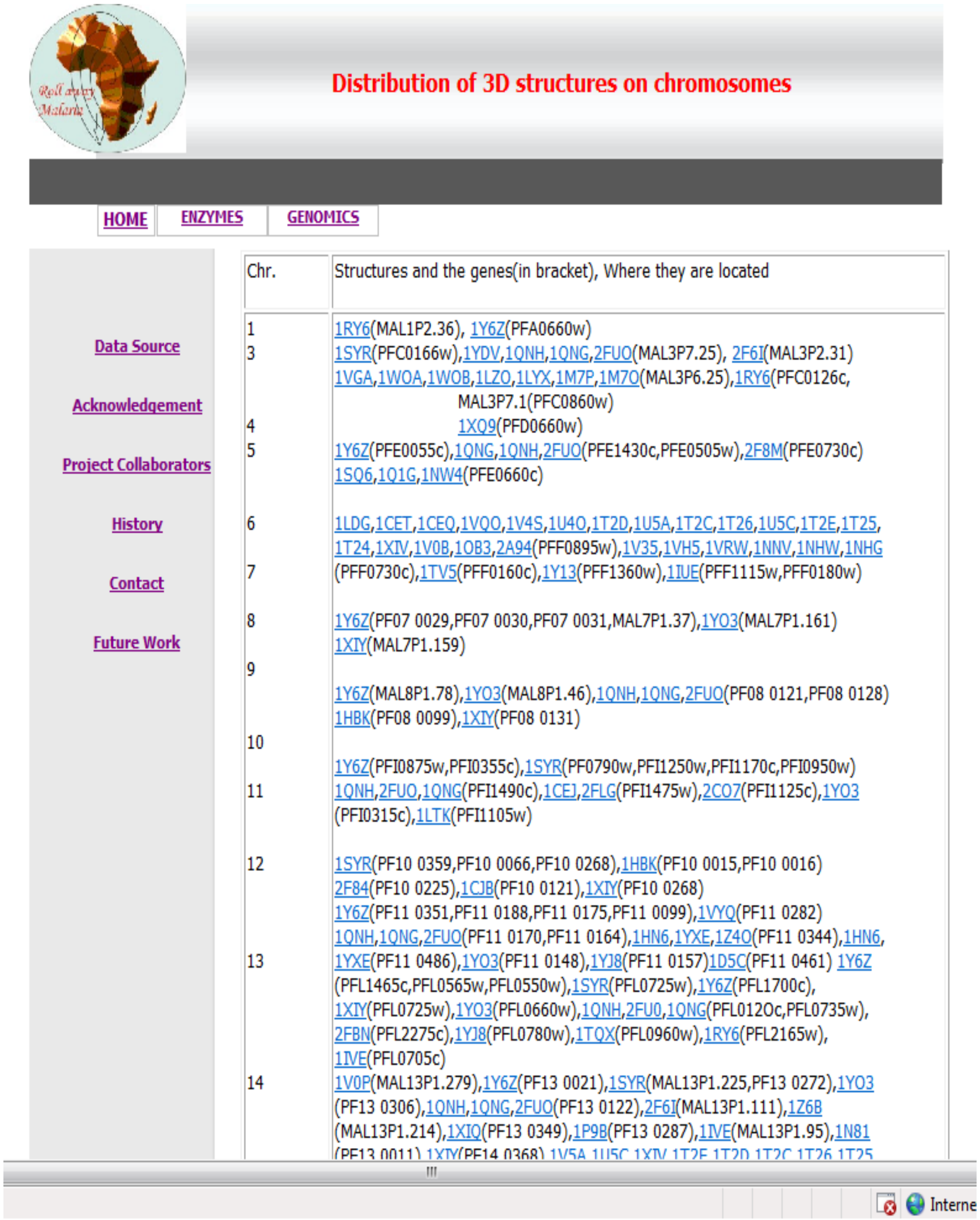
tRNA ID

tRNA ID MAL3_tRNA_Val1

Exon Location 2118283..2118359

```
tatgcgagcatggtctagtggctatgacgttcgcctcacacgcgaaagat
cccgagttcgatcctcggtgctcgtat
```

