A Data Model for the Oregon National Primate Research Center

By Samone Khouangsathiene

Presented to the Department of Medical Informatics and Outcome Research and
the Oregon Health & Sciences University School of Medicine
in partial fulfillment of the requirements for the degree of
Master of Science
May 2004

School of Medicine

Oregon Health & Science University

Certificate of Approval

This is to certify that the Masters thesis of

Samone Khouangsathiene

"A Data Model for the Oregon National Primate Research Center"

Has been approved

[redacted]

Professor in charge of thesis

[redacted]

Member

[redacted]

Member

Table of Contents

ACKNOWLEDGEMENTS

I would like to thank my advisory and exam committee for their support and guidance:

Dr. Christopher Dubay, Advisor

Dr. Paul Gorman

Dr. Judith Logan

Dr. Sergio Ojeda

I would also like to thank Daryl Morris, GRIP Research Associate, for his help in accomplishing this project by writing the Perl scripts and creating the interface architecture. Finally, a huge thanks to fellow students, friends, and family for tirelessly listening to my thesis process!

Abstract

Genetic information is being generated at the ONPRC but this information is being scattered and sequestered in different laboratories and departments. This causes many problems that involve accessibility and data integrity. In order to solve these problems, this project set out to document the information work flow, gather requirements and information needs, identify shortcomings, create a data model and a prototype information system.

The information work flow was documented through observation and informal interactions with the user groups (researchers and the Department of Animal Resources). The next step was to understand the information needs of these user groups and from their needs create requirements or guidelines. This was achieved by observation, informal interactions and personal interviews as well. Understanding the existing information work flow is crucial in identifying the gaps or shortcomings of the work flow. Based on the requirements, existing data, evaluations of different extant database schemas and applications, a comprehensive data model was created. Before a working prototype information system was created, a concept of operations was realized.

1. Introduction

The purpose of this thesis is to develop a data model that represents a usable database framework for research information at the ONPRC. A database prototype will be created based on this framework. Both processes will be based on observing and documenting the information work flow at the ONPRC. The next component will be an analysis of the needs and requirements of the project. The end deliverable will be a data model and prototype that could be evaluated and used by the ONPRC.

These goals were achieved within the past year at the ONPRC. I worked with the Department of Animal Resources (DAR) and different researchers. These different groups asked me to find and gather information for them. I did this by obtaining data files which were in different formats both locally and from the Southwest Foundation for Biomedical Research (SFBR). Initially, I centralized this information in an application, PedSys, which I will explain in greater detail later. I became very familiar with this application by inputting and manipulating data. I ran quality control and checks on the data I had received. For example, I made sure that there was no duplicate data. I also used this application to run analyses on the data to answer researchers' questions. Researchers and the DAR approached me with different questions about their information. These interactions gave me a concept of how information was created and maintained. It also gave me an understanding of what kind of answers the researchers' expected their information to produce. I documented this information work flow. In this way I gathered requirements which pointed me towards creating a central repository for this information outside of PedSys. In order to formulate how to create this central repository I evaluated resources where the information already existed, other model organism resources' approaches, and data schemas. In the Material and Methods section, I have described in greater detail the steps I followed.

2. Background and Significance

2.1. Oregon National Primate Research Center

The Oregon National Primate Research Center (ONPRC) is one of eight national primate research centers located throughout the United States. The ONPRC is mainly funded by the National Center for Research Resources of the National Institutes of Health. The ONPRC's mission is, "To advance knowledge about health and disease in humans and animals through basic biomedical research."

There are three areas of research located within the center and these are the Division of Neuroscience, Division of Reproductive Sciences, and the Division of Pathobiology and Immunology. The Division of Reproductive Sciences does research in fertility control, early embryo development and women's health. The Division of Neuroscience does research in brain development and degeneration. The Division of Pathobiology and Immunology researches newly emerging viruses including HIV.

The ONPRC also has the Department of Animal Resources (DAR) to manage its primate colony. This department is solely responsible for the primates' care and well-being. The department has eight veterinarians and 70 staff. They maintain a veterinary clinic, diagnostic labs and maintain the breeding colony. Their primary goal is to maintain a healthy and genetically diverse colony to support ongoing research.

The center maintains a large primate colony for their research. Their largest colony and main focus of research is on the Rhesus macaque (Mucaca mulatta) of which there are approximately 3277. There are also 250 Japanese macaques (Mucaca fuscata). There are another 30 vervets (Cercopithecus aethiops) being researched as well as 11 male baboons (Papio spp).[1]

2.2. Integrated Research Information Source

The Integrated Research Information Source (IRIS) was created in 1960 at the Oregon National Primate Research Center (ONPRC). It was created to maintain the copious amounts of information related to primate research and serves as a clinical and resource management system. It contains over 2.1 million animal records which includes all deceased and living primates and non-living primates. It is based on modules within different areas of research. There are fifteen modules and these are breeding management, clinical management, colony management, immunomicrobiology, medical imaging, psychological well-being, surgery management, pathology management, small animals, location management, cell/serum freezer, reports, query tool and SNOMED tool.

The types of information included are based on animal history such as their location, health, tests or projects the primate belongs to. It also supports administrative functions which are mainly centered on billing. The following is a screen shot of IRIS.

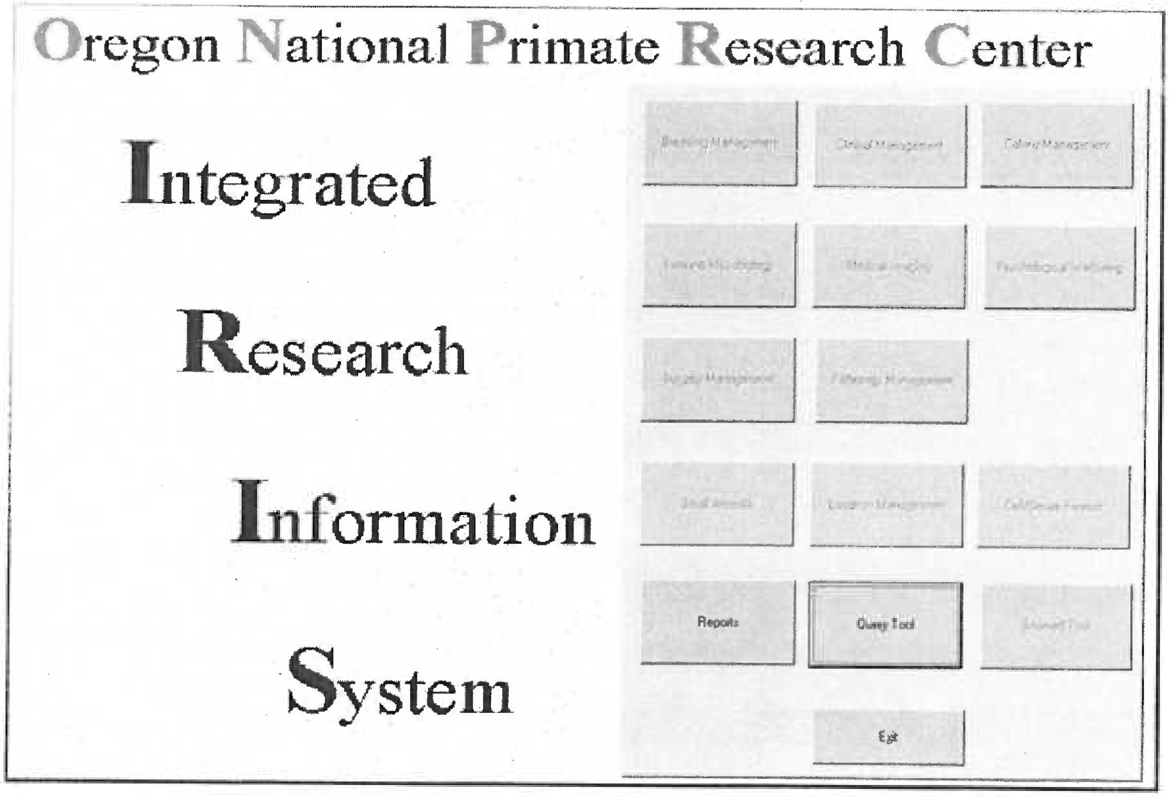


Figure 1: Screenshot of IRIS.

The IRIS database is composed of many tables but all the tables are centered on the quick reference file (QRF). The QRF maintains basic information about the animal such as animal ID (usually a five numerical digit), sex, birth date, death date, location and so forth. It is a relational database with many tables.

This database back end is based on Microsoft SQL Server. The front end or user interface is coded in Visual Basic 6.0. IRIS works on a server/client platform. The server runs on a Windows NT platform. While the client runs on a Windows 95, 98, NT, or 2000 environment. Access is password protected and can only be assessed by approved personal. Within IRIS, each module can only be accessed by appropriate personal.

IRIS is coded with built in biological and administrative triggers. An example of a biological trigger is when a researcher cannot input a birth date that precedes its parents' birth dates. An example of an administrative trigger is when a researcher cannot use a dead animal for an experiment.

