# EST Visualizer

Patrick Graves
Bioinformatics
Thursday, April 26, 2007

## 1 - ABSTRACT

Using many concepts related to bioinformatics, an application was created to visually display EST’s. Each EST was displayed in the correct position relative to the EST’s around it. Only EST’s in a user specified region of a user specified genome are displayed. Each EST was color coordinated according to its tissue type. The eventual hope is that the application will be useful in studying alternative splices and gene expression according to tissue type. Currently the application does not incorporate all of the features which would make it a useful application. While it does display the EST’s correctly, it only color codes certain tissue types and does not correctly display EST’s which are “split” on the genome. Further, the methods by which zooming is handled could be improved and many additional features such as EST filters could be added to enhance the usefulness of the program. This paper will provide a brief introduction to the bioinformatics aspects behind the application, discuss the design and implementation of the project, and then give insight into how the application could be improved for future use.

## 2 - INTRODUCTION

This project begins with understanding many of the biological terms used in bioinformatics. Before one can understand the meaning behind "BLASTing" EST's in FASTA format against genomes to obtain starting and stopping locations for these EST's on particular chromosomes, the definitions of the terms in this sentence must be defined. The following several sections introduce many of the bioinformatics terms that must be understood in order to grasp this project.

I did not have an interest in the project before I began working on it for my graduate project. However, as I worked on it, I developed more of an interest in how EST's could be displayed visually to convey the most amount of information about them. As far as I am aware, there is no existing software which accomplishes the tasks of displaying EST's over an input region.

## 2.1 - EST

An EST is an Expressed Sequence Tag. It is essentially a short sequenced of a transcribed spliced nucleotide sequence. EST's do not actually exist but are rather fabricated markers for identifying gene transcripts. They are produced by sequencing a clone of messenger RNA. The resulting sequences are limited in size by today's technology so EST's are generally on the order of 400 to 800 bases long. Since they are portions of complementary DNA they represent segments of expressed genes. [5]

Once EST's have been identified they can be physically mapped to chromosome locations using biological processes. If the genome for the organism from which the EST originated has been already mapped however, EST's can be mapped to chromosome locations using sequence alignment algorithms. One such alignment algorithm, BLAST, will be discussed later in the paper. The technique of using sequence alignment algorithms requires that the EST's for each organism be stored in a database. The EST's are currently stored in a database called dbEST which is a subset of GenBank. There are currently approximately 42 million EST's available in this database. [5]

## 2.2 - dbEST

When dbEST is referred to a database, this is somewhat of a stretch. dbEST is merely a collection of flat files in which EST's are represented in GenBank format. Each file contains millions of sequences, all of which follow this GenBank format. [4]

## 2.3 - GENBANK FILE FORMAT

The following description of GenBank format taken from http://www.psc.edu/general/software/packages/seq-intro/genbankfile.html reveals that GenBank format contains detailed information about each sequence.

File Header

- The first line in the file must have "GENETIC SEQUENCE DATA BANK" in spaces 20 through 46.
- The next 8 lines may contain arbitrary text. They are ignored but are required to maintain the GenBank format.

Sequence Data Entries

a. Each sequence entry in the file should have the following format:

first line

Must have LOCUS in the first 5 spaces. The genetic locus name or identifier must be in spaces 13 - 22. The length of the sequences is right justified in spaces 23 through 29.

second line

Must have DEFINITION in the first 10 spaces. Spaces 13 - 80 are free form text to identify the sequence.

third line

Must have ACCESSION in the first 9 spaces. Spaces 13 - 18 must hold the primary accession number.

fourth line

Must have ORIGIN in the first 6 spaces. Nothing else is required on this line, it indicates that the nucleic acid sequence begins on the next line.

fifth line

Begins the nucleotide sequence. The first 9 spaces of each sequence line may either be blank or may contain the position in the sequence of the first nucleotide on the line. The next 66 spaces hold the nucleotide sequence in six blocks of ten nucleotides. Each of the six blocks begins with a blank space followed by ten nucleotides. Thus the first nucleotide is in space eleven of the line while the last is in space 75.

last line

Must have // in the first 2 spaces to indicate termination of the sequence.[7]

## 2.4 - FASTA FILE FORMAT

While GenBank file format is very thorough in its description of a given sequence it is often beneficial to view data in other formats. FASTA file format is a much shorter and less descriptive format for representing sequences. In this project, FASTA file format is used because it takes up less storage space and contains only the necessary information for completing the tasks at hand. [6]

Fasta file format begins with a single line description of the sequence data. After this description comes the sequence data itself. Each sequence is separated by the ">" symbol which identifies the beginning of a new sequence. Immediately following the ">" symbol comes the optional description of the sequence. Information in the description is separated by "|" symbols. On the next line the sequence data is printed. An example of a protein (Amino Acid) sequence in this file format is:

```
>gi|111096|pir||D35141 T-cell receptor delta chain V region (105.211) -
mouse  (fragment)
CASGYIGGIRTDKL
```

## 2.5 - BLAST

There are two versions of  BLAST; NCBI BLAST which comes from National Center for Biotechnology Information and wuBLAST which comes from Washington University. This project uses wuBlast version 2.0 for performing sequence alignments. Particularly, BLASTn is used for comparing nucleotide sequences. BLAST stands for Basic Local Alignment Search Tool. It is an algorithm which compares biological sequences of information. BLASTp is used for comparing protein sequences while

BLASTn is used for comparing nucleotide sequences. BLAST returns results such as where sequence alignment occurs (start and stop locations) as well as what percentage of the sequence matched in the alignment region. This value is generally reported in the form of an E-value which represents the expectancy that the sequence occurred randomly and therefore is not really a match to the query sequence. In this project, BLAST was used to align each EST of a given organism to its genome. The results of the operation are used to find which chromosome the EST is located and where on that chromosome the EST is located. This information was then stored in a MySQL database for quick lookup. [4]

## 2.6 - UNIGENE

UniGene is an organized view of the transcriptome. Each UniGene entry is a set of sequence which come from the same gene or expressed gene. [4] It therefore can indicate which EST's are located on a given gene. UniGene is only mentioned in this report because it seems like it would be a good location to find where in each genome a given EST is located. This would prevent having to BLAST every EST against the genomes. It is not an ideal database to lookup information about EST’s however because it only contains information about EST’s located on genes. It would not be possible to lookup EST’s located on a given portion of the genome where a gene is not located. In addition, even looking up EST’s at a given genes location would not be ideal. UniGene contains many EST’s located on each gene but it is not guaranteed to be complete. When new EST’s are created and inserted into dbEST, it may take a while for them to be inserted into UniGene.

## 3 - METHODS:

### 3.1 - PURPOSE

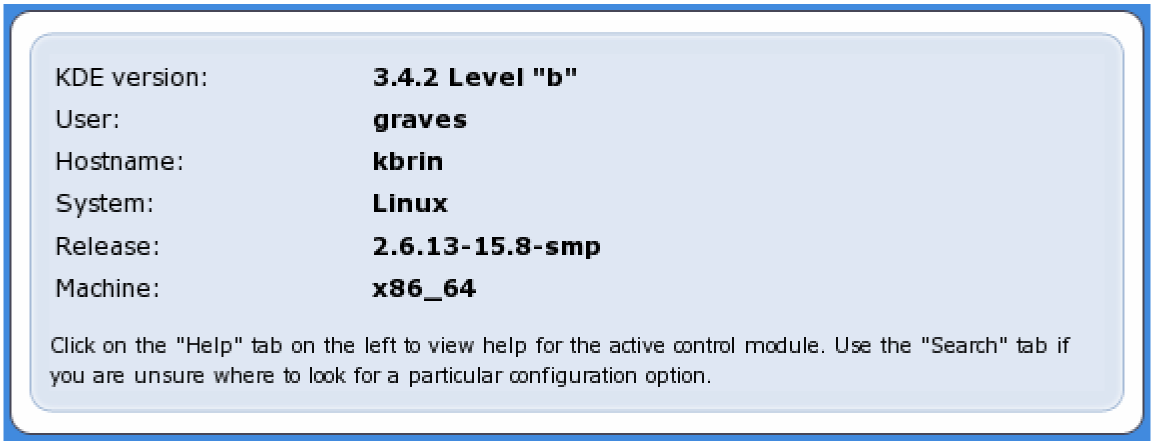
Now that minimal background information about many of the biological terms discussed throughout this project have been defined, it is necessary to talk about what the purpose of the project actually is. The purpose of this project is to locate all the EST's occurring in a given location on the genome of an organism and visually display this location along with each EST's tissue type.

The first step of this project is to download all of the EST sequences in the dbEST database and split them into files based on the organisms from which they originated. dbEST is organized alphabetically and therefore is not useful for identifying EST's which come from a particular organism. In addition, the records in the dbEST are stored in GenBank format. This is unnecessary and a waste of storage space when the only information needed is the GenBank identifier and sequence. For this reason the first step in this project is to take the alphabetical GenBank records from dbEST, parse them, and reorganize them into FASTA file format records according to the organism from which they originated.

