

NGSdb: A NGS Data Management and Analysis Platform for Comparative Genomics

Marea Cobb

A thesis

submitted in partial fulfillment of the

requirements for the degree of

Master of Science

University of Washington

2015

Committee:

Peter Myler

Neil Abernethy

William Noble

Program Authorized to Offer Degree:

Department of Biomedical Informatics and Medical Education

ProQuest Number: 1599783

All rights reserved

INFORMATION TO ALL USERS

The quality of this reproduction is dependent upon the quality of the copy submitted.

In the unlikely event that the author did not send a complete manuscript and there are missing pages, these will be noted. Also, if material had to be removed, a note will indicate the deletion.



ProQuest 1599783

Published by ProQuest LLC (2015). Copyright of the Dissertation is held by the Author.

All rights reserved.

This work is protected against unauthorized copying under Title 17, United States Code  
Microform Edition © ProQuest LLC.

ProQuest LLC.  
789 East Eisenhower Parkway  
P.O. Box 1346  
Ann Arbor, MI 48106 - 1346

©Copyright 2015

Marea Cobb

## Abstract

As researchers continue to expand the volume of Next Generation Sequencing data, the ability to store and query the data becomes increasingly important. The current approach of using spreadsheets has become too complex and the data too vast to efficiently store, view, cross query, analyze, and share among collaborators. We have created and implemented a relational database schema, **NGSdb** (*PostgreSQL*), coupled with a user-friendly web interface (Django/Python), to address this growing need. **NGSdb** currently has five core components: a **sample core**, which tracks the sample information (e.g., organism, growth phase); a **library core**, which tracks the libraries constructed from samples (e.g., library type, sequencing method, raw data files); a **genome core** which stores information about reference genomes; an **analysis core**, where the meta-information of bioinformatics analyses are stored; and a **result core** where the results of the bioinformatic analyses are stored. I have expanded **NGSdb** by developing two analysis modules; a **somy/CNV module** and **SNP module**. In addition to storing and retrieving the data, the web interface also serves as an analytical platform. The database is designed to be modular, allowing for future additions as new technology or data becomes available. The modularity enables us to query across our different data types, such as *SNP* data and *RNA-Seq* data (e.g., how does the expression level change when a gene is mutated?). We demonstrate the capabilities of our system through two separate case studies. The first recapitulates a recently published genomic analysis of two Sri Lankan strains of *Leishmania donovani*, one causing visceral disease (VL) and one causing **cutaneous** disease (CL). The second case study compares the genome of a laboratory-adapted strain of *L. donovani* with genetically modified clones derived from it: single (sKO) and double (dKO) deletions of the *dpkAR1* gene; and a derivative of dKO line that had recovered the wild type growth phenotype. We identified single nucleotide polymorphisms (SNP), copy number variation (CNV), and somy differences between these lines to expose what genomic differences may contribute to the growth phenotype recovery of the double knockouts. **NGSdb** successfully recaptured the analysis results previously published and identified a potential artifact in the second study. Through these analysis we have also established additions to **NGSdb** that we believe will further increase the usability of the system.

## Tables of Contents

<b>TABLES OF CONTENTS .....</b>	<b>4</b>
<b>SECTION 1: INTRODUCTION .....</b>	<b>6</b>
1.1 LEISHMANIA.....	8
1.2 GENOMICS.....	11
1.2.1 Next Generation Sequencing.....	11
1.2.2 Genomic Annotation.....	12
1.2.3 Genomic Analysis.....	13
<b>SECTION 2: LIMITATIONS OF EXISTING SYSTEMS .....</b>	<b>15</b>
2.1 MANAGING NEXT GENERATION SEQUENCES.....	15
2.1.1 Space Limitation.....	15
2.1.2 Sharing Sequences.....	16
2.1.3 Integration of Meta-data and Sequences.....	17
2.2 GENOMIC ANALYSIS.....	18
2.2.1 Reference Genomes.....	18
2.2.2 Comparing Results.....	19
2.3 CURRENT GENOMIC DATABASES AND APPLICATIONS.....	19
<b>SECTION 3: ARCHITECTURE OF NGSDB .....</b>	<b>23</b>
3.1 DATABASE AND SCHEMA.....	23
3.2 WEB INTERFACE DESIGN.....	27
3.3 WEB APPLICATION DEPLOYMENT.....	29
<b>SECTION 4: DEVELOPMENT OF SPECIFIC MODULES .....</b>	<b>31</b>
4.1 DEVELOPMENT OF SOMY AND CNV MODULE.....	31
4.2 DEVELOPMENT OF SNP MODULE.....	32

<b>SECTION 5: SRI LANKAN <i>LEISHMANIA</i> TROPISM .....</b>	<b>34</b>
5.1 METHODS.....	34
5.2 SOMY COMPARISON .....	35
5.3 COPY NUMBER VARIATION .....	36
5.4 SINGLE NUCLEOTIDE POLYMORPHISM.....	39
<b>SECTION 6: <i>LEISHMANIA</i> COMPARISON OF <i>DPKAR1</i> KNOCKOUT .....</b>	<b>45</b>
6.1 METHODS.....	46
6.2 SOMY COMPARISON .....	49
6.3 COPY NUMBER VARIATION .....	50
<b>SECTION 7: CONCLUSIONS.....</b>	<b>56</b>
7.1 DISCUSSION .....	56
7.2 NOVEL CONTRIBUTIONS .....	58
7.3 FUTURE DIRECTIONS.....	58
<b>LIST OF FIGURES .....</b>	<b>60</b>
<b>LIST OF TABLES.....</b>	<b>61</b>
<b>WORKS CITED .....</b>	<b>62</b>

## Section 1: Introduction

The impact of infectious disease has undoubtedly been a tremendous force over the course of human history. Tuberculosis can be found 5,000 years ago in ancient Egyptian mummies [1]. The Plague of Athens, occurring from 429 BC to 426 BC, is the first documented epidemic, most commonly thought to have been an outbreak of typhoid fever or smallpox. The Athenian plague resulted in 75,000 to 100,000 deaths, roughly 25% of Athens' population [2]. Further epidemics include the Black Death of the 1400s; smallpox, which spread like wildfire through Europe in the 1500s; and the Yellow Fever Epidemic of 1793, striking Philadelphia, Pennsylvania [3]–[5].

In the modern world infectious diseases still run rampant and are estimated to cause over 8 million deaths a year, leaving many more people suffering from their life-long effects. In particular, impoverished countries bear the brunt of these tragedies due to a lack of food, shelter, clean water, and available healthcare. In 1999 the World Health Organization (WHO) identified over 20 different infectious diseases still plaguing the modern world, including malaria, measles, and Ebola [6]. Of those 20, 17 are considered neglected tropical diseases, affecting over 1 billion people [7]. The development of accessible preventative care and treatment is crucial for reducing the hardship and loss caused by infectious disease. Research into these illnesses can help stem the tide and improve the overall health of the world's population.

Fortunately, medical research has advanced tremendously in the past half-century. In 1968, the first DNA was sequenced and by 1977, modern DNA sequencing methods were in practice [8]. In 1985, discussions began about sequencing the human genome and by 1990, a five-year research grant was presented to Congress: the beginning of the Human Genome

Project. Over the next decade, the human genome was sequenced, shedding light on the structure and function of DNA.

Mass sequencing can provide insight into human diseases, including the identification of disease biomarkers for cardiovascular disease [9] and has led to new initiatives, such as the *1000 Genomes Project*, which use mass sequencing to identify regions of the genome associated with disease traits. In 2014, the National Institute of Allergy and Infectious Disease established the Genomic Center for Infectious Disease to further sequence a variety of emerging pathogens ranging from smallpox to salmonella. It is clear that research must continue on the sources and etiologies of these diseases.

With more than 35 years of sequencing, its speed and ability has vastly improved as the technology has advanced from Sanger sequencing to the more recent parallel sequencing platforms, such as Illumina. The growing number of DNA sequences has created a need for tools to help understand and analyze the data. Duncan McCallum and Michael Smith were some of the first individuals to develop programs that assist in this process and helped establish the field of bioinformatics [8]. McCallum, with the guidance of Smith, wrote a program that numbered sequences, queried for sub-sequences, helping identify patterns unique to genes, and translated the sequence into amino acid sequences. Since then, a great many more analytical tools and sequence databases have been developed and put into standard practice. While these tools provide means to store and analyze genomic sequences, there are still many areas which can be improved upon including tools that are capable of both mass storage and genomic analysis that drive research and develop hypotheses.

This paper describes a web application and database, **NGSdb**, that can assist researchers in storing and analyzing genomic data. We explore the development and architecture of **NGSdb**,



focusing on two separate modules, the **SNP module** and the **CNV/Somy module**, and, in order to demonstrate its capabilities, perform two test studies comparing the genomes of different *Leishmania* libraries.

## 1.1 Leishmania

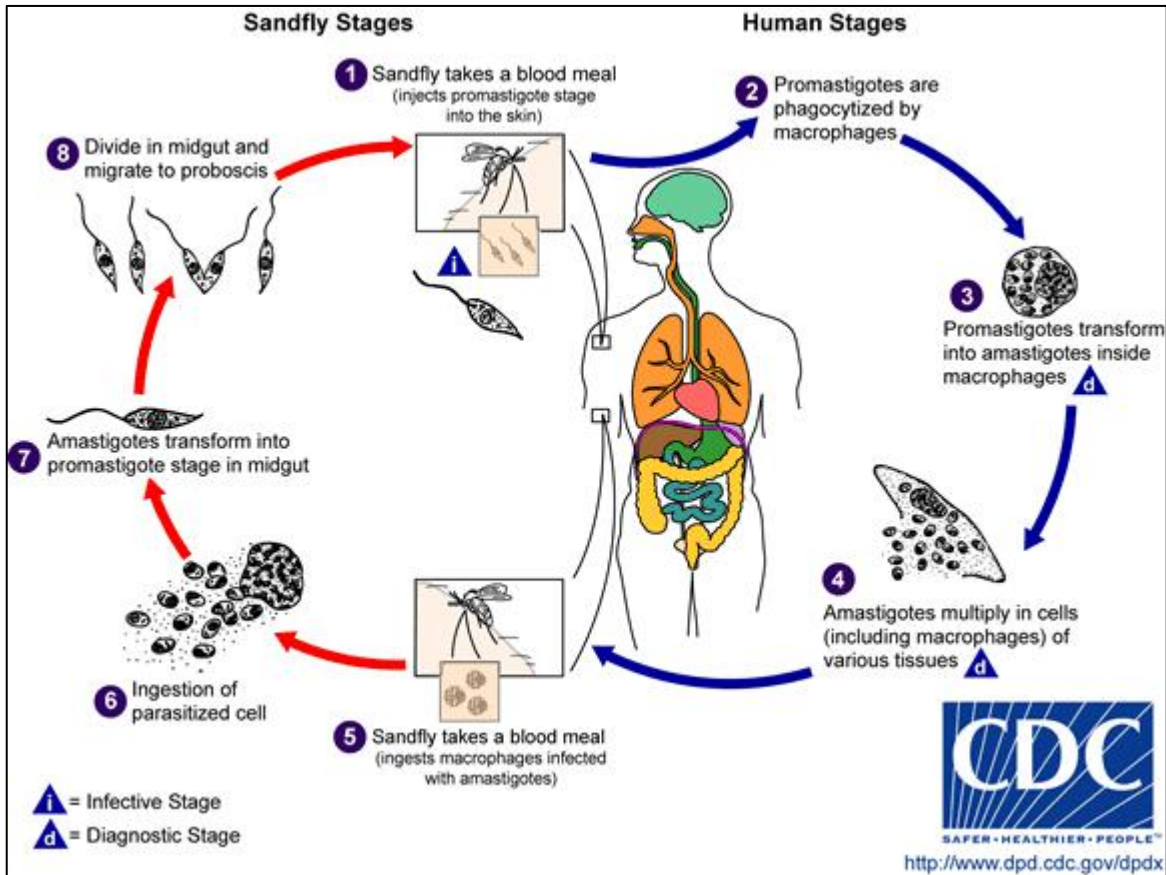
Leishmaniasis is a parasitic disease classified by the World Health Organization (WHO) as a Neglected Tropical Disease. There are over 30 different strains of *Leishmania*, the pathogen, found on every continent except Antarctica. An estimated 1.3 million new cases of leishmaniasis are reported to the World Health Organization (WHO) annually [11]. Currently, there are no vaccines or drugs capable of preventing leishmaniasis. The Center for Disease Control (CDC) provides preventative actions aimed at reducing the risk of infection [12].

*Leishmania* is part of the order Trypanosomatida. Of the 30 species of *Leishmania*, 21 cause leishmaniasis in vertebrates [13]. While the species appear morphologically similar, their enzymatic activity, antibodies, and molecular functions vary greatly. The differences between the strains increase the difficulty in developing a general preventative vaccine or treatment. Individual strains can be responsible for one of three forms: **cutaneous**, **visceral**, or **mucosal**. These forms have varying effects on the human host. **Cutaneous Leishmaniasis (CL)** is the most common type and results in skin sores. **Visceral Leishmaniasis (VL)** infects internal organs including the spleen, liver, and bone marrow [14]. This infection causes fever, weight loss, swelling of the spleen or liver, low blood counts, and low platelet counts. Without treatment, the infection becomes fatal. **Mucosal Leishmaniasis (MCL)** is a less common, secondary infection of the mucous membranes. The parasite initially causes skin sores, same as those seen in **CL**, eventually spreading to the mucous membranes of the nose, mouth, or throat [14].

The life cycle of *Leishmania* consists of the sand fly stage and the vertebrate stage.

Figure 1 displays an overview of the steps the *Leishmania* life cycle [13]. When a sand fly bites into the vertebrate host, metacyclic promastigotes are transferred into the vertebrate bloodstream. The vertebrate's immune system responds by phagocytizing the promastigotes into phagosomes within macrophages [15]. The phagosomes undergo remodeling and maturation, developing into parasitophorous vacuoles (PVs) within two to five days, varying by species [16]. Traditionally, vertebrate' macrophages protect vertebrates from infection by fusion of the phagosome and lysosome, creating an autophagolysosome, and then digesting the parasite. *Leishmania* are able to evade the host immune response by interfering with the macrophage signaling machinery and preventing the formation of autophagolysosome [17], [18].

During the formation of PVs, the promastigotes differentiate into amastigotes, the intracellular form of *Leishmania*. The amastigotes then proliferate inside the PVs and begin to spread throughout the vertebrate host. As the amastigote spreads, the likelihood of a second sand fly feeding off of the infected host's blood increases. Within the newly infected sand fly's gut, the amastigotes revert back into promastigotes. The promastigote then divides and migrates to the proboscis where it once again infects the next vertebrate host [15].



**Figure 1 - *Leishmania* life cycle [13]**

Individuals are diagnosed with leishmaniasis through a variety of lab tests, including microscopy, isoenzyme analysis, serology tests, and molecular diagnosis. The diagnosis is critical for identifying the proper treatment an individual should undergo. While microscopy can identify leishmaniasis in a blood sample, it is unable to identify the strain. Serology tests, molecular diagnosis, or isoenzyme tests are necessary to specify the strain of leishmaniasis [19].

Once a diagnosis has been made, treatment can begin. **Cutaneous disease** are generally treated through management of the skin sores and associated pain. While the sores can resolve naturally, drug treatments are recommended to prevent secondary infection and the development of **Mucosal Leishmaniasis**. **Visceral Leishmaniasis** is treated through two types of systemic therapy: parenteral and oral. Parenteral therapies include the injection of pentavalent antimonial or sodium stibogluconate for 10 to 28 days, pentamidine, paromycin, and amphotericin B

deoxycholate. Oral therapy involves the digestion of miltefosine or antifungal drugs including ketoconazole, itraconazole, or fluconazole for up to six weeks [19].