The medical histories are coded with SNOMED (Systematized Nomenclature of Human and Veterinary Medicine). SNOMED was created in 1975 as a multiaxial nomenclature system for clinical record keeping. It is a hierarchal system classification system based on 11 axes or modules. It is currently the most universally accepted system of record keeping. [2]

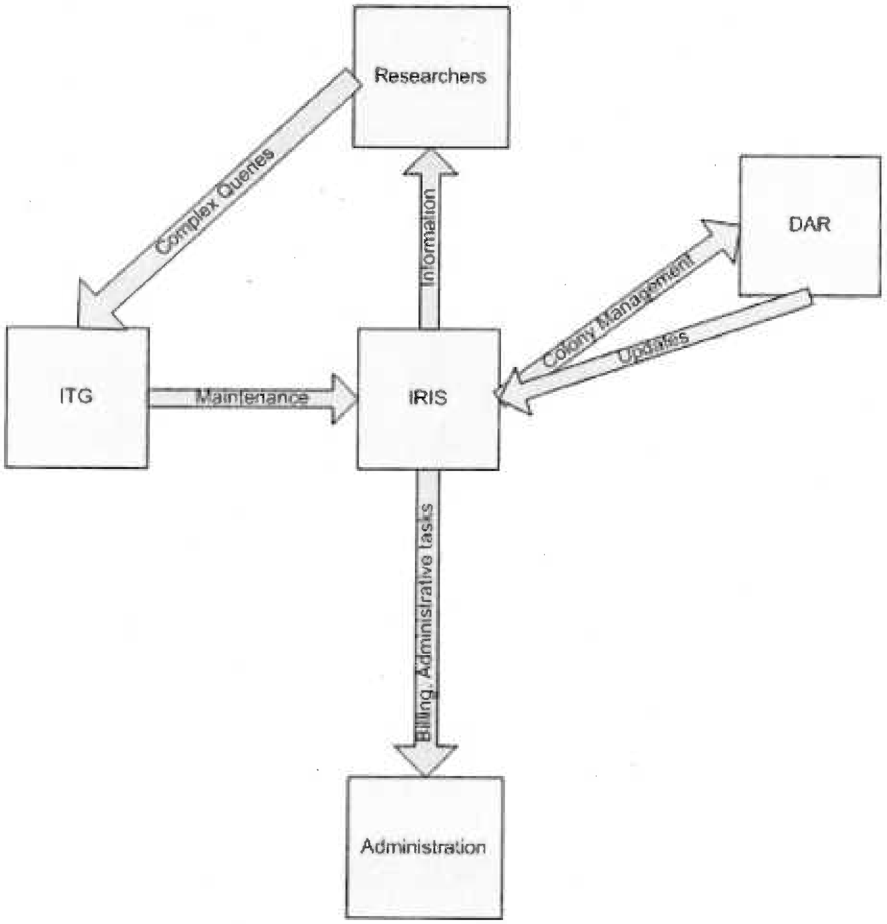


Figure 2: IRIS is an information resource for the Administration, DAR, and ONPRC researchers. Administration uses IRIS as a tool for billing and administrative tasks. The DAR uses the information from IRIS to do colony management and they are also responsible for any updates or quality checks. Researchers use IRIS to find information on a particular monkey although more in-depth complex queries have to be taken to the Information Technology Group (ITG) to be answered. ITG is responsible for the technical maintenance.

2.3. Genomic Research and Informatics Program (GRIP)

This program was started in 2003 under the direction of Dr. Christopher Dubay. The aim of the GRIP is to "…to act as a bridge to deliver appropriate information technology (IT) and bioinformatics tools to the wide variety of researchers and scientists working in and with the ONPRC." The intent is to discover if there are any information gaps within the ONPRC and to propose solutions to close these gaps. [3]

2.4. Genomics

Since the sequencing of the human genome completed in February 2001 there has been a tumultuous load of information from genomic related experiments. The vast amounts of information being produced from experiments are growing exponentially. For example, GenBank, a database of nucleic acid sequence data maintained at the NCBI (National Center for Biotechnology Information), has more than 10^{10} nucleotides. This continues to double each year. [4]

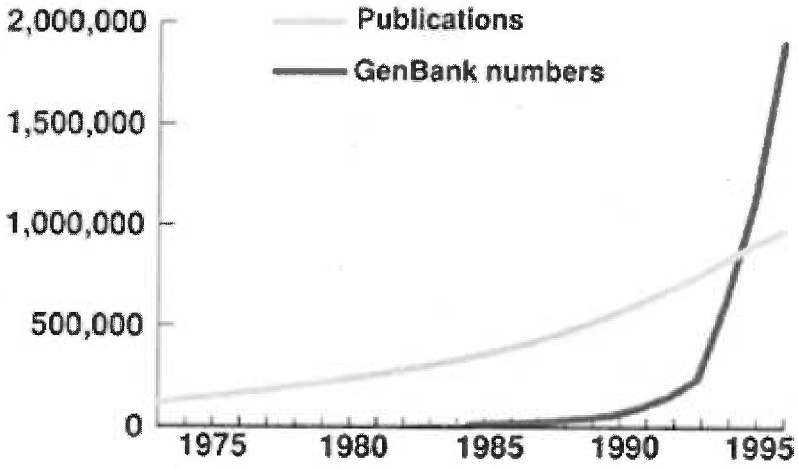


Figure 3: There is an exponential explosion of information occurring because of the human genome and advanced technologies. [4]

What is the significance of genomics and research? Genomics gives researchers incredible information to use to discover the difference between normal tissues and diseased tissues. New technologies such as microarray expression analysis, mass spectrometry, or genome wide yeast-two hybrid studies are tools that do so. [5] For example, microarray technology is used to profile gene expressions. Gene expression profiling measures the difference in a gene's expression levels. Therefore, it can be deduced which genes are more or less expressed under different conditions (diseased/non-diseased, drug/placebo, etc.) [6] This type of information gives researchers insights into human conditions. [7] The following table shows for example what gene expression technology can lead researchers to do.

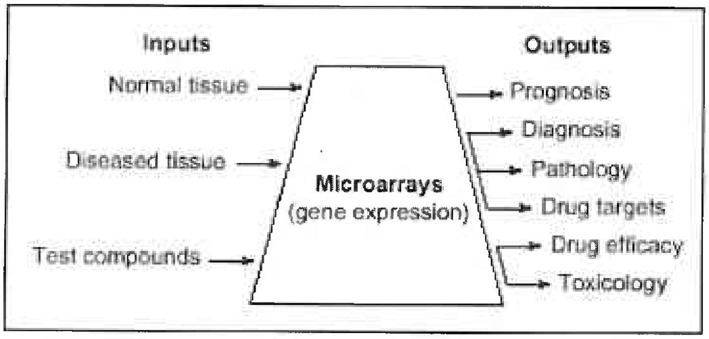


Figure 4: What researchers can achieve by using gene expression technology. [8]

Genetics is the basis of genomics. Genetics represents the macro level of genomics. Genetics includes genes and proteins. [9] There are a plethora of different types of genetic data. Some of the types of data produced from genes are single nucleotide polymorphisms (SNP), microarray expression values, gene sequences, linkage and parental markers. [10] For example, an SNP is a genetic variation that happens due to an insertion, deletion, or substitution of a single base pair (A→ T).

Another expanding area of research is proteomics which is the study of proteins. In much the same way as genes, proteins can be identified, expression levels measured, and gene linkages made. A good analogy is, "…if the genome is a list of the instruments in an orchestra, the proteome is the orchestra in the process of playing a symphony." Some of these data types are protein sequences, structures, expression values. [9]

Genetics plays an important role where non-human primates are used for biomedical research. [11] Genetic diversity is crucial in a biomedical research environment. Diversity and variability in a primate colony is the basis for assuming that this model can be bias free and generalizable for research.[12] Genetics (paternal markers) can be used to determine a pedigree for a primate colony. This pedigree helps breeding programs maintain genetic diversity within a colony.

Currently, the ONPRC has observed father (sire) and mother (dam) information on its current colony within IRIS. Paternal and maternal record keeping in IRIS is done by visual identification. In order to comprehend why this identification may be incorrect an understanding of the breeding program and the nature of primates is important. The breeding program has three different breeding groups. The first type is the largest breeding group, corrals, which typically have 7 adult males, 40 adult females, 50 one year olds, 45 two year olds. The second type are the shelter house which have 3 adult males, 25 adult females, 17 one year olds, and 14 two year olds. The third is a harem with one adult male, 8 adult females, 6 one year olds, and 5 two year olds. The last one is called time mated which means that there is a paired mating. The sexual nature of rhesus monkeys is very key to paternal misidentification because they are non-monogamous. Males and females do not start breeding until they are at least three years old. In a situation like the corral breeding group it is most likely that the sire will not be identified. Even in the time mated group it is possible to have the wrong sire because matings have been known to occur between cages. Maternal identification can also be incorrect due to infant swapping. Infant swapping is when two dams switch infants or a dam picks up an infant that is not hers. Based on these breeding groups and nature of the monkeys it is easy to understand why identification may be wrong. [12]

2.5. Case Studies

There is a growing importance to include genetic data into research and some of the ONPRC researchers realized the importance of doing so. Collaboration was started between an ONPRC laboratory headed by Michael Axthelm, PhD and a Southwest Foundation for Biomedical Research (SFBR) laboratory headed by Jeff Rogers, PhD. Axthelm was interested in breeding for presence for a particular gene for his (AIDS) research. In order to create such a breeding program he had to know which primates to breed together to get the desired gene. He sent DNA samples to SFBR and later to Davis as well. These laboratories would then send back information on the monkey's pedigree (sire and dam) and markers used to determine this. This also included information on Axthelm's gene of interest.

Later on, Judy Cameron's laboratory got interested in doing something similar. She was interested in correlating her research results with a gene(s). An example question might be something like this "Is there a correlation between shy monkeys and their genotype?" Dr. Cameron also started sending DNA samples down to SFBR only.

In order to obtain these DNA samples which are typically taken from blood samples, these laboratories usually have to request the DAR to take these blood samples. While taking these samples, Dr. Gary Heckman, DAR Manager, started to realized that this information in turn could help him greatly in his breeding colony management (e.g. breed to maintain genetic diversity). Currently, the DAR's goal is to obtain marker information for the whole rhesus colony. Over half of the colony has been typed and sample gathering is still in progress.

2.6. DNA Typing

In order to understand the type of information that is being used and received at the ONPRC, a background on what and how DNA typing is done will be briefly discussed.

