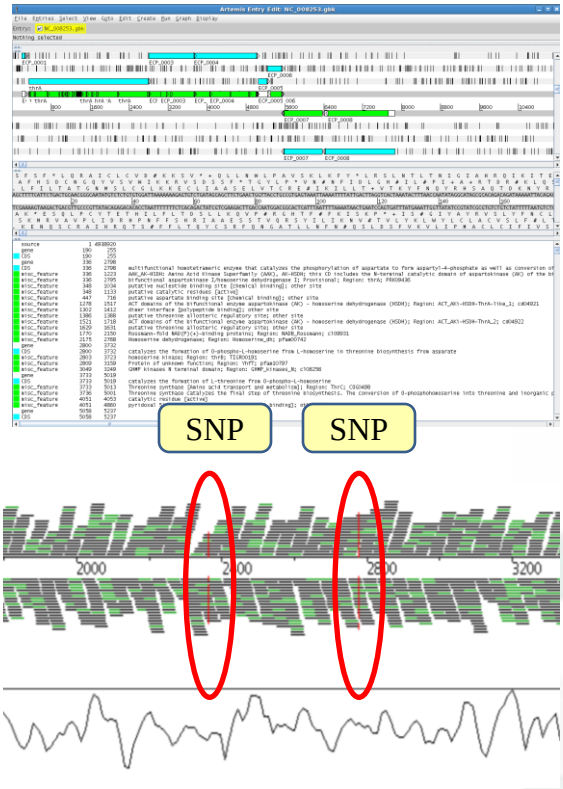
Biotechnology Information System Network (BTIS)
Department of Biotechnology
Ministry of Science & Technology, Government of India

सत्यमेव जयते

Workshop on Genome Annotation

29–30 September 2011



ICGEB INFORMATICS FACILITY

Workshop Manual

International Centre for Genetic Engineering and Biotechnology (ICGEB)
Aruna Asaf Ali Marg, New Delhi 110067 India

Workshop on Genome Annotation

29-30 September 2011

Bioinformatics Laboratory
Structural and Computational Biology Group
International Center for Genetic Engineering and Biotechnology (ICGEB)
New Delhi, India

Sponsored By:

Biotechnology Information System Network (BTIS)
Department of Biotechnology
Ministry of Science & Technology, Government of India

Staff

Principal Coordinator

Dr. Dinesh Gupta
Staff Research Scientist
E-mail: dinesh@icgeb.res.in

Guest Speaker

Dr. Mukesh Jain
Staff Scientist
E-mail: mjain@nipgr.res.in

Course Instructors

Abhinav Kaushik
Research Fellow
E-mail: abhinav@icgeb.res.in

Achal Rastogi
Research Fellow
E-mail: achal@icgeb.res.in

Sangeetha Subramaniam
Ph.D. Student
E-mail: sangi@icgeb.res.in

Zeenia Jagga
Ph.D. Student
E-mail: zeenia@icgeb.res.in

With grateful thanks, acknowledging Mayank Gupta, Anil and Babita Singh for their assistance in organizing this workshop.

Workshop Schedule

<table>
<tr><th></th><th>Thursday
September 29, 2011</th><th>Friday
September 30, 2011</th></tr>
<tr><td>10:00</td><td>Introduction of Participants</td><td rowspan="4">Guest Lecture
By Dr. Mukesh Jain (NIPGR)</td></tr>
<tr><td>10:15</td><td rowspan="3">Introduction to Genome Annotation
By Dr. Dinesh Gupta (ICGEB)</td></tr>
<tr><td>10:30</td></tr>
<tr><td>10:45</td></tr>
<tr><td>11:00</td><td>Coffee-Break</td><td rowspan="2">Coffee-Break</td></tr>
<tr><td>11:15</td><td rowspan="7"><u>Basic Module:</u> Unix, BLAST, EMBOSS
(with hands-on Session)
By Dr. Dinesh Gupta (ICGEB)</td></tr>
<tr><td>11:30</td><td rowspan="7"><u>Module 1:</u> Artemis
(with hands-on Session)
[AR]</td></tr>
<tr><td>11:45</td></tr>
<tr><td>12:00</td></tr>
<tr><td>12:15</td></tr>
<tr><td>12:30</td></tr>
<tr><td>12:45</td></tr>
<tr><td>13:00</td><td>Lunch</td><td>Lunch</td></tr>
<tr><td>14:00</td><td rowspan="4">CLC-Workbench</td><td rowspan="4"><u>Module 2:</u> Mapping Sequence Data
(with hands-on Session)
[ZJ]</td></tr>
<tr><td>14:15</td></tr>
<tr><td>14:30</td></tr>
<tr><td>14:45</td></tr>
<tr><td>15:00</td><td rowspan="2">Coffee-Break</td><td>Coffee-Break</td></tr>
<tr><td>15:15</td><td rowspan="4"><u>Module 3:</u> ACT
(with hands-on Session)
[AK]</td></tr>
<tr><td>15:30</td><td rowspan="6">CLC-Workbench continues...</td></tr>
<tr><td>15:45</td></tr>
<tr><td>16:00</td></tr>
<tr><td>16:15</td></tr>
<tr><td>16:30</td><td rowspan="3">Mop-up Session</td></tr>
<tr><td>16:45</td></tr>
<tr><td>17:00</td><td>Mop-up Session</td></tr>
</table>

Workshop on Genome Annotation

29-30 September 2011

ICGEB, New Delhi, India

The two-day workshop aims to give researchers with a working knowledge of computational sequence analysis, a firm grounding in the use of the latest genome analysis software (Artemis and ACT) developed at the Wellcome Trust Institute Pathogen Sequencing Unit (PSU) and an insight into in-silico Next Generation Sequence (NGS) data analysis.

Artemis is a powerful annotation tool and DNA viewer that allows the user to analyze sequence data from databases such as EMBL or Genbank. ACT is a comparative genomic tool that allows direct, and interactive, comparisons of multiple genomes/read sequences. This enables the user to exploit the growing number of genomes and NGS data from closely related organisms to look at genome architecture and evolution.

The course will be taught my members of the Bioinformatics Laboratory, Structural and Computational Biology Group and will take the form of a series of modules covering most aspects of sequence analysis and exploitation. Each module will be introduced with a short talk followed by ‘hands on’, to illustrate points in whole genome analysis.

Module 1 Artemis

(using *S. typhi*)

Introduction

Artemis is a free DNA viewer and annotation tool written by Kim Rutherford (Rutherford *et al*., 2000). It is routinely used by the Sanger Institute Pathogen Sequencing Unit for annotation and analysis of both prokaryotic and eukaryotic genomes. The program allows the user to view simple sequence files, EMBL/Genbank entries and the results of sequence analyses in a highly interactive and intuitive graphical format. Artemis is designed to present multiple sets/types of information within a single context. This manifests itself as the ability to zoom in to inspect DNA sequence motifs and zoom out to view local gene architecture (e.g. operons), several kilobases of a genome or even an entire genome in one screen. It is also possible to perform some analyses within Artemis with the output stored for later access.

Aims

The aim of this part of the Module is for you to become familiar with the basic functions of Artemis using a series of worked examples. These examples are designed to take you through the most immediately useful functions. However, there will be time, and encouragement, for you to explore other menus; nooks and crannies of Artemis that are not featured in the exercises in this manual. Like all the Modules in this workshop, the key is ‘if you don’t understand please ask’.

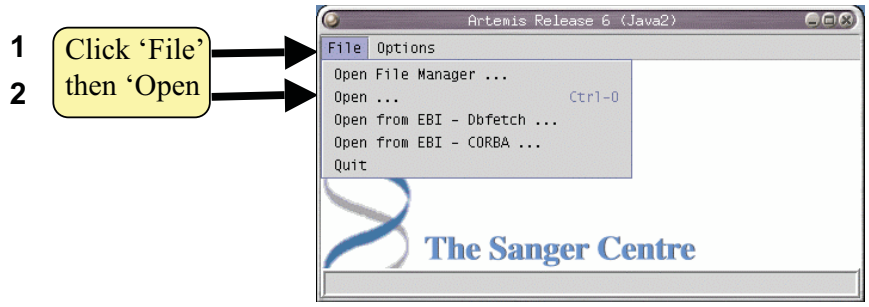
Artemis Exercise 1 Part I

1. Starting up the Artemis software

Navigate your way into the correct directory for this module
Then type:

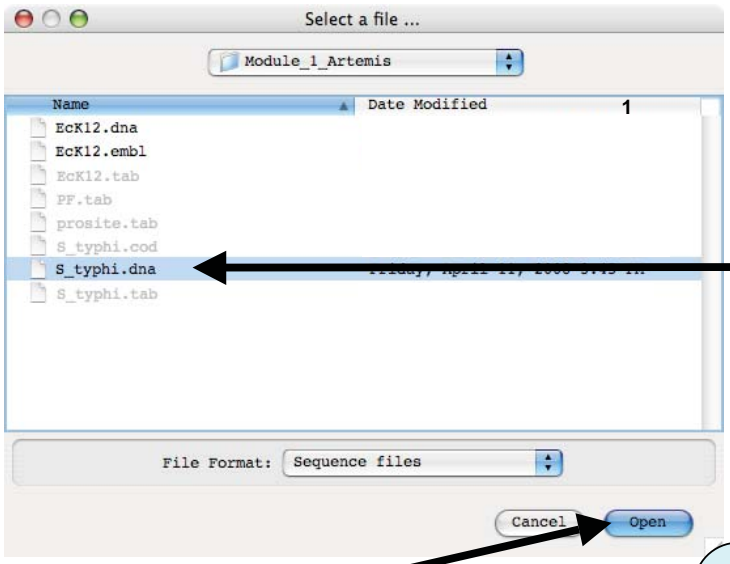
```
art & [return]
```

A small start-up window will appear (see below).
Now follow the sequence of numbers to load up the *Salmonella* Typhi chromosome sequence.
Ask a demonstrator for help if you have any problems.



For simplicity it is a good idea to open a new start up window for each Artemis session and close down any sessions once you have finished an exercise.

In the 'Options' menu you can switch between prokaryotic and eukaryotic mode.



3

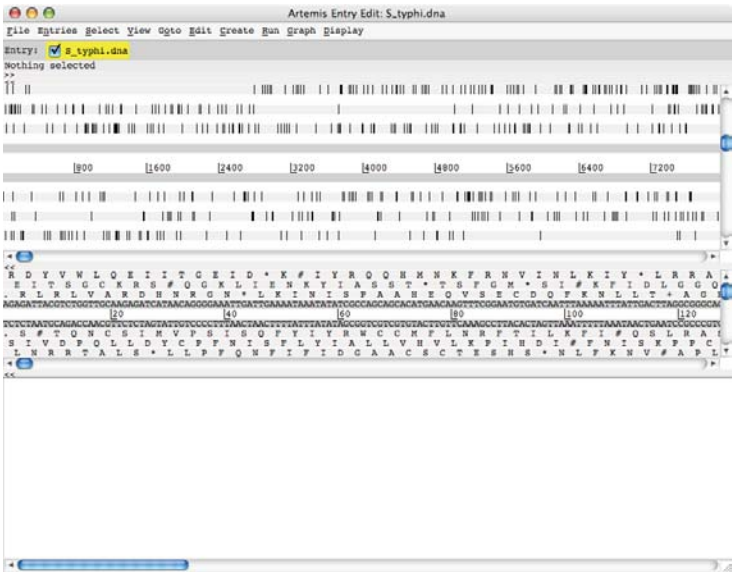
Single click to select DNA file

4 Single click to open file in Artemis then wait

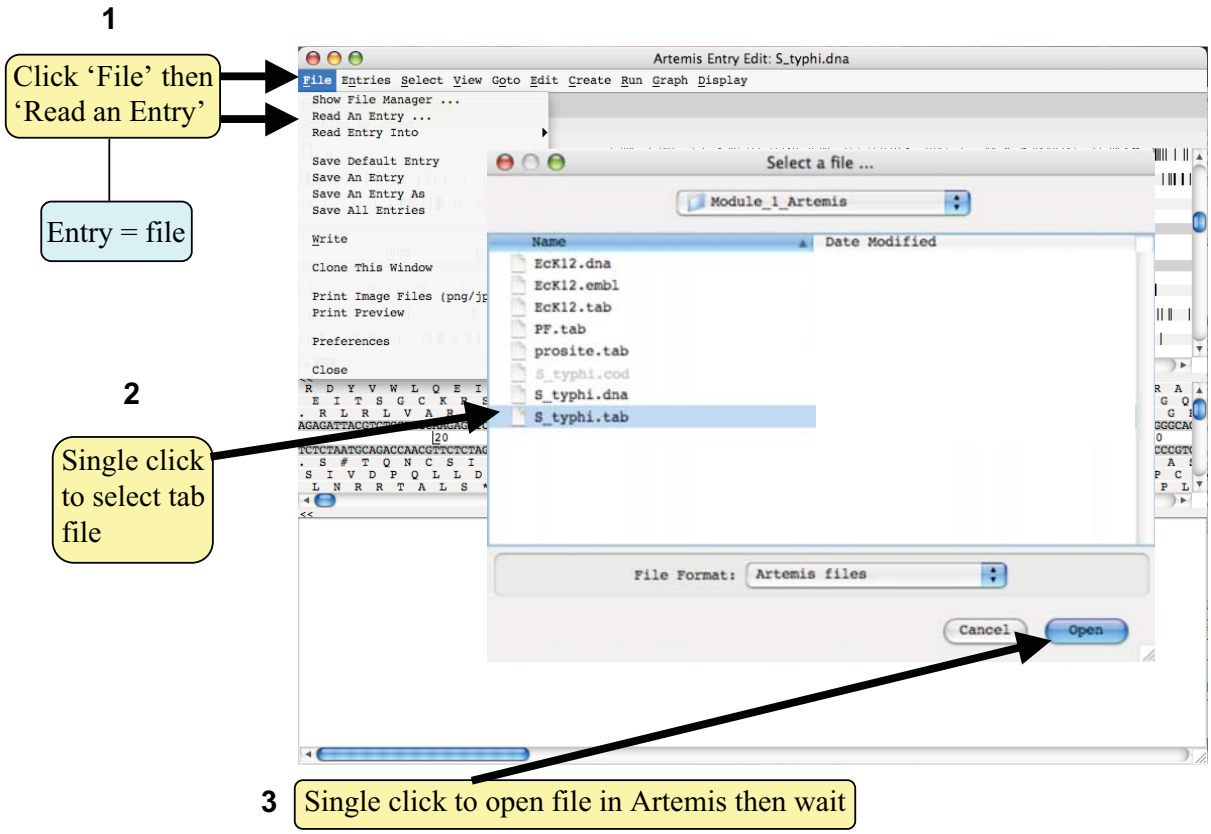
DNA sequence files will have the suffix '.dna'. Annotation files end with '.tab'. Use this feature to select the type of file displayed in this panel.