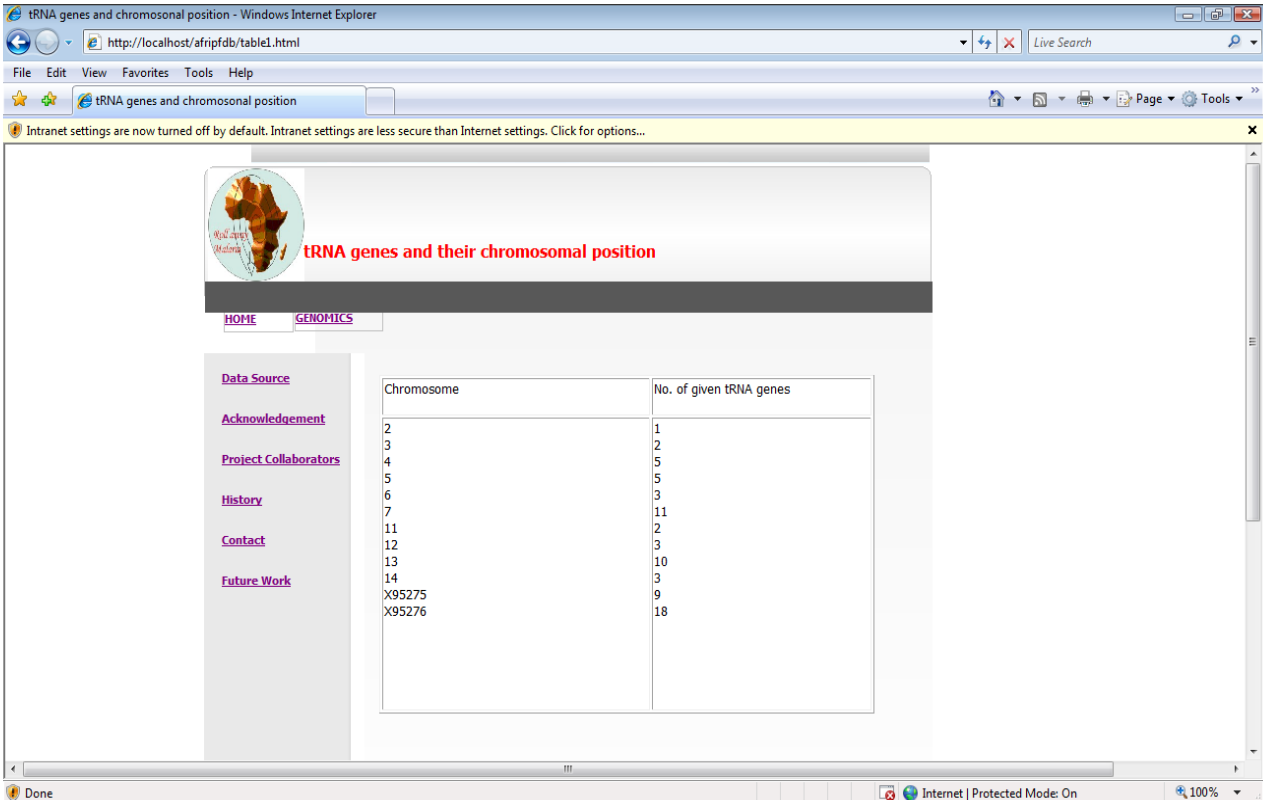
Fig 4.4 tRNA.php



Distribution of 3D structures on chromosomes

Chr.	Structures and the genes(in bracket), Where they are located
1	1RY6(MAL1P2.36), 1Y6Z(PFA0660w)
3	1SYR(PFC0166w),1YDV,1QNH,1QNG,2FUO(MAL3P7.25), 2F6I(MAL3P2.31)
	1VGA,1WOA,1WOB,1LZO,1LYX,1M7P,1M7O(MAL3P6.25),1RY6(PFC0126c,
	MAL3P7.1(PFC0860w)
4	1XQ9(PFD0660w)
5	1Y6Z(PFE0055c),1QNG,1QNH,2FUO(PFE1430c,PFE0505w),2F8M(PFE0730c)
	1SQ6,1Q1G,1NW4(PFE0660c)
6	1LDG,1CET,1CEQ,1VQO,1V4S,1U4O,1T2D,1U5A,1T2C,1T26,1U5C,1T2E,1T25,
	1T24,1XIV,1V0B,1OB3,2A94(PFF0895w),1V35,1VH5,1VRW,1NNV,1NHW,1NHG
7	(PFF0730c),1TV5(PFF0160c),1Y13(PFF1360w),1IUE(PFF1115w,PFF0180w)
8	1Y6Z(PF07 0029,PF07 0030,PF07 0031,MAL7P1.37),1YO3(MAL7P1.161)
	1XIY(MAL7P1.159)
9	
	1Y6Z(MAL8P1.78),1YO3(MAL8P1.46),1QNH,1QNG,2FUO(PF08 0121,PF08 0128)
	1HBK(PF08 0099),1XIY(PF08 0131)
10	
	1Y6Z(PFI0875w,PFI0355c),1SYR(PF0790w,PFI1250w,PFI1170c,PFI0950w)
11	1QNH,2FUO,1QNG(PFI1490c),1CEJ,2FLG(PFI1475w),2CO7(PFI1125c),1YO3
	(PFI0315c),1LTK(PFI1105w)
12	1SYR(PF10 0359,PF10 0066,PF10 0268),1HBK(PF10 0015,PF10 0016)
	2F84(PF10 0225),1CJB(PF10 0121),1XIY(PF10 0268)
	1Y6Z(PF11 0351,PF11 0188,PF11 0175,PF11 0099),1VYQ(PF11 0282)
	1QNH,1QNG,2FUO(PF11 0170,PF11 0164),1HN6,1YXE,1Z4O(PF11 0344),1HN6,
13	1YXE(PF11 0486),1YO3(PF11 0148),1YJ8(PF11 0157)1D5C(PF11 0461) 1Y6Z
	(PFL1465c,PFL0565w,PFL0550w),1SYR(PFL0725w),1Y6Z(PFL1700c),
	1XIY(PFL0725w),1YO3(PFL0660w),1QNH,2FU0,1QNG(PFL012Oc,PFL0735w),
	2FBN(PFL2275c),1YJ8(PFL0780w),1TQX(PFL0960w),1RY6(PFL2165w),
	1IVE(PFL0705c)
14	1V0P(MAL13P1.279),1Y6Z(PF13 0021),1SYR(MAL13P1.225,PF13 0272),1YO3
	(PF13 0306),1QNH,1QNG,2FUO(PF13 0122),2F6I(MAL13P1.111),1Z6B
	(MAL13P1.214),1XIQ(PF13 0349),1P9B(PF13 0287),1IVE(MAL13P1.95),1N81
	(PF13 0011),1XIY(PF14 0368),1V5A,1U5C,1XIV,1T2E,1T2D,1T2C,1T26,1T25