Short tandem repeat polymorphisms (STRPs) are also known as microsatellites. Microsatellites are regions of 2 -5 base pairs that repeat consecutively. They can be repeated anywhere from 10 – 30 times and is currently assumed to not code for any proteins (introns). [9] It has been determined that these repeats are unique to an individual and are inheritable. [13] Therefore, identifying microsatellite markers allows a pattern of inheritance to be created. [12]

ONPRC typically collects blood samples and either sends the blood samples or DNA samples directly to a center. The center runs the sample with known marker primers through a PCR (polymerase chain reaction) which amplifies markers that are present within the sample. Afterwards the presence and size of a marker is discovered by running the sample through a polyacrylamide gel. On this gel, the samples separate according to base pair sizes and then are matched to known marker sizes. The samples are further verified by running them through a genotyper machine. From these experiments, a sample will have a list of known markers and where that marker is located (allele or loci). Then to determine the sire and dam for each primate, a list of potential dams and sires is analyzed for discordant or shared alleles. At present, SFBR qualifies definitive dams by 8 shared loci and no discordant ones. Definitive sires must have at least 10 loci and no discordant ones and other potential sires must be excluded for at least two loci.

	Marker			
ID	d2s122	d10s192	d2s163	d3s1768
10591	allele/allele	182/184	230/230	126/126
11282	115/117	194/204	212/216	126/126
13713	101/107	190/204	212/230	122/126

Table 1: ID is the specific monkey identifier and is usually a five digit number. The markers are then listed with corresponding allele pairs.

2.7. PedSys

PedSys is SFBR's own program for genetic information storage and analysis. It was created by Bennet Dyke, PhD at Pennsylvania State University. He designed it principally as a research tool for use in genetic analysis of either human or non-human subjects. It was written in FORTRAN and is still written in FORTRAN. It runs in a UNIX environment command line. The data is put into a file with an associated code file which describes the format of the data. It is based on a master file which can be linked to other related files. Commands can be run on the files in the areas of:

- Data management
- File management
- Genetic analysis
- Pedigree management

It has the ability to translate PedSys files into tab delimited files or vice versa. Like IRIS it has built in biological triggers such as offspring cannot be born before parents or a mother cannot also be the father. It is still being currently used at SFBR to maintain and house all of their genetic information.[14]

2.8. Specific Project Aims

The goal of this project is to create a data model and prototype for the ONPRC participating laboratories and the DAR. In order to do this, these steps must be taken:

1) Document the current information work flow.
2) Gather common requirements from different user groups (researchers, GRIP, DAR).
3) Based on the requirements create a data model and a concept of operations (framework).
4) Using the data model create working database and information system prototypes.
5) Give future direction.

3. Materials and Methods

To create these prototypes, a software development cycle was followed. These steps were:

1) Problem Determination
2) Requirements Gathering
3) High Level Design
    a. Use cases and diagrams
    b. Data model
    c. Interface design
4) Interfaces and Components
5) Coding and Testing
6) Evaluation

Although these steps are listed and described in numerical order it does not imply that it was linear process. Like all effective software development cycles, these steps were highly iterative.

3.1. Problem Determination

As stated in the background the human genome has been mapped. The primate genome is not far behind. The primate genome is being mapped and is forecasted to be finished within one year. The mapping is being done at SFBR and the sequencing is being done at Baylor University. Will mapping of the primate genome follow the same information path as the human genome? It would seem likely that it would because primates are the most closely related to humans. Any research done on primates can be translated into a human model. [15] The National Institutes of Health recognized this fact and organized a workshop to discuss primate genomics. The workshop involved representatives from all eight primate centers, Director of the nation Center for Research Resources(NCRR), NCRR Program Director for the Regional Primate Centers, and staff from the National Human Genome Research Institute and Office of AIDS Research. In 2001, this workshop group published recommendations for future efforts in primate genomics. All these recommendations involved supporting the information that is generated from genomics research. [16]

As stated in the introduction, the DAR at the ONPRC has sent samples (DNA or blood) to SFBR or Davis and has received genetic marker information. This information is received in either tab-delimited or comma separated format. The DAR and the two separate laboratories maintain their own information in Excel or QuadroPro files. All three of them use the parentage information in breeding decisions only. There are three separate places where the same information may be housed. Sometimes, information may be sent to one laboratory but not to everyone. Then this information may reside in one place without being disseminated to the researchers who want the information.

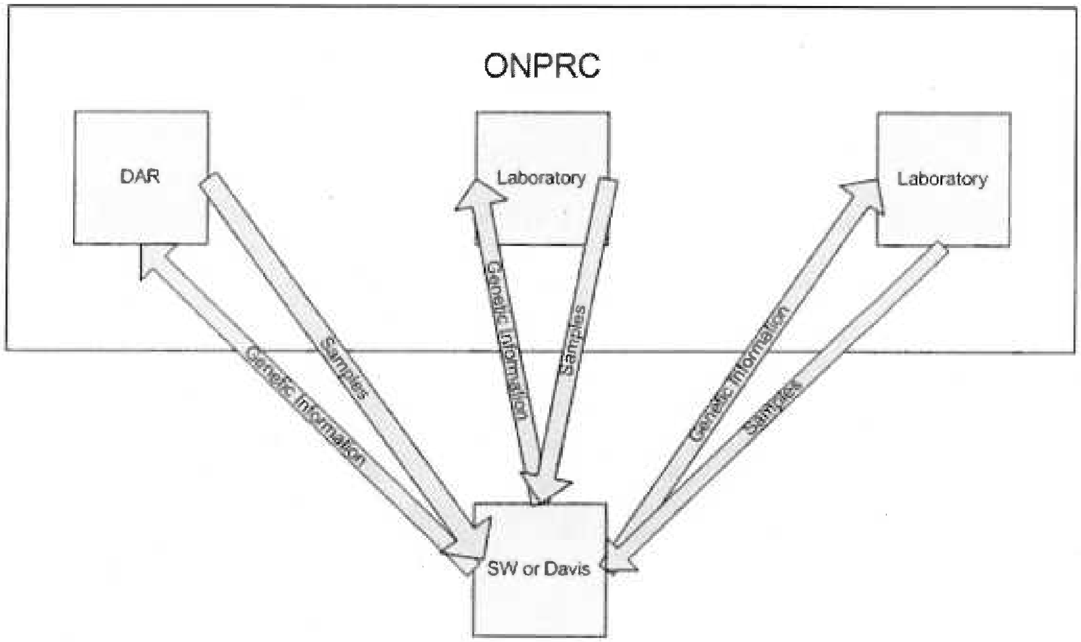


Figure 5: Samples are sent to be typed at either SFBR or Davis by separate places. Genetic Information such as parentage and markers are sent back to the originators.

As mentioned earlier in this section, the amount of information will be growing as well as the different types of information. The number of laboratories who will want to incorporate genetic data into their research will also increase. With this in mind, the current data workflow is terribly inadequate to support such an increase in information activity.

The current information flow does and will create problems. There are two main general problems that arise which are data accessibility and data integrity.

1. accessibility

As mentioned before data can be sent back to some but not all places (laboratories or the DAR). This creates confusion and work inefficiency because data can reside anywhere. For example, this past year, the DAR tried to compile a list of monkeys that have not been bled. I did this task for the DAR. In order for me to accomplish this task I had to collect all information from both Cameron's and Axthelm's laboratories and the DAR to compile a conclusive list of animals that have blood sample information for them. This process took up some time because these laboratories had to be contacted, they had to put together their lists and then send the information to me. This latter point took up the most time. After that I put all their lists into PedSys because it had functions that could help me compare files. I deduced a non-sampled list by comparing the whole colony to the list of sampled monkeys. The whole comparison time in itself did not take up very much time. It took a short time to convert Excel files into PedSys format and to run a comparison function between two lists. It was a short time compared to how long I had to wait to get definitive lists from everyone. From this experience, it was obvious that having data sources in many different places created inefficiencies and issues of accessibility.

2. data redundancy/inaccuracy

Data being housed in several different locations is not inherently inefficient in itself. It does pose problems when there are changes that need to be made for purposes of quality assurance. Changes that are required to be made in one set of data will be required to be made in all of them. Those changes might not be made if the different researchers do not communicate to each other these changes. Data inconsistencies are created and there is no current set protocol to catch them.

3.2. Requirements Gathering

The first step to develop requirements is to define the users and their needs and identify the requirements needed to meet these needs. There are three different groups of users; researchers, DAR, and GRIP. Even though they are different user groups their basic information needs overlap.

User groups	Needs	Tasks	Resources used
Researchers	-access to genetic information	Querying for specific monkey information	IRIS Excel GRIP Windows
DAR	-access to genetic information	Querying for specific monkey information	IRIS Excel GRIP Windows
GRIP	-maintaining genetic information -tracking genetic information	-querying for specific monkey information -querying for specific genetic information -QA/QC on data	IRIS PedSys Excel Windows

Table 2: The table outlines the user groups, their needs, the tasks they are involved in and the resources used to accomplish these tasks.

In 2003, Dr. Christopher Dubay arranged interviews with ONPRC researchers and the DAR. He identified one main general requirement of accessibility. It was apparent that the users were concerned that data was inaccessible due to inconsistent dispersal of information. The next group of users would be the GRIP. This group would have the same requirements as the above groups. Although this group will have added requirements since they are the group who is in charge of data collection. The following lists the general and specific requirements. The requirements were founded on interviews and informal interactions with the user groups. These informal interactions were also documented in use cases which are discussed in the following section.

General Requirements
Centralized data storage
High security with user-password access only
Scalability
Development language is specified and supported
User interface accessible
Allow for concurrent use of database
Research capabilities (e.g. genotype/phenotype correlation)
Data Entry Requirements
Capable of data entry
Creation of reports
Archiving of information
Ability to load data from different sources (IRIS, PedSys, tab-delimited format)
Data Retrieval Requirements
Querying capable of rapid access to specific data on a specific monkey
Retrieval of data about siblings
Retrieval of data on specific marker
Ability to manipulate stored data

Table 3: List of general and specific requirements.

3.3. Design

In this section an evaluation of existing possible applications will be discussed to decide on where the information should be put. Also in this section, an evaluation of existing possible schemas will be explained.