2. Loading annotation files (entries) into Artemis

Hopefully you will now have an Artemis window like this! If not, ask a demonstrator for assistance.



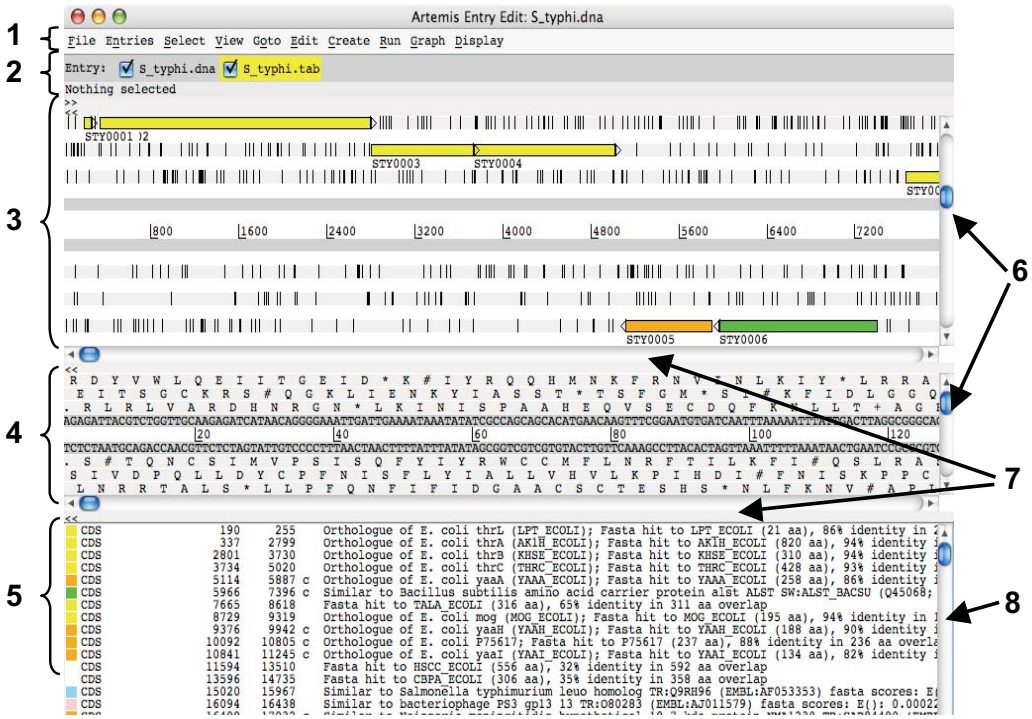
Now follow the numbers to load up the annotation file for the *Salmonella* Typhi chromosome.



What's an "Entry"? It's a file of DNA and/or features which can be overlaid onto the sequence information displayed in the main Artemis view panel.

3. The basics of Artemis

Now you have an Artemis window open let's look at what's in there.



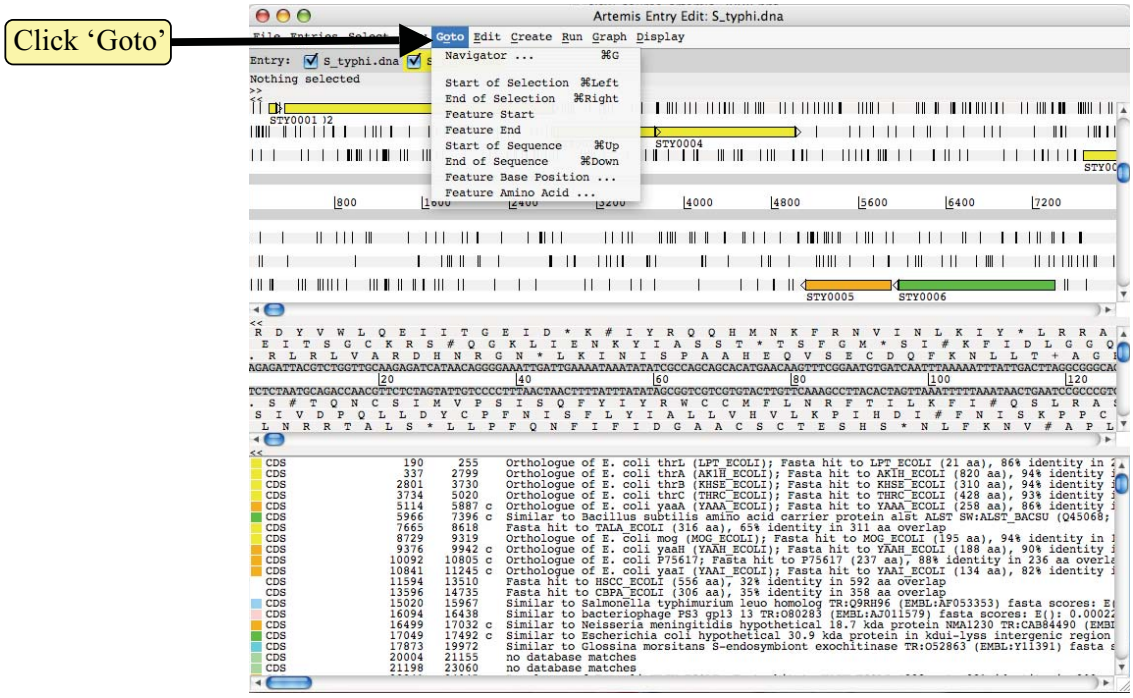
1. Drop-down menus. There's lots in there so don't worry about them right now.
2. Shows what entries are currently loaded (bottom line) and gives details regarding the feature selected in the window below; in this case gene STY0003 (top line).
3. This is the main sequence view panel. The central 2 grey lines represent the forward (top) and reverse (bottom) DNA strands. Above and below those are the 3 forward and 3 reverse reading frames. Stop codons are marked as black vertical bars. Genes and other features (eg. Pfam and Prosite matches) are displayed as coloured boxes. We will refer to genes as coding sequences or CDSs from now on.
4. This panel has a similar layout to the main panel but is zoomed in to show nucleotides and amino acids. Double click on a gene in the main view to see the zoomed view of the start of that gene. Note that both this and the main panel can be scrolled left and right (7, below) zoomed in and out (6, below).
5. This panel lists the various features in the order that they occur on the DNA with the selected gene highlighted. The list can be scrolled (8, below).
6. Sliders for zooming view panels.
7. Sliders for scrolling along the DNA.
8. Slider for scrolling feature list.

4. Getting around in Artemis

The 3 main ways of getting to where you want to be in Artemis are the Goto drop-down menu, the Navigator and the Feature Selector. The best method depends on what you're trying to do and knowing which one to use comes with practice.

4.1 The 'Goto'menu

The functions on this menu (ignore the Navigator for now) are shortcuts for getting to locations within a selected feature or for jumping to the start or end of the DNA sequence. This one's really intuitive so give it a try!



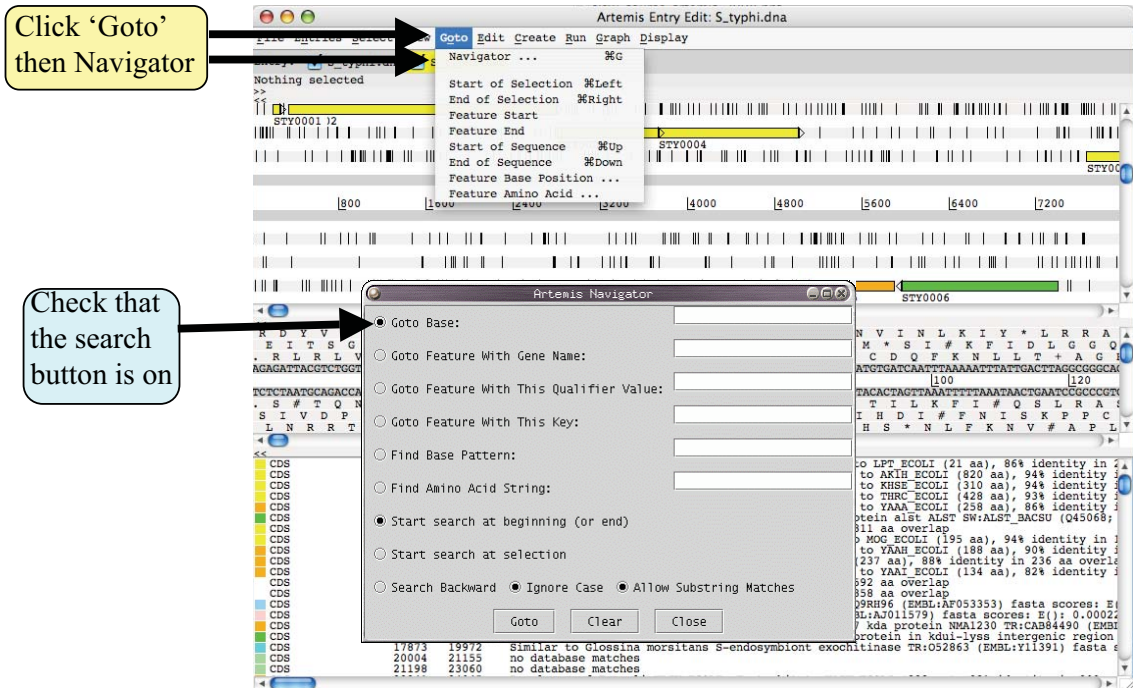
It may seem that 'Goto' 'Start of Selection' and 'Goto' 'Feature Start' do the same thing. Well they do if you have a feature selected but 'Goto' 'Start of Selection' will also work for a region which you have highlighted by click-dragging in the main window. So yes, give it a try!

Suggested tasks:

1. Zoom out, highlight a large region of sequence by clicking the left hand button and dragging the cursor then go to the start and end of the highlighted region.
2. Select a gene then go to the start and end.
3. Go to the start and end of the genome sequence.
4. Select a gene. Within it, go to a base (nucleotide) and/or amino acid of your choice.

4.2 Navigator

The Navigator panel is fairly intuitive so open it up and give it a try.



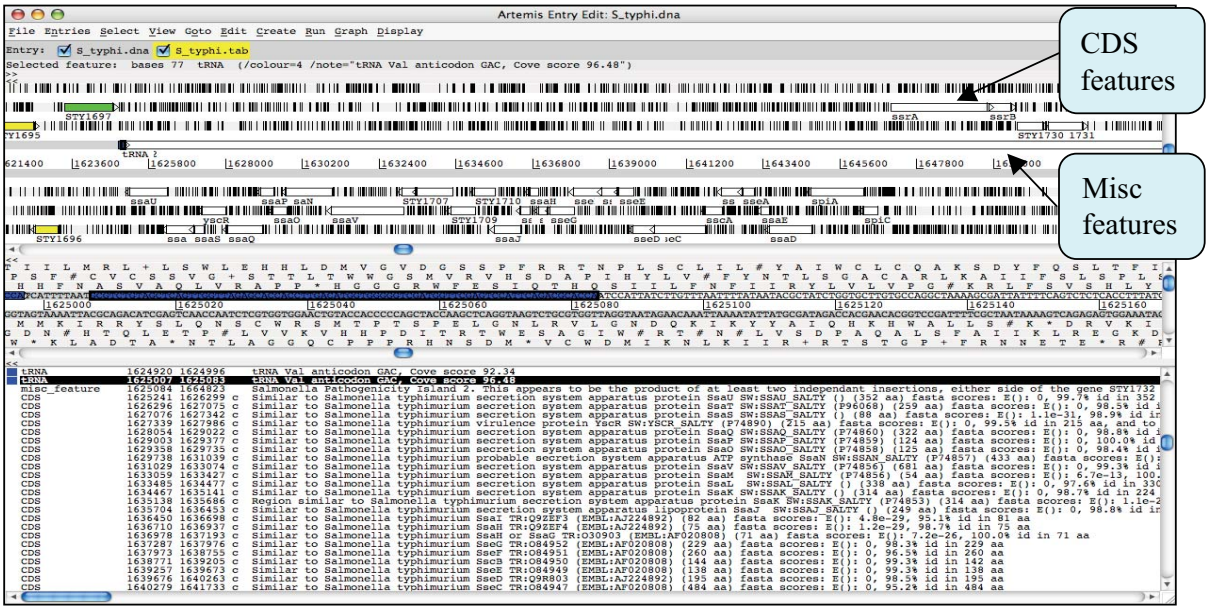
Suggestions of where to go:

1. Think of a number between 1 and 4809037 and go to that base (notice how the cursors on the horizontal sliders move with you).
2. Your favourite gene name (it may not be there so you could try 'fts').
3. Use 'Goto Feature With This Qualifier value' to search the contents of all qualifiers for a particular term. For example using the word 'pseudogene' will take you to the next feature with the word 'pseudogene' in any of its qualifiers. Note how repeated clicking of the 'Goto' button takes you through the pseudogenes as they occur on the chromsome.
4. tRNA genes. Type 'tRNA' in the 'Goto Feature With This Key'.
5. Regulator-binding DNA consensus sequence (real or made up!). Note that degenerate base values can be used (Appendix III).
6. Amino acid consensus sequences (real or made up!). You can use 'X's. Note that it searches all six reading frames regardless of whether the amino acids are encoded or not.