Fig 4.5 Distribution of 3D structure


tRNA genes and chromosonal position - Windows Internet Explorer
http://localhost/afripfdb/table1.html
Live Search
File
Edit
View
Favorites
Tools
Help
tRNA genes and chromosonal position
Page
Tools
Intranet settings are now turned off by default. Intranet settings are less secure than Internet settings. Click for options...
Done
Internet | Protected Mode: On
100%

tRNA genes and their chromosomal position

HOME GENOMICS

Data Source

Acknowledgement

Project Collaborators

History

Contact

Future Work

Chromosome	No. of given tRNA genes
2	1
3	2
4	5
5	5
6	3
7	11
11	2
12	3
13	10
14	3
X95275	9
X95276	18

Fig 4.6 tRNA genes on chromosones

Fig 4.7 Enzyme Page

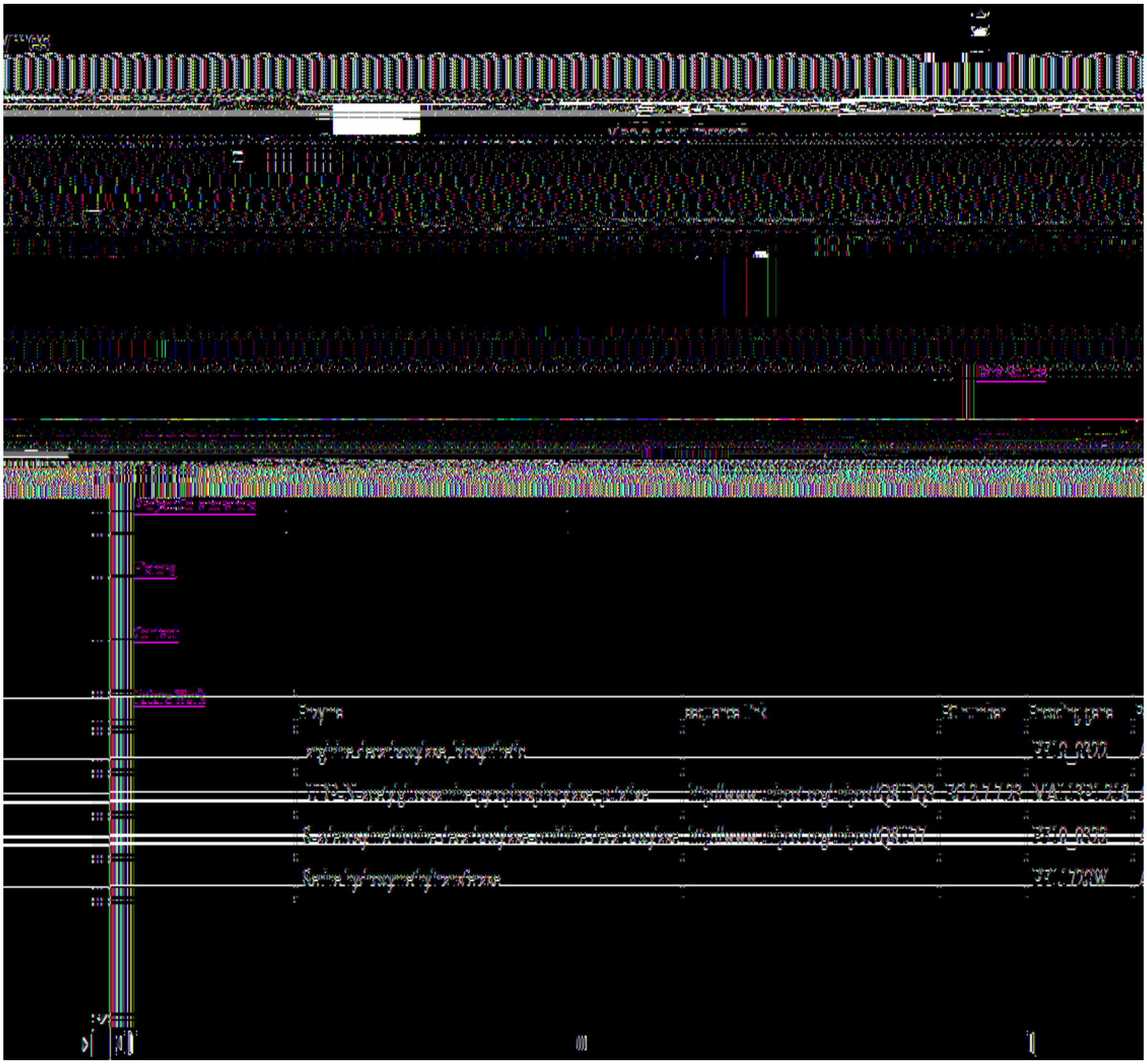
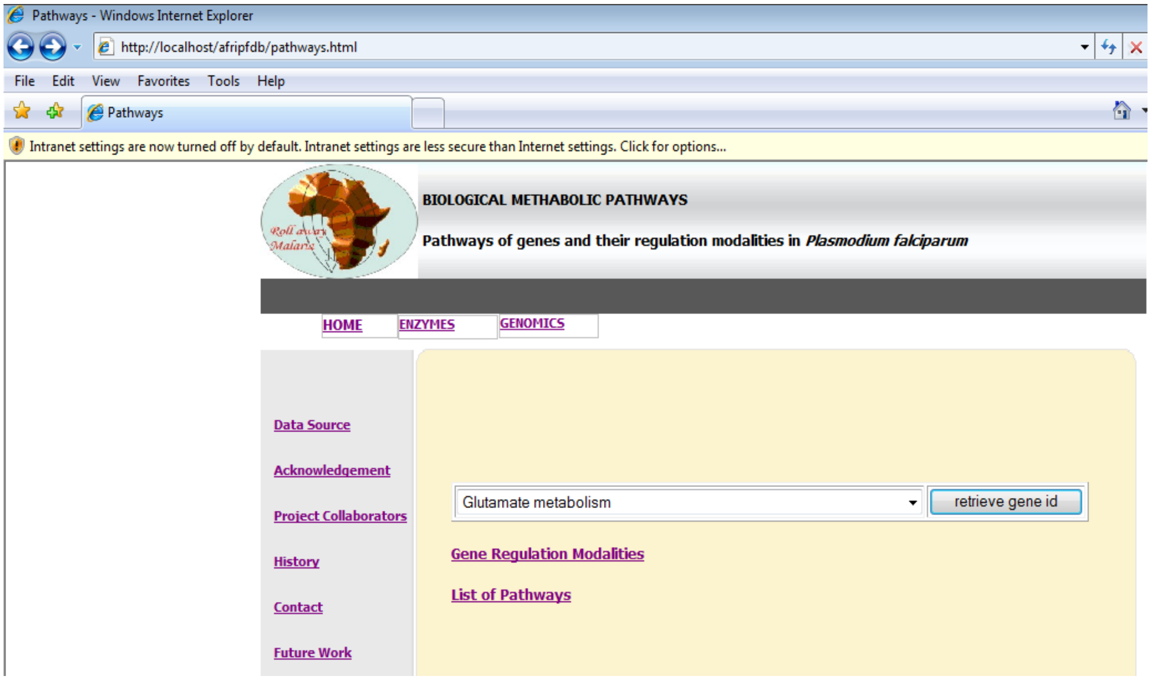