The second step in the project is to take every EST in a given organism and align it to the organism's genome. This requires the use of the alignment algorithm BLAST. The results from BLAST are then parsed and stored in a database so that they can easily be looked up at a later date. Once in the database, each EST can be identified by the organism from which it originated, the chromosome which it is located on, the start index on the chromosome of the alignment and the stop index on the chromosome of the alignment.  In addition the significance of each alignment is stored in case only sequences with a certain significance are to be displayed.

After the information has been stored in the database it can be visually displayed. This comprises the third step in the project.  With the user inputting a segment of the genome to look at (by inputting a given sequence), the program can query which EST's are located in the given region and then display these EST's on the screen.  Then each EST can be looked up in GenBank to determine its tissue type and each tissue type can be represented by a different color on the screen.

### 3.2 - HARDWARE:



KDE version: **3.4.2 Level "b"**
User: **graves**
Hostname: **kbrin**
System: **Linux**
Release: **2.6.13-15.8-smp**
Machine: **x86_64**

Click on the "Help" tab on the left to view help for the active control module. Use the "Search" tab if you are unsure where to look for a particular configuration option.

Processors:

```
processor:          0
vendor_id:          GenuineIntel
model name:         Intel(R) Xeon(TM) CPU 2.80GHz
cpu MHz:            2793.080
cache size:         2048 KB
bogomips:           5591.12
clflush size:       64
cache_alignment:    128
address sizes:      36 bits physical, 48 bits virtual

processor:          1
vendor_id:          GenuineIntel
model name:         Intel(R) Xeon(TM) CPU 2.80GHz
```

```
cpu MHz:            2793.080
cache size:         2048 KB
bogomips:           5586.21
clflush size:       64
cache_alignment:    128
address sizes:      36 bits physical, 48 bits virtual
```

RAM:

```
MemTotal:     1019724 kB
MemFree:        10204 kB
Buffers:     48684 kB
Cached:      248980 kB
PageTables:     16680 kB
```

## 3.3 - SOFTWARE:

SUSE LINUX 10.0 (X86-64)
VERSION = 10.0

Eclipse Version 3.2 – Java IDE (Integrated Development Environment)

Java 1.5.0_10 – Java Runtime Environment

## 3.4 - TOOLS:

**BioJava –** “BioJava is an open-source project dedicated to providing a Java framework for processing biological data. It include objects for manipulating biological sequences, file parsers, DAS client and server support, access to BioSQL and Ensembl databases, tools for making sequence analysis GUIs and powerful analysis and statistical routines including a dynamic programming toolkit.” It was used in this project mostly for parsing files. It was used to parse files in GenBank format as well as FASTA format. It also was used to parse vital information from the BLAST output. In addition biojava made extracting information about a given EST simple by automatically retrieving information from GenBank given a GenBank Identifier. [3]

**BLAST –** refer to background portion of the paper.

**MySQL –** MySQL is a popular open source database that makes information storage and retrieval simple. It was used because of the enormous amount of information that needed to be stored and it makes sorting information simple. Querying for EST's located in a given region so quickly would not have been possible without the aid of a database. In addition the ability to interact with a database through JAVA code using the jdbc .jar file made MySQL an ideal choice for a database.

**Gmail SMTP server –** The Gmail SMTP server was only used in this application because the extended periods of time that users of the application would have to wait to get results. After example code was found online explaining how to send emails through the Gmail SMTP server in JAVA, it was mail service of choice.

**SSH –** the application requires interaction with the server (kbrin.a-bldg.louisville.edu) at medical school. From the beginning there were two methods conceived of how this task could be accomplished. Either the application could be web based or it could be standalone and interact with the server in another fashion. Since the latter option was chosen, it was necessary to find a protocol through which the application could interact with the server securely. Secure Shell seemed ideal since there are many open source JAVA API's available for connecting to a server through SSH programmatically.

**Piccolo –** "Piccolo is a toolkit that supports the development of 2D structured graphics programs, in general, and Zoomable User Interfaces (ZUIs), in particular."

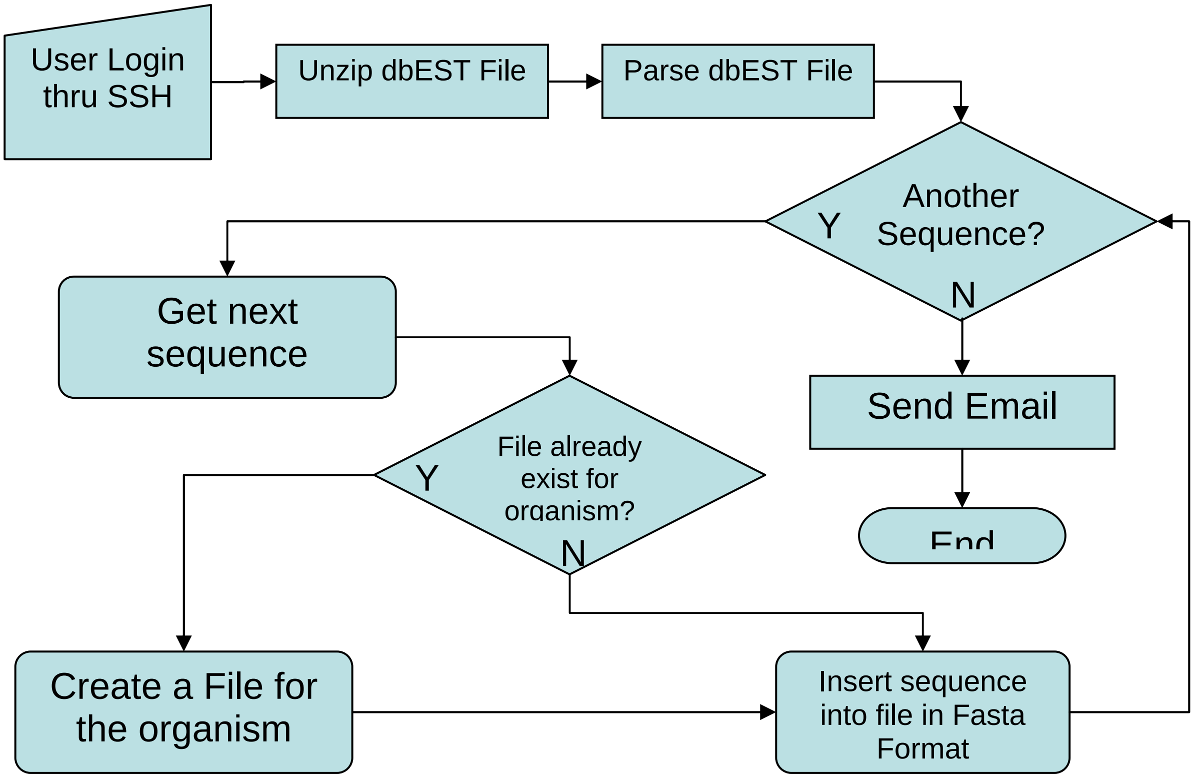
[2] Piccolo was chosen because it allows the user to build structured graphical applications without worrying about many of the low level devices employed in such an application. In addition many students were studying Piccolo in their Human Computer Interaction class and they suggested its use based on its nicely developed API. Without Piccolo, accomplishing many features of the application would have taken much more work and ingenuity.

### 3.5 - IMPLEMENTATION / DESIGN:

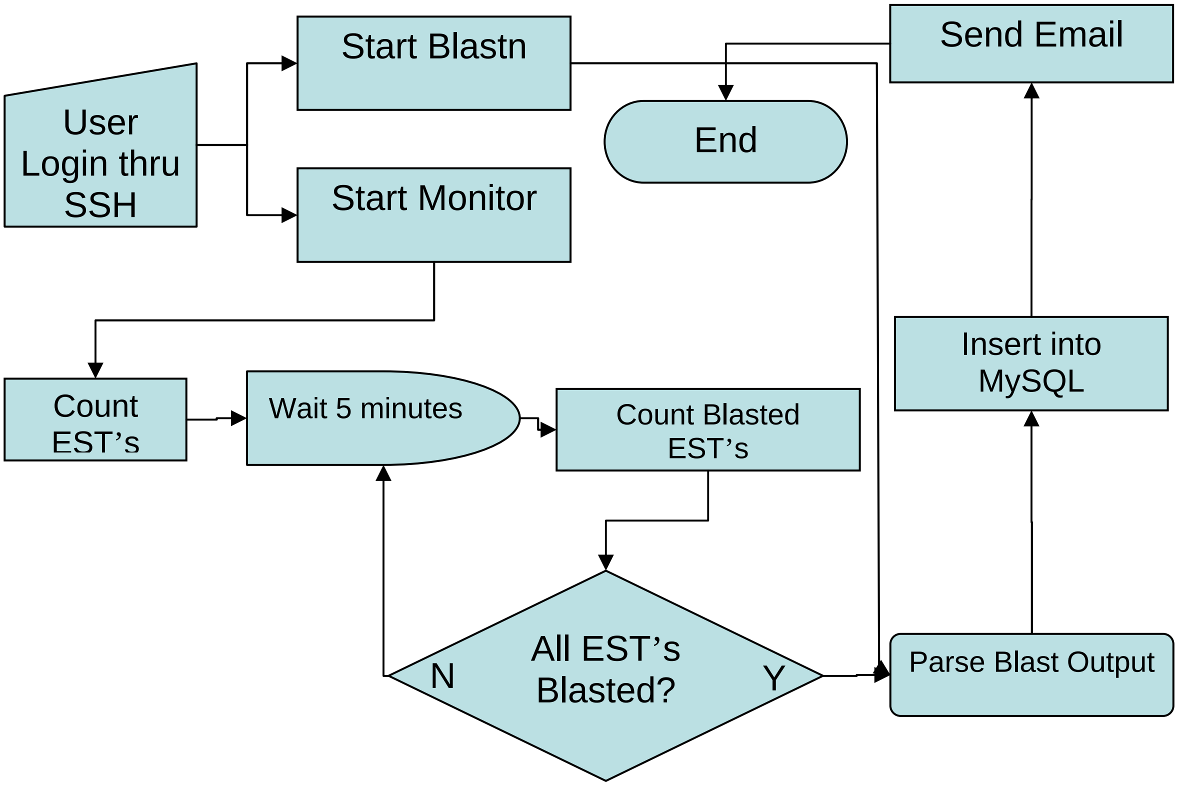
The design of the system can be broken up into three distinct modules. These modules as discussed in the purpose section of this report are as follows:

- Organize EST's according to organism
- Use Blast to align EST's to organism's genome
- Graphically display the EST's location and tissue type on a user specified region of a genome

Each of these modules is most easily explained by a flow diagram followed by a description of how the subsystem was implemented.