Over the past few decades, there has been a push to understand the underlying cellular and molecular mechanisms that allow *Leishmania* to evade the vertebrate immune system. A focus on how the parasites differentiates from their promastigote form to their amastigote form is of great interest as only amastigotes are infectious to vertebrates. Understanding how differentiation is initiated and regulated may lead to new drug targets. Genomic sequencing can help identify genes that may play a role in differentiation and potentially genomic locations or proteins that should be targeted to prevent infection of vertebrates.

## 1.2 Genomics

### 1.2.1 Next Generation Sequencing

Next Generation Sequencing (NGS) is a method of DNA sequencing, including genomic sequencing, resequencing, transcriptional profiling (RNA-Seq), and DNA-protein interactions (ChIP-Seq), which stemmed from the need for cheaper and faster sequencing. The assembly of reference genomes through sequencing of wild type samples using the Sanger method drove the emerging field of NGS.

The first step in NGS is to create a library from fragmented pieces of the DNA. Each fragment is sequenced in parallel. The reads are then reassembled against a reference genome, a representative example of a species genome, and the full alignment provides the DNA sequence. A secondary type of assembly, known as *de novo* assembly, reassembles fragments without a reference genome. This can occur when a new species is being sequenced or the assembled reference genome is known to have errors. By using parallel sequencing, a large amount of sequences can be processed in a short amount of time. As of 2013, five human genome

sequences can be sequenced in a single run, taking about a week, and costing less than \$5,000 per genome. In comparison, the Human Genome Project cost a total of \$2.7 billion and took 13 years to complete [20].

For traditional assembly of NGS, the resulting sequences are only as reliable as the reference genome. Trouble with misassembled genomes can lead to errors in future analysis. These can include false-positive or false-negative identification of SNPs, the incorrect number of gene copies, or large gaps throughout the genome. Similarly, *de novo* sequencing comes with its own challenges. Without a reference genome, the short sequence fragments produced may result in sequence gaps; areas within the sequence where no reads align, creating smaller contigs. A solution to these gaps is to use paired-end sequencing where DNA fragments are sequenced from both sides, helping remove some of them. The method of sequencing is important to track for downstream analysis as it can contribute to the quality of the analysis. For example, older sequencing techniques do not provide as accurate of sequencing, due to smaller read depths, and an overall smaller power of detecting true heterogeneity of the sequences.

### 1.2.2 Genomic Annotation

Once the DNA sequence is aligned, known genomic information is mapped to segments of the DNA through a process known as genomic annotation. This process predicts the location of protein-coding genes, tRNAs, small RNAs, pseudogenes, control regions, transposons, and other biologically relevant information. Computational tools, such as NCBI's Prokaryotic Genome Automatic Pipeline, have been developed to assist in these predictions [21].

These computational tools compare the sequenced genome to an orthologous annotated genome and/or identify patterns unique to genes, protein-coding regions, etc [22]. Basic Local Alignment Search Tool (BLAST) is commonly used to identify homologous genes within

phylogenetically related genomes. BLAST can be used as a first pass to identify known genes. Hidden Markov models can be used to predict the probability of a smaller DNA segment belonging to a protein-coding region after they have been trained on a known genome [23]. A few tools, including Ensembl's genome annotation pipeline and NCBI's Prokaryotic Genome Automatic Pipeline, combine these methods to provide a comprehensive annotation while the majority are standalone tools that researchers can combine for a comprehensive annotation. Analysis platforms must be able to account for the various types of annotations and the software which determined them.

### 1.2.3 Genomic Analysis

Without interpretation or analysis, sequences do not provide researchers with biologically relevant information. Genomic analysis consists of identifying various features of genomes including structural variation, sequence variation, and functional variation. Common features include calculating some and copy number variation (CNV) values, determining individual gene expression values, and identifying single nucleotide polymorphisms (SNPs) within the genome. Moreover, many researchers are interested in comparing across genomes to identify differences amongst samples. Publicly available algorithms are available to assist researchers in these analyses.

Somy is a subset of aneuploidy and represents the number of individual chromosome copies in a genome. Trisomy 21 is a human example where only one chromosome is found to have three copies. CNV is similar to somy but refers to the number of copies a specific section of a chromosome contains. CNVs are typically calculated for genes but can also be determined for ranges of the chromosome. SNPs are traditionally defined as a single base difference from the

reference genome but now is commonly accepted as changes in short regions (<10 basepairs) of the DNA. These changes can also include small insertions and deletions (indels).

## Section 2: Limitations of Existing Systems

### 2.1 Managing Next Generation Sequences

#### 2.1.1 Space Limitation

One of the largest challenges of working with sequencing data is how to manage it in a space-efficient manner. Illumina sequencing can result in more than 100 gigabytes of raw reads per lane [24]. Developing a data management system that allows for effective analysis has been identified as one of the most pressing needs in bioinformatics [25]. As DNA sequencing has become cheaper, labs have begun to utilize it at a significantly higher rate such that storing the data has become a major portion of the total budget [26]. Workflows have been improved to remove unneeded raw data as soon as possible, decreasing the long-term storage requirements. To assist, tools such as *bgzip* have been created to compress raw sequence files minimizing the space required for long-term storage [27].

The bioinformatics community has developed three approaches to handle the growing space requirement: “(1) add storage; (2) throw away some data (“triage”); and (3) compress the stored data” [26]. Many argue that there is no need to store the raw sequencing data long term but that it is sufficient to only store the output of the analysis. Adding storage is becoming cheaper, but many smaller labs are unable to afford or justify the price of increased storage. Scientists are hesitant to throw older data away in case they wish to return to it. The third approach is the most common amongst bioinformaticians. By leveraging compression algorithms, they are able to dramatically reduce the storage footprint of genomic data. Testing has shown that *bgzip* reduces standard SNP data to roughly 10 to 20 percent of its original size [28]. Guy



### 2.1.2 Sharing Sequences

Separate from the concern of data storage is that of data transfer. The sheer size and quantity of genomic data creates many difficulties in sharing amongst collaborators. The most accessible solution, basic email, is not well suited for this function. Standard email servers only allow up to about 20 megabytes in the attachments. A more viable solution is to store the sequences on a centrally available server that users can access from internal or external networks. This is also advantageous because it allows whole machines to be dedicated to storing genomic data. Certain institutions such as the University of California Santa Cruz (UCSC) and the European Molecular Biology Lab's European Bioinformatics Institute (EMBL-EBI) provide public access to databases containing genomic databases and software stacks to leverage it. These projects, *UCSC Genome Browser* [29] and *Ensembl* [30] respectively, may also be downloaded and hosted on the lab's own servers.

While there are many services that allow published sequences to be shared and/or stored, internal sequences are another matter. Prior to publication, researchers may need to share their sequence files with collaborators without submitting them to public databases. The internal file systems of a remote machines requiring credentialed access is one solution. Typically, due to security concerns, the creation of a remote machine requires the assistance of IT specialists and separate hardware, both of which smaller labs may be unable to fit into their operating budgets.

For labs that are unable to afford an IT specialist, they can turn to newer technology and store their data in the cloud [31]. Cloud based services, such as Amazon Web Services (AWS), DNAnexus, and the Google Cloud Platform, allow users to upload their data and run their analysis on these service provider's hardware. Users are charged on a monthly basis for the amount of storage and CPU they utilize, allowing them fine-grained control over their budget.

Storing the information on the cloud allows all collaborators to access the data from any computer with Internet access. One drawback to these cloud services is the limited control over cyber security. While research has shown that the cloud provides a more secure environment, researchers must give up control of potentially sensitive information [32]. Depending on the grants funding the research, storing the data in the cloud may not be a possible solution.

While storing sequencing data is a hurdle for many labs, they must also consider how they may be sharing the data internally and externally. These requirements must be addressed when considering how best to store the raw data as well as the analysis. Remote machines have been used for years in practice and offer a strong solution for larger labs that can afford an IT professional. Cloud computing is now providing a similar but cheaper solution for researchers. Both of these systems address how to access the data but do not consider the best methods to begin analyzing the data.

### **2.1.3 Integration of Meta-data and Sequences**

Genomic data consists of not only raw sequencing data but also meta-information that includes, but is not limited to, the sample information (organism, date sequenced, treatments, life stage, etc.), sequencing information (version, model, software, etc.), and analysis information (software, software version, etc.). It is important to track this information for downstream analysis, visibility during publication, and reproducibility. Typically, metadata is stored in separate text files that researchers must track and associate with their corresponding dataset.

An individual sample may be processed and analyzed using a range of methods, producing numerous result files that must remain connected with the original samples. Furthermore, the same analysis tool may be used on the same sample but with different parameters (e.g. setting the number of mismatches allowed in an alignment). File naming

conventions can help researchers track the sample but can become unreadable if too many parameters are required to distinguish the files. Most bioinformatic tools add metadata, including the analysis command, to the output file's header or to a separate metafile to help address this. However, metafiles may be named ambiguously resulting in further confusion. One solution to tracking metadata is to use a database to store the association between the samples and analysis files.

## **2.2 Genomic Analysis**

### **2.2.1 Reference Genomes**

Reference genomes are continually being updated and improved upon. For example, in 2014 alone, *TriTrypDB* released three different updates to their reference genomes. It is important to recognize that genomic comparison results are dependent on the reference genome and version used. Small changes to the genomes can result in major differences in the final analyses.

The constant changes to the references results in short-lived genomic analyses. With improved annotations and updated genomes, analyses should be run with the most recent genome. However, researchers may not want to re-run or replace their old analyses - it's possible that older publications may be reliant on those data sets or they may not expect the update to cause a significant enough change to the analysis to invalidate upcoming work. It is important to track which version of the reference genome is used for each analysis to ensure reproducibility. Additionally, some reference genomes are of such poor quality, that a researcher may use a slightly less related species that has a more accurate reference for their analysis.

### 2.2.2 Comparing Results

The reference genomes are not the only source of differences between datasets. Even if the same sample and reference genome are used for genomic analysis, the software may differ. All such differences must be considered when comparing across datasets. For example, SNPs that are identified using one analysis tool, e.g. *HaplotypeCaller*, may not be the same as those used through another analysis tool, e.g. SAMtools' *mpileup*. Researchers must consider how these differences are contributing to their overall analysis.

The University of Nebraska compared 11 different variant callers specifically focusing on *HaplotypeCaller* and SAMtools' *mpileup* [33]. Each of these tools use a Bayesian model to identify variants but do not end up identifying the same number of variants. *HaplotypeCaller* identified 21,631 true positive variants and 273 false positive variants while SAMtools' *mpileup* identified 21,930 true positive variants and 1,030 false positive variants. The discrepancy can be attributed to the different methods and filtering each tool uses to identify the SNPs. What is important here is to consider how a researcher would consider the discrepancies biologically. Analyses must acknowledge which of these SNPs do not overlap and may be untrue SNPs and take additional care in identifying their potential effect.

## 2.3 Current Genomic Databases and Applications

Genome databases store and organize full genomic sequences, providing public access to view and analyze the information. Two of the largest genomic databases are the *Genome* [34] database from the National Center for Biotechnology Information (NCBI) and *Ensembl* [30] from the European Molecular Biology Laboratory's European Bioinformatics Institute (EMBL-EBI). Smaller databases that are unique to a subset of organisms or a specific organism, such as *TriTrypDB* [35] are also available for researchers. These databases are important for research as

they act like an encyclopedia containing all past sequences and variations of these sequences allowing researchers access to work already completed.

NCBI contains over 30 databases storing a range of biological resources [36]. In particular, the NCBI databases, *Genome*, *dbSNP*, and *dbVar*, store information specific to NGS analysis. The *Genome* database contains the full genomes of over 10,000 species, including 20 *Leishmania* genomes. *dbSNP* contains sequence variations discovered through user-submitted research. The database contains SNPs from over 300 organisms. *dbVar* contains genomic structural variations including inversions, copy number variations, and translocations. As of March 2015, *dbVar* contains information on 11 organisms. The advantage of using NCBI databases is that they are interconnected. For example, one could identify a SNP of interest and then follow one link to run a BLAST against this gene or another link to explore the dataset that the SNP was found within. Variations are also automatically annotated from the *RefSeq*, *GenBank*, and other NCBI databases [37].

While NCBI is a widely used resource for biologists and bioinformaticians, it is limited in scope and capability. While the *Genome* database contains over 10,000 species, the limited number of organisms found in *dbSNP* and *dbVar* restrict which researchers can benefit from these databases. As the database grows this will become less of a concern, but for now, the absence of certain variation data makes it of limited use to researchers focused on those areas. For those that are able to find relevant information the databases are great for exploring published data, but they are not meant for a full-scale genomic analysis. Data of interest can be downloaded for further exploration, but must be done so offline.

The other variety of tools available is web applications that allow users to upload their own data and run queries on these datasets. An example of this is *Galaxy* [38]– [40], a web-

based application that allows users to analyze datasets from *UCSC Genome* or to upload their own datasets. This application has the capability to answer complex biological questions, such as, which SNPs are found within one genomic library or sample but not in another. *Galaxy* allows users to create and save a workflow to answer these common questions. Users submit jobs to the server, which are then added to an online queue. The time to complete a job varies by complexity and the number of jobs already present in the queue. Local implementations of *Galaxy* can be installed to prevent long queue times. *Galaxy* also allows the user to save their own workflows that they can then share with collaborators. This provides an easy means for reproducing the analysis.

While *Galaxy* is widely used for computational biomedical research, it is not intended to store data long-term. As of 2015, *Galaxy* has set a space limit to of 250 gigabytes for registered users and 5 gigabytes for unregistered users. This means that a registered user could only store between 500 to 15,000 analysis files. Additionally, the data is stored in files ordered by the time uploaded, making it difficult for the user to track individual files. Relying on *Galaxy* to store the raw data would be impractical. Secondly, uploading the data to *Galaxy* can also be time consuming and/or impossible. Files over 50 gigabytes are unable to be uploaded, potentially limiting the files that can be analyzed. *Galaxy* cautions users to not upload local files that are larger than 2 gigabytes through the interface but rather to use the command line File Transfer Protocol (FTP) [41]. This requires the user to register with *Galaxy* and have knowledge of FTP clients. Another solution would be to use a local *Galaxy* server to allow for quicker access to large datasets. Unfortunately, using a local instance does not solve the problem of how best to store and/or organize multiple files.

Other applications are available to store and analyze data for a cost. Google has developed *Google Genomics* that charges for data storage and queries. For each gigabyte stored on the Google Cloud the user must pay \$0.022 and an additional \$1 for every million API calls. While this is cheap, costing roughly \$25 per year to store a human genome, labs, which are computationally heavy, may not want to pay for each API call. Additionally, some labs may not be able to use the cloud due to security concerns. The large drawback to *Google Genomics* is that it is currently aimed at programmers. Researchers who do not have extensive programming skills may find the tool difficult to use.

Current databases and applications provide a variety of tools for researchers. The databases allow users to view vast quantities of published information about genes, proteins, and more, while applications allow researchers to analyze their own unpublished data without the overhead of storage. However, most are standalone tools that do not allow the user to both store and analyze their own data, and they are often designed for individuals with programming knowledge.

## Section 3: Architecture of NGSdb

### 3.1 Database and Schema

NGSdb stores the sample and analysis information in a relational database using *PostgreSQL*, an open source relational database system [42]. The database is divided into five different cores: **sample**, **library**, **genome**, **analysis**, and **results**. Additional analysis modules, including **RNA-Seq/ribosome profiling**, **spliced leader**, **SNP**, and **CNV/somy** are attached to the results table. Dividing the database into these cores allows for easier tracking of the meta-information of libraries and samples while also providing the capability of adding new analysis modules in the future without needing to make changes to the core tables. The full schema is depicted in the supplementary files.

The **sample core** contains information regarding the biological sample prior to library creation. This includes meta-information ranging from the sample organism to information about the sample's storage. Tracking this information is important for understanding the full biological scope of experiments as these specifics can change how one interprets analysis results. Researchers are able to enter this information online before they begin preparing the library.