Fig 4.8 Enzyme.php

Pathways: this module gives more information on the pathways in the metabolism of *Plasmodium falciparum*.It also provides links to the Gene Regulation Modalities and list of pathways sub modules. See fig 4.9



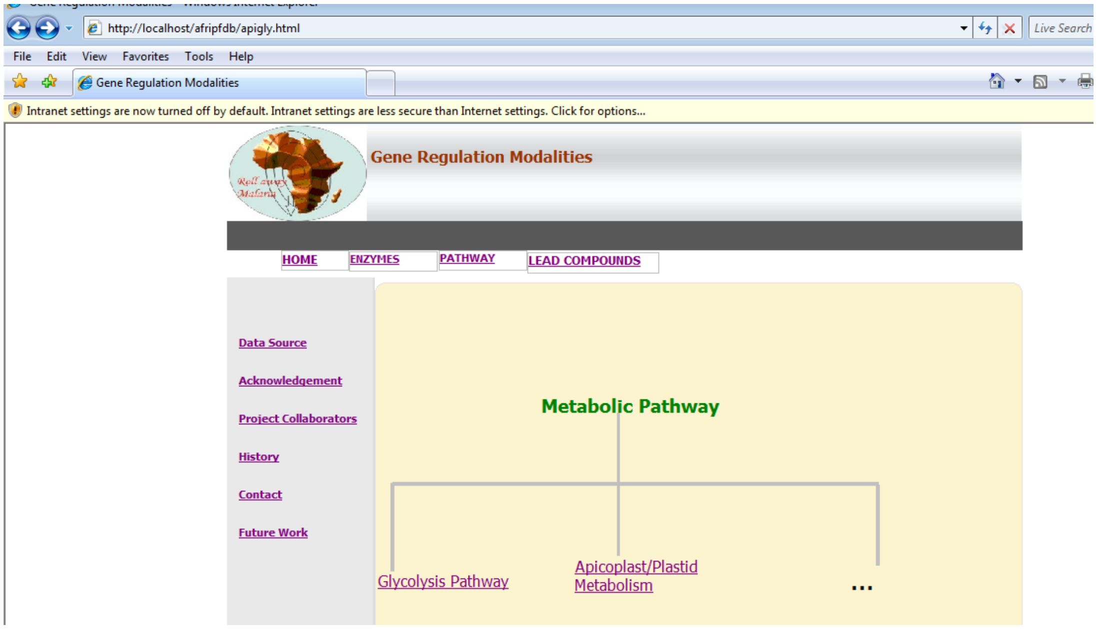
Intranet settings are now turned off by default. Intranet settings are less secure than Internet settings. Click for options...