Clearly there are many more features of Artemis which we will not have time to explain in detail. Before getting on with this next section it might be worth browsing the menus. Hopefully you will find most of them easy to understand.

Artemis Exercise 1 Part II

This part of the exercise uses the files and data you already have loaded into Artemis from Part I. By a method of your choice go to the region located between bases 1625084 to 1664823 on the DNA sequence. This region is bordered by *ribE* gene which codes for riboflavin synthase alpha chain. You can use either the Navigator, Feature Selector or Goto functions discussed previously to get there. The region you arrive at should look similar to that shown below.



Most of you will recognise this region as Salmonella Pathogenicity Island 2 (SPI-2)

Once you have found this region have a look at some of the information that is available to you:-

Information to view:

Annotation

If you click on a particular feature you can view the annotation attached to it:
select a CDS feature (or any other feature) and click on the Edit menu and select Edit Selected Feature. A window will appear containing all the annotation that is associated with that CDS. The format for this information is constrained by that which can be submitted to the EMBL database where it is stored within 'keys' and 'Qualifiers' see Appendix II.

Viewing amino acid or protein sequence

Click on the view menu and you will see various options for viewing the bases or amino acids of the feature you have selected, in two formats i.e. EMBL or FASTA. This can be very useful when using other programs that are not integrated into Artemis e.g. those available on the Web that require you to cut and paste sequence into them.

Plots/Graphs

Feature plots can be displayed by selecting a CDS feature then clicking 'View' and 'Show Feature Plots'. The window which appears shows plots predicting hydrophobicity, hydrophilicity and coiled-coil regions for the protein product of the selected CDS.

Load additional files

The results from Prosite searches run on the translation of each CDS should already be on display as pale-green boxes on the grey DNA lines. The results from the Pfam protein motif searches are not shown, but can be viewed by loading the appropriate file. Click on 'File' then 'Read an Entry' and select the file PF.tab. Each Pfam match will appear as a coloured blue feature in the main display panel on the grey DNA lines. To see the details click the feature then click 'View' then 'View Selection' or click 'Edit' then Edit Selected Features'.
Please ask if you are unsure about Prosite and Pfam.

Further information on specific Prosite or Pfam entries can be found on the web at http://ca.expasy.org/prosite and http://www.sanger.ac.uk/software/Pfam/tsearch.shtml

In addition to looking at the fine detail of the annotated features it is also possible to look at the characteristics of the DNA covering the region displayed. This can be done by adding in to the display various plots showing different characteristics of the DNA. This information is generated dynamically by Artemis and although this is a relatively speedy exercise for a small region of DNA, on a whole genome view (we will move onto this later) this may take a little time so be patient.

To view the graphs:

Click on the 'Graph' menu to see all those available. Perhaps some of the most useful plots are the 'GC Content (%)' (**1**) 'GC Deviation' (**2**) and 'Karlin signature plots' (**3**) as shown below. To adjust the smoothing of the graph you change the window size over which the points on the graph are calculated, using the sliders shown below. If you are not familiar with any of these please ask.

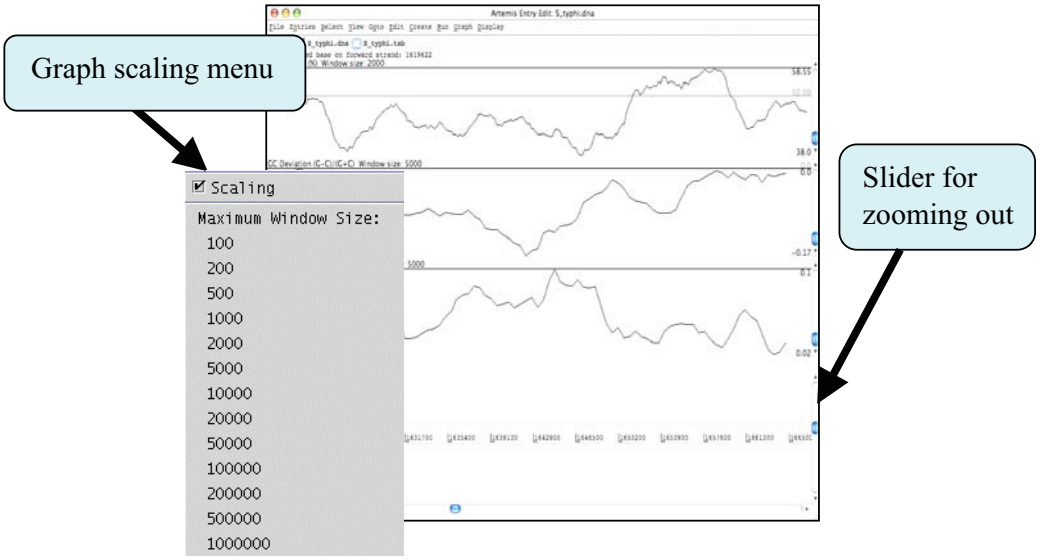
Notice how several of the plots show a marked deviation around the region you are currently looking at. To fully appreciate how anomalous this region is move the genome view by scrolling to the left and right of this region. Notice also that the nucleotide profile of SPI-2 appears to split the region into two segments.

As well as looking at the characteristics of small regions of the genome, it is possible to zoom out and look at the characteristics of the genome as a whole. To view the entire genome use the sliders indicated below. However, be careful zooming out quickly with all the features being displayed, as this may temporarily lock up the computer. To make this process faster, and clearer, switch off stop codons by clicking with the right mouse button in the main view panel. A menu will appear with an option to de-select stop codons (see below). If you have any problems ask a demonstrator.

To de-select the annotation click here.

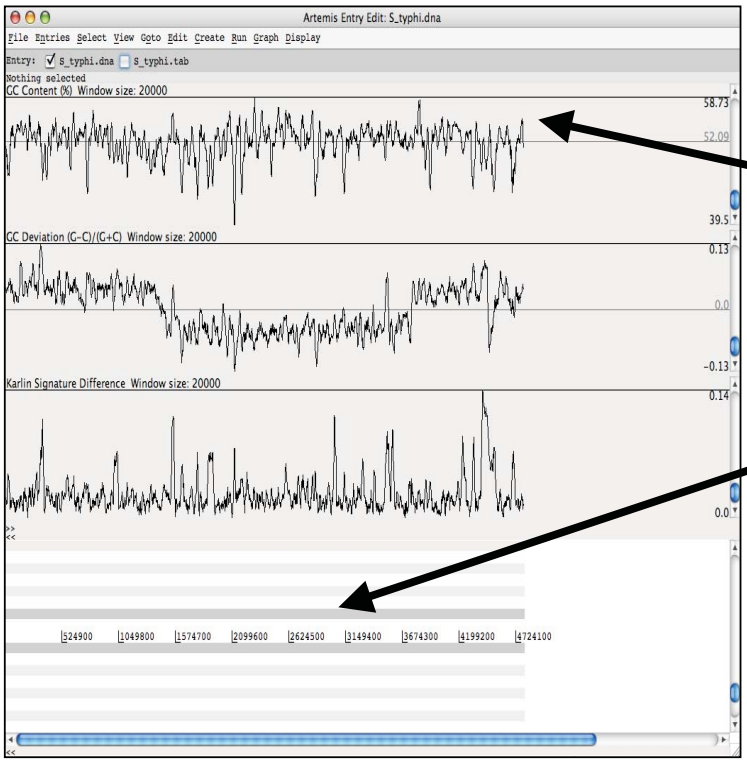
You will also need to temporarily remove all of the annotated features from the Artemis display window. In fact if you leave them on, which you can, they would be too small to see when you zoomed out to display the entire genome. To remove the annotation click on the S_typhi.tab entry button on the grey entry line of the Artemis window shown above.

Your Artemis window should now look similar to the one shown below.



One final tip is to adjust the scaling for each graph displayed before zooming out. This increases the maximum window size over which a single point for each plot is calculated.
To adjust the scaling click with the right mouse button over a particular graph window. A menu will appear with a series of values for the maximum window size (see above), select 20000. You should do this for each graph displayed.

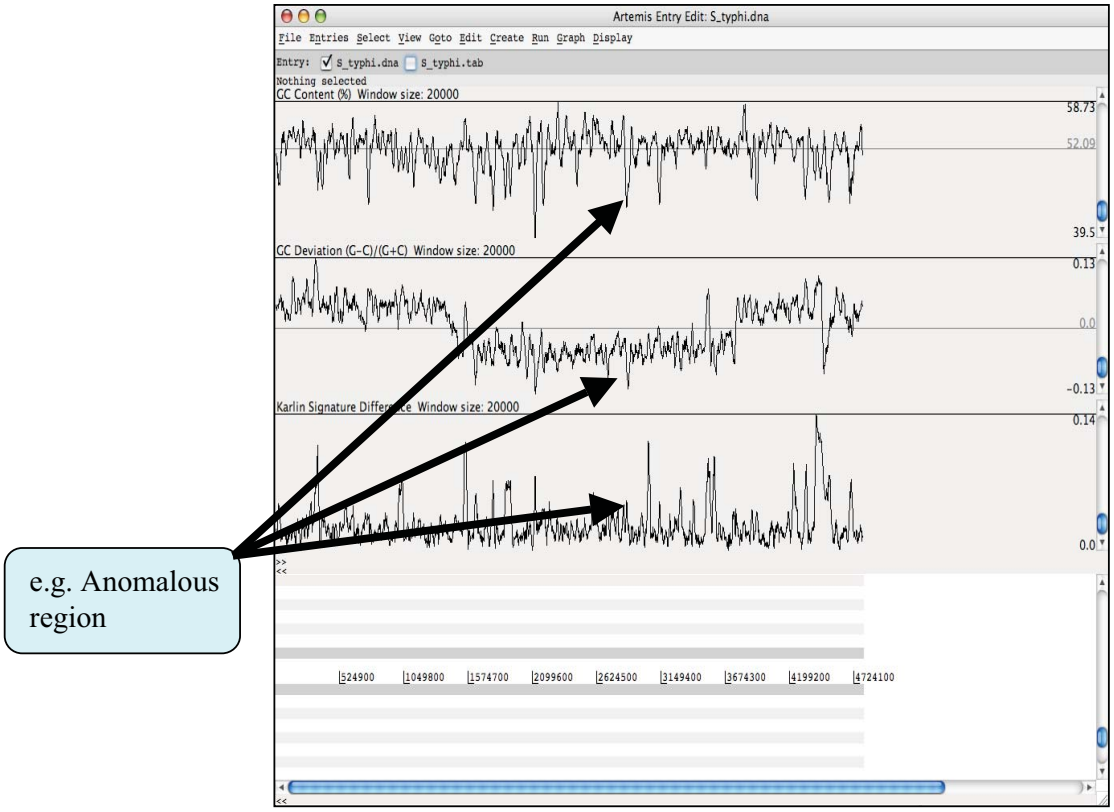
You are now ready to zoom out by dragging or clicking the slider indicated above.
Once you have zoomed out fully to see the entire genome you will need to adjust the smoothing of the graphs using the vertical graph sliders as before to have a similar view to that shown below.



Click with the left mouse button in a graph window. A line and a number will appear. The number is the relative position within the genome (bps).

Click and drag to highlight a region on the main DNA line. Notice that the boundaries of this region should now be marked in the graph windows that your previously clicked in.

Artemis Exercise 1 Part III



The graphs can be used to look at other regions within the genome that stand out by having an atypical G+C content or Karlin signature (di-nucleotide frequency). You will see from the whole genome view of *S.* Typhi that there are many other examples of anomalous regions of DNA within a genome, many of which will have been laterally acquired. Since it has been shown in many bacteria that laterally acquired DNA carries important genes for virulence and lifestyle it is worth spending a little time exploring some of these regions. You should identify a region which you feel is interesting based on the graphs and zoom into look at the genes encoded within this region (one example is shown above).

So firstly zoom back into the genome to look in more detail at the first of these three peaks. Zoom into this position by first clicking on the DNA line at approximately the correct location. If you then use the vertical side slider to zoom back in, Artemis will go to the location you selected. Remember that in order to see the CDS features lying within this region you will need to turn the annotation (*S*_typhi.tab) entry back on.