IRIS Evaluation

IRIS has no present plans to incorporate another module into their schema. Although, according to Wayne Borum, ONPRC's Senior Systems Analyst, it would be easy enough to do so. Is this the answer to maintaining all the incoming genetic data? To answer, we should also ask what IRIS's main goal is. When it was created in 1960, IRIS was made mainly for administrative purposes, for tracking information on all animal subjects. IRIS can show information on a particular monkey and simple queries can also be run. An example of this is to find all ages of all monkeys. IRIS cannot do any complex queries that genetic data types were meant for (e.g. creating pedigrees). IRIS would be able and useful to house genetic parentage information. Outside of that, it would not be useful to house such a large amount of data into IRIS because the data would not be able to be analyzed the way it should.

PedSys

PedSys is another system to evaluate because it was created to house and analyze genetic data. The first obvious benefit to using PedSys is the ease of transferring data from SFBR to ONPRC. No data translation would be required. Another benefit of PedSys is that it already has built in biological triggers. Yet another benefit is that it has genetic analysis tools such as kinship coefficient and inbreeding functions.

There are many disadvantages as well to using PedSys. A major disadvantage is that the file and data management is difficult and cumbersome. Inputting and updating information cannot be done quickly in one step. Querying for information across files is laborious and takes several steps to produce output. Although, it is not suggested that PedSys not be used altogether, it should just be used specifically for genetic analyses.

Program Evaluation Summary

Since IRIS or PedSys did not seem to be sufficient to house or maintain the genetic data being generated at the ONPRC, a separate database was created. At the same token, some if not all of the information could still be housed in either place. For example, IRIS could house the parental information (genetic dam and sire). PedSys could house the genetic data because it has genetic analysis functions.

Schema Evaluations

Entity Attribute Value Schema

An entity-attribute-value (EAV) schema is more dynamic and flexible for heterogeneous data. [17] For example, in a table called phenotype there is an entity column named monkey id (primary key), an attribute column named phenotype and the last column named value. Monkey ids are listed in the monkey id column. Corresponding to those ids are their different phenotypes in the next column such as hair color, eye color, weight, height. In the next column of value there would be values that correspond to the phenotypes. The following table illustrates this.

Monkey id	Phenotype	Value
12345	Hair Color	Brown
12345	Eye Color	Brown
12345	Weight	124
12345	Height	123

Table 5: An EAV table where monkey id is an entity. Phenotype is an attribute. Value lists the values for each attribute.

In addition to this table another table must be maintained to keep track of the different phenotypes and values. This is referred to as a metadata table which describes the types of information that is housed in another table. In theory, the phenotype table can maintain as many attributes and values as possible. It must be noted here that each table must be strongly typed in order for queries to work. For example, all phenotypes that have integer values must be in a separate table from phenotypes with character values. This type of schema is used repeatedly in electronic patient records. [18]

A major benefit of this type of schema is flexibility because many attributes and values can be added without much alteration to the whole schema. [18] This schema allows for sparse data types thus decreasing required storage space. For example, not all monkeys will have phenotype for a certain disease. If we create this attribute in a conventional schema (i.e. relational) then this attribute would be empty for most monkeys.

There are some drawbacks to using an EAV schema. The EAV schema is used as a schema to physically store data. This physical schema is different from how a user conceptualizes data to be stored (logical schema). Therefore, the physical schema must be restructured to be in a logical schema because the user expects this. This involves much more coding then in a conventional relational schema where the physical schema and logical schema are very similar. [19] Complex attribute queries are inherently slower because of the schema.

An EAV schema is typically not used for simple or static data structures so that is why it was not incorporated into the project's schema. An EAV schema would have too much of an overhead to create if the data is just as easily housed in a conventional relational schema. In PhenoDB, a pharmacogenetics database, which uses a mixture of EAV and conventional relational tables, the database houses genetic information in conventional relational tables. [17]

A perfect example of where an EAV schema would be appropriate is a phenotype table. This table as illustrated in the above example would possibly have several different data types and not all phenotypes would apply to all monkeys.

Generic Model Organism Project (GMOD)

The Generic Model Organism Project is a consortium of other model organism groups which include: WormBase, FlyBase, MGI (Mouse Genomic Informatics), SGD (Saccharomyces Genome Database), Gramene, Rat Genome Database, EcoCyc (Encyclopedia of Escherichia coli), and TAIR (Arabidopsis Information Resource). Their aim is to, "...develop reusable components suitable for creating new community databases of biology." [20]

GMOD has a schema that they have taken the liberty to name, Chado, which is the way of the tea, a Japanese tea ceremony. The Chado schema is based on modules (much like that of classes) and these modules are:

cv	-controlled vocabularies
sequence	-dna, rna and protein features
companalysis	-computational analyses
genetic	-genetic and phenotypic data
expression	-rna or protein expression data
organism	-species data
pub	-publication and references

These modules have dependencies between each other. The links between modules are usually whole tables. The schema, instructions, and a generic browser is free and available on the web to download. The schema is suppose to be database management system (DBMS) independent but so far works only on PostgreSQL. It is used through an Apache 1.3 server. Instructions are included on how to configure the computer to handle Chado and how to populate the database from existing data sets. It is in an alpha production stage which means that it is still being developed and there are bugs that still need to be dealt with.

This whole package from GMOD looks very promising as a guideline for future use. Since it is collaboration between model organisms resources it would seem that it would include many requirements that this project is seeking. Yet, it is difficult to evaluate such a comprehensive product when this project is only looking to handle such a static and simple data type. Perhaps in the future when a more comprehensive information structure is desired, GMOD, would be an appropriate platform or guideline to use.

Relational Schema

In a relational schema, there are attributes and relationships. For example, a monkey table has attributes of monkey id, sex, birth date, death date, status, and source. Each of these attributes is represented by values.

Monkey_id	Sex	Birth date	Death date	Status	Source
12345	F	01/01/2003		Alive	Local

Table 4: Simple relational table example. There are attributes of monkey identification, sex, death date, status, and source.

Each table has a relation to another table. For example, the monkey table might have a relation to another table called samples. Samples may have an entity of sample id with attributes of date, monkey id (who the sample was taken from), storage location, etc. [21]

A relational schema is the most widely used database schema. [22] Many model organism databases use a relational schema. [23]Some examples of these are the Mouse

Genomic Informatic group from Jackson Laboratories, Rat Genome Database, FlyBase, and ZebraFish International Resource Center. Note that some of these resources are the same resources who are working in collaboration on GMOD.

Schema Evaluation Summary

Both the EAV and GMOD schemas appeared to be useful for future guidelines. Both of these schemas were created with data that already existed. Data and data types were examined from existing databases then these schemas were created from that. This is the protocol that was used to create the database at the ONPRC. It would be difficult to state if these other schemas would fit the model at the ONPRC simply because we are working with one data type, genetic.

Based on a majority of the model organisms resources, a relational schema was used in this project. An EAV schema was also incorporated where it seemed appropriate.

3.4. Use Cases and Diagrams

Use cases are a textual description of a specific scenario. Use case diagrams are a visual description of a specific scenario. There can be many different scenarios. Creation of use cases and diagrams will help in understanding the different scenarios that are encountered. [24] Based on these use cases and diagrams, a data model can be created. Refer to appendix A.

The next step is to design the user interface. Since the end product is only a prototype this step is not significant.

3.5. Interfaces and Components

In this step decisions were made on the technical aspects. Since this is a developmental project tools were used that were inexpensive and widespread. The most inexpensive database management system is MySQL. It is an open source developmental tool which means that it is free and available on the internet. It is a DBMS that is widespread in the bioinformatics community. [25] It is based on SQL so it could easily be moved to another DBMS if that was deemed necessary in the future. Refer to Appendix B for SQL script. An Apache/Tomcat server was installed. Unfortunately, it was run on a Windows environment which is not necessarily inexpensive but is widespread especially in an academic setting. A Perl script was written to read PedSys data into the database. It was unnecessary to create a script to read in Excel files because this can be done through PedSys. For the user interface, a basic JSP (JavaServer Pages) was used. Refer to Appendix C for the architecture.

3.6. Coding and Testing

The coding involved programming in Perl, JAVA, and SQL. The coding created the database, imported data, and any interfaces. The database was tested to see if it could supply answers to different queries. There were 'debugging' cycles. The end result was a working prototype.

4. Results

4.1. Concept of Operations

From the observations and interactions with the user groups, a concept of operations was created. This was an important step before creating the data model. [23]

Along with the requirements, it gave me the understanding of how and where the data model would work best.

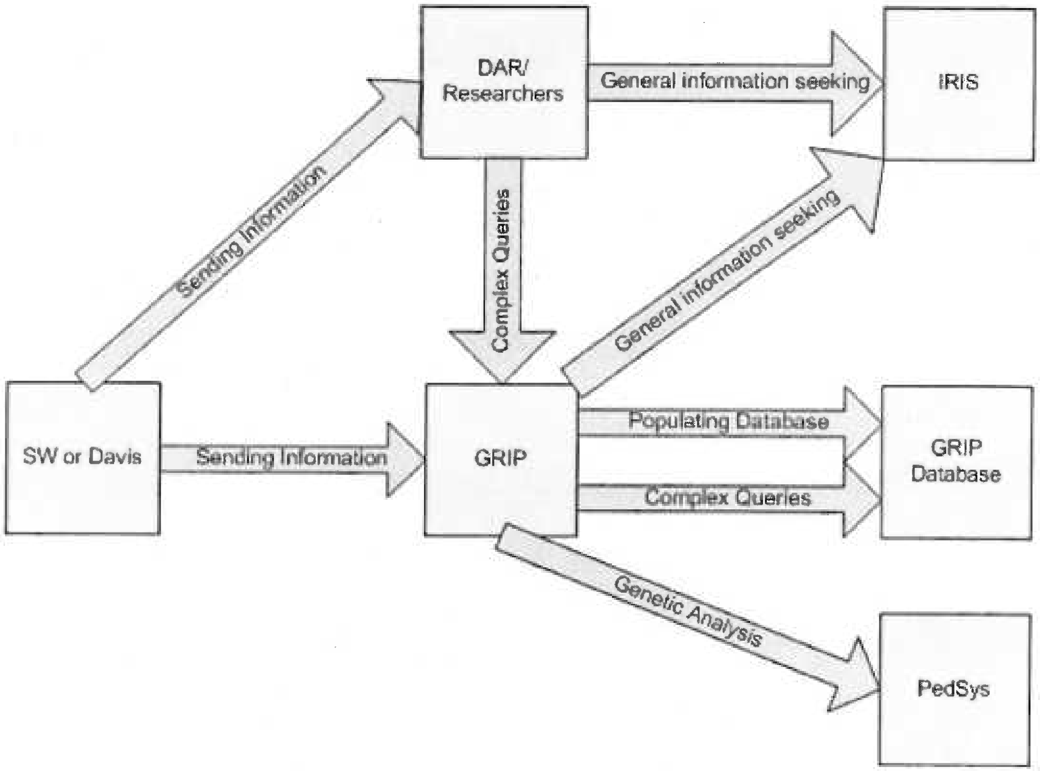


Figure 6: The current information flow after inclusion of GRIP.