BIOLOGICAL METHABOLIC PATHWAYS

Pathways of genes and their regulation modalities in *Plasmodium falciparum*

Glutamate metabolism

Fig 4.9 Pathways Page



Intranet settings are now turned off by default. Intranet settings are less secure than Internet settings. Click for options...

Fig 4.10 metabolic pathways

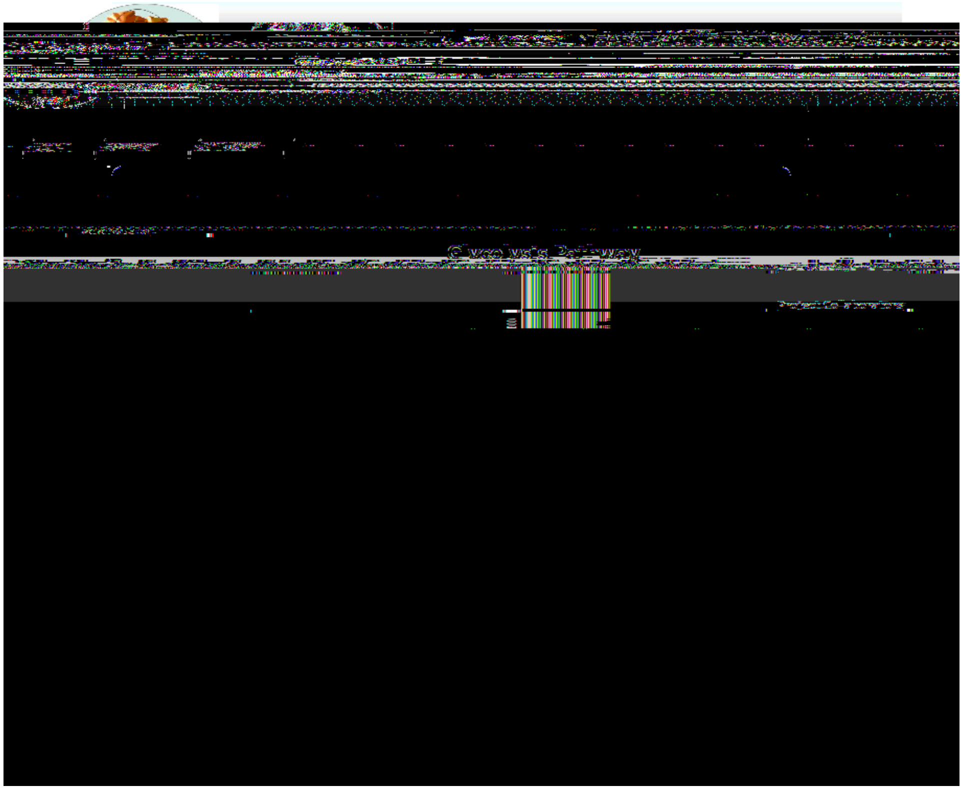
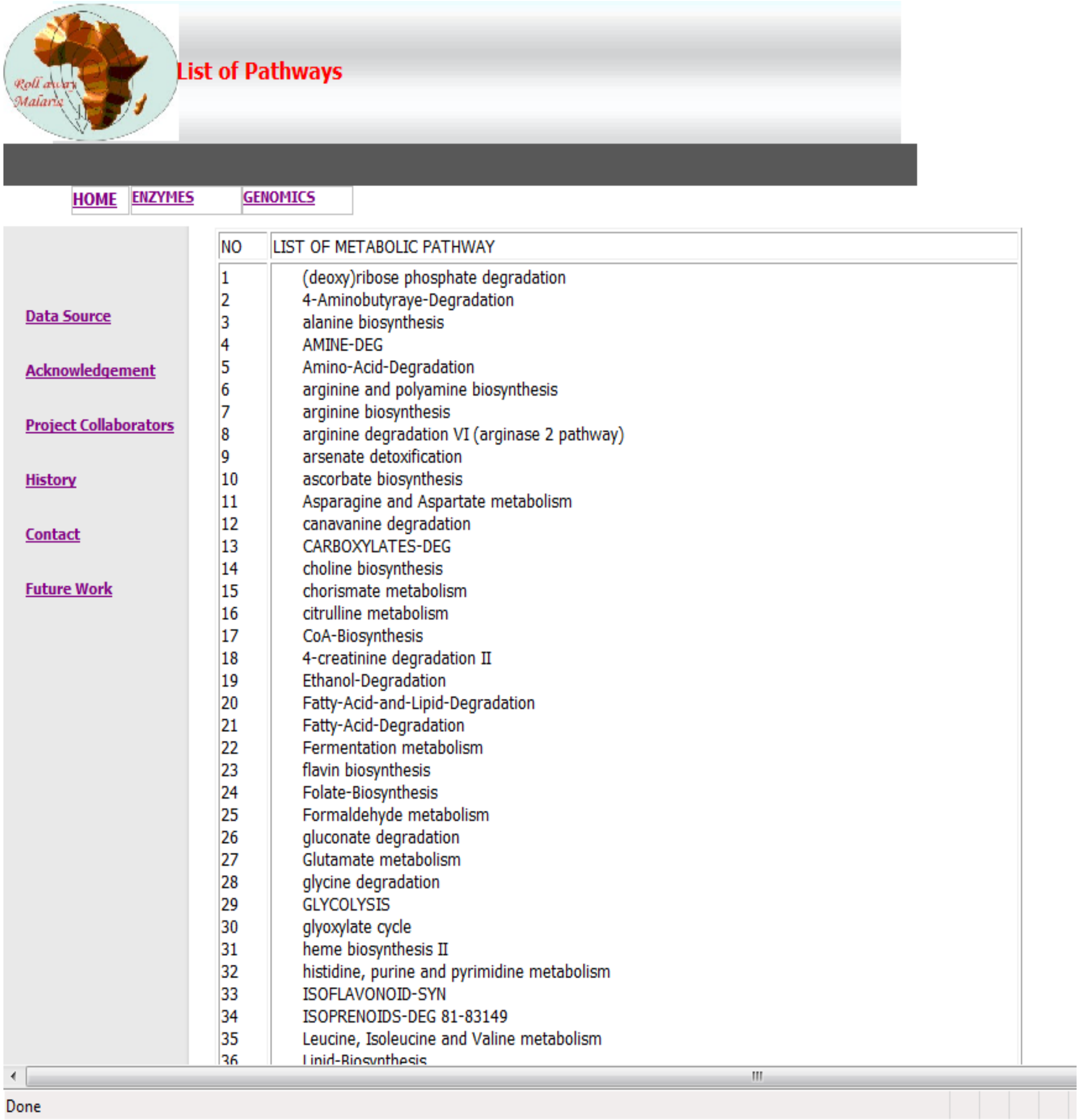