It is also worth looking for other markers of lateral acquisition such as integrase genes and tRNA integration sites.

If you have time, and as a cautionary note, you should also go and have a look at Karlin and G+C plots for the region centred around genome position 4231500.

Module 2
Mapping Sequence Data

Introduction

Improvements in DNA sequencing technology have lead to new opportunities for the studying organism at the genomic and transcriptomic levels. Applications include studies of the genomic variation within species and gene identification. In this module we will be using simulated data from *E.coli* genome, although all though the techniques you will learn are applicable all the technologies (e.g., Illumina Genome Analyzer, 454 GS FLX and ABI SOLiD). A single machine can produce around 20 Gigabases of sequence data from the Illumina machine comes as relatively short stretches of 35-100 base pairs (bp) of DNA- around 300 million of them. These individual sequences are called sequencing reads. The older capillary sequencing methods produces longer reads of ~500bp, but is much slower and more expensive.

One of the greatest challenges of sequencing a genome is determining how to arrange sequencing reads into chromosomes. This process of determining how the reads fit together by looking for the overlaps between them is called **genome assembly.** Capillary sequencing reads (~500bp) are considered long enough for the genome assembly. Genome assembly using sequence reads of >100bp is not possible in many cases due to the high frequency of repeats longer than the read length. Therefore, new sequencing technologies are mostly used where a **reference genome** already exists is called **resequencing**.

When resequencing, instead of assembling the reads to produce a new genome sequence and then comparing the two genomes sequences, we map the new sequence data to the reference genome. We can then identify **Single Nucleotide Polymorphism (SNPs)**, insertions and deletions (indels) and Copy Number Variants (CNVs) between two similar organisms.

In this module

The example used in this module is a set of 1000, 35bp simulated reads from throughout the *E.coli* genome and for finding SNPs is a set of simulated reads to cover 10,000 bases of the *E.coli* genome.

Module summary

A. File formats
B. Making index files
C. Mapping the data
D. Converting bowtie output to BAM
E. Visualizing the mapped reads in Artemis
F. Identifying the Single Nucleotide Polymorphism

A. File formats

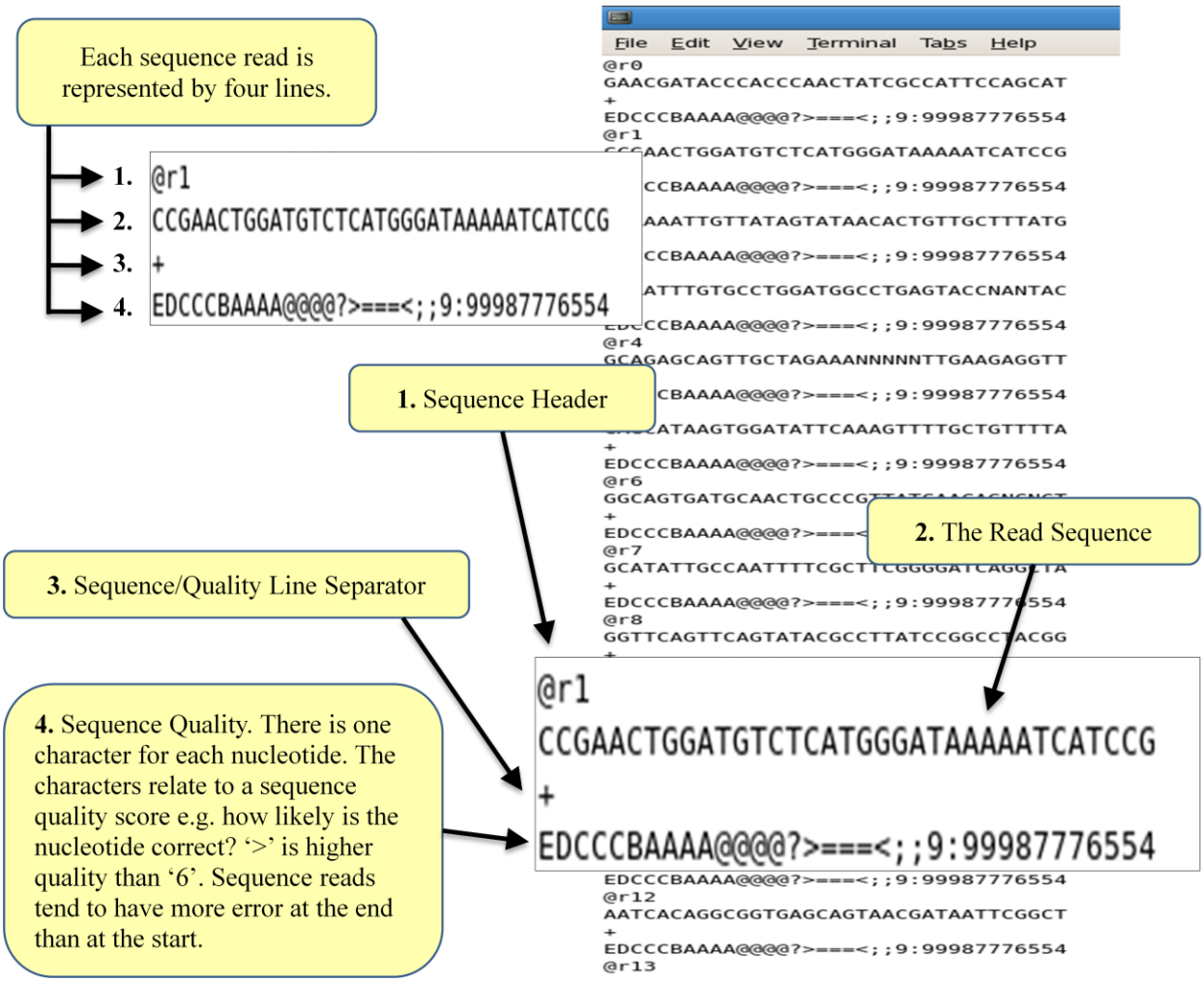
You have the reference file of *E.coli* strain 536 (NC_008253.fna), a strain known to cause urinary tract infections. You also have two files of sequence reads simulated from that genome (a set of 1000, 35-bp reads). Look in both the reference file and read files.

Open up a terminal and navigate to Module_2_MSD directory, then type:

```
$ cd reads/
$ head NC_008253.fna
```

Compare the reference file above to the files of sequencing reads:

```
$ more e_coli_1000.fastq
```



B. Making index files

Now we will make index from the *E.coli* genome sequence for the strain 536, which is known to cause urinary tract infections using short reading mapping program Bowtie (Langmead *et al.*, 2009).

Bowtie is an ultra fast, memory efficient short read aligner for aligning large sets of short DNA sequences (reads) to large genome. Bowtie indexes the genome with a burrows-wheeler index to keep its memory footprint small.

Navigate to module 3 reference genome directory, then type:

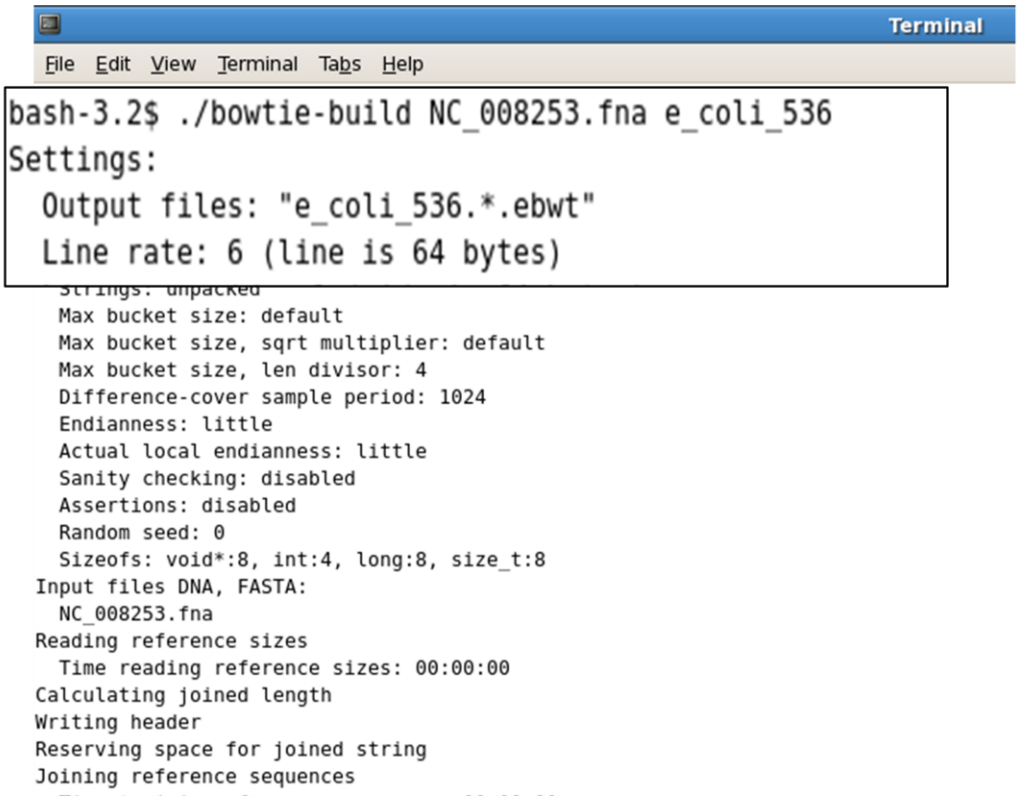
```
$./bowtie-build NC_008253.fna e_coli_536
```

The command should finish quickly , and print several lines of status messages. When command has completed, note that current directory contains 4 new files named *e_coli_536.1.ebwt, e_coli_536.2.ebwt, e_coli_536.3.ebwt, e_coli_536.4.ebwt, e_coli_536.1.rev.1.ebwt and e_coli_536.1.rev.2.ebwt*.

These files constitute the index. Move these files to the indexes subdirectory to install it. To test that the index is installed properly, on the terminal type:

```
$./bowtie -c e_coli_536 GCCTGAGCTATGAGAAAGCGCCACGCTTCC
```

If the index is installed properly, this command should print a single alignment and then exit.



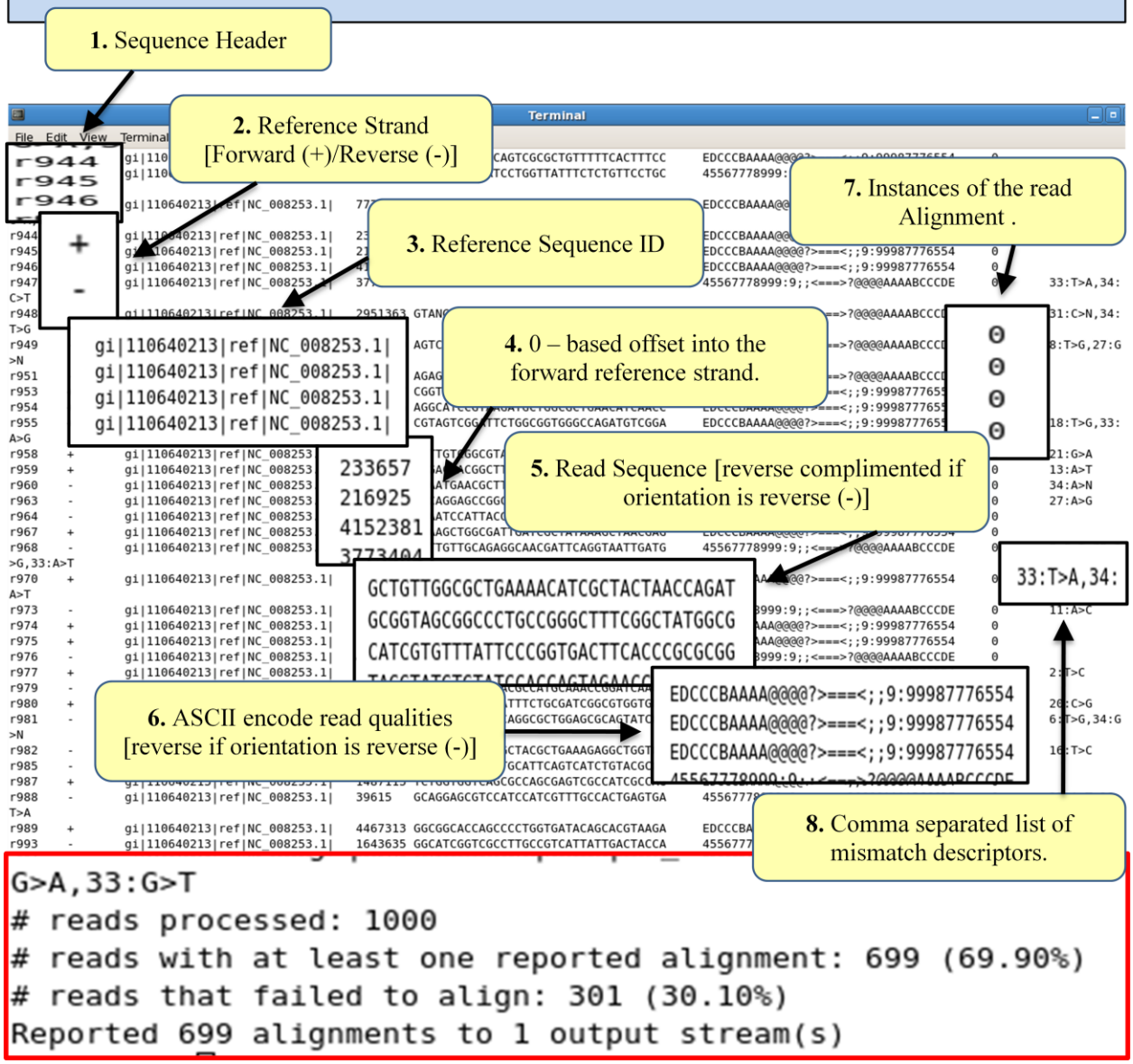
C. Mapping the data

Now we will map reads to the *E.coli* reference genome using Bowtie (Langmead *et al.*, 2009).