In this concept of operations, it would be ideal if the typing centers sent information to the GRIP and the respective laboratories. The GRIP can use the data to populate the database and PedSys. The GRIP can use PedSys to do genetic calculations and analyses. The database would be used as an information tracking tool. It is easier to manipulate and change data in the database as compared to PedSys. The database would also be used for more complex data queries. (E.g. Which primates have blood samples and who do not?) The GRIP can also use IRIS to supplement any information that the database should have to complement queries. The separate laboratories can keep track of their own information or they can rely on the GRIP and the database to maintain it for them.

4.2. Database Design Process

The first step of this database design process was to identify the requirements which was already described in an earlier section. The next step was to create from the requirements a conceptual design or an entity-relationship (ER) data model. Refer to Appendix D. An ER model is a description of the data involved in terms of objects and their relationships to each other. [22] The next step was to convert the conceptual design into a logical database design or a logical schema. Refer to Appendix E.

A central table much like that in IRIS, *monkey*, was composed of its five digit unique identifier (as a primary key), sex, birth date, death date, and status (live, dead, sold). The next table, *acquired*, tells if a monkey was brought in from somewhere else. It has attributes of monkey_id (primary key, foreign key to *monkey*), source, and date (acquired). This was a separate table because most monkeys were not acquired. Another table, *geneticrelationship* contains monkey id as a primary key which is a foreign key to the monkey table. It has genetic sire (sire_id), genetic dam (dam_id) and the date that this information was created (observation_date). Another similar table, *observedrelationship* also has a primary key and foreign key of monkey id. The table contains observed sire(sire_id) and observed dam(dam_id) as well as the date that the observation was made(observation_date). Two separate tables were created for this because all monkeys (estimated 23,000 monkeys) will have observed relationships but not all monkeys will have genetic relationships (estimated 2,000). Therefore, instead of having many nulls within a table it was deemed more efficient to create separate tables.

The next few tables that will be described follows the information flow. To state the process simply, we want to capture the information from when samples are sent out to be typed and when the samples are returned with marker information. The first table is to capture the sample information, *samples*. Each monkey will have samples that are sent away to be typed. This table has a primary key of sample_id. A foreign key of monkey_id points to the monkey table. The table also has sample_type, sample_date, and storage(location). It is possible that a monkey can have more than one sample taken. The next table, *markers* has a primary key of sample_id (foreign key to samples) and marker. This table includes abs_allele1, abs_allele2. Abs_allele1 and abs_allele2 are foreign keys to another table, allele. The allele table has primary keys of allele, marker, and source. The other attributes are abs_allele and bp_size. The last table in this series is the *MarkerInfo* table which has a primary key, abs_allele. This is also a foreign key to abs_allele in the *allele* table. This table has attributes of location, primer, and annotation.

It must be explained that the allele table is necessary to keep track of where the markers come from. Even though the same markers can be used by different centers, the alleles may be called something different. For example, marker Ds1106 may have alleles 120:120 according to SFBR but the same marker may have alleles 119:119 from Davis. A way to store this information is to create a pseudo name, abs_allele and map it back to different allele names on the same marker. This makes doing allele searches or calculations easier by coercing the same alleles with different names into one common allele. By incorporating an allele table that keeps track of the allele name and source we can maintain data integrity with no loss of data.

The last two tables, phenotype and nomenclature, were inserted for possible future use. These tables are structured in an EAV schema. The phenotype table consists of an entity (monkey), any attributes with its associated value and the date the observation was made. Based on this table another table was created, nomenclature, to keep track of any attributes and the possible values for those attributes.

The user interface was based on a simple JSP because it was server and platform independent. Initially, the user logs in by an html form, login.html. This launches a JSP, gripdbstart.jsp, that asks the user for the type of query they want to run. Another page pops up asking for the query parameters, getqueryparams.jsp. This query parameter gets passed to the database (via postqueryparams.jsp) which returns output to another JSP, query1_o.jsp, which is shown to the user.

5. Data Model and Prototype Analysis

It must be kept in mind that this project was an initial development project. Although, a formal evaluation was not undertaken, an analysis was done to see if the requirements were met by this development project. An analysis was undertaken for this data model and prototypes to verify if the end results fulfilled the requirements that were specified.

General Requirements	**Requirements Met By**
Accessibility	Creating central database
High security with user-password access only	MySQL administrative security tasks
Data Integrity	Creating a central database
Scalability	Relational schema is amendable
Development language is specified and supported	Documentation
User interface accessible	Web user interface (widespread technology)
Allow for concurrent use of database	Creating central database
An active prototype database	Creating central database and web user interface
Data Entry Requirements	
Capable of data entry	MySQL DBMS
Creation of reports	MySQL DBMS
Archiving of information	MySQL DBMS
Ability to load data from different sources (IRIS, PedSys, tab-delimited format)	Perl Script
Automatic biological triggers	Has not been met. Good future project.
Data Retrieval Requirements	
Querying capable of rapid access to all data on a specific monkey	MySQL, web interface
Retrieval of data about siblings	MySQL, web interface
Retrieval of data on specific marker	MySQL, web interface
Ability to manipulate stored data	MySQL

Table 6: Listed requirements correlated with how they were fulfilled.

Most of these requirements were met by creating a central database. Creating a central database improves accessibility and data integrity. [21] It also allows concurrent use and allows for a web interface to be created (data access). Using a DBMS that is based on Standard Query Language (SQL) which is a widely used database language makes it scalable. [26] MySQL is an ideal DBMS for development because it is free, widely used and fairly robust. [27] Data entry and retrieval requirements were met by testing these parameters in house.

Unfortunately, one requirement was not met and these were built in biological triggers. This is a requirement that was taken from existing database examples, IRIS and PedSys. Biological triggers would enhance data integrity by enforcing correct data entry. It was deemed at this point that since information was taken from IRIS and PedSys that the data integrity was already in tact because these databases already have biological triggers. It would seem logical that the information taken from these databases and put into another database would also have data integrity. Information is only taken from these places to put into the database. The database is not the initial point of data entry therefore, biological triggers were not coded in.

Presently, the framework is being used. The DAR and at least one researcher has approached the GRIP with their queries. I have been able to answer their questions using PedSys and will start using the database as well. I have been receiving positive feedback especially from the DAR. Their queries usually deal with colony management and marker information (who has samples and who do not).

6. Discussion

The presented data model and prototype is meant to act as a spring board to begin to capture all the incoming information which currently is being scattered around at the ONPRC. This was an initial step toward gathering and maintaining data integrity in this specific information model. The next step would be to put the prototype through a more rigorous and qualitative evaluation. This step will naturally be taken because the prototype will be used by the GRIP. This is a type of evaluation when a system can be used in real world processes and the real bugs and problems become apparent. Also, as mentioned before as information becomes increasingly diverse this data model will need to change and adapt. The timing and the way that this information comes in will dictate how the data model will change.

As suggested in an earlier section, this database is not meant to supplant IRIS. In fact, it should work very well to complement it. IRIS should still be the central repository for information but it should not be the only place for it. It would be ideal for IRIS to maintain information about the genetic parentage but would also point to the database if any further information were required. PedSys should also still be utilized since it is a very useful tool for genetic analysis. It is assumed that any further information created from DNA typing will be sent directly to GRIP in PedSys format. Using the Perl script this information will then be inputted directly into the database. A system needs to be put into place to put genetic parentage into IRIS.

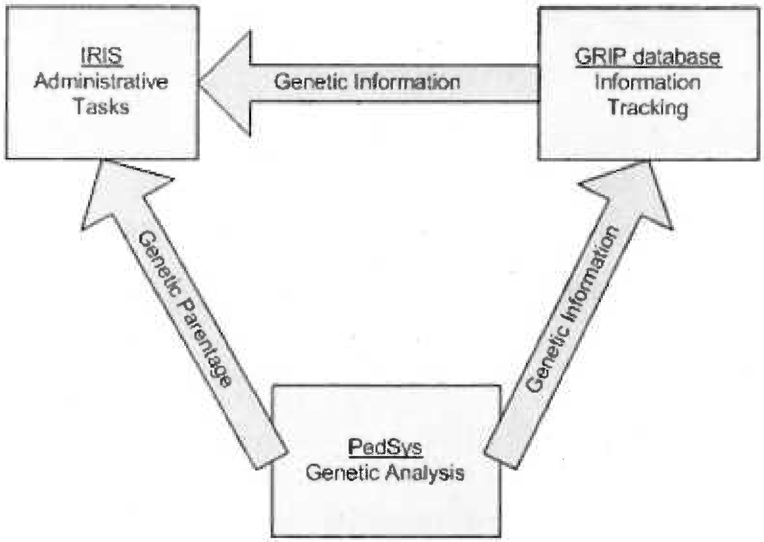


Figure 6: A proposed federation of databases between IRIS, PedSys and the GRIP database.