Fig 4.11 Glycolysis Pathway

This image cannot currently be displayed.

Fig 4.12 Apicoplast Metabolism



List of Pathways

NO	LIST OF METABOLIC PATHWAY
1	(deoxy)ribose phosphate degradation
2	4-Aminobutyraye-Degradation
3	alanine biosynthesis
4	AMINE-DEG
5	Amino-Acid-Degradation
6	arginine and polyamine biosynthesis
7	arginine biosynthesis
8	arginine degradation VI (arginase 2 pathway)
9	arsenate detoxification
10	ascorbate biosynthesis
11	Asparagine and Aspartate metabolism
12	canavanine degradation
13	CARBOXYLATES-DEG
14	choline biosynthesis
15	chorismate metabolism
16	citrulline metabolism
17	CoA-Biosynthesis
18	4-creatinine degradation II
19	Ethanol-Degradation
20	Fatty-Acid-and-Lipid-Degradation
21	Fatty-Acid-Degradation
22	Fermentation metabolism
23	flavin biosynthesis
24	Folate-Biosynthesis
25	Formaldehyde metabolism
26	gluconate degradation
27	Glutamate metabolism
28	glycine degradation
29	GLYCOLYSIS
30	glyoxylate cycle
31	heme biosynthesis II
32	histidine, purine and pyrimidine metabolism
33	ISOFLAVONOID-SYN
34	ISOPRENOIDS-DEG 81-83149
35	Leucine, Isoleucine and Valine metabolism
36	Lipid-Biosynthesis

Fig 4.13 List of Pathways

CHAPTER FIVE

SUMMARY AND CONCLUSION

5.1 SUMMARY

The database afriPFdb has been successfully developed and it is to be deployed on the internet and it supports multi users in a networking environment. This allows for quickly access anytime, anyplace, anywhere and by anyone who needs to make use of the information. It is platform independent and can run on any machine with the specified operating system.

The database has also well structured in terms of its table design such as to room for easy update of the information in the database. We have also included our findings and results gotten so far from the application of built *in-silico* tools which we believe will be useful especially for cases where results from the lab is not available.

The web site has been designed to be very simple, user friendly and very easy to navigate because we are aware that the intended users (especially the biologists) of the system may not be advanced users of the computer.

5.2 RECOMMENDATION

The *Biocyc* database which contains the *Humancyc and Plasmocyc* flat files among others are in text format which cannot be uploaded directly into the database. We recommend that further releases or updates should be done in spreadsheet for easier uploading into the database.

5.3 FUTURE WORK

The database has been successfully designed and developed. In the course of this project work we discovered more ideas on how this work can be expanded and they include;

- Due to the fact that work is still on-going in malaria research and the search for a potent cure/prevention of malaria, there will be a need for constant updates and review of information on the website and the database. As new releases come from the sequencing laboratories the database administrator will be required to upload these into the database.

- For easy uploading of the EMBL flat file, the XML version can be used so this will eradicate the need to convert first to CSV before uploading into the database tables.

5.4 CONCLUSION

In this project work, we have designed and developed a new database for the most deadly malaria parasite, *Plasmodium falciparum* called The African *Plasmodium falciparum* Database (afriPFdb) which contains information on the entire genome of *Plasmodium falciparum* (which includes its metabolic pathways, enzymes, genes, tRNA genes, chromosome list, e.tc.). The database also incorporates *in-silico* results for areas where no experimentsl results are available which will then later be experimentally validated.