On the command line of the aligned read appears in the left , type:

```
$ ./bowtie reference_genome/indexes/e_coli_536 reads/e_coli_1000.fq
```

The first argument to bowtie is the base name of the index for the genome to be searched. Second argument is the name of FASTQ files containing the reads. You will see bowtie print many lines of output .
Each line is an alignment for the read.



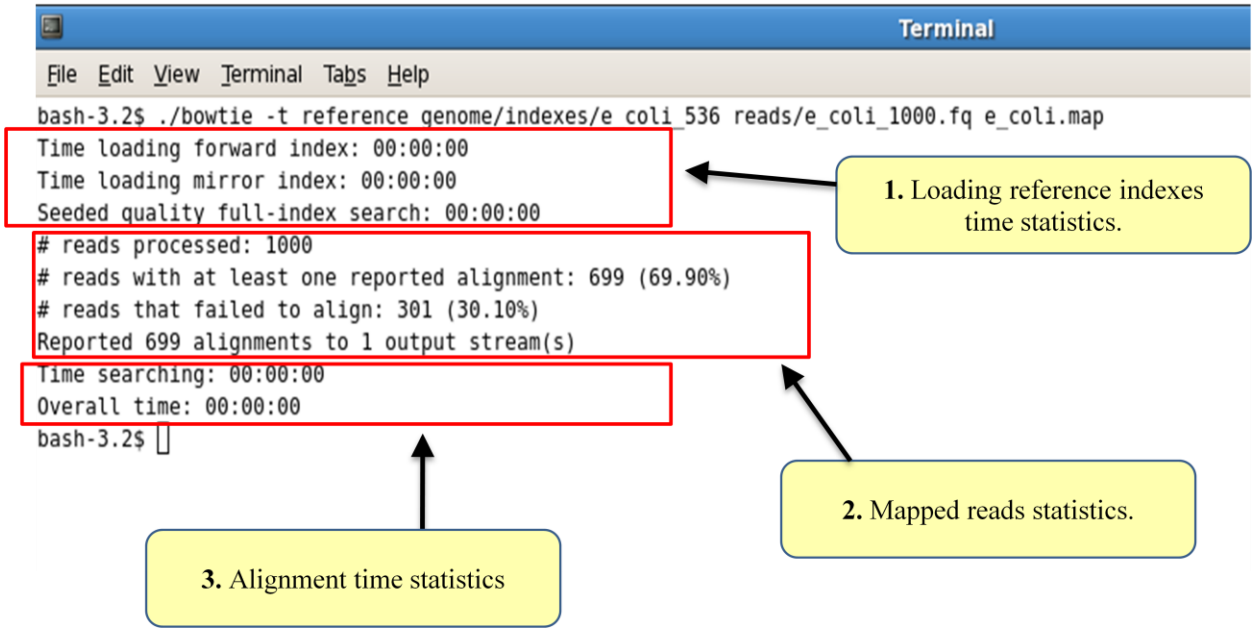
C. Mapping the data..continued…

Next, on the command line type :

```
$ ./bowtie -t reference_genome/indexes/e_coli_536 reads/e_coli_1000.fq
e_coli.map
```

This run calculates the same alignments as the previous run, but the alignments are written to *e_coli.map*(the final argument) rather than to screen.

–t option instructs Bowtie to the print timing statistics.



D. Converting Bowtie output to BAM

We are going to view the map reads in Artemis using Artemis BAM view. However the mapping result is not currently in BAM format. Therefore, we will convert Bowtie output to BAM using SAMtools (Li *et al.*, 2009).

SAM tools is a suit of tools for storing, manipulating, and analyzing alignment such as those output by Bowtie.SAM understands alignment in either of the two complementary formats: the human readable SAM format, or the binary BAM format.

Bowtie can output SAM (using `-S/--sam` option), and SAM can be converted to BAM using SAMtools .

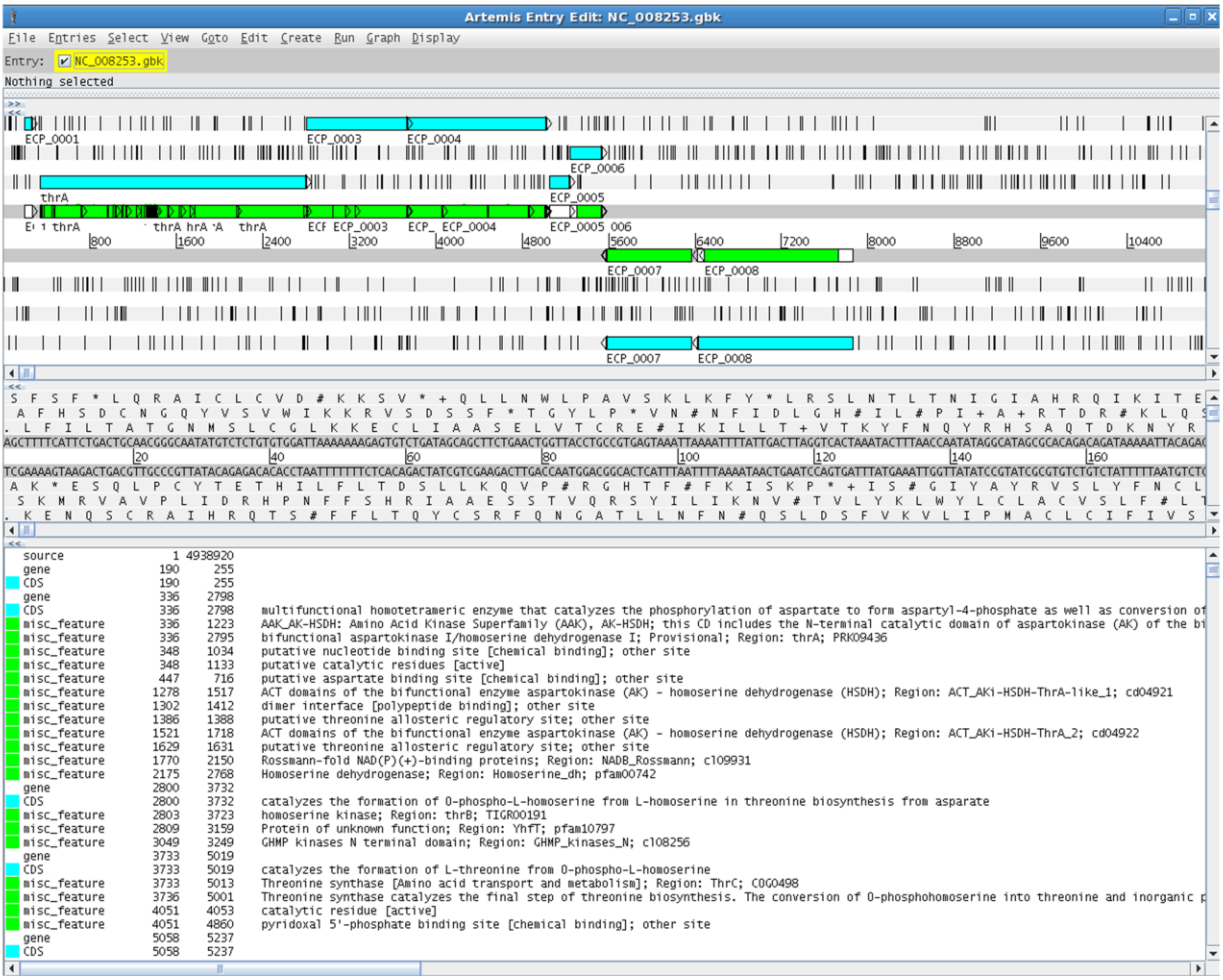
```
$ ./bowtie -S reference_genome/indexes/e_coli_536 reads/e_coli10000snp.fq
e_coli_snp.sam

$ ./samtools view -bt reference_genome/NC_008253.fna.fai e_coli_snp.sam >
e_coli_snp.bam

$ ./samtools sort e_coli_snp.bam e_coli_snp.sorted

$ ./samtools index e_coli_snp.sorted.bam
```

E. Visualizing the mapped reads in Artemis

We will now examine the read mapping in Artemis using the BAMview feature.

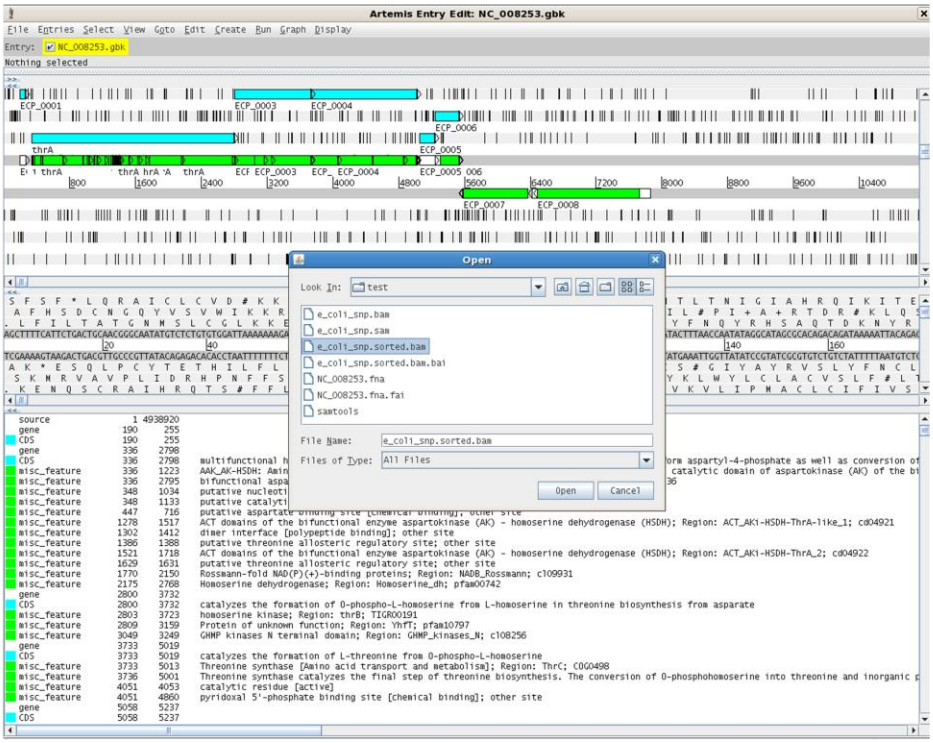
Open Artemis and load NC_008253.gbk from reference_genome directory. This contains exactly the same sequence as NC_008253.fna, but also has genome annotation so we can see the gene models.

You should see the Artemis window appear as in the screenshot below.

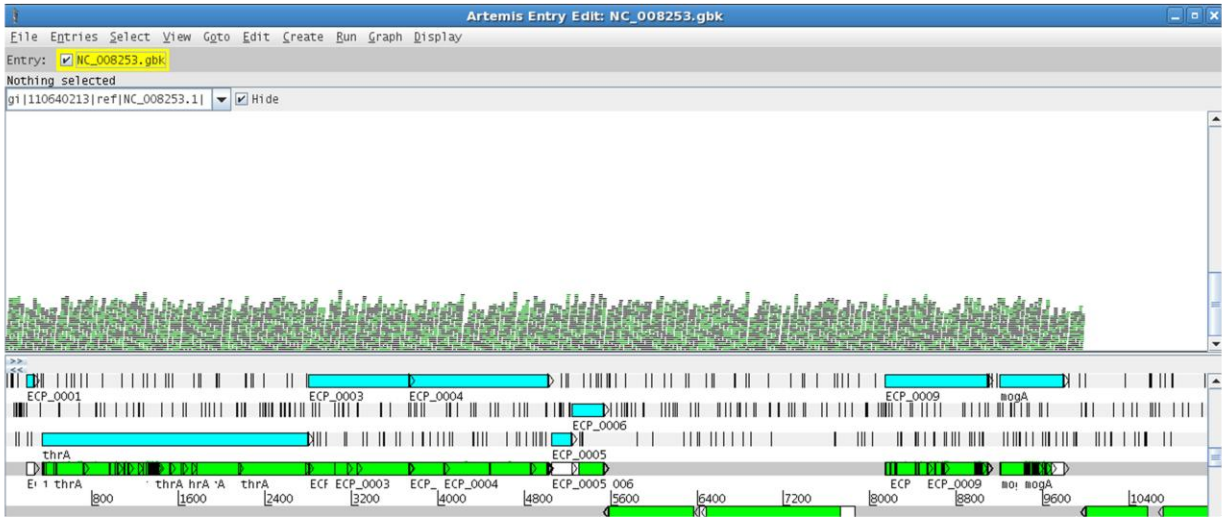


E. Visualizing the mapped reads in Artemis..continued…

From the Artemis file menu, select 'Read BAM', then locate the file *e_coli_snp.sorted.bam* from the module 3 data directory.