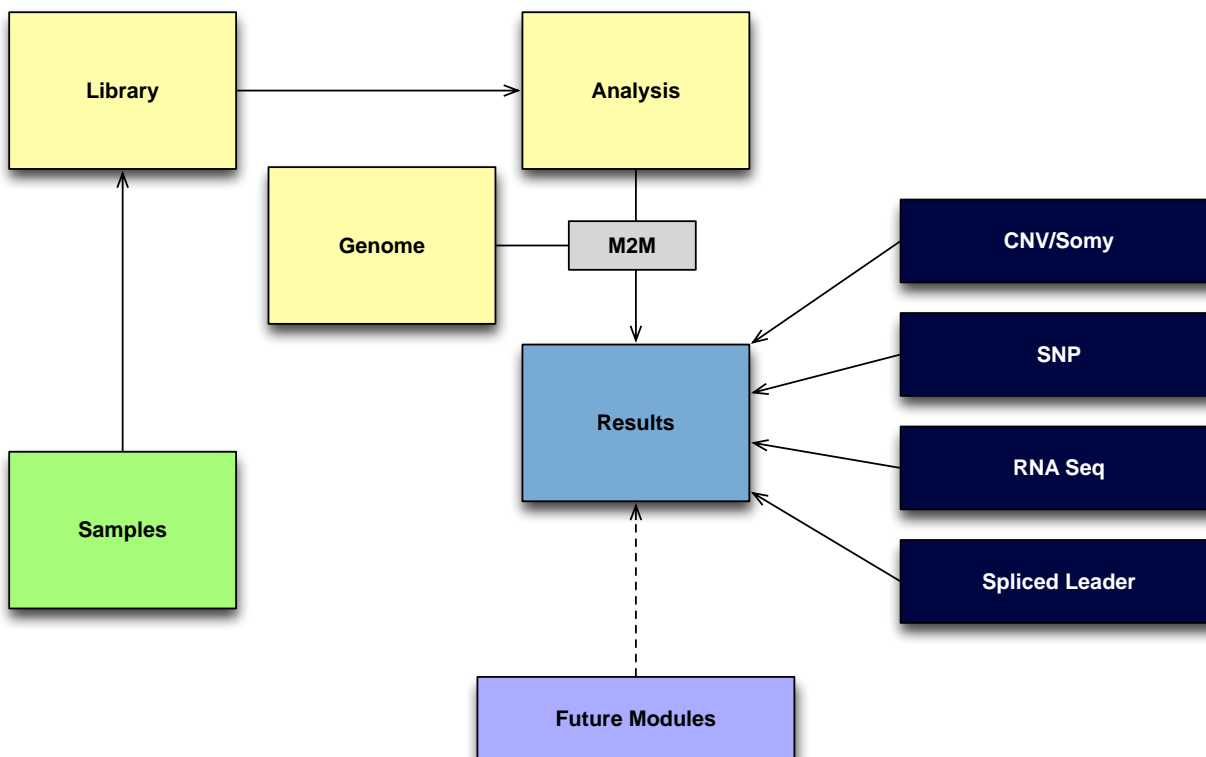
The **library core** contains the meta-information after the sample has been prepared into a library. This includes but is not limited to information about how the library was prepared, the date it was created, and the library's author. Each library entry references its original sample. Libraries may also be grouped into experiments for easier analysis. For example, the libraries that are used in each test case can be grouped together into two separate experiments. When the libraries are run through various analytical options, the libraries can be selected as an experiment. This relieves researchers from having to remember the library and/or sample names between each experiment.



The **analysis core** stores meta-information about bioinformatic analyses that have been completed on a library. This information includes the type of analysis, any software tools used, and more. The **genome core** contains information about reference genomes. It is connected to a **feature table**, which stores all of the annotated information about the genome, i.e. genes, and chromosomes. We use the **genome core** to track the reference genome in each analysis.

The **spliced leader (SL) module** contains information about the chromosomal locations where mRNA is trans-spliced. Three different tables, **resultraw**, **resultsite**, and **resultsgene**, store the information. **Resultraw** stores raw read counts for every position in the genome while **resultsite** associates these spliced leader sites with the nearest downstream genes. **Resultsgene** sums all of the associated reads to create a total read count for each associated gene. Similarly, the RNA-Seq and Ribosome Profiling module contains the table **resultsriboprof** that holds the total read count for genes that were actively being transcribed during the profiling.

The **results core** connects four modules, **spliced leader**, **RNA-Seq/ribosome profiling**, **CNV/somy**, and **SNP**, to the analytical results of a specific library. Each module contains tables summarizing the results, which are referenced by the **library id**, **genome id**, **analysis id**, and a unique **result id** as seen in Figure 2. By using a separate **result id** for each analysis, a single library can undergo multiple analyses and each result can be considered separately. For example, if a new version of a reference genome is released, we do not need to remove our old results but can rather mark them as deprecated. This allows for comparisons across different results and provides the opportunity for the researcher to better control the data they are looking at. The **CNV/somy** and **SNP modules** were used in the case studies, which will be further discussed in **Section 5** and **Section 6**.



**Figure 2 – Overview of NGSdb's schema**

Specific libraries, samples, and analysis are organized into experimental groups. Information about each experiment is stored across five different tables, which contain information ranging from the experimental setup to all libraries associated with the experiment. The experiments are given a group name for researchers to reference. Experiments are organized by the type of libraries, e.g. RNA-Seq or DNA-Seq.

**NGSdb** was designed modularly with the intent of adding modules as new biological data become available. For example, we first developed the SNP module and application views before adding the CNV and somy data. Only once we had created the basic SNP views and became interested in querying the corresponding CNV and somy data did we develop the CNV and somy modules. This demonstrates how **NGSdb** can expand and evolve with new genomic technology or data.

To increase the speed of **NGSdb** we have used common vocabulary (CV) tables to reduce the redundancy of common terminologies. As an example, we consider how one could store the statistics of an individual SNP. VCF files contain a variety of quality scores and statistics including allele frequencies, z-score from the Wilcoxon rank sum test, read depth, maximum likelihood expectation of allele frequency and count, and the phred-quality score. Each statistic could be stored as seen in Table 1. This method has the statistics term and description repeated for each entry. This means that every time the user updates the database, they may be required to update more than one column per entry. A better option can be seen in Table 2 where we split it into two separate tables. This allows each statistics term to be accessed through a foreign key. If a statistics entry were updated, only the foreign key would need to be altered rather than both the statistics term and description.

Statistics ID	SNP ID	Statistics Term	Statistics Description	Value
1	1	FS	Phred-scaled p-value using Fisher's exact test to detect strand bias	2
2	1	DP	Approximate read depth; some reads may have been filtered	149
3	1	MQRankSum	Z-score From Wilcoxon rank sum test of Alt vs. Ref read mapping qualities	-4.814
4	2	FS	Phred-scaled p-value using Fisher's exact test to detect strand bias	27.95 6
5	2	DP	Approximate read depth; some reads may have been filtered	201

**Table 1 - Statistics table without CV table**

Statistics ID	SNP ID	Statistics Term	Value
1	1	1	2
2	1	2	149
3	1	3	-4.814
4	2	1	27.956
5	2	2	201

Statistics Term	Statistics Description
FS	Phred-scaled p-value using Fisher's exact test to detect strand bias
DP	Approximate read depth; some reads may have been filtered
MQRankSum	Z-score From Wilcoxon rank sum test of Alt vs. Ref read mapping qualities
FS	Phred-scaled p-value using Fisher's exact test to detect strand bias
DP	Approximate read depth; some reads may have been filtered

**Table 2 - Statistics Table with CV table**

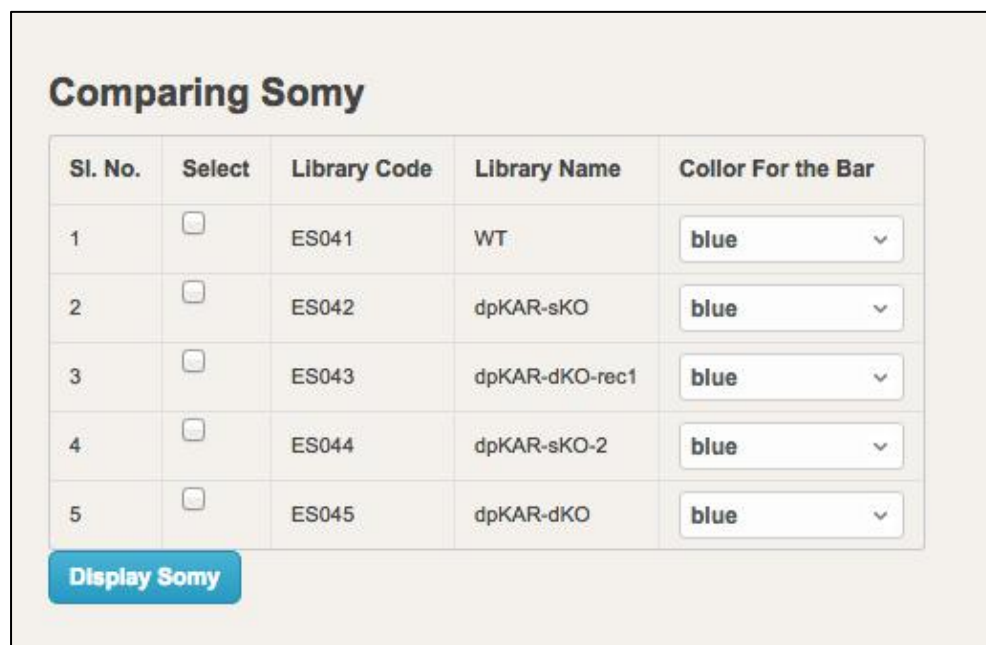
PostgreSQL automatically indexes primary keys to increase the retrieval speed. To further increase the speed of commonly queried attributes, we have indexed additional columns of our database. These columns include all common vocabulary terms, sample names, library codes, library category, spliced leader position, gene names, SNP position, SNP effect type, and SNP effect value. Indexing these values increases the retrieval time but does result in additional overhead when inserting new values or updating existing values.

### 3.2 Web Interface Design

The web interface was developed using *Django*, a Python web framework and *Bootstrap*, a front-end framework [43], [44]. *Django* was developed for database-driven web applications and manages the website's views, HTML templates, and database queries. *Bootstrap* provides

HTML, CSS, and JavaScript for a clean and responsive interface design. *Django* and *Bootstrap* have allowed us to focus on developing our web interface with a user-based approach. All views were designed while simultaneously analyzing genomic data to ensure that the application focused on displaying and answering biologically relevant questions.

We have designed the interface with the researcher in mind and have included functionality that allows them to manipulate queries and outputs as they wish. For example, when comparing some values across libraries, the user is capable of selecting the individual libraries they want to display, as well as the line-color and line-style [Figure 3]. This flexibility gives the researcher the control to easily explore their own questions.



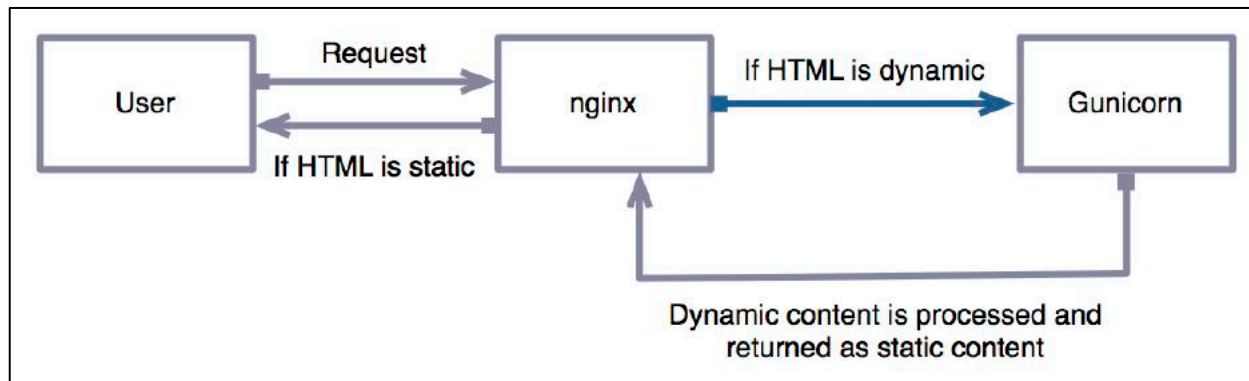
Sl. No.	Select	Library Code	Library Name	Collor For the Bar
1	<input type="checkbox"/>	ES041	WT	blue
2	<input type="checkbox"/>	ES042	dpKAR-sKO	blue
3	<input type="checkbox"/>	ES043	dpKAR-dKO-rec1	blue
4	<input type="checkbox"/>	ES044	dpKAR-sKO-2	blue
5	<input type="checkbox"/>	ES045	dpKAR-dKO	blue

**Display Somy**

**Figure 3 - Somy options for Graphical Display**

### 3.3 Web Application Deployment

The database and Django-application runs on an Ubuntu 14.04.1 LTS virtual machine, 45 MHz with 1024 MB of RAM and 64 GB of disk space. All of the required software to run the application was also downloaded onto this machine. Table 3 provides the full list of required software and Table 4 lists all of the required python packages. We used *Fabric*, a Python library, to deploy our application [45]. The application's dynamic files are hosted by *Gunicorn*, a Python WSGI HTTP web server for UNIX, while the application's static files are hosted by *nginx*, a high-performance HTTP server [Figure] [45].



**Figure - NGSdb Application Deployment**

Software	Version	Download URL
PostgreSQL	9.0.5	<a href="http://www.postgresql.org">www.postgresql.org</a>
Nginx	1.4.6	<a href="http://www.nginx.org">www.nginx.org</a>
Python	2.7.6	<a href="http://www.python.org">www.python.org</a>
bcftools	1.1	<a href="http://www.samtools.github.io/bcftools.html">www.samtools.github.io/bcftools.html</a>
SamTools	1.1	<a href="http://www.samtools.sourceforge.net">www.samtools.sourceforge.net</a>
VcfTools	1.12b	<a href="http://www.VCFtools.sourceforge.net">www.VCFtools.sourceforge.net</a>
Snpeff	4.1a	<a href="http://www.snpeff.sourceforge.net/index.html">www.snpeff.sourceforge.net/index.html</a>

**Table 3 - Software required for NGSdb**

<b>Python Software</b>	<b>Version</b>	<b>Download URL</b>
Gunicorn	19.1.1	<a href="http://www.gunicorn.org">http://www.gunicorn.org</a>
Django	1.5.4	<a href="http://www.djangoproject.com">http://www.djangoproject.com</a>
Fabric	1.8.0	<a href="http://www.fabfile.org">http://www.fabfile.org</a>
South	0.8.2	<a href="https://pypi.python.org/pypi/South">https://pypi.python.org/pypi/South</a>
django-auth-ldap	1.1.5	<a href="https://pypi.python.org/pypi/django-auth-ldap">https://pypi.python.org/pypi/django-auth-ldap</a>
django-debug-toolbar	0.9.4	<a href="https://django-debug-toolbar.readthedocs.org">https://django-debug-toolbar.readthedocs.org</a>
django-filter	0.7	<a href="https://pypi.python.org/pypi/django-filter">https://pypi.python.org/pypi/django-filter</a>
django-grappelli	2.4.9	<a href="https://django-grappelli.readthedocs.org">https://django-grappelli.readthedocs.org</a>
django-tables2	0.14.0	<a href="https://django-tables2.readthedocs.org">https://django-tables2.readthedocs.org</a>
django-tables2-reports	0.0.9	<a href="https://pypi.python.org/pypi/django-tables2-reports">https://pypi.python.org/pypi/django-tables2-reports</a>
ecdsa	0.10	<a href="https://pypi.python.org/pypi/ecdsa">https://pypi.python.org/pypi/ecdsa</a>
paramiko	1.12.0	<a href="http://www.paramiko.org">http://www.paramiko.org</a>
psycopg2	2.5.1	<a href="http://www.initd.org">http://www.initd.org</a>
pycrypto	2.6.1	<a href="https://pypi.python.org/pypi/pycrypto">https://pypi.python.org/pypi/pycrypto</a>
python-ldap	2.4.13	<a href="http://www.python-ldap.org">http://www.python-ldap.org</a>
six	1.4.1	<a href="https://pypi.python.org/pypi/six">https://pypi.python.org/pypi/six</a>
wsgiref	0.1.2	<a href="https://pypi.python.org/pypi/wsgiref?">https://pypi.python.org/pypi/wsgiref?</a>
xlwt	0.7.5	<a href="https://pypi.python.org/pypi/xlwt">https://pypi.python.org/pypi/xlwt</a>
django-mathfilters	0.3.0	<a href="https://pypi.python.org/pypi/django-mathfilters">https://pypi.python.org/pypi/django-mathfilters</a>
GChartWrapper	0.8	<a href="https://pypi.python.org/pypi/GChartWrapper">https://pypi.python.org/pypi/GChartWrapper</a>
Pillow	2.5.3	<a href="https://pypi.python.org/pypi/Pillow">https://pypi.python.org/pypi/Pillow</a>
django-boolean-sum	0.1	<a href="https://pypi.python.org/pypi/django-boolean-sum">https://pypi.python.org/pypi/django-boolean-sum</a>
PyVCF	0.6.7	<a href="https://pyVCF.readthedocs.org">https://pyVCF.readthedocs.org</a>
numpy	1.9.0	<a href="http://www.numpy.org">http://www.numpy.org</a>
PyYAML	3.11	<a href="http://www.pyyaml.org">http://www.pyyaml.org</a>

**Table 4 - Python Software required for NGSdb**

## Section 4: Development of Specific Modules

The specific modules that I developed contain three different analyses type: somy, CNV, and SNP. Each of these modules is connected to the previously mentioned meta-information through a unique **result id** and **library id**. It is important to note that a library can have more than one **result id**, regardless of analysis type. For example, a user may be interested in storing SNPs found through the *HaplotypeCaller* and through *mpileup*. By allowing a one-to-many relationship, both analyses can be stored and queried. Additionally, each portion is connected to a table that stores information about the chromosome that the somy, CNV, or SNP are found within. Each chromosome is specific to a genome and genome version.