REFERENCES

1. A. Bahl, B. Brunk, R. L. Coppel, J. Crabtree, S. J. Diskin, M. J. Fraunholz, G. R. Grant, D. Gupta, R. L. Huestis, J. C. Kissinger, *et al.*: *PlasmoDB: the Plasmodium genome resource. An integrated database providing tools for accessing, analyzing and mapping expression and sequence data (both finished and unfinished),* Nucleic **Acids** Research, January 1, 2002; 30(1): 87 - 90.
2. A Bairoch and R Apweiler : *The SWISS-PROT protein sequence data bank and its supplement TrEMBL,* Nuclear Acids Research, (1999)
3. C. Batini : Conceptual database design: an entity-relationship approach, Benjamin/Cummings Pub. Co., Redwood City, **USA**, (1991).
4. C. J. Date: *Databases in Depth: The Relational Model for Practitioners* O'Reilly Media, Incorporated (2005)
5. Clare Sansom: *Database Searching with DNA and Protein Sequences*: *An Introduction,* Briefings in Bioinformatics v. 1 i. 1 p. 22 - 32
6. D. A. Benson, I. Karsch-Mizrachi, D. J. Lipman, J. Ostell, and D. L. Wheeler: *GenBank: update*, Nucleic Acids Research, January 1, 2004; 32(90001): D23 - 26.
7. D. L. Wheeler, D. M. Church, R. Edgar, S. Federhen, W. Helmberg, T. L. Madden, J. U. Pontius, G. D. Schuler, L. M. Schriml, E. Sequeira, *et al.*, Database resources of the National Center for Biotechnology Information: update**,** Nucleic **Acids** Research, January 1, 2004; 32(90001): D35 - 40.
8. Ewan Birney, Michele E. Clamp: *Biological database design and implementatio*n. Briefings in Bioinformatics 5(1): 31-38 (2004).
9. **Florence Horn[*], Gerrit Vriend[1] and Fred E. Cohen:** *Collecting and harvesting* ***biological*** *data: the* ***GP****CRDB and NucleaRDB information* ***systems*** , Nucleic Acids Research, 2001, Vol. 29, No. 1 346-349 (2001) Oxford University **Press** Jan L. Harrington: *Object-oriented Database Design Clearly Explained,* Academic Press, USA, (2005)
10. H Ogata, S Goto, K Sato, W Fujibuchi, H Bono, and M Kanehisa :KEGG: Kyoto Encyclopedia of Genes and Genomes Nuclear Acids Research. (1999)
11. Jan L. Harrington, Harrington: *Relational Database Design Clearly Explained, Morgan Kaufmann Publishers (2002)*

12. Jeffery L. Whiten, Lonnie D. Bentley, Kevin C. Dittman ; *Systems Analysis and Design Methods*, McGraw-Hill/Irwin, New York, U.S.A (2001)

13. Jeffrey Sachs & Pia Malaney Review article: *The economic and social burden of malaria Nature* 415, 680-685 (2002)

14. Jérôme Wojcik, Vincent Schächter: *Proteomic Databases and Software on the Web* , Briefings in Bioinformatics v. 1 i. 3 p. 250 - 259 Roger S. Pressman 2005 Software Engineering: *A practitioner's Approach*, McGraw-Hill/Irwin, New York, U.S.A (2005)

15. Lisa J. Mullan: *Tutorial section: Biological Sequence Databases*, Briefings in Bioinformatics v. 4 i. 1 p. 75 - 77

16. Margaret Gardiner-Garden, Timothy G. Littlejohn: *A Comparison of Microarray Databases,* Briefings in Bioinformatics v. 2 i. 2 p. 143 - 158

17. Peter Rob, Carlos Coronel: Database *Systems: Design, Implementation, and Management*, Cengage Learning (2008)

18. Philip E. Bourne, John D. Westbrook, Helen M. Berman: *The Protein Data Bank and lessons in data management*, Briefings in Bioinformatics v. 5 i. 1 p. 23 - 30

19. Rolf Apweiler: *Functional Information in SWISS-PROT: the Basis for Large-scale Characterisation of Protein Sequences* , Briefings in Bioinformatics v. 2 i. 1 p. 9 - 18

20. Russ B. Altman Editorial: *Building successful biological databases*, Briefings in Bioinformatics v. 5 i. 1 p. 4 - 4

21. Selina S. Dwight, Rama Balakrishnan, Karen R. Christie, Maria C. Costanzo, Kara Dolinski, Stacia R. Engel, Becket Feierbach, Dianna G. Fisk, Jodi E. Hirschman, Eurie L. Hong, Laurie Issel-Tarver, Robert S. Nash, Anand Sethuraman, Barry Starr, Chandra L. Theesfeld, Rey Andrada, Gail Binkley, Qing Dong, Christopher Lane, Mark Schroeder, Shuai Weng, David Botstein, J. Michael Cherry : *Saccharomyces genome database: Underlying principles and organization*, Briefings in Bioinformatics v. 5 i. 1 p. 9 - 22

22. Simon Bennett, Steve McRobb and Ray Farmer : *Object Oriented Systems Analysis and Design using UML*, McGraw-Hill/Irwin, New York, U.S.A (2002)

23. Tim J. P. Hubbard, D. Andrews, Mario Cáccamo, Graham Cameron, Yuan Chen, Michele E. Clamp, Laura Clarke, G. Coates, Tony Cox, Fiona Cunningham, Val Curwen, Tim Cutts, Thomas Down, Richard Durbin, X. M. Fernandez-Suarez, James Gilbert, Martin