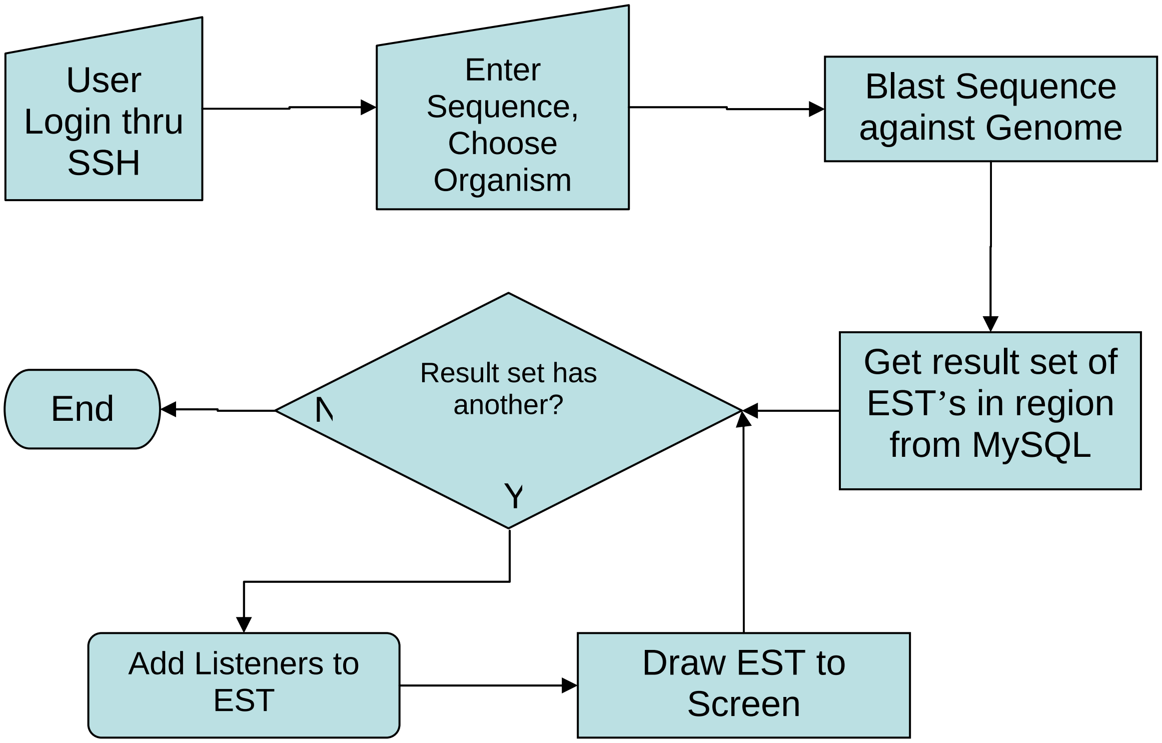

User Login thru SSH
Unzip dbEST File
Parse dbEST File
Another Sequence?
Y
N
Get next sequence
Send Email
File already exist for organism?
Y
N
End
Create a File for the organism
Insert sequence into file in Fasta Format

This module of the program essentially splits a list of alphabetically ordered GenBank sequences into a list of FASTA sequences ordered by organism. The first step is to login to the server through SSH. This allows a user to start a job on the server from any computer. After the user logs in and issues the command to split a list of EST files, each EST file is unzipped and then iterated through. If a file already exist for the organism from which the sequence comes from then the sequence will be appended to that file in FASTA format. However, if a file does not already exist for the organism from which the sequence comes, then a file will be created and named according to that organisms scientific name. When there are no longer any files to split, the program sends an email to the user notifying them that their process has complete. This is necessary because the process of splitting all EST's in the dbEST database takes approximately 10 hours to complete. This is also the reason that it is necessary to run this par of the program in the background on the server. By running the process in the background a user can start a job and then logoff the server without fear that his job will be terminated.



The second module of the program essentially "BLASTs" each EST of a given organism against that organism's genome. This process, like the one mentioned above can also take a long period of time to complete. Relatively small organisms which have a small genome and relatively few EST's can complete fairly quickly. However, other organisms can take a very long period of time. Based on data collected it is known that Yeast requires 1 hour, and it is estimated that Cow would require 1 month, and Human up to 4 months. Because the amount of time required for these processes to run it is necessary for them to run in the background of a server. This prevents the user from having to dedicate a machine to running the process.

While the "BLASTing" is being run, a separate process to monitor the BLAST program is also started in the background. This monitor essentially checks to see if every EST has been blasted yet. It only checks every 5 minutes to prevent the processor from overworking just to monitor the progress. Once the monitor finds that all EST's have been blasted, it parses the results and inserts them into a MySQL database. At this point a table called EST_Indexes contains a record for every EST that was blasted. The fields used for each EST are Genbank Identifier, Organism, Chromosome, Start, Stop, E-value. From these fields the location of each EST within the genome can quickly be located. After each sequence has been entered into the database an email is sent to the user who started the process to notify that the job is complete.

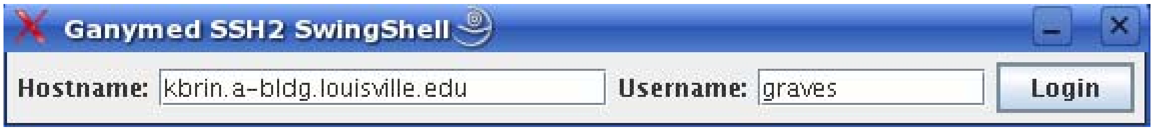

User Login thru SSH
Enter Sequence, Choose Organism
Blast Sequence against Genome
Get result set of EST’s in region from MySQL
Result set has another?
N
Y
End
Add Listeners to EST
Draw EST to Screen

The last module of the program displays the results.  It begins as the other two modules do with the user logging into the server.  After the user logs into the server, they can load a sequence in FASTA format and Blast it against the organisms genome.  This will identify a region in the genome where EST's should be located.  Using the chromosome, stop and start indexes of this region, the program queries the MySQL database and retrieves the EST's which reside in this region.  Each EST is then looked up in GenBank by its GenBank identifier and its tissue type is parsed out.  Based on the tissue type the EST is colored so that the user can easily determine the tissue type of all the EST's.  In addition a mouse listener is added to each rectangle representing an EST on the screen so that the user can obtain additional information about any given EST.  By clicking on an EST all the information in GenBank about that sequence is retrieved and displayed in a Dialog Box on the screen.

## 4- RESULTS:

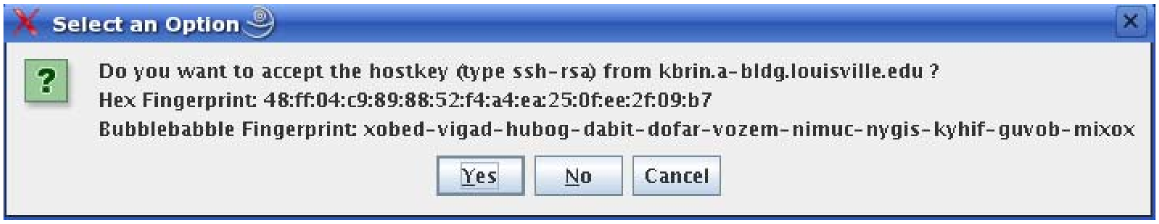
The results from the project are promising.  While there still many ways in which the application can be improved it does offer a framework which ties many concepts together.  Both Saccharomyces Crrevisiae (Yeast) and Caenorhabditis elegans (Worm) EST’s are already in the database.  With this data the program appears to be working.  EST’s are displayed in the correct regions and certain coloring schemes are performed correctly.  In addition information from GenBank is retrieved about each EST when it is clicked on.  While zooming could be handled better (only zoom in one dimension) the feature does work and is beneficial.  The following screenshots demonstrate a few of the capabilities of the application.