A point to remember if the actual implementation goes along these lines is that keeping data in different places poses a question about data integrity. A system should be put into place where the information in all three places is identical. This could be solved by using PedSys at all times to read information into both IRIS and the database.

As different data types are being created and entered into the information flow a re-evaluation of what is required for this data model will be inevitable. As mentioned before there are schemas that are present to handle this. An EAV schema may be a possibility for the future if the ONPRC receives an influx of new data and new data types that fit the requirements of an EAV schema.

7. Summary

Although, this project has documented the information flow and has created a working database and information system prototype, there are many more tasks that should be accomplished to answer the ONPRC's information needs. This project is a decent initial foray into answering those information needs. As stated in the methods section, this whole process is highly iterative. The next steps for this project should include:

- user evaluation of prototype
- built in biological triggers
- link to IRIS

There are also ongoing processes that should occur for the whole duration of the project. These are:

- requirements gathering

- evaluation of other schemas and databases
- re-evaluation of current information model

This project will not end here but continue on until, hopefully, a complete user friendly database is working.

It should be mentioned that in the bigger scope, marker information is just one data type that exists at the ONPRC. There are many other data types that are being housed in separate laboratories. For example, Dr. Eliot Spindle researches the effects of nicotine during pregnancy and how nicotine effects lung tumors. He uses microarray technology to help him identify genes that are affected. He has microarray expression data and gene sequences. Another example is of Dr. Michael Conn who uses proteomics to research gonadotropin releasing hormone (GnRH). GnRH is an important factor in raising or lowering the levels of steroids. Dr. Conn has proteomic data types. [1]

A future direction would be to include more user groups and information systems for the GRIP to analyze. The GRIP has the ability to help ONPRC researchers to safely store and efficiently utilize their information. Ideally, the GRIP should be able to create information repositories much like the Mouse Genome Informatic (MGI) group at Jackson Laboratories. MGI has been able to create an online central resource for mouse genomic research. Users are able to do dynamic searches that are linked to other important databases such as GenBank. Users are also able to submit their own information into the database. In short, it acts very much like a laboratory management system (LIMS). MGI has effectively provided a resource to the mouse model organism research community. In much the same way, the GRIP is starting down the same path.

REFERENCES

1. ONPRC website. 2004.
2. Bakker, A.R., et.al., Handbook of Medical Informatics, J.H. van Bemmel, Editor. 1999, Springer-Verlag Heidelberg: Heidelberg, Germany.
3. Dubay, C.J., et.al., Genetics Resource & Informatics Program (GRIP). 2003, Oregon National Primate Research Center: Portland.
4. Ermolaeva, O., et al., Data management and analysis for gene expression arrays. Nat Genet, 1998. 20(1): p. 19-23.
5. Roos, D., Bioinformatics--Trying to Swim in a Sea of Data. Science, 2001. 291(5507):p. 1260-1261.
6. Kettunen, E., et al., Differentially expressed genes in nonsmall cell lung cancer: expression profiling of cancer-related genes in squamous cell lung cancer. Cancer Genet Cytogenet, 2004. 149(2): p. 98-106.
7. DeWeerdt, S.E., What's a Genome?, ed. B.J. Culliton. 2000, Rockville, Maryland: The Center for the Advancement of Genomics (TCAG).
8. Schena, M., et al., Microarrays: biotechnology's discovery platform for functional genomics. Trends Biotechnol, 1998. 16(7): p. 301-6.
9. Lesk, A.M., Introduction to Bioinformatics. 2002, New York, New York: Oxford University Press Inc.
10. Kohane, I.S., et. al., Microarrays for a Integrative Genomics. 2003, Cambridge, Massachusetts: The MIT Press.
11. Eichler, E.E. and P.J. DeJong, Biomedical applications and studies of molecular evolution: a proposal for a primate genomic library resource. Genome Res, 2002. 12(5): p. 673-8.
12. Williams-Blangero, S., J.L. VandeBerg, and B. Dyke, Genetic management of nonhuman primates. J Med Primatol, 2002. 31(1): p. 1-7.
13. Morin, P.A., S. Kanthaswamy, and D.G. Smith, Simple sequence repeat (SSR) polymorphisms for colony management and population genetics in rhesus macaques (Macaca mulatta). Am J Primatol, 1997. 42(3): p. 199-213.
14. Dyke, B., PedSys User's Manual, in A Pedigree Data Management System. 1999, Southwest Foundation for Biomedical Research: San Antonio, Texas.
15. Stone, W.H., Genetic research with nonhuman primates: serving the needs of mankind. Symposium summary and future prospects. Genetica, 1987. 73(1-2): p. 169-77.
16. Rogers, J., et.al., Recommendations for Future Efforts in Primate Genomics. 2001, National Center for Research Resources: Seattle, Washington. p. 14.
17. Nadkarni, P.M., et al., Organization of heterogeneous scientific data using the EAV/CR representation. J Am Med Inform Assoc, 1999. 6(6): p. 478-93.
18. Anhoj, J., Generic design of web-based clinical databases. J Med Internet Res, 2003. 5(4): p. e27.
19. Marenco, L., et al., Neuronal database integration: the Senselab EAV data model. Proc AMIA Symp, 1999: p. 102-6.
20. Cain, S., Generic Model Organism Database Construction Set. 2004, National Institutes of Health.

21. Rolland, F.D., The Essence of Databases. 1998, Harlow, England: Prentice Hall. 226.
22. Ramakrishnan, R., et. al., Database Management Systems. International Edition 2003 ed. 2003, Singapore: McGraw-Hill Education. 1065.
23. Rubin, D.L., et al., Representing genetic sequence data for pharmacogenomics: an evolutionary approach using ontological and relational models. Bioinformatics, 2002. 18 Suppl 1: p. S207-15.
24. Fowler, M., et. al., UML Distilled. Second ed, ed. G. Booch, et. al. 2000, Reading, Massachusetts: Addison-Wesley. 185.
25. Stein, L.D., et al., The generic genome browser: a building block for a model organism system database. Genome Res, 2002. 12(10): p. 1599-610.
26. Kriegel, A., et.al., SQL Bible. 2003, Indianapolis, Indiana: Wiley Publishing.
27. Gilfillan, I., Mastering MySQL 4. 2003, Alameda, California: SYBEX.

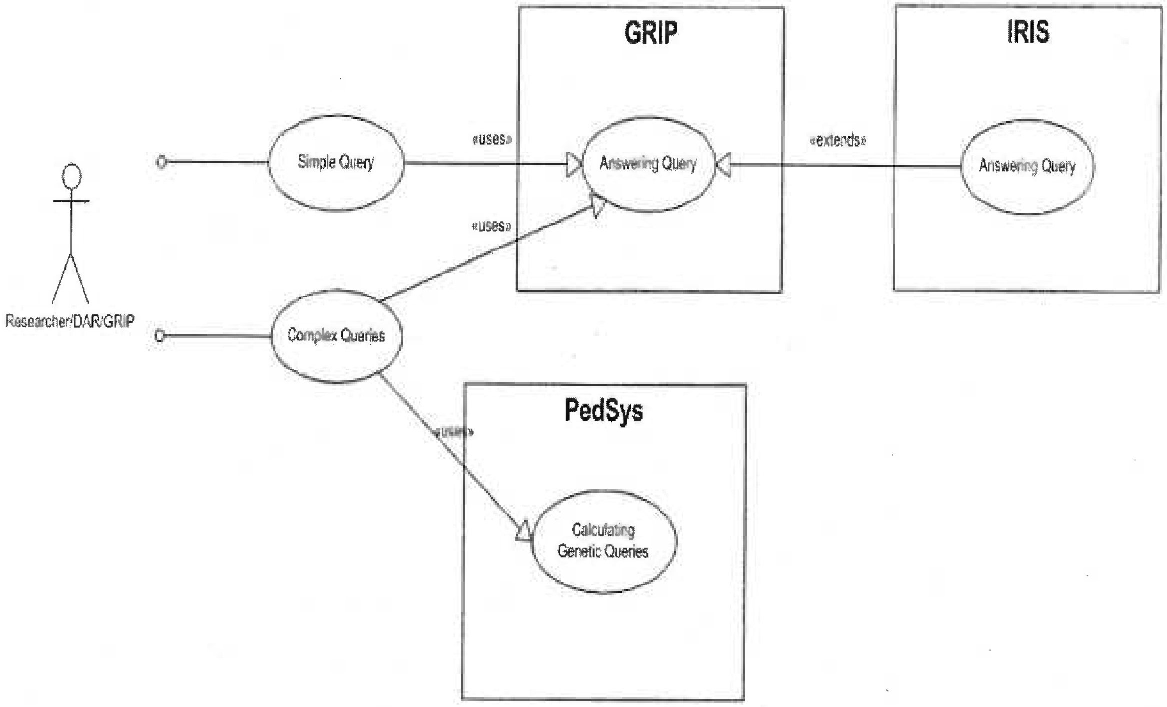
APPENDIX A – Use cases and diagrams

Use Case: Simple query for primate information such as sex, birth date, marker information.
Actors: Researcher/DAR/GRIP
Main Flow:

1. Actor accesses database by logging in with user name and password.
2. Actor types in primate identifier and selects information desired.
3. Application goes to local database and does search.
4. Application returns desired information to the actor.

Alternative Flow: *Login Failure*

1a. At step 1, database fails to recognize user name and password. Allow user to re-enter user name and password.