### 4.1 Development of Somy and CNV Module

To prevent unnecessary complexity, we regarded somy and CNV values to be a continuum, with somy being stored as a large CNV read. This allows us to store their data into the same tables and eliminates the size of our schema. This also increases **NGSdb's** flexibility by not limiting the window size of read coverage and providing multiple levels of granularity.

Somys and CNVs are stored across three different tables. The CNV table stores the position range and window size. This can correspond to a full chromosome (somy) or a segment of the chromosome (CNV). The CNV value corresponds to the calculated CNV or somy ratio. The second table is a common vocabulary (CV) table that indicates which type of analysis is stored: somy or CNV.



## 4.2 Development of SNP Module

In order to maintain flexibility in our querying capability of SNPs, we chose to individually store each SNP in the database and store the location of the VCF files they were uploaded from. This allows us the ability to use SQL queries and to use standard analysis software when deemed appropriate. SNPs are stored across 12 different tables as seen in Figure 4 and contain the same information present in the VCF file. The **effect table** contains information about what affects the SNP had on the genome that were found from *SnEff*. The filter table stores information about the quality of the SNP and whether it passes any specified filter. Additionally, the statistics table stores all statistics calculated from *HaplotypeCaller*. Four of the tables are CV tables, which allow for faster querying. Additionally, attributes that are commonly queried were indexed to increase the query speed. These attributes include the SNP position, the effect string storing a gene name that was impacted by the SNP, and the effect class storing the SNP's impact type.

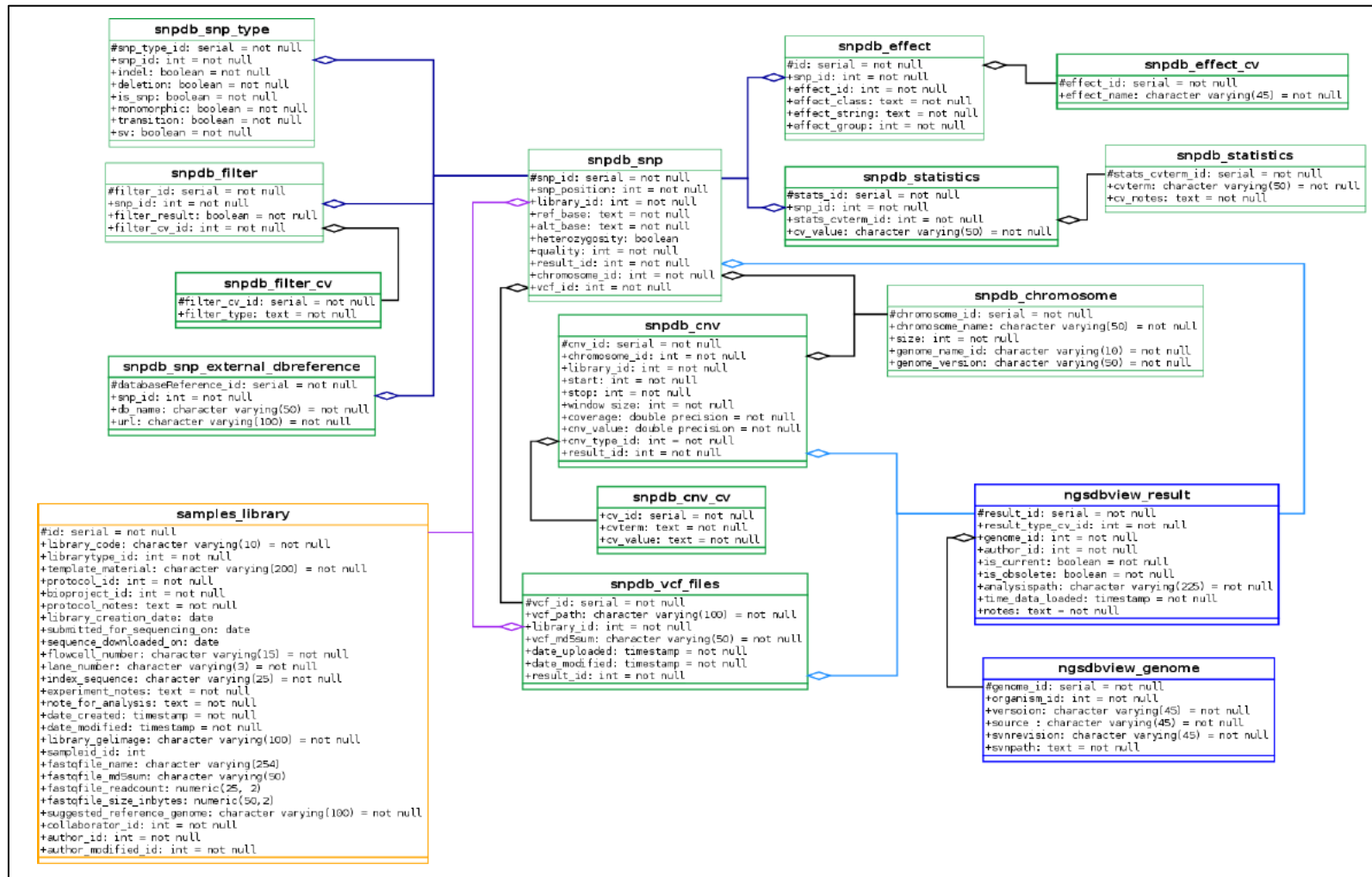


Figure 4 - Comparative Genomic Module

## Section 5: Sri Lankan *Leishmania* Tropism

*Leishmania donovani* typically causes visceral leishmaniasis. A Sri Lankan *L. donovani* strain, MON-37, is responsible for infecting thousands with cutaneous leishmaniasis [46]– [48]. Visceral leishmaniasis is very uncommon in Sri Lanka but of the four cases identified from 2007 to 2014, all have been caused by *L. donovani* MON-37 [49]. Our previously published research questions whether the same or different sub-strains of *L. donovani* MON-37 are responsible for the two types of infection. Using the **Comparative Genomic** module, we have reproduced the genomic analysis we previously published in *Genetic Analysis of Leishmania donovani Tropism Using a Naturally Attenuate Cutaneous Strain* [50].

### 5.1 Methods

A full description of the materials and methods can be found in our previously published work, *Genetic Analysis of Leishmania donovani Tropism Using a Naturally Attenuate Cutaneous Strain* [50].

#### Sample Collection

We isolated *L. donovani* parasites from two Sri Lankan patients. A visceral leishmaniasis sample was taken from the bone marrow of a 53 year old male and the cutaneous leishmaniasis samples was taken from the skin lesion on the nose from a 28 year old male. The samples were immediately placed into *Leishmania* promastigote culture medium.

#### Library Preparation

Genomic DNA was collected from the samples and sheared into 100 to 1,200 base pair long fragments. Paired end segments were created using Illumina's *Genomic DNA Sample Preparation Kit*.

## Sequencing

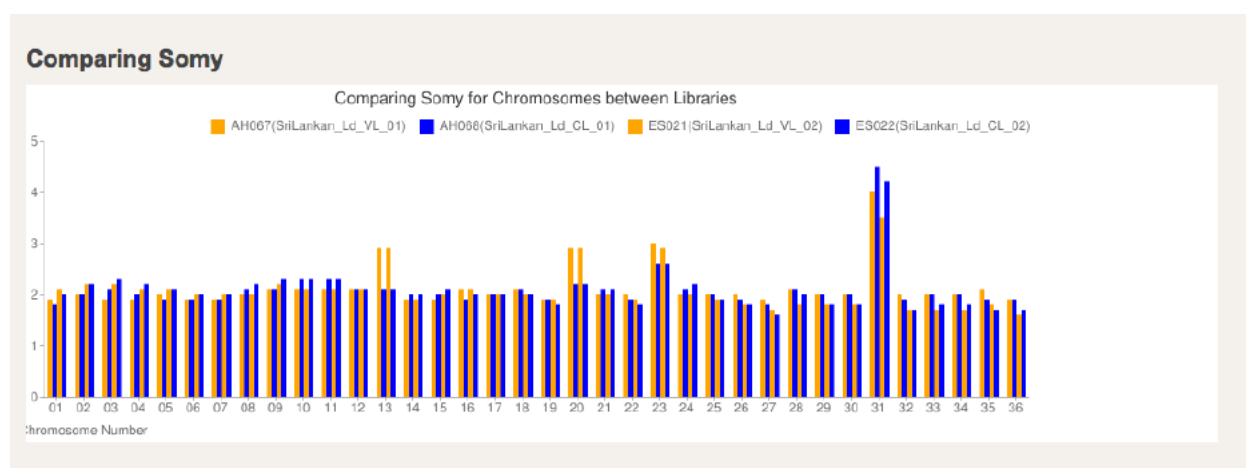
Genomic libraries were sequenced using Illumina's *Genome Analyzer IIx* at the High Throughput Genomics Unit at the University of Washington generating 100-nucleotide long paired-end reads. Reads that had a quality below 30 were removed. The reads containing *Illumina* adaptor sequences were trimmed off using *cutadapt (v1.2)* software [51]. The *L. donovani* (BPK282/Ocl4 cloned from Nepal) reference genomes were collected from *Welcome Trust Sanger Institute's* ftp site (<ftp://ftp.sanger.ac.uk/pub/pathogens/Leishmania/donovani/>). The SAM files were then converted into sorted BAM files using SAM tools (v0.1.18) [27].

## 5.2 Somy Comparison

Somy values ranged from 1.58 to 4.48 across all libraries. Table 8 summarizes the range of each individual library. We leveraged a single **NGSdb** view to identify the differences in somy values. This view allows the user to select which libraries they would like to view and their associated colors [Figure 3]. We chose to display the two VL libraries in yellow and the two CL libraries in blue. The resulting graph [Table 5] shows that of the 36 chromosomes, 32 are disomic across all libraries. Chromosome 23 is trisomic for all libraries while chromosome 13 and 20 are trisomic for both visceral libraries (**AH067** and **ES021**) and disomic for cutaneous libraries (**AH068** and **ES022**). Chromosome 31 is tetrasomic for all libraries.

Library	Minimum Somy (chromosome)	Maximum Somy (chromosome)	Average Somy
AH067	1.86 (Ld01)	4.00 (Ld31)	2.13
AH068	1.79 (Ld01)	4.48 (Ld31)	2.10
ES021	1.58 (Ld36)	3.47 (Ld31)	2.06
ES022	1.64 (Ld27)	4.22 (Ld31)	2.08

**Table 5 – Overview of Somy Values**



**Figure 5 – Somy values for VL and CL libraries**

### 5.3 Copy Number Variation

NGSdb contains a similar view for CNV as described for somy. The user is able to choose which library and line color they would like to display. The CNV values across the 36 chromosomes share a similar pattern for all libraries with a few exceptions [Supplementary Figures]. An additional view allows users to compare groups of libraries and identify regions that

differ by a user-selected amount [Figure 6]. This view found that chromosomes 1, 2, 3, 7, 25, 30, and 32 did not contain any differences between the visceral and cutaneous libraries greater than 0.75. The remaining libraries typically had one continuous region, varying in length that differed between the two groups. Figure 7 shows three base pair ranges, 34000 to 39000, 126000 to 130000 and 334000 to 336000, found on chromosome 20 that differ between the VL and CL libraries by more than 0.75. We previously identified ten chromosomal locations that differed in gene copy number variation between the visceral and cutaneous libraries [50]. The **Comparative Genomics** module is currently unable to look at CNV values by genes, but instead identifies 1,000 base pair regions that differ by more than 0.75. Of the ten regions previously identified, we captured large portions of each of them. Small regions of the genes identified were not determined to be different using the **Somy/CNV module** because that specific 1,000-basepair segment was less than 0.75. **NGSdb** was able to identify additional 1,000-basepair segments that were not earlier mapped to genes.

### Comparing CNVs

#### Group 1

Sl. No.	Include	Library Code	Library Name
1	<input checked="" type="checkbox"/>	AH067	SriLankan_Ld_VL_01
2	<input type="checkbox"/>	AH068	SriLankan_Ld_CL_01
3	<input checked="" type="checkbox"/>	ES021	SriLankan_Ld_VL_02
4	<input type="checkbox"/>	ES022	SriLankan_Ld_CL_02

Color :

Line Style :

#### Group 2

Sl. No.	Include	Library Code	Library Name
1	<input type="checkbox"/>	AH067	SriLankan_Ld_VL_01
2	<input checked="" type="checkbox"/>	AH068	SriLankan_Ld_CL_01
3	<input type="checkbox"/>	ES021	SriLankan_Ld_VL_02
4	<input checked="" type="checkbox"/>	ES022	SriLankan_Ld_CL_02

Color :

Line Style :

Data Summarizing Mode :

Minimum CNV value difference between groups:

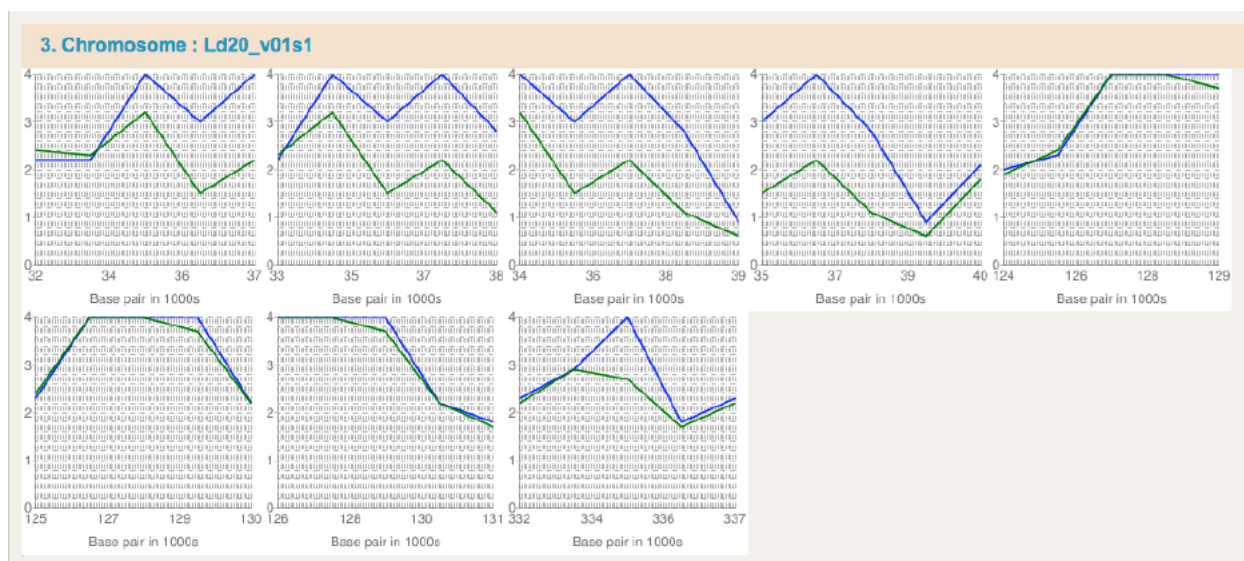
Set it to Zero to browse entire chromosome manually

Number of datapoints to plot in a horizontal picture:

More the points (value) messier picture will be. 100 is optimal. Very few points, many pictures.