Login screen for the server:



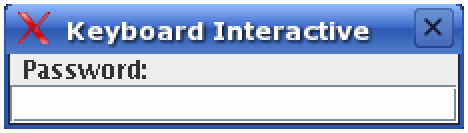
Hostname: kbrin.a-bldg.louisville.edu Username: graves

Display of any responses received from the server:

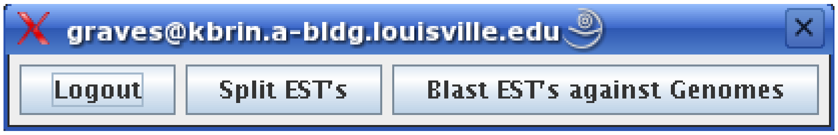


Do you want to accept the hostkey (type ssh-rsa) from kbrin.a-bldg.louisville.edu ?
Hex Fingerprint: 48:ff:04:c9:89:88:52:f4:a4:ea:25:0f:ee:2f:09:b7
Bubblebabble Fingerprint: xobed-vigad-hubog-dabit-dofar-vozem-nimuc-nygis-kyhif-guvob-mixox

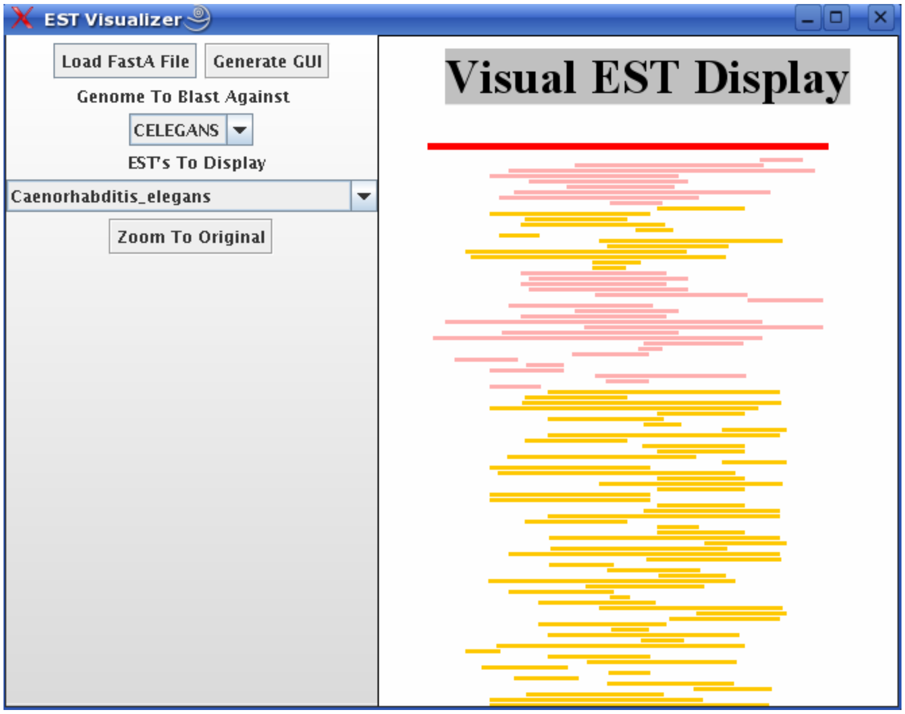
Interactive Password input:



Graphical User Interface for parsing alphabetically ordered EST's into EST's organized by organism and BLASTing each EST against the genome.

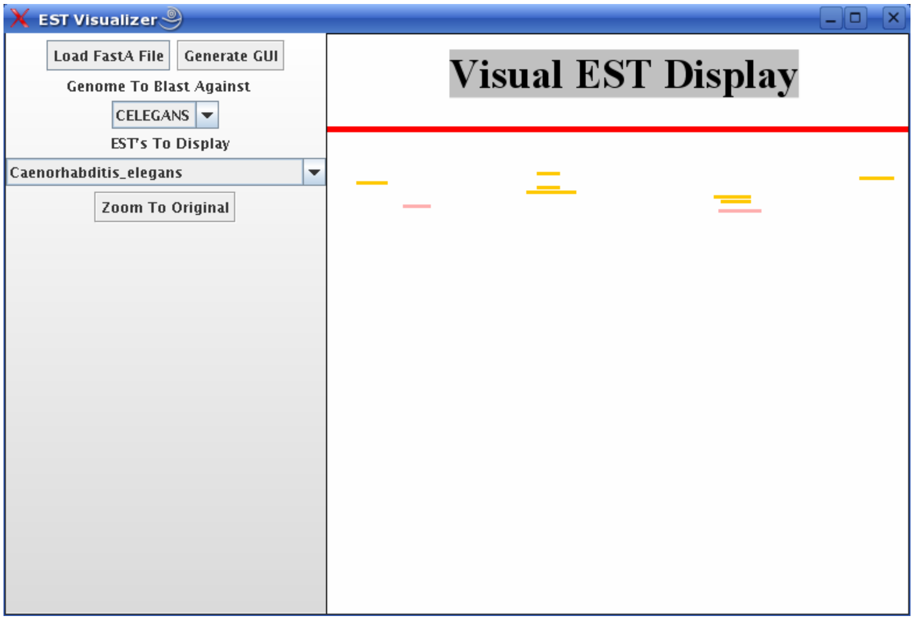

graves@kbrin.a-bldg.louisville.edu
Logout
Split EST's
Blast EST's against Genomes

Display of EST visualization portion of the program:



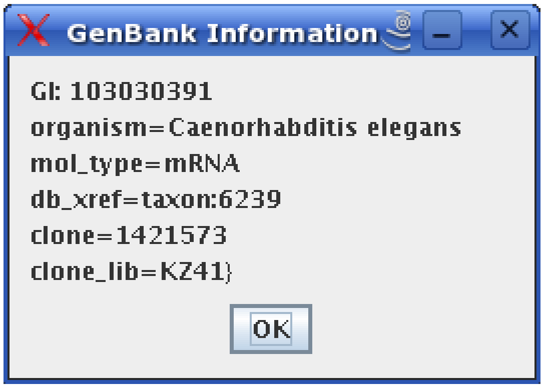
Genome To Blast Against
CELEGANS
EST's To Display
Caenorhabditis_elegans

Display of EST visualization portion of the program with different input:



Genome To Blast Against
CELEGANS
EST's To Display
Caenorhabditis_elegans

Description Dialog box which displays when an EST is clicked on:



GI: 103030391
organism=Caenorhabditis elegans
mol_type=mRNA
db_xref=taxon:6239
clone=1421573
clone_lib=KZ41}

## 5 - TESTING:

Testing the application proved to be especially difficult. The reason for this is the sheer size of data involved. Each genome can be up to 3.2 billion base pairs long and each EST file can contain up to 8 million EST's. Yeast has approximately 33,000 EST's associated with it and Worms have approximately 120,000 ESTs associated with them. Because of the large amounts of EST's, "BLASTing" them requires some time. Therefore, testing the application on many different organisms would be a lengthy process.

Another difficulty in testing is the lack of data to test on. Since the only two organisms currently available to test on are Yeast and Worms, testing certain features is difficult. Yeast EST's do not have a tissue type associated with them and Worms do not have many tissue types associated with them. Therefore testing to see if EST's are displayed in correct colors was difficult.

The application was tested for several test sequences and it appeared to display the EST's in the correct places. In addition parts 1 and 2 of the application appear to be working properly. Alphabetically organized EST's are properly split into EST's organized by organisms and each of these EST's is blasted against the appropriate genome to produce EST data which is inserted into a MySQL database. Both these features were tested over and over again to ensure that they worked for all organisms.

## 6 - APPENDIX

### 6.1 - MANUAL

The application is fairly simple to use. From the screenshots above, it is evident that the design is self explanatory. The first step when using the program is to log on to the server. The next step is to load a FASTA file that you wish to map EST's to. After this FASTA file has been selected, the organism to which the sequence belongs should be selected. Generally the same organism will be selected from which the EST's should be pulled. If information about an EST is desired then that EST should be clicked. Information from GenBank will then be displayed about that EST in separate window.

### 6.2 – DIFFICULTIES

Most of the difficulties overcome in this project were associated with the learning curve required to begin development of the application. In order to complete this project, information related to bioinformatics had to be known. Because the project was started at the beginning of the semester before the Bioinformatics class had covered many of the topics useful to this particular application, time was wasted trying to figure things out. Also, most of the files associated with bioinformatics are very large. The EST files alone are several hundred gigabytes. This alone forces the use of memory management techniques which often times do not need to be considered when programming. In addition the amount of time required to BLAST several hundred thousand EST's at once presented a problem. Without extensive knowledge of the Linux operating system, running this process in the background proved to be somewhat challenging. Running the process though JAVA’s exec command even with suppressing all output lead to undesirable results. The method employed to finally achieve the desired results required several days of testing and even it is not a particularly good design.

With more time to work on the application many of the design choices may be reconsidered. More tables would be introduced into the database to aid in storing information so that the user does not have to input as much information into the system.

### **6.3 – SUGGESTED IMPROVEMENTS**:

There are several improvements which could be made to the application. First, the application currently does not display alternative splices. This feature could be added to the application by storing not only the chromosome start and stop locations in the database for each EST but the start and stop locations for each EST as well. Then when EST’s are displayed they could be connected by a line if they have the same GenBank Identifier and if the stop location of one of the EST’s is the same as the start location of the other.