Hammond, J. Herrero, H. Hotz, Kevin L. Howe, V. Iyer, K. Jekosch, Andreas Kähäri, Arek Kasprzyk, Damian Keefe, S. Keenan, Felix Kokocinski, D. London, I. Longden, Graham P. McVicker, Craig Melsopp, Patrick Meidl, Simon C. Potter, Glenn Proctor, Mark Rae, D. Rios, M. Schuster, Stephen M. J. Searle, J. Severin, Guy Slater, Damian Smedley, James Smith, William Spooner, Arne Stabenau, Jim Stalker, R. Storey, S. Trevanion, Abel Ureta-Vidal, J. Vogel, S. White, Cara Woodwark, Ewan Birney: *Ensembl 2005*, Nucleic Acids Research 33(Database-Issue): 447-453 (2005)

24. T. Kulikova, P. Aldebert, N. Althorpe, W. Baker, K. Bates, P. Browne, A. van den Broek, G. Cochrane, K. Duggan, R. Eberhardt, *et al.*: *The EMBL Nucleotide Sequence Database*, Nucleic Acids Research, January 1, 2004; 32(90001): D27 - 30.
25. Toby J. Teorey, Tom Nadeau, Sam S. : *Database Modeling and Design: Logical Design Lightstone*, Elsevier Science & Technology Books (2005)
26. World Health Organisation; *World Malaria Report 2008,* WHO Press, Geneva , Switzerland.

WEBSITES VISITED

1. http://www.tech-faq.com/database.shtml : What is a Database?
2. http://www.library.uq.edu.au/training/skills/what_dbase.html : What is a Database?
3. http://en.wikipedia.org/wiki/Database_management_system : Database management system
4. http://publib.boulder.ibm.com/infocenter/zoslnctr/v1r7/index.jsp?topic=/com.ibm.zmiddle.doc/zmiddle_46.html : What is a database management system ?
5. http://www.wisegeek.com/what-is-a-database-management-system.htm : What is a database management system?
6. http://www.exampleessays.com : Analysing the Functions of a database
7. http://dipastro.pd.astro.it/planets/tngproject/TechRep/rep55/node5.html
8. http://en.wiktionary.org/wiki/database_management_system database management system
9. http://publib.boulder.ibm.com/infocenter/zoslnctr/v1r7/index.jsp?topic=/com.ibm.zmiddle.doc/zmiddle_46.html : Why use a database?
10. http://instructor.mstc.edu/instructor/dcolby/images/WBEA/WBEA_SeminarWeb/Terminology.htm The Database Behind Your Web Page
11. http://en.wikipedia.org/wiki/Hierarchical_model : Hierarchical model
12. http://en.wikipedia.org/wiki/Network_model : Network model
13. http://www.summersault.com : Why We Use PostgreSQL For Web Application Development

14. http://techdocs.postgresql.org/techdocs/supportcontracts.php : Finally, a list of big companies using PostgreSQL for serious projects
15. http://www.geocities.com/bioinformaticsweb/data.html : Biological Database/Molecular biology database/bioinformatics database
16. http://www.oxfordjournals.org/nar/database/cap/ : NAR Database Categories List
17. http://ansit.wordpress.com/2007/02/06/biological-databases/ Biological Databases
18. http://www.millenniumpromise.org/site/PageServer?pagename=gi_malaria_poverty : Malaria and extreme poverty
19. http://www.cdc.gov/ncidod/eid/vol4no3/nchinda.htm : Malaria: A Reemerging Disease in Africa Thomas C. Nchinda
20. http://ubio.bioinfo.cnio.es/Cursos/doctoradoUAM2008/Imedina/babel_presentation.pdf Biological Databases and Babelomics
21. http://en.wikipedia.org/wiki/KEGG : KEGG
22. http://lane.stanford.edu/howto/index.html?id=_1159 : What is the KEGG database?
23. http://www.genome.jp/kegg/genes.html KEGG: Kyoto Encyclopedia of Genes and Genomes
24. http://en.wikipedia.org/wiki/GenBank : GenBank
25. http://en.wikipedia.org/wiki/Swiss-Prot : Swiss-Prot
26. http://www.ebi.ac.uk/swissprot/ : UniProtKB/Swiss-Prot
27. http://en.wikipedia.org/wiki/Plasmodium : Plasmodium
28. http://www.sanger.ac.uk/ : *Plasmodium falciparum* Genome Projects
29. http://www.google.com.ng/imgres?imgurl=http://www.clongen.com/images/plasmodium_falciparum.jpg&imgrefurl=http://www.clongen.com/plasmodium_falciparum.php&h=369&w=441&sz=132&tbnid=1WtT0g4GRKsJ::&tbnh=106&tbnw=127&prev=/images%3Fq%3Dplasmodium%2Bfalciparum&hl=en&usg=___bVVm-pUttajcoSWpoe0rx20vWE=&sa=X&oi=image_result&resnum=6&ct=image&cd=1 : Plasmodium falciparum, a causative agent of Malaria
30. http://en.wikipedia.org/wiki/Plasmodium : Plasmodium