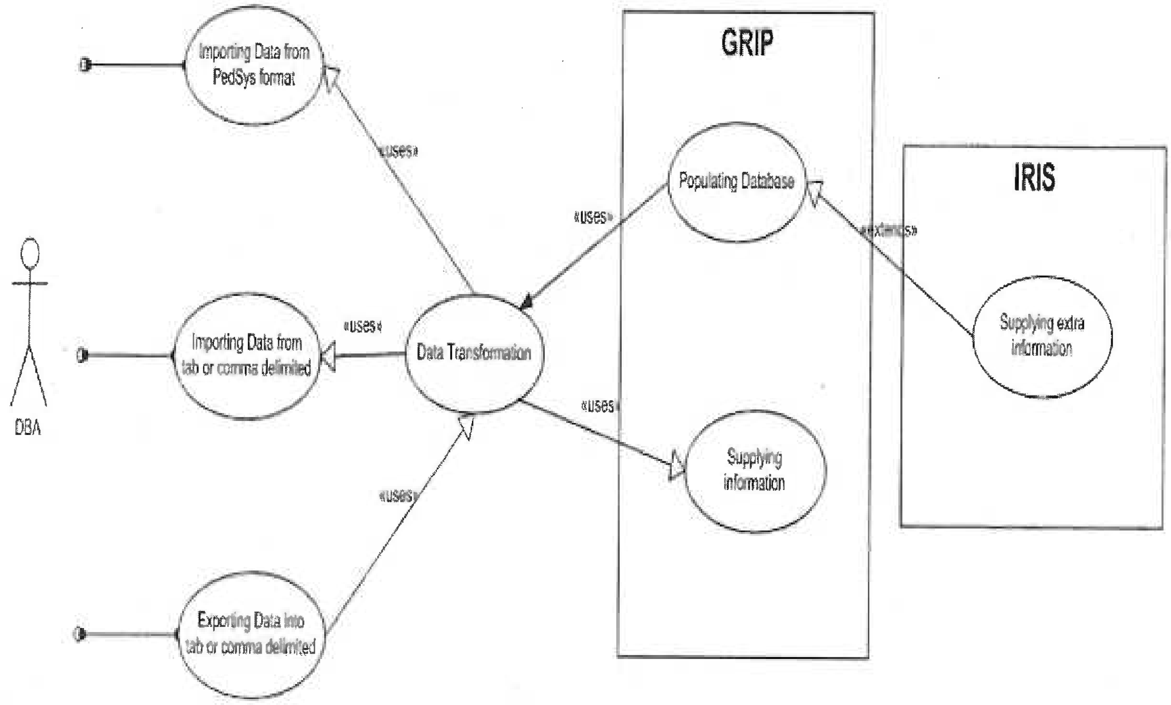


APPENDIX A – Use cases and diagrams

Use Case: Importing/exporting data into or out of different sources.
Actors: GRIP
Main Flow:
1. User receives information in PedSys format.
2. User uses a Perl script to read in this data into database.

Alternative Flow: *Fixed formats*
1a. User receives information in a fixed format.
1b. User translates this data into PedSys format using PedSys utility.
2a. User uses a Perl script to read in this data into database.

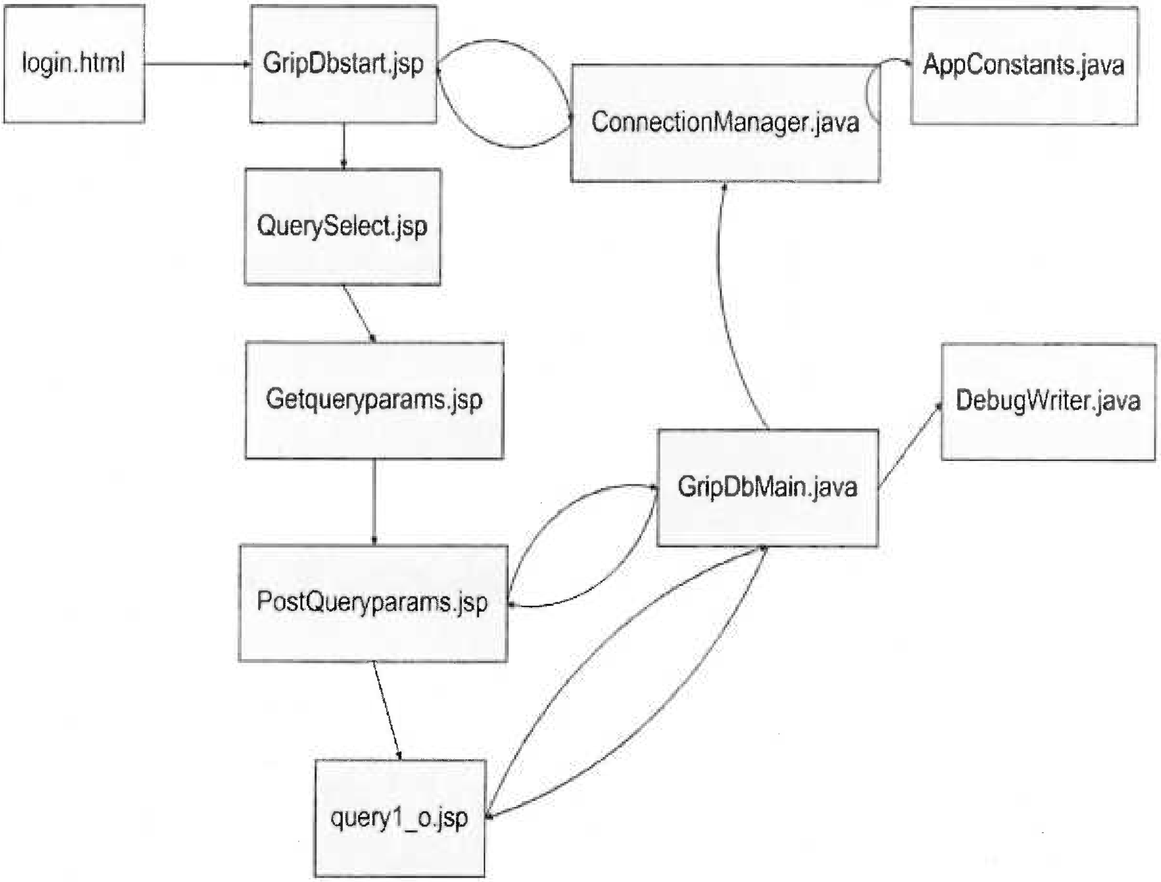
Alternative Flow: *Exporting data*
1a. Instead of receiving information user wants to export information.
1b. User puts database output into fixed format.


Importing Data from PedSys format
«uses»
GRIP
Populating Database
«uses»
IRIS
«extends»
Supplying extra information
Importing Data from tab or comma delimited
«uses»
Data Transformation
«uses»
Supplying information
«uses»
Exporting Data into tab or comma delimited
DBA