The color coordination for EST’s of a given tissue type could also be improved. Currently only a select few tissue types are color coordinated. As new tissue types are found in EST’s from organisms not currently stored in the database, colors should be assigned to them.

Filters should be added to allow the user only to view EST’s of a certain tissue type or EST’s of a certain length. These filters could be added dynamically in a toolbar on the right side of the application. This way the filters would only apply to features that exist in a given query.

Semantic zooming and one dimensional zooming could also be added to the application to increase its usability. As the user zooms in on an EST, the start and stop indexes could be displayed, the E-Value associated with the sequences hit in BLAST, or even the sequence itself.

Currently when the user opens the application it first has to query the database to determine which organisms have EST's in the database. As more and more EST's are being added to the database, this process is becoming increasingly slow. If the organisms in the database were tracked in a file instead, this would drastically improve the initialization of the application.

A table could be added to the database to link organisms with the genome they are associated with. Currently the user is responsible for choosing both the organism to BLAST the input sequence against and which EST's to map to the results. If there were a table in the database which told the application which genome to use and where to find the genome this would make the application more user friendly and reduce user errors.

## 6.4 - INPUT SEQUENCES

The following two sequences listed in FASTA format were used as input to the application to produce the screenshots presented in the results section of the report.

```
>gi|1532279|gb|C07208.1|C07208 C07208 Yuji Kohara unpublished cDNA:Strain N2 hermaphrodite
embryo Caenorhabditis elegans cDNA clone yk107f12 3', mRNA sequence
GTNTNCTTGGTTTCCAGAGAACAGNGGNGTTGGTGTTTCTCTGTGGGCATGGAACAATATTGA
GCATCTTGTCCTGGCTCTCCTTCCTCTCCTGGACTTCCAGCGGCTCCATGTGGNCCATCATCTC
CCTTCTCTCCCTGTGGTCCACGCTCTCCTGGTGGTCCAGCAATTCCAGTTGCTCCACGATCTCC
TTGCTCTCCTTGATCTCCAGACTCTCCTGGTGCTCCTGGAGTTCCCTTGGCTCCTGGAAGTCCG
ATTTGGTGTACAANATCGNCTCCTG GCTCTCCAGGAGGTCCNTCT
```

```
>gi|71997203|ref|NM_001026683.1| Caenorhabditis elegans Variable
ABnormal morphology family member (vab-10) (vab-10) mRNA, complete cds
TGTAATACCAAAACATATATAACCATCATAGTTACCCATATGGATACAAGAAATGGTTGGAGAGAGGTAT
CGTAGTCATCACGCAGTCGATCCATCGGATGCTCAGGAGAAGGAGGTGCTCGATCATTATGAGCTGAATA
GGGAGAAATATAATGATGAGCGAGACAATGTGCAGAAGAAGACGTTCACAAAATGGGTTAACAAGCACTT
GTCAAAGACGGATCACAAAATTGACGATTTATTTGTCGATTTACGGGACGGATATGCACTGATTGCGTTG
```

```
TTGGAGGCTCTTACTGGCGAGCGAATTCAAAAAGAGAACGGCTACACACGATTCCATCGGATTCAGAATG
TACAGTACTGCTTGGACTTTTTGAAAAAGAAAAATATCAAATTGGTAAACATTCGACCCGAGGACATTGT
CGAAGGAAACGGAAAACTGACACTCGGACTGATTTGGACAATTATTCTCAACTTCCAGGTCTCAGTAATC
CGCCAACGTCTCCTTTTGGAGTCGTCACAACACGAACAAATGAGTGCAAAACACACTACCACCAACTCAC
AGGTCTCCCTACACGGAAGCGATGCGACATCGGCGCGCGACGCATTGCTCCAATGGGCCAGACGGGTGAC
TGCTGGTTATCCACGTGTTAATGTGAACAACTTTTCGAGCTCATGGCGTGATGGACTCGCCTTTAATGCA
ATTCTCCATCGATACAGAAGCAGTGCTATCGATTGGAATAAGATCAGTTCGGACTCAGTTTCCAACACTG
AACGCCTGAACAACGCGTTCGCCGCGGCGGACCGCGAATTCGGAGTCGAACGTCTTCTTGATGCCGAAGA
CGTTGACACCAACAATCCAGACGAGAAGTCGATTATCACCTACGTCTCGTCACTCTACAATGCTCTTCCA
CACGAGCCGGAGATGAGCAGACTGCAAAAAGTGCAAGAAGAGTACATTGAAGAGGCCTATGAATGGCGTG
AGTGGGTGGTTCGAGCTATCCAGCTCGTCGACGATCGTCACCTCCAGGGAACTGCTTCGGAGCTCATCTA
CGAGCTTCAACGATTCCGAGAGGATGATCTACCGCCGAGAGAAGAGCAGAAGCGACGGTTGACACTTGTC
TATGAACACTTGGAGAAGGTGATGAGGTCGACGGAGCTCTTTGCGATCCCACACGAGCTGAGTGCTCCGGA
GCTTCAACGGGTGTGGAATGAGCTTCAGAACTCGATAGATCGGAGGTTTGATGTTCTCGAGAGACATCG
TATTCAGGAGGGCAACTCGAACGACCTTCTCAGCCGCCTTGCCCGCGGCATTGGCATCACCAACGAAAAG
CTCGACCTGATTCTGAAGAGAATCGAGGATGTGGAGGCTCGCGTCGACACATCACCACCGGCAGCCGTTG
AGCGTACCGTCTCGGAGATTGTCGACGATTTGAATGCTCTCGAGAGCCCGATCGCCAGATTCTTTGAAGA
CGTTGAAGAGCTGAAATCGATGCAACATCCGGAAGCCAACGATTTCTACAAACAAGTGTACGGACTTCAT
CAGCGAAGAACAACCTACCTGGATCGTCTGACGAATCAGATTCTGGTTCGCCTTGGCGTCCGAACTGACA
GTCTTCACAAGGAAAATCAGCAGAGACTCGAAAATATGAGAAAGACATCGTTCAGTCGTGTTGAGGAGTG
CATCGAGTGGGTTCGTGTTCGTATGGAGAAGCTCACGACCATGGAGTTCCTGGAGGACTTGGAAACATTG
GAGCACGTCTTTGAGCAACACAAGTTTGACAATCGAGACATCCAGGACTTCCGTCAAAATGTTGATGAAT
GCATTGCGAGACAAGCTGAAGTCTCGGCGGAGGACACTTATGAGTATTGCGAGCTTCTTCGCGTGCTCGA
GAGTGAATACCAACAACTCCGTGACCTGTCTGCCGGAAGAATGCTCGATTTGGACTCCCTCATCGCTTTT
GTCCGTGCTGCTCAACTCGAGCTCATTTGGGTTTCCGAGAGAGAATCCATTGAGGTCACCAGAAACTGGA
GTGATATCAAGCAGTTGGATCTTCCGATGCTCACCAACTACTACAAGCAGCTGCTCCACGAGATGGAACT