**Display CNV**

**Figure 6 - Options to Compare Groups of CNV**



**Figure 7 – CNV Differences between VL and CL Libraries**

## 5.4 Single Nucleotide Polymorphism

A total of 158,540 SNPs were identified across all four libraries. The **NGSdb** view depicted in Figure 8 provides a summary view of the SNP types for each library. Additional SNP information about all libraries that have an associated SNP result can be viewed on the SNP dashboard. Of the 158,540 SNPs found across the VL and CL libraries, 697 caused changes to the coding sequence length, 24,086 were non-synonymous changes, 16,847 were synonymous changes, and 153,906 occurred in intergenic regions [Table 6].

Library Code	Reference Genome	SNP Total	Heterozygosity Count	Homozygous Count	Indel Count	Transition Count	Transversion Count	Snp Density
AH067	LdoSPK	32048	2371	29577	6412	17750	14298	0.0005
AH068	LdoSPK	32204	2990	29214	6228	17792	14412	0.0005
ES021	LdoSPK	46609	7087	39542	14952	21219	25360	0.0007
ES022	LdoSPK	47679	7987	39712	15334	21631	26048	0.0007

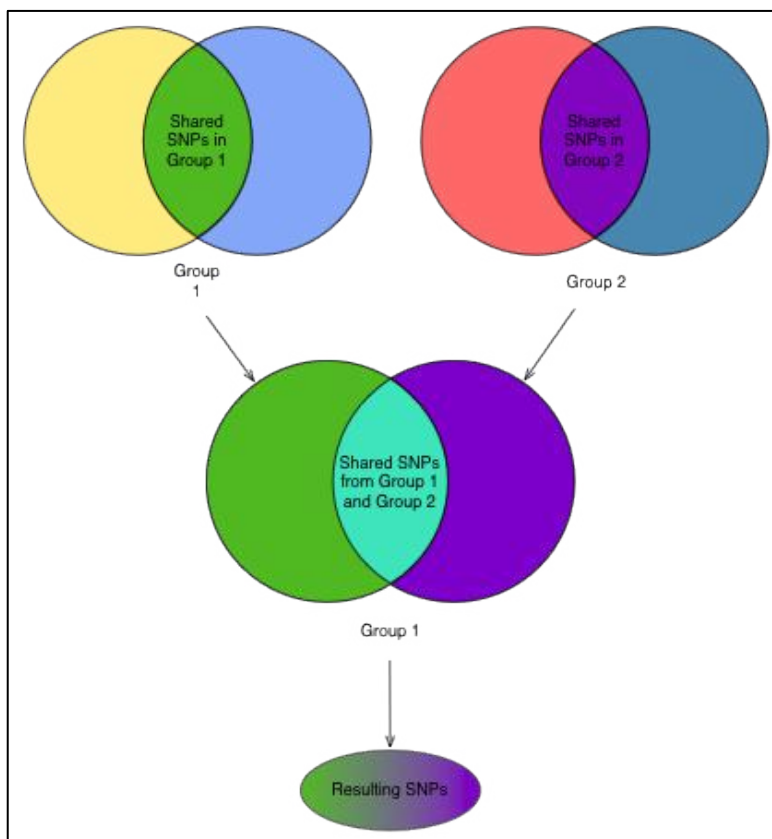
**Figure 8 – Overview of Library SNPs**



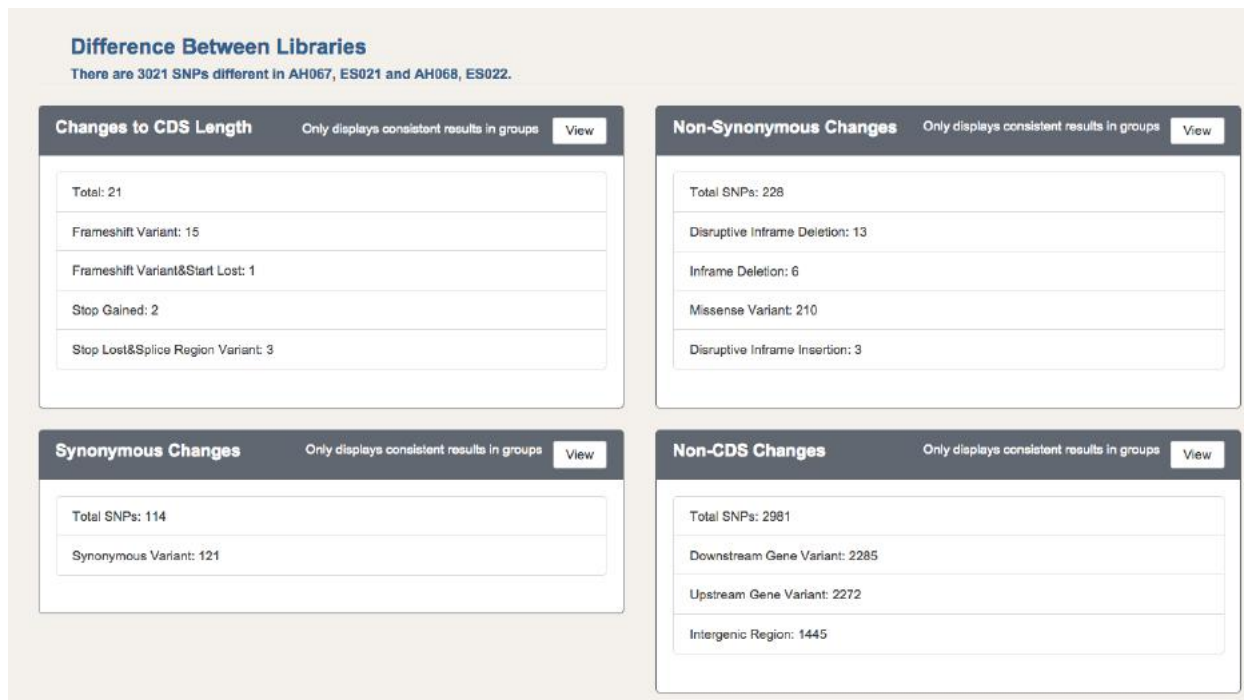
Sample ID	Changes to CDS lengths	Non-Synonymous Changes	Synonymous Changes	Intergenic Regions
AH067	77	5,720	4,068	31,024
AH068	81	5,827	4,143	31,182
ES021	269	6,258	4,312	45,298
ES022	270	6,281	4,324	46,402
<b>Total</b>	697	24,086	16,847	153,906

**Table 6 – Summary of SNPs by Library**

To narrow down which of these SNPs may contribute to the difference in disease pathology, we compared the visceral-causing libraries (**AH067** and **ES021**) with the cutaneous-causing libraries (**AH068** and **ES022**). This **NGSdb** query utilizes *bcf-isec* from *BCFtools* to determine the complement between the two groups of libraries [55]. The application first identifies the intersections or common SNPs between libraries of the first group and libraries of the second group. The SNPs that are shared within each individual group are then queried to identify their complements [Figure 9]. The resulting view displays a count of the complementing SNP's effect divided by the putative impact types [Figure 10]. From here, the user can choose a specific putative impact to focus on and navigate to a more specified view. This view contains the chromosome, position, quality, gene impacted, gene product name, gene length, distance of SNP from start codon, reference and alternate bases for all SNPs and libraries. SNPs are flagged if the quality score is below 50 and if the CNV of the region the SNP falls within is outside the 1.72-2.25 range [Figure 11].



**Figure 9 - Logic of comparing groups of libraries**



**Figure 10 – SNP Overview by Impact**

We have identified 21 SNPs that result in a change of the CDS region. A full list of these SNPs is provided in the supplementary documents. Figure 11 provides a sample of the list as seen through the **Comparative Genome** interface. Of these 21 SNPs, we removed 3 SNPs that had a quality less than 30. The five previously identified SNPs were all present in the remaining group of 19 SNPs [Figure 11]. Two of these SNPs are unique to the cutaneous libraries while three are unique to the visceral libraries.

The SNPs unique to the cutaneous-causing libraries were found on chromosome 30 and chromosome 31. The heterozygous SNP on Chromosome 30 at position 579030 causes a premature stop codon, truncating 52% of the protein. The homozygous SNP on chromosome 31 at position 602800 causes a stop loss from a non-synonymous change.

The SNPs unique to the visceral-causing libraries were found on chromosome 31 and chromosome 32. The heterozygous SNP on Chromosome 31 at position 1000447 causes a frameshift due to a single nucleotide insertion, affecting 72% of the gene. The homozygous SNP on chromosome 32 at position 9103 causes a frameshift due to a single nucleotide insertion, affecting 27% of the gene. The final heterozygous SNP found on chromosome 22 at position 174033 results in a stop gained from a non-synonymous change.

Of the 92 non-synonymous SNPs previously identified to be unique to the cutaneous-causing libraries, we correctly identified 91 of these SNPs and found an additional 20 SNPs. Our analysis found that the SNP located at position 130897 on chromosome 32 was only present in **ES022**, removing it from our results. We previously narrowed down SNPs further by determining if the SNP were found in a functionally important region of the gene. The **Comparative Genomics**' module currently does not have this capability and would require manual review to complete this process. We plan on integrating this information in the future.

Ld30_v01s1	579030	306.77	<a href="#">LdBPK_301640.1</a>	Hypothetical Protein, Conserved	696	333	C	Ref 1	Ref 1	Ref, T 1	Ref, T 1
Ld31_v01s1	602800	3120.77	<a href="#">LdBPK_311390.1</a>	Hypothetical Protein, Conserved	1406	0	C	Ref	Ref	A	A
Ld31_v01s1	1000447	509.73	<a href="#">LdBPK_312060.1</a>	Succinyl-Diaminopimelate Desuccinylase-Like Protein	471	129	A	Ref, AG 1	Ref, AG 1	Ref 1	Ref 1
Ld32_v01s1	9103	754.75	<a href="#">LdBPK_320030.1</a>	Hypothetical Protein, Unknown Function	367	266	C	CG 1	CC 1	Ref 1	Ref 1
Ld32_v01s1	174033	61.77	<a href="#">LdBPK_320480.1</a>	Hypothetical Protein, Conserved	939	239	C	Ref, T 1	Ref, T 1	Ref 1	Ref 1

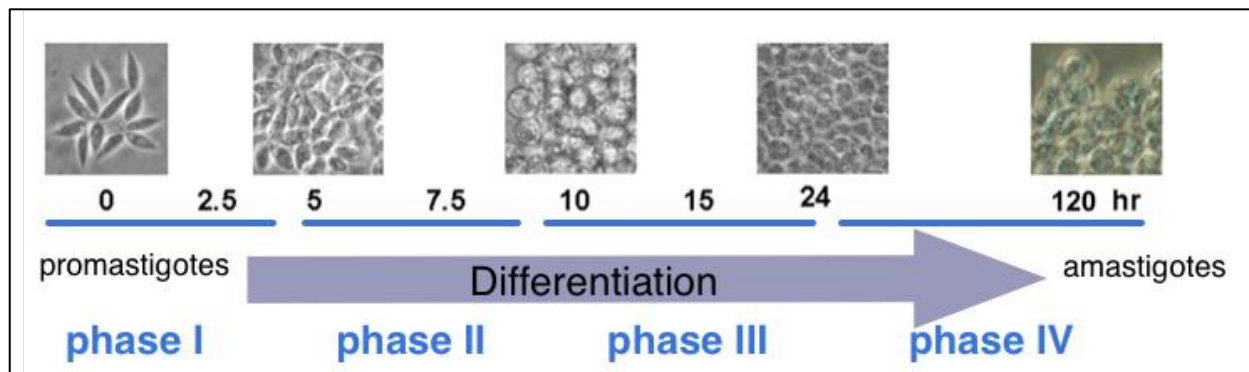
**Figure 11 – SNPs unique to VL or CL Libraries**

## Section 6: *Leishmania* Comparison of *dpKAR1* Knockout

To demonstrate the capabilities of the **Comparative Genome** module, we will examine five *Leishmania*. Specifically, we will focus on demonstrating how the **Comparative Genome** module can be used to identify genomic differences between the double knockout library and the double knockout recoveries using *NGSdb* as our analysis platform.

Differentiation of *Leishmania* from promastigotes to amastigotes occurs in four phases, which are represented in Figure 12. During the first five hours inside the human macrophage (Phase I), the promastigote is exposed to a change in pH and temperature, which signal the pathogen to begin differentiation. The second phase occurs five to ten hours after infection. Promastigotes stop moving and begin to aggregate inside the macrophage. From 10-24 hours, the cells undergo morphological and biochemical modifications including changes in the level of phosphorylation [56]– [58]. The pathogen lose their flagella and the cells become rounded, the morphological traits of amastigotes. Finally, over the next four days, the cells mature to amastigotes [59].

An increase in phosphorylation during Phase III of differentiation supports the idea that phosphorylation plays a role in differentiation and is probably regulated by protein kinases and phosphatases [56]. Quantitative proteomics (iTRAQ) has identified multiple protein kinases that exhibit differences in phosphorylation among the phases of differentiation, including *LinJ.05.0390*, *LinJ.25.2450*, and *LinJ.35.1070*. In addition, two regulatory subunits of protein kinase A (*PKAR*) were identified (*LinJ.13.0160* and *LinJ.34.2680*). The study detected three different trends of phosphorylation for *LinJ.34.2680*, referred to as *PKAR'* (*PKAR prime*) in the rest of this study. This suggests that *PKAR'* plays a role in differentiation.



**Figure 12 - Promastigote Differentiation**

## 6.1 Methods

### Sample Collection

Five *Leishmania donovani* samples were collected from two different experiments. The first experiment collected the samples **ES041**, **ES042**, and **ES043**. Sample **ES041**, was not treated and serves as our wild type. Sample **ES042** and sample **ES043** contain a single knockout and double knockout recovery of *PKAR'*, respectively. Samples **ES044** and **ES045** were collected seven months later and contain a single knockout and double knockout, respectively. A summary of the libraries is displayed in Table 7.

Sample ID	Phenotype	Date Library Creation	Sample Batch	Life Cycle Stage	Cell Morphology
ES041	Wild type	2013-05-29	1	Mixed	Looks normal
ES042	<i>PKAR'</i> Single Knockout	2013-05-29	1	Mixed	Chubby with short flagella
ES043	<i>PKAR'</i> Double Knockout Recovery	2014-01-15	1	Mixed	Larger than WT
ES044	<i>PKAR'</i> Single Knockout	2014-01-15	2	Mixed	Larger than WT
ES045	<i>PKAR'</i> Double Knockout	2014-01-15	2	Mixed	Similar to WT

**Table 7 - List of *Leishmania donovani* samples**

## Library Preparation

Genomic DNA from each sample was fragmented into 100 to 200 base pair segments. Paired-end libraries were constructed using Illumina's *Genomic DNA Sample Preparation Kit*.