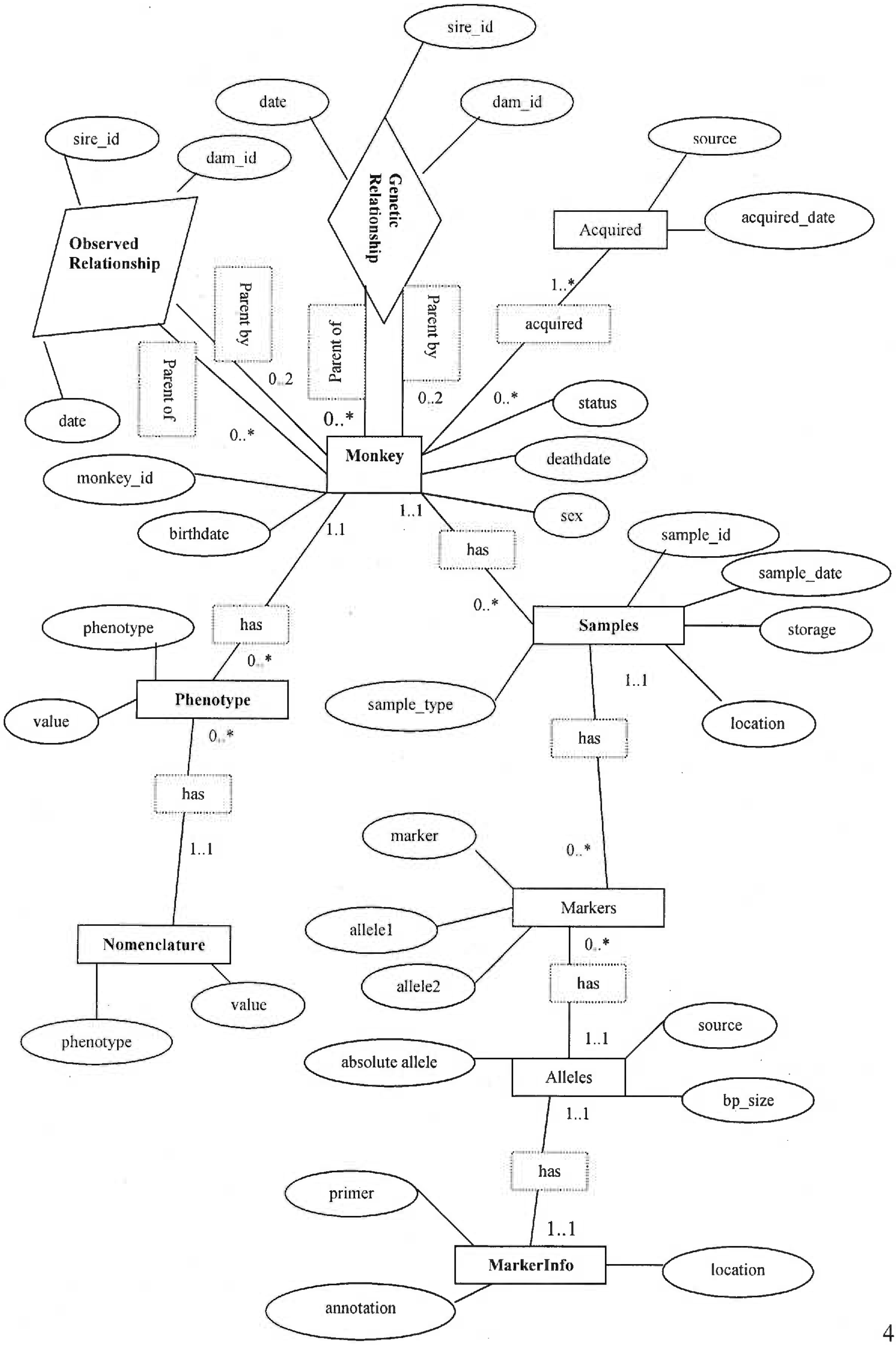
APPENDIX B – SQL

```
# GRIP MySQL Database Creation Script
#
# Host: localhost    Database: GRIP
#-------------------------------------------------------
# Server version

create database GRIP;
use GRIP;

#
# Table structure for table 'monkey'
#

CREATE TABLE monkey (
  monkey_id char (5) NOT NULL,
  birthdate date,
  deathdate date,
  sex enum ('M', 'F', 'H', 'U') default 'U',
  status enum ('Dead','Sold','Live'),
  PRIMARY KEY (monkey_id)
) TYPE=InnoDB;


#
# Table structure for table 'observedrelationship'
#

CREATE TABLE observedrelationship (
  monkey_id char (5) NOT NULL,
  INDEX mon_id(monkey_id),
  sire_id char (5),
  dam_id char (5),
  observation_date timestamp NOT NULL,
  PRIMARY KEY (monkey_id),
  CONSTRAINT X1 FOREIGN KEY (monkey_id) REFERENCES monkey
(monkey_id)
) TYPE=InnoDB;


#
# Table structure for table 'geneticrelationship'
#

CREATE TABLE geneticrelationship (
  monkey_id char (5) NOT NULL,
  INDEX mon_id (monkey_id),
  sire_id char (5),
  dam_id char (5),
  observation_date timestamp NOT NULL,
  PRIMARY KEY (monkey_id),
  CONSTRAINT X2 FOREIGN KEY (monkey_id) REFERENCES monkey
(monkey_id)

) TYPE=InnoDB;

#
# Table structure for table 'samples'
#

CREATE TABLE samples (
  sample_type enum ('blood') default 'blood',
  sample_id varchar (5)NOT NULL,
  monkey_id char (5)NOT NULL,
  INDEX mon_id (monkey_id),
  sample_date date,
  storage varchar (10) default NULL,
  location varchar (10) default NULL,
  PRIMARY KEY (sample_id),
  CONSTRAINT X3 FOREIGN KEY (monkey_id)REFERENCES
monkey(monkey_id)
) TYPE=InnoDB;

#Table for 'acquired'

CREATE TABLE acquired (
  monkey_id char (5)not null,
  INDEX mon_id (monkey_id),
  acquired__date date,
  PRIMARY KEY (monkey_id),
  CONSTRAINT X5 FOREIGN KEY (monkey_id)REFERENCES monkey
(monkey_id)
) TYPE = InnoDB;


#Table for 'Phenotype'

CREATE TABLE phenotype (
  monkey_id char (5)not null,
  INDEX mon_id (monkey_id),
  phenotype varchar (50)not null,
  value varchar (50)not null,
  date date,
  PRIMARY KEY (monkey_id, phenotype),
  CONSTRAINT X6 FOREIGN KEY (monkey_id)REFERENCES
monkey(monkey_id)
) TYPE = InnoDB;

#Table for 'alleles'

CREATE TABLE allele (
  abs_allele char (5)not null,
  allele char (5)not null,
  marker varchar (50)not null,
  source varchar (50)not null,
  bp_size varchar (50),
  PRIMARY KEY (allele, marker, source),
  KEY (abs_allele),
) TYPE = InnoDB;


#
#Table structure for table 'Markers'
#

CREATE TABLE markers (
  sample_id varchar (5) NOT NULL,
  INDEX sam_id (sample_id),
  marker varchar (50) NOT NULL,
  abs_allele1 varchar (10) NOT NULL,
  abs_allele2 varchar (10) NOT NULL,
  INDEX abs (abs_allele1),
  INDEX abs2 (abs_allele2),
  PRIMARY KEY (sample_id, marker),
  KEY (marker),
  CONSTRAINT X4 FOREIGN KEY (sample_id)REFERENCES
samples(sample_id),
  CONSTRAINT X8 FOREIGN KEY (abs_allele1)REFERENCES allele
(abs_allele),
  CONSTRAINT X10 FOREIGN KEY (abs_allele2)REFERENCES allele
(abs_allele)
) TYPE = InnoDB;


#Table for 'Marker Info'

CREATE TABLE MarkerInfo (
  abs_allele varchar (10)not null,
  INDEX abs (abs_allele),
  location varchar (50)not null,
  primer varchar (50),
  annotation varchar (50),
  PRIMARY KEY (abs_allele),
  CONSTRAINT X7 FOREIGN KEY (abs_allele) REFERENCES allele
(abs_allele)
) TYPE = InnoDB;

#Table for 'nomenclature'

CREATE TABLE nomenclature (
  phenotype varchar (50)not null,
  INDEX pheno (phenotype, value),
  value varchar (50)not null,
  comment varchar (50),
  PRIMARY KEY (phenotype),
  CONSTRAINT X9 FOREIGN KEY (phenotype) REFERENCES
phenotype(phenotype)
)TYPE = InnoDB;
```

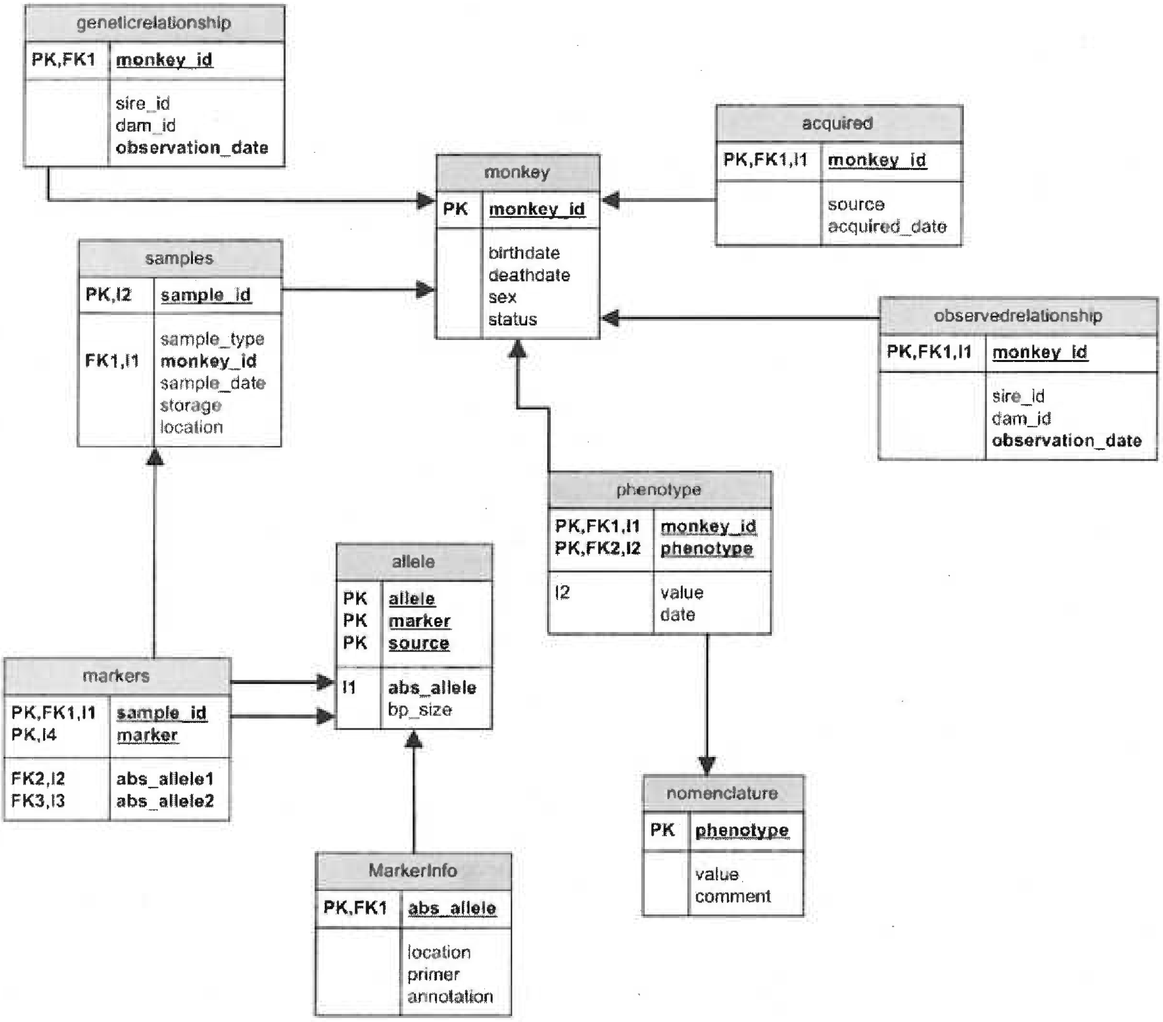
APPENDIX C – Architecture of database interface



APPENDIX D – Entity Relationship Diagram



APPENDIX E – Database Schema


geneticrelationship
PK,FK1
monkey_id
sire_id
dam_id
observation_date
monkey
PK
monkey_id
birthdate
deathdate
sex
status
acquired
PK,FK1,I1
monkey_id
source
acquired_date
samples
PK,I2
sample_id
FK1,I1
sample_type
monkey_id
sample_date
storage
location
observedrelationship
PK,FK1,I1
monkey_id
sire_id
dam_id
observation_date
phenotype
PK,FK1,I1
PK,FK2,I2
monkey_id
phenotype
I2
value
date
allele
PK
PK
PK
allele
marker
source
I1
abs_allele
bp_size
markers
PK,FK1,I1
PK,I4
sample_id
marker
FK2,I2
FK3,I3
abs_allele1
abs_allele2
nomenclature
PK
phenotype
value
comment
MarkerInfo
PK,FK1
abs_allele
location
primer
annotation