CCGTGAGAAGCAGTACAATGATGTGCACAATCAAGGAGCGGCTCTTCTCAATCAGGGACATCCAGCGATT
CGTGTCATCGAAGTCTATCTCCGACAAATGCAAAGTCAATGGGATTGGCTTCTTGCGTTAAGCAAGTGCC
TTGAGGAGCACCTCAGAGATGCGCTCAACCTCAAGTCCTTTATGGAGGAGGCTTCCGACGCTGAAGCCTG
GATCCAGGAGCAATCCGTCAGGCTTGAGAACAACTACAATCGGACGGACTTCTCGTTGGAAGAAGGAGAA
CGATTCTTGCGGGAGCTCGACGAGATCAAGGAGATTCTCAACAAGTATCATCAAGTGCTCATGGCTCTCA
CCGAGCGATGTGCAAGCATCTCACCACTCTGGCAGAGAGGAGAACGTATCCCGCATCCGATCAAGGTTAC
CGCACTCTGTGACTATTCCGATGAGAATGTGACGATCAAGGCTGGAGATGATGTCTACCTTCTGGACAAC
TCTGATTTGATCAAGTGGACGATTCGTGACATCTCTGGAGCCGAGGGACAAGTCCCATCAGTGGTTTTCC
GTATCCCACCGACTGATGCTCGTCTCACTGCACTCTTGAACCGACTTCTTCAACAATTCGAGAAGCTCAA
GAAGCTCTGGGACAAGAAACACAGGATGGTTCGATTCAACATGGTGCTCAACACGATGAGAACTATTCAA
GGATGGGATCTCGACACGTTCAACTCGATTGACCCTGATCAACGTGACGCTATCATGAAGGCACTGAACG
ATGATGCTAACAAACTGCTCAGCGAGCTCGATCCAAATGATCCATTGGCGTTGCGTCTTCGCGAAGAGCT
CCGAAGAACCAACGAGCACTTCTGGAACTTGCTCAATGCGAGTCAAAAGCCTCCAGAGCCAGATTGGGCTT
CTCAGTACGATCAGAAGATGGCCGAGTTGCTGAAGAAGTTGGAGGAAGCTTGGCGTGAGCTTAATGATG
CTGTCGGAAAGCCAATCTCCCGATCGCCGGAAGATCTTGAACGAGTCATTCATGCTCACAAGAGATTTGA
GGATGCTCTTCAAGCCTTGGACAGTGACGTGGCCAATGTCAAGGAGCTCTTCCGTCAACTTCCGAATCCA
ACACCAACCCAGCGTGTCAATCATGACCGCTTGAATGGGCTTTGGGATGACTTGTGGGATTTGTCAAGGA
TGTACGTCGAGCGAATCAAGGTATTGGAGTCGGTTCTGAATGGAATGGTCGAAGTCGCTGACATCGTGAG
GCAACACGAGATTACGTTGAACTCATTCGACGATCTGCCGGCTGCGCTCGATAAACTGCGAGGACACCAC
TCGCAGTTGCTCGAGATTAACATGGTGCTCAAGGTGAGCAAGTTGGGGGCTCAAAATGAGCATTTTTCAC
AACAACAAACCGTCATTGACCAACTCAACAGGAATGTTGCACTTCTCCGTCAACACGTCTCGAGAACCCG
TATTAACGAGGGACACCATCCTGACGTTGACGCCATCGAAGACGAGGTGCAAAAGCTGAATGTCCGCTGG
GAAAATGTGAACTCCCAAATAGCTTCTCGCTTGCTCGCCGTCGAAAGTGCCCTCCAAATCCAAATGGTCT
ACCGATCCGAGTACGAGACTGAAATGTCGTGGTTGGACACTGTCGAGGAGACGATCAATCGTCTGAGAAA
GCCAGAAGAGCTCCGCCCTGAGCAATACCAACAGCAACTCGACATGCTCATCGCCGAATACACAAACCTT
CAAGAGCACACTCAAGCGATTGAGCACGTGAACAAGGAGGGTGGTCGGTTCATTCATGAAGCCAAGATTT
TCGACGCGAAACTCGGACAGTACTCTGATGGAATTGTTGGAATCCACGGACCCGGTATCAAGTCGGAATT
CCGTCGTACTAAGCCACAGCCGAAGAATGGATCACAGATTGTTACTGAGGAGCTTGAGCTCCTGAACCGT
CGATTTGCTCAACTGAGCTCCCTGATCCTGGAACGCCGCAACACAATGCAAGTGCTCATCCAGAACTGGA
AACGTCAGAAGCAGGAAAATGTGACTCAAGTGGTATCGTTCCGTGAAGCTGAGGTGTCTGGCATGATGAC
GGACTTGACGAGGTTCCGACAGGAGATCTTTACGACGCATCTCACGTTCAACTCGAATCCAGAGTCAATT
```

```
GATGCGGCGACGAAGAATGTTCAGAACGTGAAGCAGTCGCTCGACTCGTGGCGTGACCGAATCAAGGAAC
GGCTGGATGAGATTGATCGGCTTTGCACAGAAGAAGGCGACTCGTTGACACCGGAGCAGTATAGTGCGTT
GAGAGAGATGAGACGGCAACTTGCCGATGAATATGATACGGTTTTGAGGACTGTCGAGGGTATTCACACG
AGACTCAACATCCTCTCCGCCTTGCTCATTGAGTTCTCATCAGTGACGTCATCAATGCAATCATGGATGA
CTGATAGAACACGTCTTGCTGGAGACATTCGTCACAAGTCGGGAGACCCGATGAGAATTGATGAAGCTCG
ATTTGAAGCCAAATCTCTGATGGATGAAGTTATTCGAGAAGAGTCGCGGCTCAAGACGATTGGAGCATCG
GTGCTCAAGATTGAGCAGGAGATTTCTGCGATGCGGGATGACGTGAGAGCCAGTGGATCGACGGATGATGT
TGGAATTTCGGTGGACGAGGTTTACGAGACGAGGCGACGAGTTGAAGATGACTACATGCAACTTCTTCG
TCAGTGTCAGGATCTTATTTCGTTCCAAAATCGTCTTCATGCAATGAATGATGAGCATTCTGAGCAAGCC
CGACGAGCAGATGAGTGGCTTCAAATGCTCCAGAATGATGTGGAGGATGTAGACCAAGACCCTAGATTTC
AGAGAGACGAGGATCGGATTCAAAGGATCGAGGAGCTGAATCGAATGGCTGCAGGTGGATCTTCACAGCT
TGATGATGCTGAGCAGGCGTCGAGGAGACTTCTGACGGCTCTTGAAGGAACGAATGTCGCGAATGATGTC
AGAGCTCGGCACGAGGAACTAGCCAACTTGAGAAGAGGGAAGCATCAGAAAGTGATTGATCGGCTATCGC
AGAATATGATGGAGGCGGCATCCCGAAAGGCTGAAGCTGAAGGAGTGAAGCAGGCGGTGGAGAACTTGAG
GCAATGGAGTGAGCAAACTGCTCAACGGACTCGGCAACCGGTGCAGCTGCCGCTCACCGAGCTGGATCTT
CATGAGGCTAGAAAGGACGAGCAGGTTCTTCACGGAGAGATTGAAAATCGCTTGGCTCTGATTGAAGAGC
TCGAGAAGAAGGCTGCAGATGTTGGAGACCACGCTTCCCTCGCCGAACTTCAGGAGTGTAAGATGAAGCT
CAAACGGAGCAATTCTGATCTCAAAGGTCTTCGAGACAACATTTTCGATGCGATCAATGGACTTCAAACT
GTCAATTCGGAAGGTGAAACTCTCTCCAGAGCCGTGGATTCTGCTGGAGCCAAGATTCGATCTGCTCGCC
TGCCAGAGGCTCAATCGGAAGTTGAAGCTCTCCAAGATCAGGCGGACAACCTTGAGAGGATCACCAATAA
CTTGTGCAACATTCCAAATGTCACCCGAACCGAGCCAGTTATCCAGAAGAGCAAGGATCTCAGGAAACGA
GTTGATTCGTGTGCTCAGGAGCTTGATGCTCGAATGGGAAAGCTCGCGGAGTTGGAATCGTTGGATGCAG
AGTTTGATGGTGCAAAGAACAAGTTGAGCTCATTTATTGGAGCATTTGATGATGAGTTGAAAGGATTGGA
GAAAGTATCAATTGATAAGGAGAAGCTCGCCGAGCAACGCCGGCAAACTCAAGATCTCGTCGATAAGCAC
TCTGAAGGAAATGCAATTCTTGATGATGTTGAGGCGATTGCGCAGAAAGTCACTGCAGAAGATCCATCAA
AGACTGGAAGTGCTCAAAAATCTGTTGGAGAACTTGGAGCACGACTTCAGCGTCAAGCCAGTGAGCTCAA
GGCTCGTGGAGATAAGATCAACAAGCTCGACTCGAAAGCCACGTCATTCGCTGAATCAGAAGCCGCTGTG
CTCGGATATATTGAGAAGCAGAAGGATCAGCTGTCGACAGGATTCCCAGTTCCAGCTACGAAAGAAGGAG
TGAAATCTCAACTTTTGGACTTGGAAAGAATGAATAAAACTGGAAAAGAAGAGCAACGACGTGTGGATGA
TGCTCGTCACTCGGCTAGAGAGCTCGCAAGAGAAGCATCTGTTGAGAAGGAGGTCCAAGATATGAATCAA
CGAGAGAAGAAGCTTCTTGACGAGTGGGAAGATTTGGCCGATCAATTCGATGCTGTAAGATCCCGAGCCA
ATAAGGCTGAGCAGGTACTCAACGAGTGTGCTCAGATGGAAAAATATATTGGTGCCAAGAAGAACATGTT
GGAAGGAATCGGAGCACCGAGCACGGAGCCCGGAGTCGCCAAGGCAAACCGTGCACAGATTCAGTCGATGA