## Sequencing

DNA libraries were sequenced using the Illumina's *Genome Analyzer IIx* at *Covance Inc., Seattle* generating 100-nucleotide long paired-end reads. *FastQC* was used to verify the quality and GC content of each library. Reads that had a quality below 30 were removed. The reads containing *Illumina* adaptor sequences were trimmed off using *cutadapt (v1.2)* software [51]. The *L. donovani* (BPK282/Ocl4 cloned from Nepal) reference genomes were collected from *TriTrypDB* 9.0 ftp site [52]. The paired-end reads were locally aligned using *bowtie2 (v2.2.4)* [53]. Alignments with inserts larger than 1,000 bases were considered a discordance alignment.

## Copy Number Variance and Somy

Copy number variation and somy values for each chromosome were calculated independently using a Perl script from "Genetic Analysis of *Leishmania donovani* Tropism Using a Naturally Attenuated Cutaneous Strain" [50]. CNVs were calculated by splitting each chromosome into 1,000-base non-overlapping tiles. The median read coverage of each segment was divided by the median read coverage of the entire chromosome [Equation 1].

$$CNV = \frac{\text{median}(\text{segment read coverage})}{\text{median}(\text{chromosome read coverage})}$$

**Equation 1 - Equation to find CNV values**



Somy values were determined in a similar process. The median read coverage of each chromosome was determined. Next, the median of all chromosomal medians was determined and then divided by two to represent the median read coverage for a haploid allele of a chromosome. The median coverage of the chromosome was then divided by the median haploid chromosome read coverage [Equation 2].

$$somy = \frac{\text{median}(\text{chromosome read coverage})}{\text{median}(\text{genome read coverage})/2}$$

**Equation 2 - Equation to find Somy values**

### Single Nucleotide Polymorphisms

To minimize the number of false positive SNPs, we ran GATK's *IndelRealigner* to locally realign areas where indels are suspected to be the cause for differences between each alignment and the reference genome. SNPs were then identified using *HaplotypeCaller* from the Broad Institute for each of the five libraries mentioned above. Alignment files (bam format) were compared against **LdonovaniBPK282A1 version 9.0** from *TriTrypDB* [35]. High quality SNPs were identified by limiting the identification to regions with a total coverage greater than 350. SNPs with a phred-scaled confidence threshold between 20 and 50 were flagged as low quality while those below 20 were disregarded. Genotypes were identified by choosing the most likely alternate allele.

The SNPs were then annotated using *SnpEff* and the **LdonovaniBPK282A1 version 9.0** genome file [54]. We created a Python script to automatically upload the resulting Variant Call

Format (VCF) file into the database. This script requires the user to input the reference genome, genome version, library code, and path to the VCF file.

To address how to store samples and their sequencing data in a way that allows biologists to easily and efficiently query the data, we developed **NGSdb**, a web application that sits on top of a database. Below we describe how we organized the data in a modular way in anticipation of new technology and data types. We also discuss how the database and web application were implemented.

## 6.2 Somy Comparison

Somy values ranged from 1.10 to 4.39 across all libraries. Table 8 summarizes the range of each individual library. Of the 36 chromosomes, 13 are disomic across all libraries. Chromosome 8, 12, and 23 are trisomic while chromosome 31 is tetrasomic for all libraries except **ES041**, which presents as trisomic. The remaining chromosomes have small differences across each library, reflecting the variance discovered in previous research [52], [60], [61].

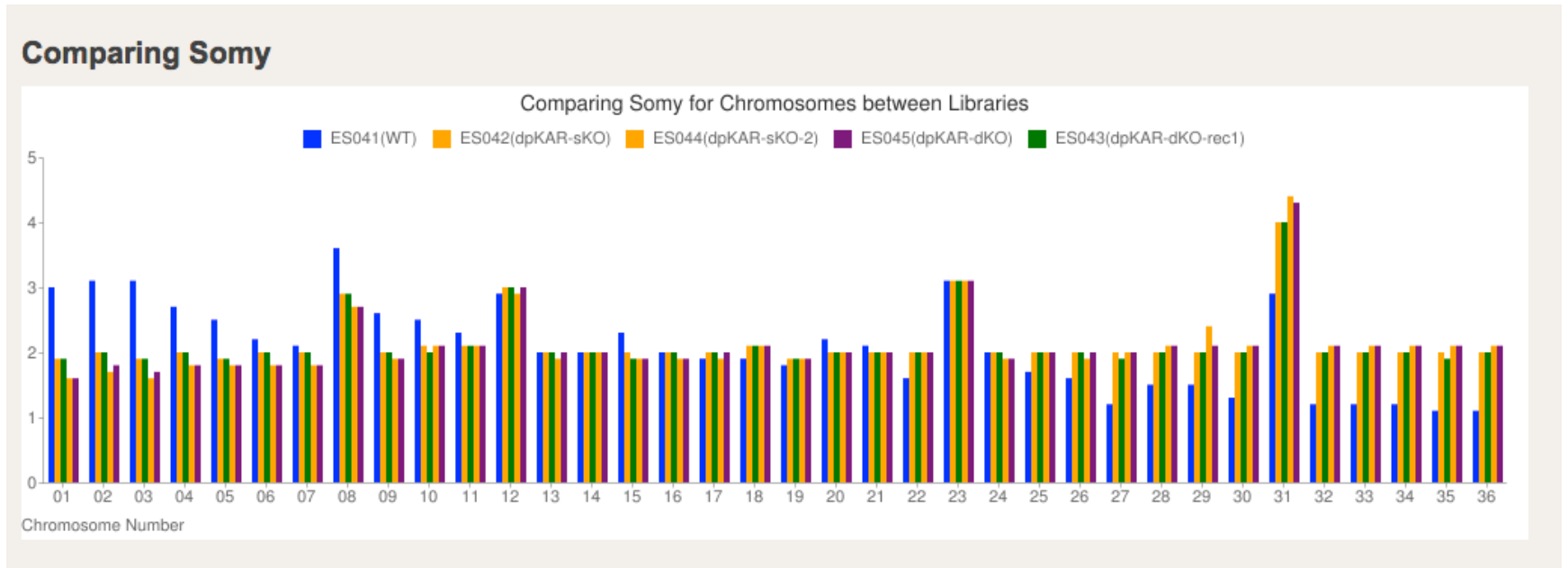
**ES041** has higher somy values for chromosomes 1-6, 9, and 10 and exhibits lower somy values in chromosomes 25-36 [Figure 13]. Unpublished research presented at the 2015 Kinetoplastid Molecular Cell Biology Conference by the *Wellcome Trust Centre for Molecular Parasitology* and the *University of Glasgow* has found evidence that each *Leishmania* chromosome has a single origin of replication. We hypothesize that this may be the cause for the unexpected pattern seen in the wild type. The single replication may mean that larger chromosomes are under replicated while the smaller chromosomes are over replicated.

<b>Library</b>	<b>Minimum Somy (chromosome)</b>	<b>Maximum Somy (chromosome)</b>	<b>Average Somy</b>
ES041	1.10 (Ld36)	3.61 (Ld08)	2.08
ES042	1.87 (Ld01)	4.01 (Ld31)	2.13
ES043	1.89 (Ld01)	4.05 (Ld31)	2.13
ES044	1.58 (Ld01)	4.39 (Ld31)	2.12
ES045	1.61 (Ld01)	4.34 (Ld31)	2.12

**Table 8 - Summary of somy**

### 6.3 Copy Number Variation

The CNV values across most of the 36 chromosomes share the following pattern: libraries **ES042** and **ES043** follow the same pattern of chromosomal variance while libraries **ES044** and **ES045** share a different pattern. **ES044** and **ES045** are always read more frequently in locations that differ. Figure 14 demonstrates this pattern similarity in chromosome 7. Chromosome 11 and chromosome 30 [Figure 15] did not exhibit any differences between the libraries. **ES041** showed some differences from all libraries in chromosome 31 [Figure 16]. We also noticed areas where no reads were found across all libraries in chromosome 35 [Figure 17]. The differences in CNV values do not make biological sense. We believe that that we may have discovered an amplification bias or other method error.