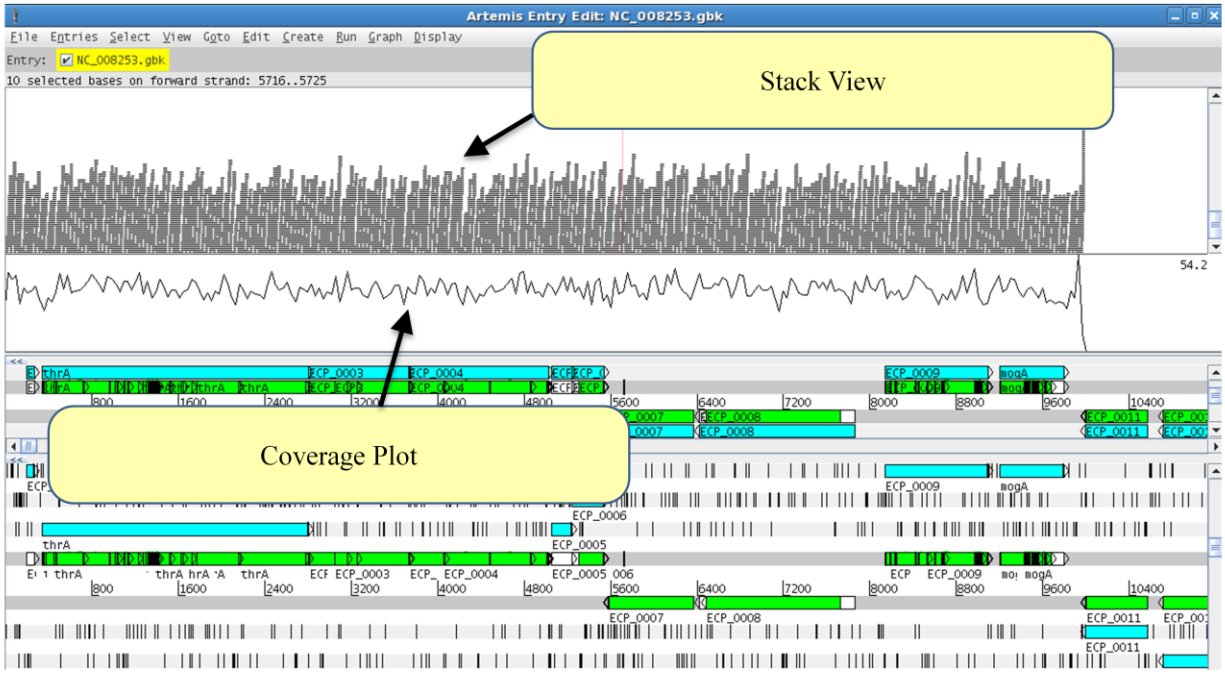
You should see the BAM window appear as in the screenshot below.



E. Visualizing the mapped reads in Artemis..continued…

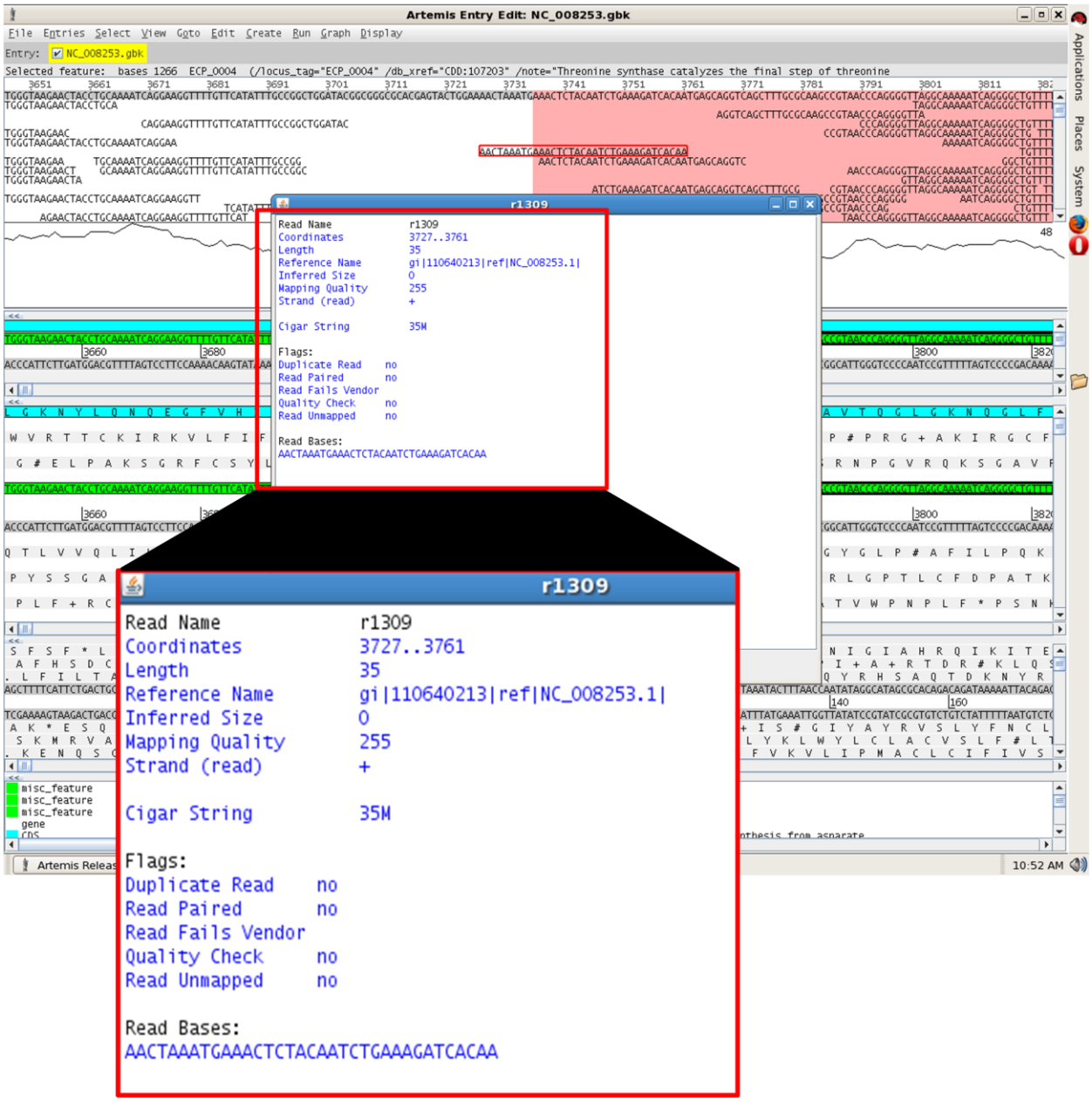
Now we want to view the coverage of the reads mapped to the reference genome.

You should see the coverage plot in the BAM window appear, as in the screenshot below.



E. Visualizing the mapped reads in Artemis..continued…

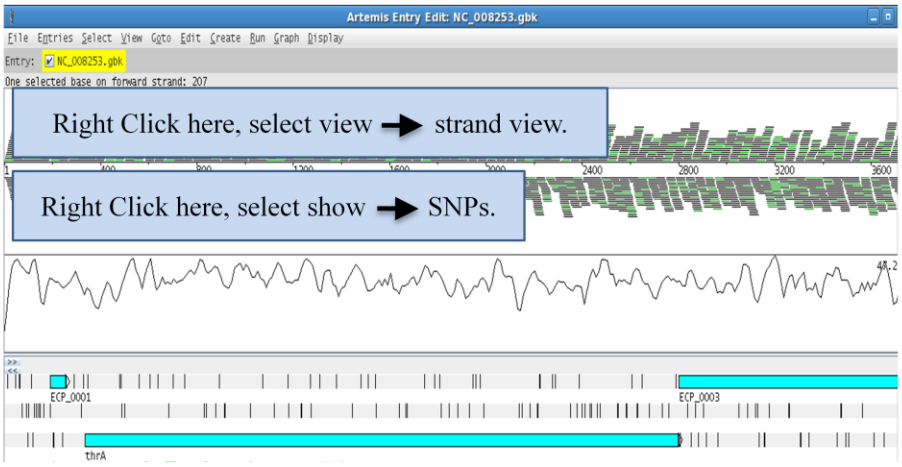
Zoom in and right click on a read. Select '*show details of …*'



Notice the "*mapping quality*". The maximum value for this is 255. the mapping quality depends on the accuracy of the sequence read and the number of mismatches with the reference. A value of zero means that the read mapped equally well to atleast one other location and therefore is not reliably mapped. The flags described the reads mate pair mapping.

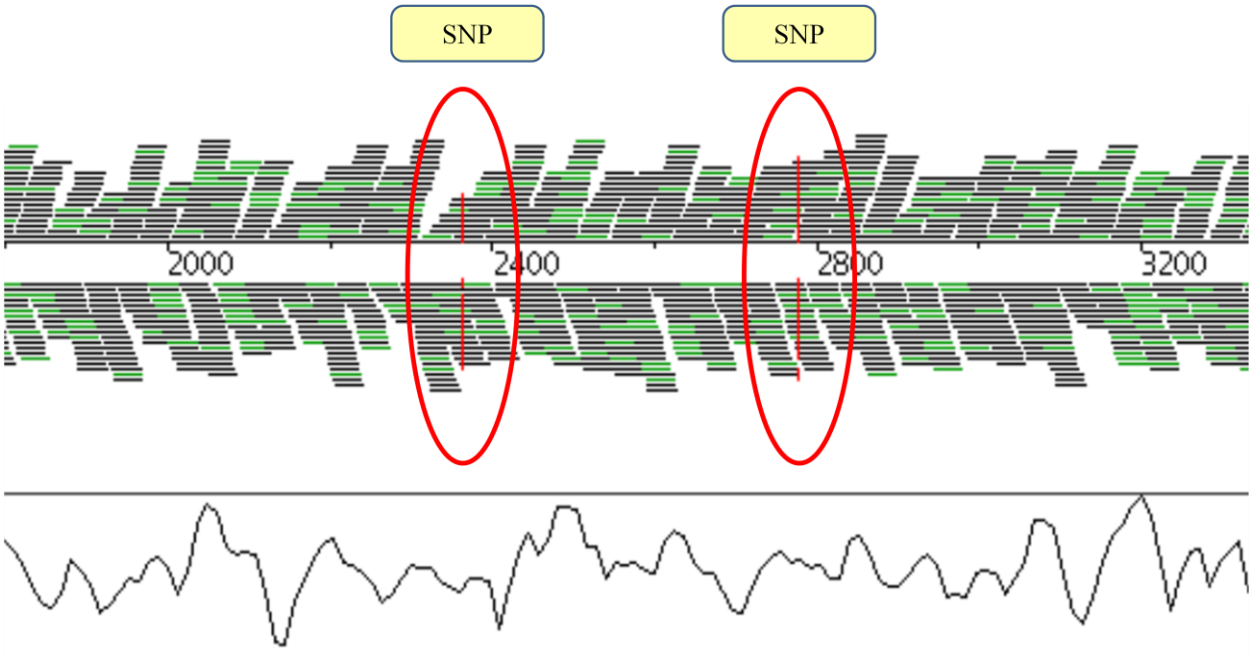
F. Identifying Single Nucleotide Polymorphism

Some differences between the reference and the mapped reads are due to sequencing errors. On average, 1 in every 1000 bases in the reads is expected to be incorrect. However if reads mapping to the same location consistently have a base which is different from the reference, it is likely that this base is mutated in the other genome.