AAGCGGAGACGGAAGGGGAGAAGTCTGCACTGGAACATGTGAACTCTCTGGCGAATGAGCTGATTGCTG
ATGGAGGAGCCAATGTTGAGGAGTTGATGAAGAAGATGGATAGGCTGAACCGGAAATGGCACTCGCTGGA
AAGTGGATTGGATGAAAATGCTGGAAGGGTGGAGGAGGCTGCGAAGTTGGGACAAGAGCTCAAGGATATT
CAGAAGGAGTTGAGAAAAGAACTCGGAGAGCTCGAATCGAATGTTGAAAAGGCGTCTGCCATGTCATCCA
ACGACATTGGAGATCAACTTGCCACCCTTGACTCTTTGAAATCCAGATTCGGAGGCGTCGATAAGGCTTT
GGAGAAGCTGAAGGGTATCCTGGAAGCTACTGAAGAGCTCGAAGTAGATGCGACGAATCGAGCTGAAATT
CAAGAGCAACTGGAAACAACTCAGAAGAAGGCTGATGAACTAGAGAGAAAGATTGAGAATGTGAAGAAGG
CGGCGTTGAACGCTCAGAATGAGGGCCTGGAGCTAGAGAAGAAGCTCGACGAGCTCATCGGAACAGTCAA
CTCGGCGGAAAATGAGCTTGAGCTGGCTGCACCGATCGCTGCGGAATCTTTGAAACTTGCGGATGAACTG
AAACGAGCTGAAGAACTATTCCAGAAGCTTATTGAAAACGAGGGAGATGTCTCTTTGATCAGAGCAAAAG
TTGCAGAGGAGCTCAAGAAGAAACCAGACGCGGAGCTCAAGAAGAAGCTGGAGTTGCTCTATCAGAAATG
GCCGAAAGCTCTCGGAGCAGCTAGAGATCGCAAGGATTTGGTGTCGAAGGCTGGAGATCTTGTGAAACAG
TTTGGCGATCAGGTGCAAGCACTCGAGCAGCGACTTCAAGGAGATCAGGCTGAACTCGATGAGCTACTGG
CTTCGGATAAGGCTCACGATCCAGAAGTTTGTGATGCTCTGAAGCTTGTGGAGTTGACGATGGCTAGACG
GCTGGCGGATGTTGATGCTCTTAACGCTGTGATGAATAGAATCGAGTCAAGTGCTCCAGGACCCGATGCG
AATCGGCTCAGAAGACGCGCTGATACACTGTCGGATGATGCGAAGGGAATGGCGAAGAAGGCTCGAACGG
CTGCAGATTTGGCACAGCGGAAGCAAGGATTAGCTAAGAAATTTGAGCGACTCTGTGACGAGGTGTCCCA
ATTTACTGAGAATCAGAAAGCCGAGATTCAGGATGCTATTGAGAAGGACCTGTTGAATGCCGAGAGAGTT
CAGAGCAAGCTCAACAAAATCGATGACTTTTGGTCTTCAAACTCCCGTGAGCTCAAGAACGTCGGTGATG
AGATCAAAATCGACGCCACCCCAGAAGATGCTCAAGCAGTTGATACTAAGCTCGCCGAGCTTCAAGCAGG
AATCGATGGGCTCCTCGCCACGCTACAGGAGCAAAATGTGCACCTGGAAGAAAAACGAGAGCAAGCAAAT
CGTGTTCAGTCGGAAAGCCAAAAAGCTGCTGGAAAGATCAACTCTTTGGTCGCAGAAATTGCGGATTTGG
ATCCAATCGGGAGAAGTCGCGATGAGCTTCAAAAGCAGAAGAAGGAAGTCGTCGAGTTGGCAGGAGATTT
GGGTTCGGCTCAGACAAAGATGTTGGAGCTCGGAGCTGAATGGGAAGCTGCTCTCGGAGCCGGAATCGTT
```

```
GCTCAGCCAGTGTTTGAGATGAATCGAGCGGCAACCGATGAGTTGAACAAGCTCGCCGCCCGTGCTGGAAA
ACGCCTTGCCCAACGCGAGAAGAAGATTACCGAGACGGAAGATGAGATTGACAAGCTTCACGCGGATGC
CGATCAGATTGTCGGTGCTCTTGAAGCAATCGCCAAGGATGAAGCCCTCCAAGGAGCCCCGTCACAGCTT
TTGGACCCGAAGCAAGTCTCTGAAAAAGTTCGACAATTGAAAGAGTCGCTGAAGCCTGTTGGAGAGAAGA
TGGATGCTTTCAACACGGACTGTAAGCTTCTGATAAAGACTGCTGGGCCTGAATCTGATACAAAAGAGCT
GGATTCTTTGCTGAAAAAGGTTGGAGACGCGTATTCGGATGTTGTTGGAAAGGTTTCCGATAAGGAGATG
AGTGTGGATGCGGCTGTTCAGCAACAGGGAAAGGTTGAAGATGCCTATCGAGCGCTGCTTAATTGGCTGG
AGGAGACGGAGGAGATGATGGAGAATCGGAAGAAACCATCTGCAGATGCAAAGGTTGCCAAGGCTCAACT
CCATGATTATGAGGTTCTGATGAAGCATGTGGAGGATAAGAAGCCCAGTGTTGATGGATTCAAAGCCATG
ATTGAGAAGATTGTCGCAGAAGCTTCCAGTGATGAGGAGAAGAAGGCACTTGGAAATAAGAATGCACAGA
TTGAAGATCGATACAAGGATCTTCTCAACTCGGCAGTGGATCGTCAACGGAAGCTGCTGGATGCAGTGGA
TCTTGCGGAACGTCTTCAAGAAGTTACGATTCCACTGGATTCCTGGCTTCAAAGCGCTGATAAACGACTT
CAGGCTCTTGCAAAAGTACCTATTACTGTTGAAAAGGCTGAGGAGATGATCGGAGAGCAGGAAGCCTTGC
AAGATGAGCTCGAGCATAAATCCGACGATCTTAAGGATGTTTTGGAGATTGCTCCGATGTTAGCCTCCCT
GGTTAGTGTTGAGGATGCCAACTCCATTAGCGGACAGGTGAACCAGTTGGAAGCACGAGCGAGAGCTCTT
GACGCAGGAATTACCAACATGAGACCACTGCTAGAGTCGTTCCTTCAACAGATTCAAGATTTTACGCTGG
ACGCTGAAGATATGACTCAATTTGTTGGAGAGACTGAGGTGAAGCTTGGCGAGCTCGATGAGCTGCCGAT
TGAGCCGGATGATTTGGTGGAGCAGACAAATATTCTTGCGGAAATTGCTGTTTCGATTGCGGATCGAGAT
GAAATGATGGCGAATATCTTTGAAGTCGGGAAGCAGCTTGCGATCCAGGGAGAACCAGAAGAAGCTCTGA
TTGCACAGAAGAAGCTTGATGATTTGAAGTTTCGATATGCTGATCTGATGACATCTGCTGATGAGAAGAT
TGCACTTCTTGCCAAGGCGATTCCATTATCGGAAGGATTCCATGAAGGATTTGATACTGTCATGCAAGTT
CTGGAGGATATGGATCGTGATTTGCAAACTATTGATGAAGAGGATCCCGAGACACAGGCTGAACTCATTT
TCCTTCTCGAAGAAGATATTTCTCAAAAAATGCGCCCATCCGTGGACGAGCTTACTGCTCTTTCCAACCA
GCTTCAAGTTCTGTGCTCTGCTGATAAGGCTGATGAGCTTCAGACCAACACGATTGCCATGAACAAGCTG
GTGAACTCGGTAGCGGATAGAGTTGCTCGGCGAGCTGAGAGAATTGAGATGGCATCGAAACAATCGAGAG
CTGTTCTGGACGATCTACAGTATCTTATTGAATGGTTCAGTGCGGCTAGAGAGCGAATTTTGGAAGGAGC
ACCGCCATCGCTGGATTTGGAAGTTCTCAAGTCACAGCTGAAACATCAAAGAATCACGAATGAAGAAGCCA
GTGCTAATAAGGTTCAGTTCAGAAATGTTGCTGGAGAAGCGAAGAAGGTTGCTCGGCAGTTGGGAATGG
AAGGAAATGAGGCGAATGAGAAGATCTCAGACACAGTTGACGAGGGAAAGGAGTTGGTTGAAGAAGTGAT
GGCTCTTTGTGCAGATCGGACGGAGACGTTGGAACGAGCTCTTGCATTGATGGAGCAGCTTACATCACAA
TTTGATGAACTCAACAAGTGGCTGGATCAGATGGATGCCGAACTTCAAGCTTCTCCATCAGTGACGACAG
CGACACCTGCTGCGGAACTCAGAGAAATGCACGATCACAATGAAGAGTTGGCACGAATGGTGGCTGCTTA
TCGACCGATTATCGAAGGATTCAAGTCTGATGTCGGATCTCTTCACGAGGTTCTTGCCGAAGATCAGGCT
CCGCTTTTGGAATCAGTTGCTGGAGAACTTGTGCAAGGATATGAAGAAGTTCGGGAGGCGGTAAGAGCAC
GTGGACATGCAATTGATAATATGATGGGTGCGACGATTGGATTCGGAGAACGACTTGAGACATTGGTGGC
GAATCTTCAAGGAGCTGCTGATCGACTCAGGGAGAATGAAGGAATATCTGCGGATCCGAGTGTTCTTGAA
TCGAGGCTTGCAGAGAATCGATCCATTGTTGAGAGTCTTCGTGATAAGCAGAATGCCTATGATGCTCTCA
AGCAGACGGCTTCAGAACTTCTTGCGTCAGCTCCAGAAGGTGATGCTGCTGCTGGGGATGTTGAGAACAA
GCTCAATCGACTCGAGAAGTTATGGAAGGAAATTGAACGAGAAGCTGTTGATCGGGGAGTTCTTTTGGAG
GATGTTCTAGACAAGGCGAAGCACTTTTGGAGTGAGCTCGATTCGTGTCAGAAGGCTGTTGATGACTTGA
GAAATCGTCTTGAGCTTGTGGAGCCCGCTACTGGGCATCCGGAGCAGTTGGCTGATCAGCAGGAGATTAT