**Figure 13 – Somy by Library**

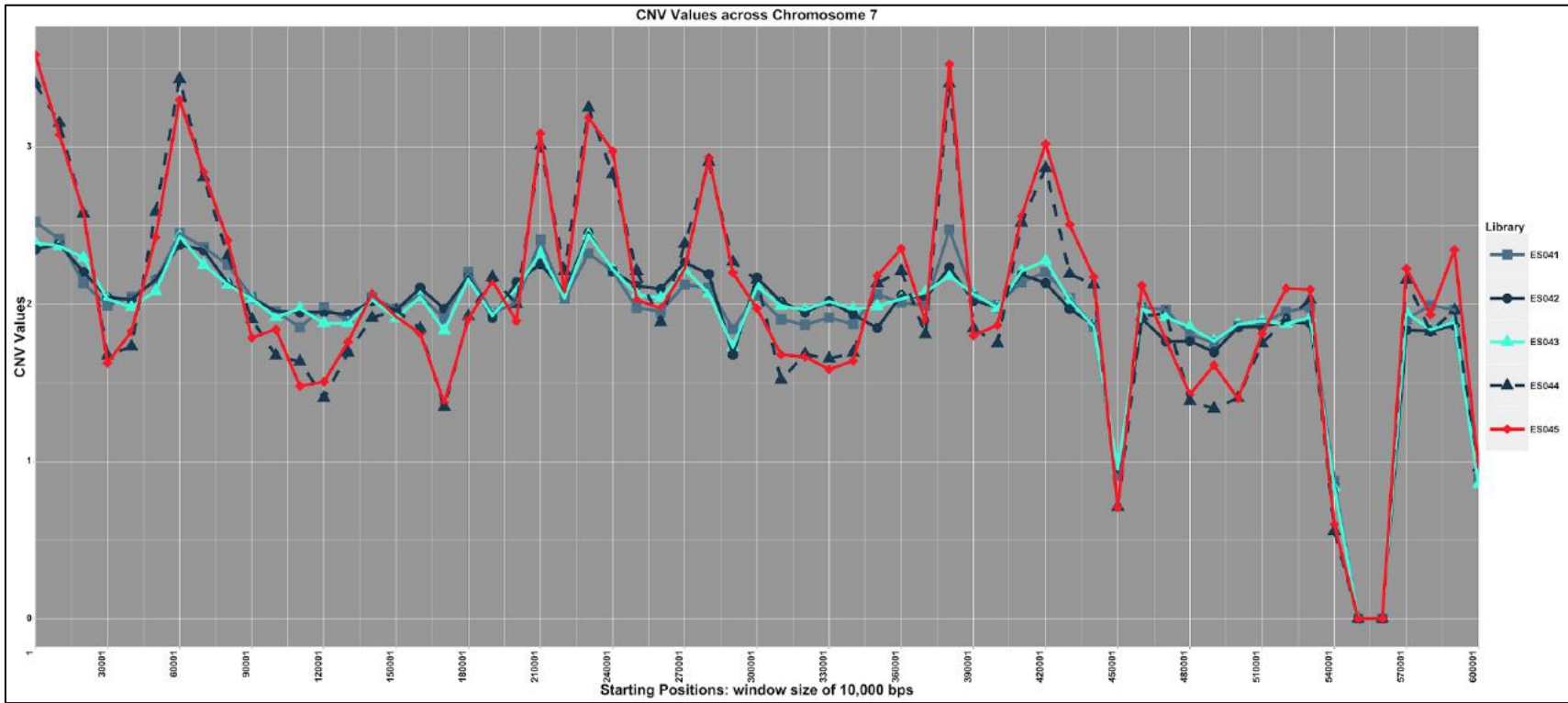


Figure 14 - CNV Values across Chromosome 7

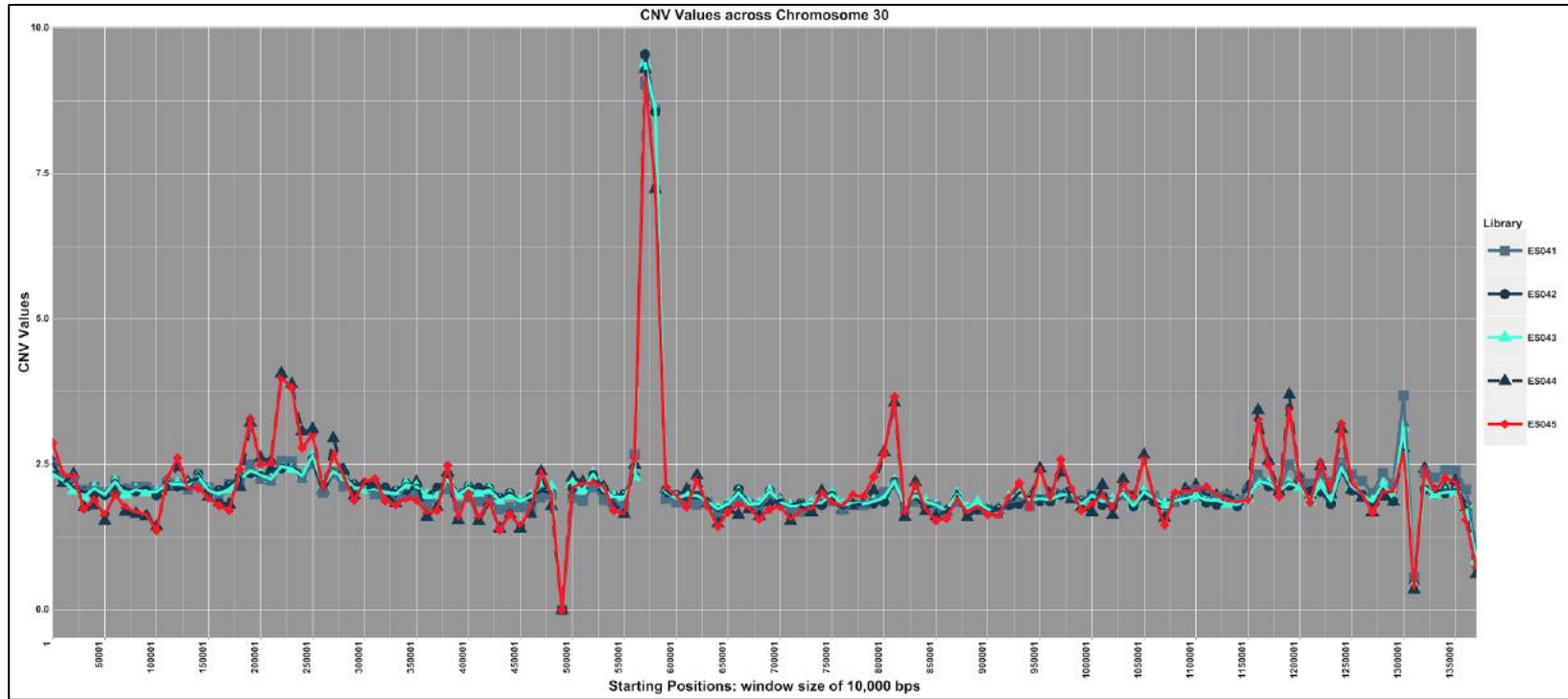


Figure 15 - CNV Values across Chromosome 30

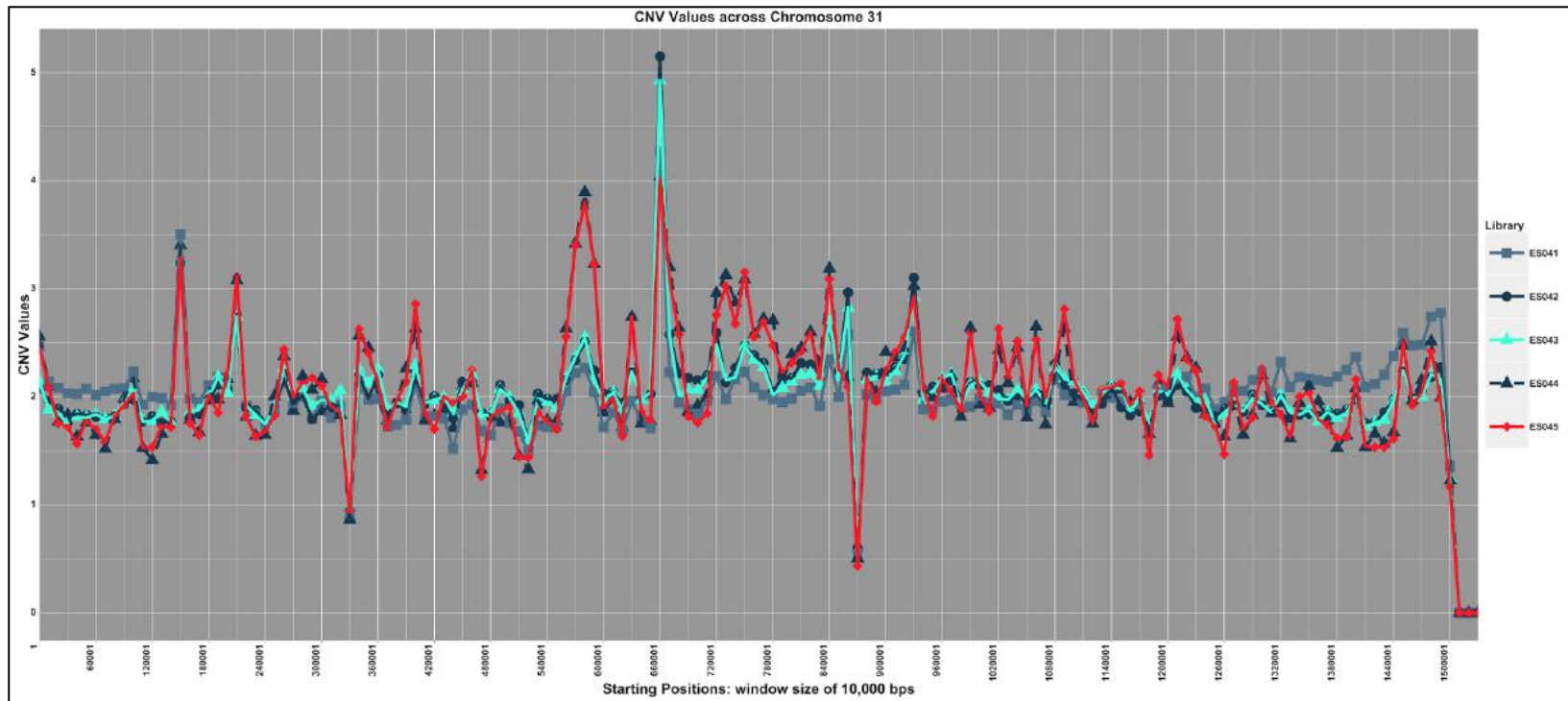


Figure 16 - CNV Values across Chromosome 31

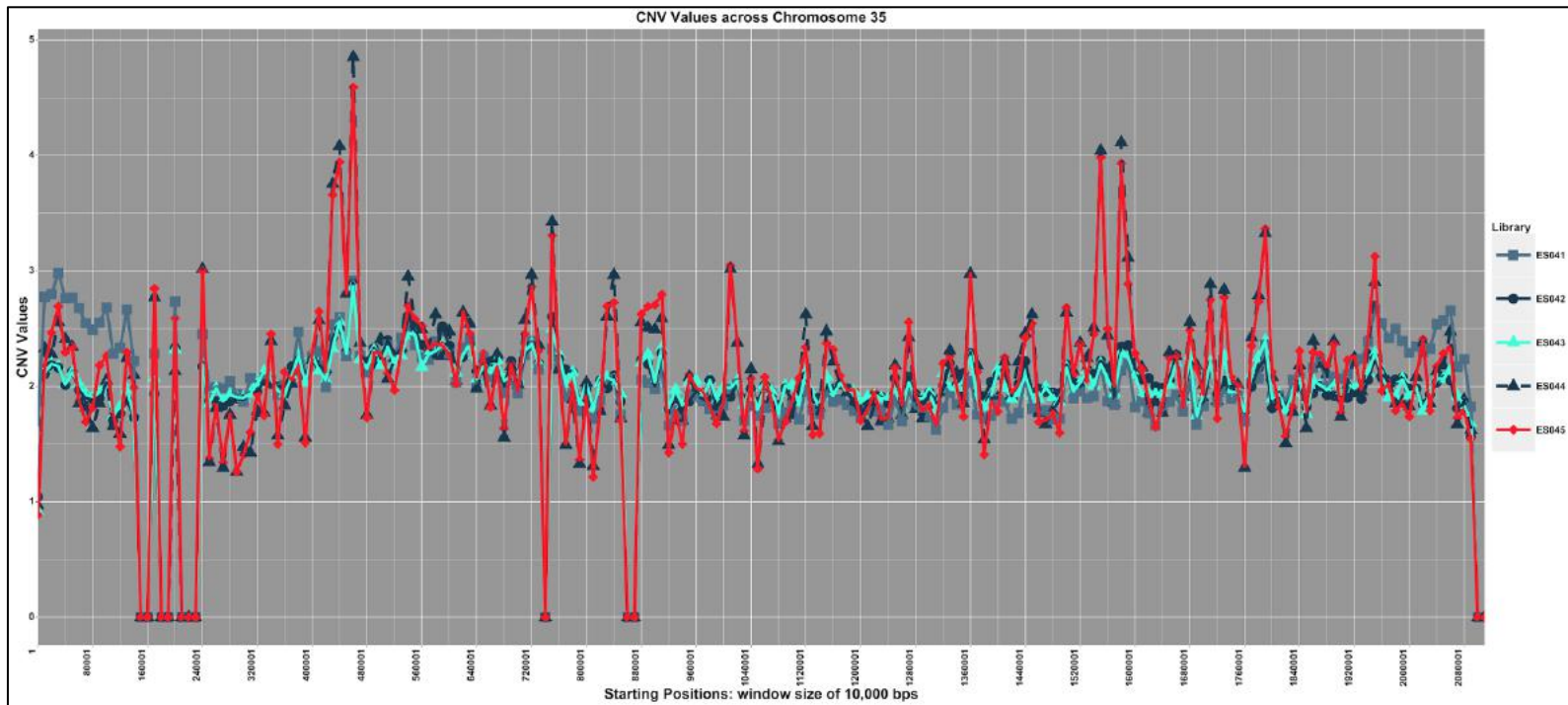


Figure 17 - CNV Values across Chromosome 35



## Section 7: Conclusions

Databases and analytical platforms are highly beneficial to the bioinformatics field because they provide a simple means of tracking many biological samples, storing large datasets, and comparing across multiple libraries. Current publically available databases and applications are geared towards one of these functionalities but are not a do-all system. **NGSdb** provides the community with a tool that encompasses all of these requirements. We demonstrated the capabilities of **NGSdb** and the **Somy/CNV** and **SNP modules**, through two test cases. **NGSdb** was able to reproduce the same results we previously published and successfully led us to the same conclusions we determined through manual review. Additionally, we illustrated how our application can be used to identify poor quality data and batch effects.

### 7.1 Discussion

**NGSdb** is open source and available online at <https://github.com/bifxcore/ngsdb>. Any researcher is able to download their own copy of **NGSdb** and with minimal setup, have a fully functioning copy. We have included all of our python scripts including those we use to automatically upload analysis results and the script we use to calculate the somy and CNV values. We are actively updating the available package as we expand and improve our database and interface. We have currently made over 250 updates to the code.

As previously mentioned, we have organized samples into unique experiments. This allows users to avoid wading through all loaded samples and to quickly identify those they are interested in querying. The organization also allows for an experience tailored to the user, as they will not be exposed to unnecessary information. For example, certain queries require the user to identify samples and/or libraries that they are interested in exploring. If the user does not identify a specific experiment, they will be required to choose the samples/libraries from all present in the

database. If the user has chosen a specific experiment, they will only be required to choose from the list of samples/libraries associated with this experiment. Additionally, each experiment page has links that send the user to queries specific to the type of experiment. If the experiment contains DNA libraries that have an associated SNP analysis then it will be linked to the queries that explore SNPs. Similarly, RNA libraries with spliced leader analyses will link to queries that examine this data.

We decided to store SNP data in both the SNP module and by storing the path to the annotated VCF file. We recognize that storing SNP data in two places has the potential to cause problems in the future but believe that certain queries can be completed much quicker through publicly available bioinformatic tools. Difficulties may arise if a user would like to update a SNP, as they must remember to update in both locations or there will be discrepancies. Conversely, the user may update the VCF file without updating the database. To protect ourselves from this, we have denied access to users to update the tables and have copied the VCF files to a secure location where only database administrators are granted access. Our upload script ensures that both tables are entered before committing the data. Additionally, we compute the MD5 hash for each VCF file. At any time, we can check the integrity of each VCF file to ensure that it has not changed since being uploaded.

Before we committed to storing the path to the VCF files, we created SQL queries to compare SNPs across libraries. While these queries returned the expected information, they were very slow, taking up to five minutes. In contrast, *VCFtools*, a public program designed to query, filter, compare, and summarize VCF files, is able to run the same query in under 30 seconds

[55]. We determined that the increased efficiency using *VCFtools* outweighed the potential risk of storing the VCF files. This is a potential area for improvement in the future regarding both query capabilities and data consistency.

## 7.2 Novel Contributions

We have demonstrated the following contributions to the bioinformatics community from the development of **NGSdb** and the **Comparative Genomics** module.

1. A flexible and adaptable infrastructure and database storage system for genomic sequences. The database consists of modules connecting to central tables allowing for easy expansion as new technological advancements bring new data types and analyses. The system is agnostic to organisms allowing any researcher to store their own genomic sequences.
2. A lightweight web application allowing researchers to view and explore their data sets. The application functions as an analytical platform with the capability to query the aforementioned database.
3. The analysis of five different *Leishmania* libraries using the web application to demonstrate the capability of the **Comparative Genomics** module.

## 7.3 Future Directions

We have demonstrated how **NGSdb** can be used to drive research by helping develop hypotheses, analyze genomic data, and provide an easy means of combining data types. The modular structure of the underlying database allows for future additions to be seamlessly

implemented. We anticipate that new genomic technologies will become available within the next few years that will require new modules. Dr. Jean-Claude Dujardin's research group from the Institute of Tropical Medicine, Antwerp, Belgium, has expressed interest in using **NGSdb** to integrate metabolomics data with their current genomic results. We are working with them to download and install **NGSdb** locally.

Not only do we predict new modules being implemented, but have intentions of improving current queries and adding additional queries. We plan to collect feedback from our collaborators in order to identify areas of improvement. These suggestions may range from visual changes to displaying additional data. User testing will be important in order to maintain and increase the use of the application. Additionally, we plan on including the option to run a variety of statistics to estimate the confidence of an analysis. With this plan, we will need to add more samples to the database to further increase the power of the statistics.

The genomic analysis of the five *Leishmania* libraries did not lead to a strong candidate list but was able to identify 10 SNPs across four genes that may explain the differences between the double knockout and double knockout recovery. These genes can be further explored in the lab to determine if they are contributing to the phenotypic differences.

## List of Figures

FIGURE 1 - <i>LEISHMANIA</i> LIFE CYCLE [13].....	10
FIGURE 2 – OVERVIEW OF NGSDB’S SCHEMA.....	25
FIGURE 3 - SOMY OPTIONS FOR GRAPHICAL DISPLAY .....	28
FIGURE 4 - COMPARATIVE GENOMIC MODULE.....	33
FIGURE 5 – SOMY VALUES FOR VL AND CL LIBRARIES.....	36
FIGURE 6 – OPTIONS TO COMPARE GROUPS OF CNV.....	38
FIGURE 7 – CNV DIFFERENCES BETWEEN VL AND CL LIBRARIES.....	39
FIGURE 8 – OVERVIEW OF LIBRARY SNPs.....	39
FIGURE 9 - LOGIC OF COMPARING GROUPS OF LIBRARIES.....	41
FIGURE 10 – SNP OVERVIEW BY IMPACT .....	42
FIGURE 11 – SNPs UNIQUE TO VL OR CL LIBRARIES .....	44
FIGURE 12 - PROMASTIGOTE DIFFERENTIATION .....	46
FIGURE 13 – SOMY BY LIBRARY .....	51
FIGURE 14 - CNV VALUES ACROSS CHROMOSOME 7 .....	52
FIGURE 15 - CNV VALUES ACROSS CHROMOSOME 30.....	53
FIGURE 16 - CNV VALUES ACROSS CHROMOSOME 31.....	54
FIGURE 17 - CNV VALUES ACROSS CHROMOSOME 35.....	55

## List of Tables

TABLE 1 - STATISTICS TABLE WITHOUT CV TABLE.....	26
TABLE 2 - STATISTICS TABLE WITH CV TABLE .....	27
TABLE 3 - SOFTWARE REQUIRED FOR NGSDB.....	29
TABLE 4 - PYTHON SOFTWARE REQUIRED FOR NGSDB.....	30
TABLE 5 – OVERVIEW OF SOMY VALUES .....	36
TABLE 6 – SUMMARY OF SNPS BY LIBRARY .....	40
TABLE 7 - LIST OF <i>LEISHMANIA DONOVANI</i> SAMPLES.....	46
TABLE 8 - SUMMARY OF SOMY.....	50

## Works Cited

- [1] T. M. Daniel, “The history of tuberculosis,” *Respiratory Medicine*, vol. 100, no. 11, pp. 1862–1870, Nov. 2006.
- [2] R. J. Littman, “The plague of Athens: epidemiology and paleopathology,” *Mt. Sinai J. Med.*, vol. 76, no. 5, pp. 456–467, Oct. 2009.
- [3] C. Platt, *King Death: The Black Death And Its Aftermath In Late-Medieval England*. Routledge, 2014.
- [4] A. M. Geddes, “The history of smallpox,” *Clinics in Dermatology*, vol. 24, no. 3, pp. 152–157, May 2006.
- [5] K. D. Patterson, “Yellow fever epidemics and mortality in the United States, 1693-1905,” *Soc Sci Med*, vol. 34, no. 8, pp. 855–865, Apr. 1992.
- [6] “WHO | International Classification of Diseases (ICD),” *WHO*. [Online]. Available: <http://www.who.int/classifications/icd/en/>. [Accessed: 06-May-2015].
- [7] WHO | World Health Organization, “Neglected tropical diseases,” *WHO*, 2015. [Online]. Available: [http://www.who.int/neglected\\_diseases/diseases/en/](http://www.who.int/neglected_diseases/diseases/en/). [Accessed: 23-Jan-2015].
- [8] C. A. Hutchison, “DNA sequencing: bench to bedside and beyond†,” *Nucleic Acids Res*, vol. 35, no. 18, pp. 6227–6237, Sep. 2007.
- [9] F. Xu, Q. Wang, F. Zhang, Y. Zhu, Q. Gu, L. Wu, L. Yang, and X. Yang, “Impact of Next-Generation Sequencing (NGS) technology on cardiovascular disease research,” *Cardiovasc Diagn Ther*, vol. 2, no. 2, pp. 138–146, Jun. 2012.
- [10] T. 1000 G. P. Consortium, “A map of human genome variation from population-scale sequencing,” *Nature*, vol. 467, no. 7319, pp. 1061–1073, Oct. 2010.
- [11] “WHO | Leishmaniasis,” *WHO*. [Online]. Available: <http://www.who.int/mediacentre/factsheets/fs375/en/>. [Accessed: 23-Jan-2015].
- [12] C.-C. for D. C. and Prevention, “CDC - Leishmaniasis.” [Online]. Available: <http://www.cdc.gov/parasites/leishmaniasis/>. [Accessed: 06-May-2015].
- [13] C.-C. for D. C. and Prevention, “CDC - Leishmaniasis - Biology.” [Online]. Available: <http://www.cdc.gov/parasites/leishmaniasis/biology.html>. [Accessed: 23-Jan-2015].
- [14] C.-C. for D. C. and Prevention, “CDC - Leishmaniasis - Disease.” [Online]. Available: <http://www.cdc.gov/parasites/leishmaniasis/disease.html>. [Accessed: 23-Jan-2015].
- [15] S. M. Gossage, M. E. Rogers, and P. A. Bates, “Two separate growth phases during the development of *Leishmania* in sand flies: implications for understanding the life cycle,” *Int. J. Parasitol.*, vol. 33, no. 10, pp. 1027–1034, Sep. 2003.
- [16] J. C. Antoine, E. Prina, T. Lang, and N. Courret, “The biogenesis and properties of the parasitophorous vacuoles that harbour *Leishmania* in murine macrophages,” *Trends Microbiol.*, vol. 6, no. 10, pp. 392–401, Oct. 1998.
- [17] I. Mitroulis, I. Kourtzelis, V. P. Papadopoulos, K. Mimidis, M. Speletas, and K. Ritis, “In vivo induction of the autophagic machinery in human bone marrow cells during *Leishmania donovani* complex infection,” *Parasitology International*, vol. 58, no. 4, pp. 475–477, Dec. 2009.