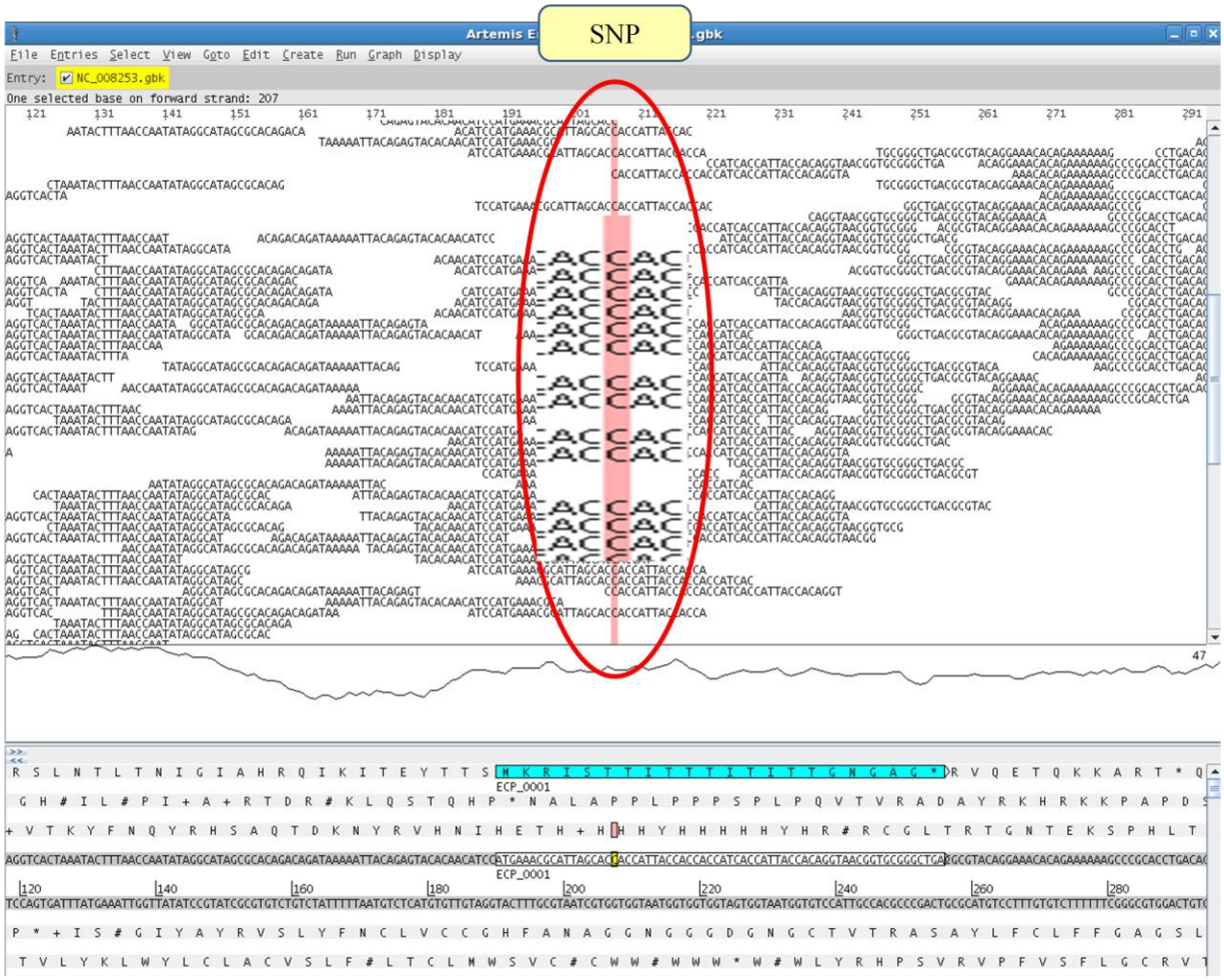
Strand View

Red marks appear on the stacked reads highlighting every base in a read which does not match to the reference. If you zoom in you can distinguish SNPs as vertical red lines. Red lines, while the random sequencing errors or mismatches would be more disperse.



F. Identifying Single Nucleotide Polymorphism..continued…

Zoom in as far as you can go to the one of the vertical red line of the mismatch. You can see that while the sequence of the reads (in black) is generally the same as the reference sequence. The vertical red lines identify the consistent difference as in the case below.



What could be the consequence of the SNPs you have identified? Many SNPs will have no effect – Why is this?

The SNP examples are quite clear with this dataset, however this is not always the case. What if the read depth is very low? If there are only two reads mapping, the reference is T and both reads are C. is this enough evidence to say that the genomes are different?

Module 3 ACT

(using prokaryotic example)

Introduction

The Artemis Comparison Tool (ACT), also written by Kim Rutherford, was designed to extract the additional information that can only be gained by comparing the growing number of sequences from closely related organisms (Carver *et al*. 2005). ACT is based on Artemis, and so you will already be familiar with many of its core functions, and is essentially composed of three layers or windows. The top and bottom layers are mini Artemis windows (with their inherited functionality), showing the linear representations of the DNA sequences with their associated features. The middle window shows red and blue blocks, which span this middle layer and link conserved regions within the two sequences, in the forward and reverse orientation respectively. Consequently, if you were comparing two identical sequences in the same orientation you would see a solid red block extending over the length of the two sequences in this middle layer. If one of the sequences was reversed, and therefore present in the opposite orientation, there would be a blue 'hour glass' shape linking the two sequences. Unique regions in either of the sequences, such as insertions or deletions, would show up as breaks (white spaces) between the solid red or blue blocks.

In order to use ACT to investigate your own sequences of interest you will have to generate your own pairwise comparison files. Data used to draw the red or blue blocks that link conserved regions is generated by running pairwise BLASTN or TBLASTX comparisons of the sequences. ACT is written so that it will read the output of several different comparison file formats; these are outlined in Appendix II. Two of the formats can be generated using BLAST software freely downloadable from the NCBI, which can be loaded and run a PC or Mac. Whilst having a local copy BLAST to generate ACT comparison files can be very useful, it means that you are tied to a particular computer. Another way of generating comparison files for ACT is to use either of the WebACT web resource (Appendix V). Both of these sites allow you to cut and paste or upload your own sequences, and generate ACT readable BLASTN or TBLASTX comparison files.

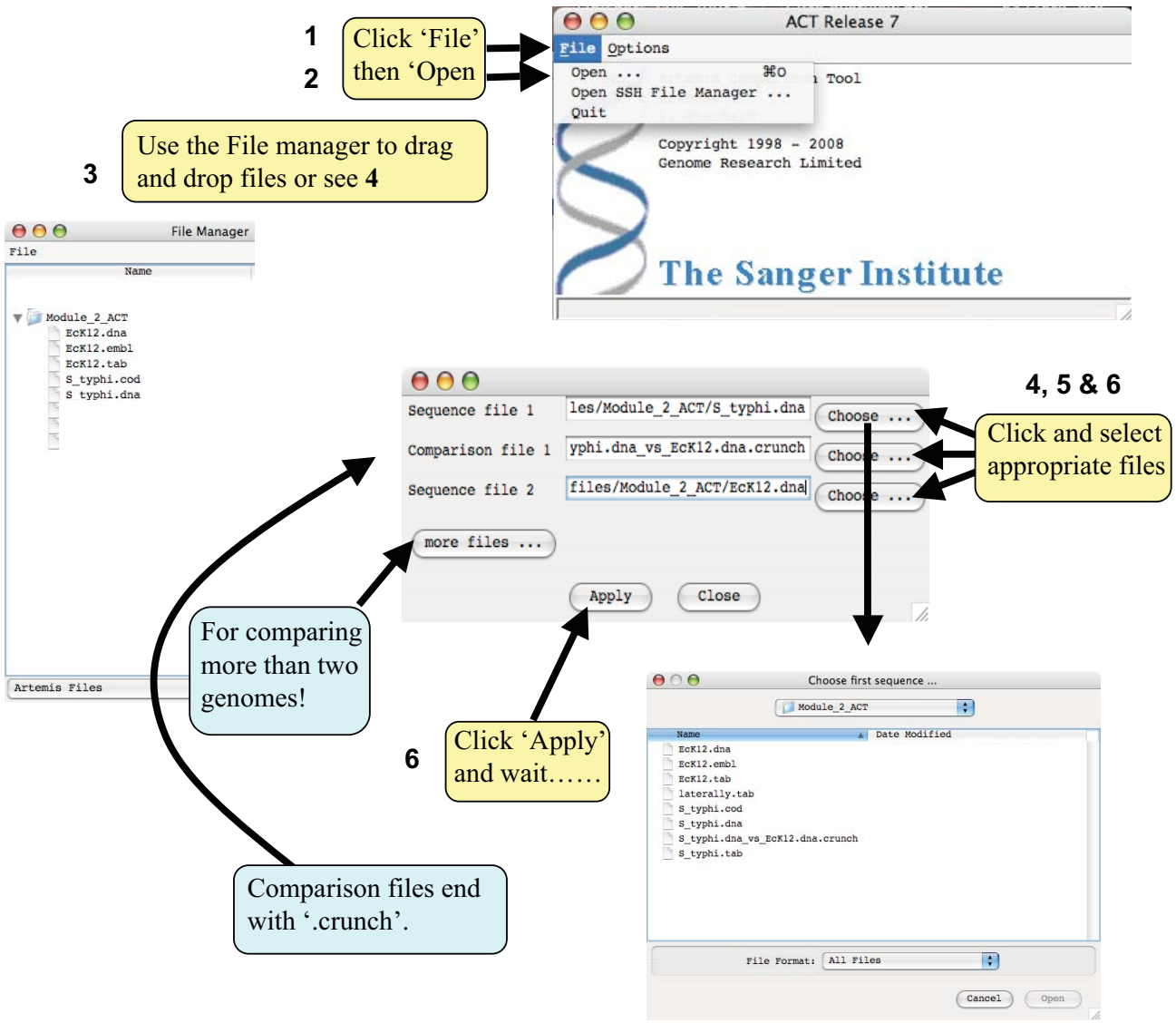
Aims

The aim of this Module is for you to become familiar with the basic functions of ACT by using a series of worked examples. Examples will touch on exercises that were used in the previous Artemis Module, this is intentional. Hopefully, as well as introducing you to the basics of ACT these examples will also show you how ACT can be used for not only looking at genome evolution. You will also be shown or use a web resource, WebACT to generate your own comparison files and view them in ACT, depending on the time available.

1. Starting up the ACT software

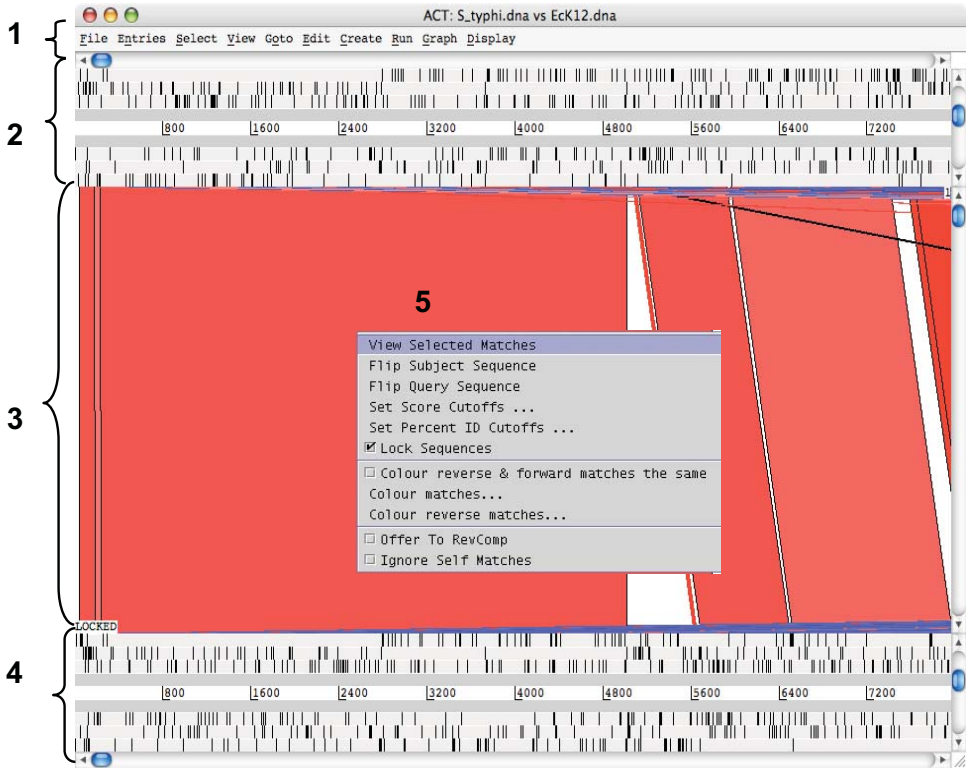
Make sure you're in the **Module_3_ACT** directory.
Then type
`act &` `[return]`
A small start up window will appear.
Now let's load up a *S.* Typhi versus *Escherichia coli* comparison.

The files you will need for this exercise are: *S_typhi.dna*
S_typhi.dna_vs_EcK12.dna.crunch
EcK12.dna



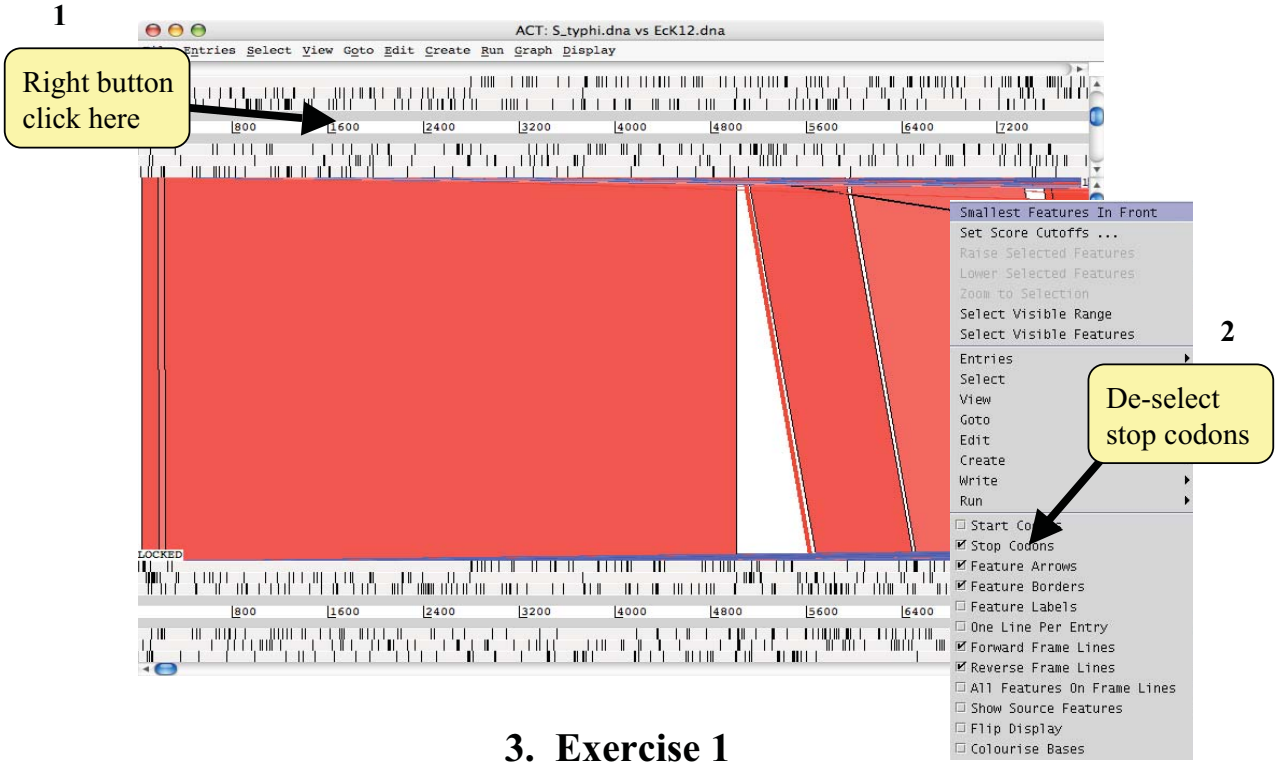
2. The basics of ACT

You should now have a window like this so let's see what's there.