GGCTCAAGTTGCCAGCGAGATGGAGCGAGCTCGCCCACGCATCGAAGCTCTGAGCATTGCTGGAAAACAG
CTTGCTGACTATGTTCCAGATGATGAGAAGGCTGTGATTGAGAATCAAGTGGCGAATGTTCGTGGCGGAT
TCTCGACGATTACGGGACTTTTCGCTGAGAAAAAGAGAGATTTGATTGCGGCGATGGAAGAGGCTATGAC
ATTCCATGGAGATCTTCAGGAACTTCTCAAATGGTTGGATATGGCAGAGCAGAAGCTTCTCAAGATGAGC
CCCGTAGAGCATGCCAAGCACATGACGGAGATTGAGCAATTATTGAAGGAGTTGCACACATTCAAAGACG
AGGTGCACGAAAGAGGAGTAGCCAAGGAGCAAGTCGTCGCTACAGCTCTCCAGCTCGCTGCAGATGCTCC
ACCGCACCTGGCGGCTACCGTAAGACAACCAGTCGCCGACCTGAATACTAGATGGAGCCGATTGAATGCA
GCACTTGCCGAACGAGAGCATAAGCTCGAGAATTTGATGCTCCAAATGGGAAAACTGGCGAGCACAATTG
CTCAGTTGACGGCTTGGATGGATAAGACGAGAGCCACTTTGAAGGATATTGCTCCGCCGAAGAATGCGGT
GAACTTGAGAGATATTGAAATTGCTCAATGTAAACTCGTTGTGCTCAGTAATGATATTCATGCTCATCAG
GATAGTGTGAATGCTGTGAATCGTGCTGCTCAAAAGTACATCCAAACTTCTGGAGCTCTGGACGCTGAAAC
TTCTGATTCTCTGAAATCTATGAACTTGAAATGGGAGGATATTCAGAAAGTCTTGGAATCACTGGCTTT
TGATATGGAAGTCGCTAAAAAAGAGGCTGAAAATGTTGGAGGAGAAGTCGAGAAATGGCAGAGATGGCTG
GAAGAAACTGAATCTGCTCTGCTTTCCACTAAACCAACGGGAGGATTACCGGAGACTGCTGAATTCCAGC
TCGACGAATTCAAGGCCCTGAAGCTCGACGTTGAGCACAATGCTTCGCCACTTGAAGCACATCTTCATGC
CACTGAGCAACACTTGAAGGAAGAACCCCAGGACGCTGACACATGGCTTTCCAAGACTCACGGAGCAATG
```

```
AAGACGAAGTGGAATAAGGTGAAGGAGCTGCTCGTTGATAGAGAGAAGAAGCTGCAGGTGGCTTATGAGC
AAGCGGTTGCTCTTGAGAGTGCGTTGAATGATATGGAAGATTGGATTATTGCCGCTGAACGCAAGCTCAC
CGATCAGCCATCGATCTCACGTCTCCCGGATGTAATTGAAAAGCAGCTCGCCGAGCACGAATCATGGATG
GAAGAGGTTGCTGGACGAAAGATGGCAATGACGAAACATCAAGCATCAGGAGTCCATATGCAGTATTATT
GTGAGAAGAAGGATGCCATCCCGATCAAGAATCGTCTTGTTTCCCTGAAACATCGCGTTGAAAAGATCTC
TGGAAGAACTGCAGAACGTGCGAAGCAGCTGGCTGTCACTCGAGATGAGGTTGCCACGTGGCAAGACGGG
CTTCATGATTTGGAGCATTTCATCTCGGATGTCCTTGTGAAAATTGCTCCTGAACCAAATACCACTAGCT
CACTGGAAAAGCTCAAGGCGAAGCTGGAGGAGGTGAAGGAAGCTCAGCGAGATGTGACCGCCAAGCAGAC
ACTGTTCGATGTGACTAGAAAACGTGGAATCGGACTTGCAGAGCGAGCAACTCGGTCGGAGTACAAACAA
ATTTCGATGACAAACGAGAAGATGAGCAAGAAGTGGGCTGAGATGTTGAAGAAGTTGAGAGATCGACTGA
GAGAAGCCGAGCAGGCAGTTCTGGAAGGTGGAGCATTTGAGGAGTCGATGAATGATCTGGAATCTTGGGT
TGATGATGAGCTGGAGAGATATCAGAAAGCTGAGCATGAGCCTGTATTTGCGGATATTGATGGAGTTAGA
GCACTTGTTGATGAAGAATCCAGAAGATCTGCTGAAAGAAAAACGAAGGAAAATGGAGTGAAGACGGTGG
TAAAGAAGGCAGATGCTCTGATGGCTTCAGGAGTTGATGAGAAGGATTCGATAGCTCAGGCTAAAGAACG
ACTTGTTGAGAAGTGGAATCAAGTAGAAGAGGCTGCAAGACATCGTGGAAATAGTATCAAAGAGGCTGAA
CAAGCAGCTGAAGAGTTTGATGCAAAGACGCATGCTCTGCTGGATTGGCTGGCGGTTGAAGAGCAGAAGC
TCAAGGCTTCAGGTCTGGATGAGGTGGAAGGTGTGAAGCAGGAGATGGATGAAGCCAAGGGAAGATATCA
AGAATGCTTGAAGAAGGGAGAAGAGATTCTTTCAAAGTGTCAGCCGGCTGCTGAGCCGATTCTCCGGAAT
TGGATGCGAGTCGTTGAGGCACGATGGAAGGAGGTCTCGGAGAAGGTTGACGAGCGGGAGTTTACACTTT
TGGAGCAAGAGCAGAAGGCTAAGGAGCAGAATGAGCAGATTGAGAAGCTTGCCAAGTTTGCTGCGCAGAA
GAGAGAAGAGCTCAATCGGATGATTGAGCAGCCACCGGCTCAGGATCTTGATACCATGGAGCAGAACATTT
GTGACTTTGCTAACCTCGACTCGGAGCTCCGCGAGCAACAACCCGAGGTAGACGCCGCTTGTAAATCTG
CAAAGAAGGGAGCCAGGAATCCAGCTGCAGAGATGCTCTCGACAGAATGGAAGAAGCTTTGGTTGGATGC
AATGGGTCTTCAATCATCGCTGGACAATCAAAAGGCACTACTGGAAGAGATGAAGAGGCTTGAAGGATGG
AAGTGGGAAGATTGGAAGGAGCGATATGTGGAGTGGAATGATCACGCGAAGGCACGTGTCAATGATTTGT
TCCGACGAATCGATCGACTTCACACTGGAAATGTGCCAAGACAGGTGTTCATCGATGGAATTATTGGATC
CAAATTCCCAACTTCCCGACTCGAAATGGCCAAAGTTGCTGATCGATTTGACAAGGGTGATGGAATGATC
AATGCCAAGGAATTCATCAATGCTCTCCGATTTGATGCTTCTAATAGAAACGCCAAGCCACAAACGGACA
CTGAAAAGATCACGCATGAGATTGAGCTGCAGAAGAAGACGTGCAGCTGCTGTACCCCATACCAAATTGA
GAAGATCTCTGAGAATCACTATCGGTTCGGAGACACCCATATCAAGAGAATGGTCCGCATCCTCCGATCG
ACTGTGATGGTTCGAGTCGGAGGTGGATGGGAGTCGCTGGACGAGTTCCTCCACAAACATGATCCATGTC
GTGCCAAGGGACGTCTTAACATCAACATGTTCCCAGAAGCTCGTCCGATTCATGCGCTTGACTCGATGCG
ATCTTTCACCAAGAATCGACACGGCAAGCAGCTTCCGACCACCGGAACGCCTGGTCCGATTATGAAGATT
CGCGAGAAGACCGACAGAAGTGTGCCGATGAGCGGAGGGCTCGGCGGTACAGCTGGCTATACGGTTACCA
CTGATTCGCATAGACACACTGACGCTCGCCCATCCAGAATTCCACGTGCTCCATCGGACATGAGCGCTGG
GCGCCTGAGCCGTGTCGGAAGTGTCTCCAACTCGAAAAACTCGATCGTCGACTCGTCGACACCTTCCCGA
CCAGAATCCCGAGCCTCTTCGGATGCTGGAGACCGACAAACGAGGATTCCCTCGTTGAGAGCACGTAAAG
GACAAAGATATATCCCACAAGGACCATCATCATCGTCTTCAAAGTGATTTAATTGATTCGATTTAATTTT
TTTTCCTATGTTTTTGAATGATTTTACTTTTTTTTTTCTTAATTTTTTAATATGTACCCCCACCCCTTTC
CCCCCAAAAAATTATCCAGTTTTTCCCGGCCTACAGTTGCGCGCCCCCATCACGTCATCTTGGGTTACTG
TAGCTGGGATTACTGGGCTCTTTCCACCCAAAAAAAAACTGATTTTTCTAGACACCGGTACCCCCATCTC
TCTTTTTGTGAATATCATCATTTCCGATTTTTCCCCTTCTCAATGCCTTTTTTGTTTTAATTGTATAAGA
TTAATTTCTAATATGAAATAATTTATTGAAA
```

## REFERENCES

[1] Ann E. Loraine and GreggA. Helt . Visualization Techniques for Genomic Data. Affymetrix, Inc.

[2] http://www.cs.umd.edu/hcil/jazz/

[3] http://biojava.org/wiki/Main_Page

[4] http://www.ncbi.nlm.nih.gov/

[5] http://en.wikipedia.org/wiki/Expressed_Sequence_Tag

[6] http://en.wikipedia.org/wiki/Fasta_format

[7] http://www.psc.edu/general/software/packages/seq-intro/genbankfile.html