- [18] M. Shadab and N. Ali, "Evasion of Host Defence by *Leishmania donovani*: Subversion of Signaling Pathways," *Molecular Biology International*, vol. 2011, p. e343961, Apr. 2011.
- [19] C.-C. for D. C. and Prevention, "CDC - Leishmaniasis - Resources for Health Professionals." [Online]. Available: [http://www.cdc.gov/parasites/leishmaniasis/health\\_professionals/index.html#tx](http://www.cdc.gov/parasites/leishmaniasis/health_professionals/index.html#tx). [Accessed: 23-Jan-2015].
- [20] I. H. G. S. Consortium, "Finishing the euchromatic sequence of the human genome," *Nature*, vol. 431, no. 7011, pp. 931–945, Oct. 2004.
- [21] T. Tatusova, M. DiCuccio, A. Badretdin, V. Chetvernin, S. Ciufu, and W. Li, "Prokaryotic Genome Annotation Pipeline," in *The NCBI Handbook*, 2nd Edition., National Center for Biotechnology Information (US).
- [22] E. V. Koonin and M. Y. Galperin, "Genome Annotation and Analysis," 2003.
- [23] G. Spencer, "Background on Comparative Genomic Analysis," Dec. 2002.
- [24] "HiSeq 2500 Specifications," Illumina, Inc, 2015.
- [25] J. Zhang, R. Chiodini, A. Badr, and G. Zhang, "The impact of next-generation sequencing on genomics," *J Genet Genomics*, vol. 38, no. 3, pp. 95–109, Mar. 2011.
- [26] M. Hsi-Yang Fritz, R. Leinonen, G. Cochrane, and E. Birney, "Efficient storage of high throughput DNA sequencing data using reference-based compression," *Genome Res.*, vol. 21, no. 5, pp. 734–740, May 2011.
- [27] H. Li, B. Handsaker, A. Wysoker, T. Fennell, J. Ruan, N. Homer, G. Marth, G. Abecasis, R. Durbin, and 1000 Genome Project Data Processing Subgroup, "The Sequence Alignment/Map format and SAMtools," *Bioinformatics*, vol. 25, no. 16, pp. 2078–2079, Aug. 2009.
- [28] P. Cock, "Blasted Bioinformatics!?: BGZF - Blocked, Bigger & Better GZIP!," *Blasted Bioinformatics!?*, 08-Nov-2011. .
- [29] W. J. Kent, C. W. Sugnet, T. S. Furey, K. M. Roskin, T. H. Pringle, A. M. Zahler, and D. Haussler, "The human genome browser at UCSC," *Genome Res.*, vol. 12, no. 6, pp. 996–1006, Jun. 2002.
- [30] F. Cunningham, M. R. Amode, D. Barrell, K. Beal, K. Billis, S. Brent, D. Carvalho-Silva, P. Clapham, G. Coates, S. Fitzgerald, L. Gil, C. G. Girón, L. Gordon, T. Hourlier, S. E. Hunt, S. H. Janacek, N. Johnson, T. Juettemann, A. K. Kähäri, S. Keenan, F. J. Martin, T. Maurel, W. McLaren, D. N. Murphy, R. Nag, B. Overduin, A. Parker, M. Patricio, E. Perry, M. Pignatelli, H. S. Riat, D. Sheppard, K. Taylor, A. Thormann, A. Vullo, S. P. Wilder, A. Zadissa, B. L. Aken, E. Birney, J. Harrow, R. Kinsella, M. Muffato, M. Ruffier, S. M. J. Searle, G. Spudich, S. J. Trevanion, A. Yates, D. R. Zerbino, and P. Flicek, "Ensembl 2015," *Nucleic Acids Res.*, vol. 43, no. Database issue, pp. D662–669, Jan. 2015.
- [31] P. M. 27 and 2014 inShare24 Email Print Comments, "Bioinformatics Infrastructure Got You Down? Head to the Cloud, Rent a Supercomputer! | Biocompare: The Buyer's Guide for Life Scientists." [Online]. Available: <http://www.biocompare.com/Editorial-Articles/158629-Bioinformatics-Infrastructure-Got-You-Down-Head-to-the-Cloud-Rent-a-Supercomputer/>. [Accessed: 06-May-2015].



- [32] J. Perkel, “Cybersecurity: How safe are your data?,” *Nature News*, vol. 464, no. 7293, pp. 1260–1261, Apr. 2010.
- [33] A. Cornish and C. Guda, “A Comparison of Variant Calling Pipelines Using Genome in a Bottle as a Reference,” *BioMed Research International*.
- [34] *Genome*. National Center for Biotechnology Information.
- [35] M. Aslett, C. Aurrecochea, M. Berriman, J. Brestelli, B. P. Brunk, M. Carrington, D. P. Depledge, S. Fischer, B. Gajria, X. Gao, M. J. Gardner, A. Gingle, G. Grant, O. S. Harb, M. Heiges, C. Hertz-Fowler, R. Houston, F. Innamorato, J. Iodice, J. C. Kissinger, E. Kraemer, W. Li, F. J. Logan, J. A. Miller, S. Mitra, P. J. Myler, V. Nayak, C. Pennington, I. Phan, D. F. Pinney, G. Ramasamy, M. B. Rogers, D. S. Roos, C. Ross, D. Sivam, D. F. Smith, G. Srinivasamoorthy, C. J. Stoeckert, S. Subramanian, R. Thibodeau, A. Tivey, C. Treatman, G. Velarde, and H. Wang, “TriTrypDB: a functional genomic resource for the Trypanosomatidae,” *Nucleic Acids Res.*, vol. 38, no. Database issue, pp. D457–462, Jan. 2010.
- [36] *The NCBI Handbook*, 2nd ed. National Center for Biotechnology Information (US), 2013.
- [37] E. W. Sayers, T. Barrett, D. A. Benson, E. Bolton, S. H. Bryant, K. Canese, V. Chetvernin, D. M. Church, M. Dicuccio, S. Federhen, M. Feolo, I. M. Fingerman, L. Y. Geer, W. Helmsberg, Y. Kapustin, S. Krasnov, D. Landsman, D. J. Lipman, Z. Lu, T. L. Madden, T. Madej, D. R. Maglott, A. Marchler-Bauer, V. Miller, I. Karsch-Mizrachi, J. Ostell, A. Panchenko, L. Phan, K. D. Pruitt, G. D. Schuler, E. Sequeira, S. T. Sherry, M. Shumway, K. Sirotkin, D. Slotta, A. Souvorov, G. Starchenko, T. A. Tatusova, L. Wagner, Y. Wang, W. J. Wilbur, E. Yaschenko, and J. Ye, “Database resources of the National Center for Biotechnology Information,” *Nucleic Acids Res.*, vol. 40, no. Database issue, pp. D13–25, Jan. 2012.
- [38] J. Goecks, A. Nekrutenko, J. Taylor, and \$author firstName \$author.lastName, “Galaxy: a comprehensive approach for supporting accessible, reproducible, and transparent computational research in the life sciences,” *Genome Biology*, vol. 11, no. 8, p. R86, Aug. 2010.
- [39] D. Blankenberg, G. Von Kuster, N. Coraor, G. Ananda, R. Lazarus, M. Mangan, A. Nekrutenko, and J. Taylor, “Galaxy: a web-based genome analysis tool for experimentalists,” *Curr Protoc Mol Biol*, vol. Chapter 19, p. Unit 19.10.1–21, Jan. 2010.
- [40] B. Giardine, C. Riemer, R. C. Hardison, R. Burhans, L. Elnitski, P. Shah, Y. Zhang, D. Blankenberg, I. Albert, J. Taylor, W. Miller, W. J. Kent, and A. Nekrutenko, “Galaxy: a platform for interactive large-scale genome analysis,” *Genome Res.*, vol. 15, no. 10, pp. 1451–1455, Oct. 2005.
- [41] J. Jackson, “File Upload via FTP,” 14-Nov-2013.
- [42] *PostgreSQL*. The PostgreSQL Global Development Group.
- [43] *Django Web Framework*. Django.
- [44] M. Otto and J. Thornton, *Bootstrap*. Twitter Bootstrap.
- [45] J. Forcier, *Fabric*. 2015.

- [46] N. D. Karunaweera, F. Pratlong, H. V. Y. D. Siriwardane, R. L. Ihalamulla, and J. P. Dedet, "Sri Lankan cutaneous leishmaniasis is caused by *Leishmania donovani* zymodeme MON-37," *Trans. R. Soc. Trop. Med. Hyg.*, vol. 97, no. 4, pp. 380–381, Aug. 2003.
- [47] S. Ranasinghe, R. Wickremasinghe, A. Munasinghe, S. Hulangamuwa, S. Sivanantharajah, K. Seneviratne, S. Bandara, I. Athauda, C. Navaratne, O. Silva, H. Wackwella, G. Matlashewski, and R. Wickremasinghe, "Cross-sectional study to assess risk factors for leishmaniasis in an endemic region in Sri Lanka," *Am. J. Trop. Med. Hyg.*, vol. 89, no. 4, pp. 742–749, Oct. 2013.
- [48] H. V. Y. D. Siriwardana, H. A. Noyes, N. J. Beeching, M. L. Chance, N. D. Karunaweera, and P. A. Bates, "Leishmania donovani and cutaneous leishmaniasis, Sri Lanka," *Emerging Infect. Dis.*, vol. 13, no. 3, pp. 476–478, Mar. 2007.
- [49] S. Ranasinghe, W.-W. Zhang, R. Wickremasinghe, P. Abeygunasekera, V. Chandrasekharan, S. Athauda, S. Mendis, S. Hulangamuwa, G. Matlashewski, and F. Pratlong, "Leishmania donovani zymodeme MON-37 isolated from an autochthonous visceral leishmaniasis patient in Sri Lanka," *Pathog Glob Health*, vol. 106, no. 7, pp. 421–424, Nov. 2012.
- [50] W. W. Zhang, G. Ramasamy, L.-I. McCall, A. Haydock, S. Ranasinghe, P. Abeygunasekera, G. Sirimanna, R. Wickremasinghe, P. Myler, and G. Matlashewski, "Genetic Analysis of *Leishmania donovani* Tropism Using a Naturally Attenuated Cutaneous Strain," *PLoS Pathogens*, vol. 10, no. 7, p. e1004244, Jul. 2014.
- [51] M. Martin, "Cutadapt removes adapter sequences from high-throughput sequencing reads," *EMBnet.journal*, vol. 17, no. 1, pp. pp. 10–12, May 2011.
- [52] T. Downing, H. Imamura, S. Decuypere, T. G. Clark, G. H. Coombs, J. A. Cotton, J. D. Hilley, S. de Doncker, I. Maes, J. C. Mottram, M. A. Quail, S. Rijal, M. Sanders, G. Schönian, O. Stark, S. Sundar, M. Vanaerschot, C. Hertz-Fowler, J.-C. Dujardin, and M. Berriman, "Whole genome sequencing of multiple *Leishmania donovani* clinical isolates provides insights into population structure and mechanisms of drug resistance," *Genome Res*, vol. 21, no. 12, pp. 2143–2156, Dec. 2011.
- [53] B. Langmead and S. L. Salzberg, "Fast gapped-read alignment with Bowtie 2," *Nat. Methods*, vol. 9, no. 4, pp. 357–359, Apr. 2012.
- [54] P. Cingolani, A. Platts, L. L. Wang, M. Coon, T. Nguyen, L. Wang, S. J. Land, X. Lu, and D. M. Ruden, "A program for annotating and predicting the effects of single nucleotide polymorphisms, SnpEff: SNPs in the genome of *Drosophila melanogaster* strain w1118; iso-2; iso-3," *Fly (Austin)*, vol. 6, no. 2, pp. 80–92, Jun. 2012.
- [55] P. Danecek, A. Auton, G. Abecasis, C. A. Albers, E. Banks, M. A. DePristo, R. E. Handsaker, G. Lunter, G. T. Marth, S. T. Sherry, G. McVean, R. Durbin, and 1000 Genomes Project Analysis Group, "The variant call format and VCFtools," *Bioinformatics*, vol. 27, no. 15, pp. 2156–2158, Aug. 2011.
- [56] P. Tsigankov, P. F. Gherardini, M. Helmer-Citterich, G. F. Späth, and D. Zilberstein, "Phosphoproteomic Analysis of Differentiating *Leishmania* Parasites Reveals a Unique

- Stage-Specific Phosphorylation Motif,” *J. Proteome Res.*, vol. 12, no. 7, pp. 3405–3412, Jul. 2013.
- [57] M. A. Morales, R. Watanabe, C. Laurent, P. Lenormand, J.-C. Rousselle, A. Namane, and G. F. Späth, “Phosphoproteomic analysis of *Leishmania donovani* pro- and amastigote stages,” *Proteomics*, vol. 8, no. 2, pp. 350–363, Jan. 2008.
- [58] M. A. Morales, R. Watanabe, M. Dacher, P. Chafey, J. Osorio y Fortéa, D. A. Scott, S. M. Beverley, G. Ommen, J. Clos, S. Hem, P. Lenormand, J.-C. Rousselle, A. Namane, and G. F. Späth, “Phosphoproteome dynamics reveal heat-shock protein complexes specific to the *Leishmania donovani* infectious stage,” *Proc. Natl. Acad. Sci. U.S.A.*, vol. 107, no. 18, pp. 8381–8386, May 2010.
- [59] E. Barak, S. Amin-Spector, E. Gerliak, S. Goyard, N. Holland, and D. Zilberstein, “Differentiation of *Leishmania donovani* in host-free system: analysis of signal perception and response,” *Mol. Biochem. Parasitol.*, vol. 141, no. 1, pp. 99–108, May 2005.
- [60] M. B. Rogers, J. D. Hilley, N. J. Dickens, J. Wilkes, P. A. Bates, D. P. Depledge, D. Harris, Y. Her, P. Herzyk, H. Imamura, T. D. Otto, M. Sanders, K. Seeger, J.-C. Dujardin, M. Berriman, D. F. Smith, C. Hertz-Fowler, and J. C. Mottram, “Chromosome and gene copy number variation allow major structural change between species and strains of *Leishmania*,” *Genome Res.*, vol. 21, no. 12, pp. 2129–2142, Dec. 2011.
- [61] Y. Sterkers, L. Lachaud, N. Bourgeois, L. Crobu, P. Bastien, and M. Pagès, “Novel insights into genome plasticity in Eukaryotes: mosaic aneuploidy in *Leishmania*,” *Mol. Microbiol.*, vol. 86, no. 1, pp. 15–23, Oct. 2012.