1. Drop-down menus. These are mostly the same as in Artemis. The major difference you'll find is that after clicking on a menu header you will then need to select a DNA sequence before going to the full drop-down menu.
2. This is the Sequence view panel for 'Sequence file 1' (Subject Sequence) you selected earlier. It's a slightly compressed version of the Artemis main view panel. The panel retains the sliders for scrolling along the genome and for zooming in and out.
3. The Comparison View. This panel displays the regions of similarity between two sequences. Red blocks link similar regions of DNA with the intensity of red colour directly proportional to the level of similarity. Double clicking on a red block will centralise it. Blue blocks link regions that are inverted with respect to each other.
4. Artemis-style Sequence View panel for 'Sequence file 2' (Query Sequence).
5. Right button click in the Comparison View panel brings up this important ACT-specific menu which we will use later.

1



2

De-select stop codons

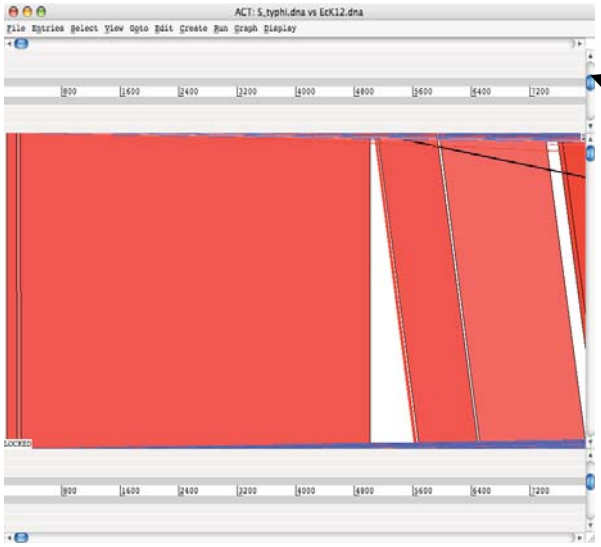
3. Exercise 1

Introduction & Aims

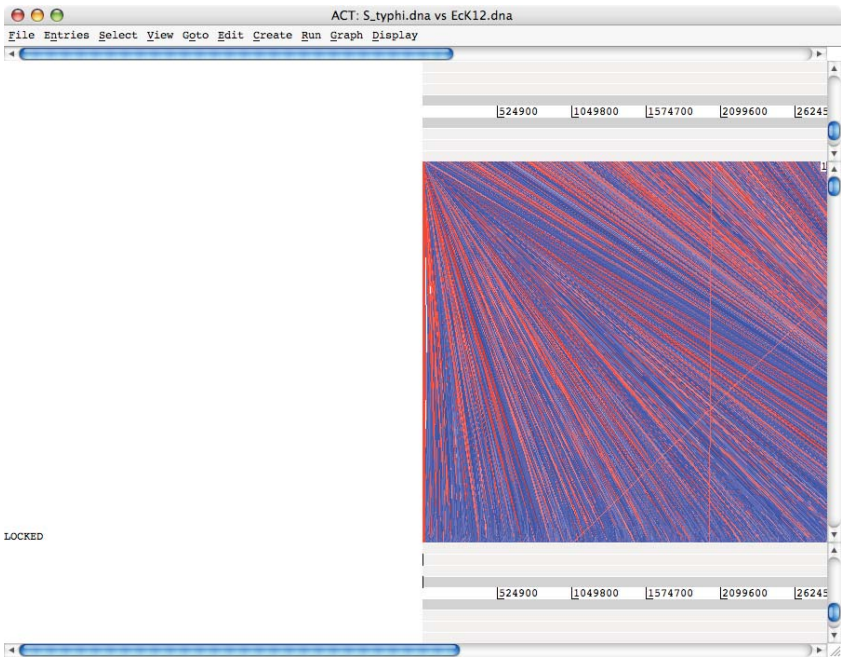
In this first exercise we are going to explore the basic features of ACT. Using the ACT session you have just opened we firstly are going to zoom outwards until we can see the entire *S.* Typhi genome compared against the entire *E. coli* K12 genome. As for the Artemis exercises we should turn off the stop codons to clear the view and speed up the process of zooming out.

The only difference between ACT and Artemis when applying changes to the sequence views is that in ACT you must click the right mouse button over the specific sequence that you wish to change, as shown above.

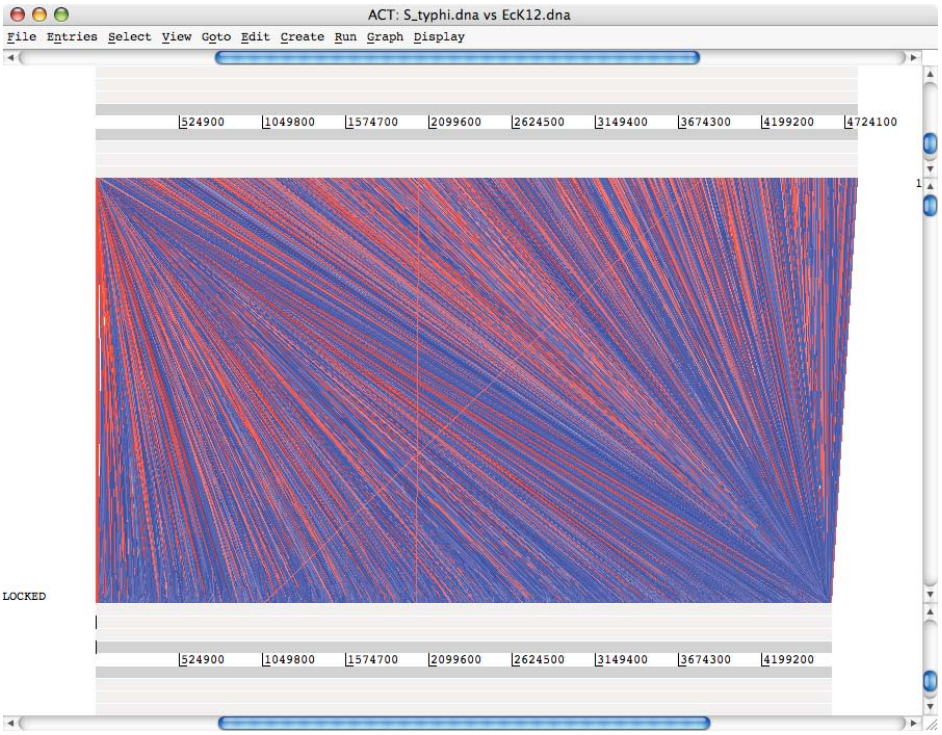
Now turn the stop codons off in the other sequence too. Your ACT window should look something like the one below:



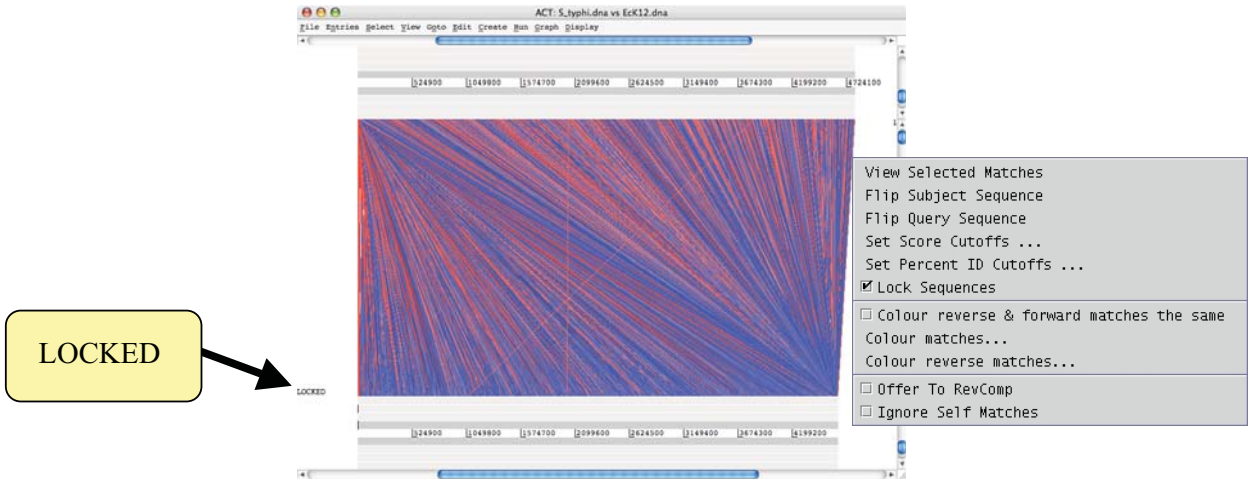
Use the vertical sliders to zoom out. Drag or click the slider downwards from one of the genomes. The other genome will stay in synch.



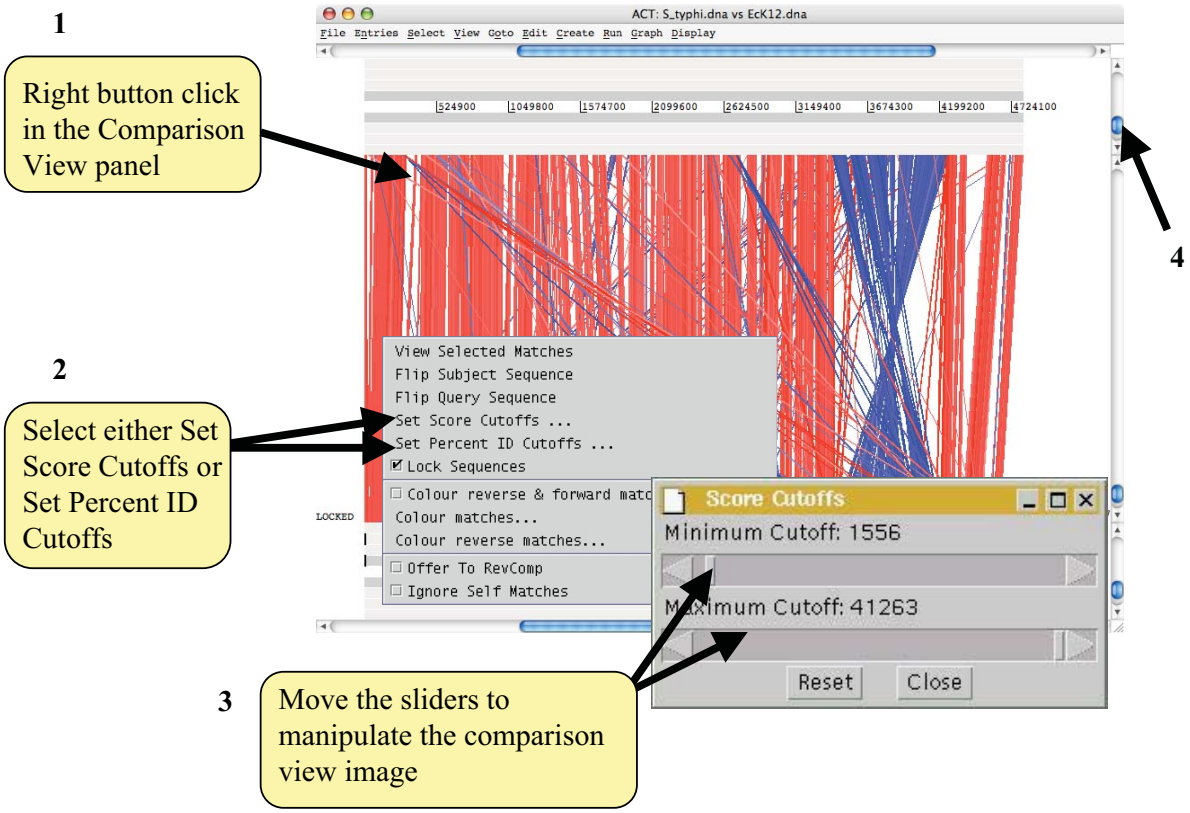
Once zoomed out your ACT window should look similar to the one shown above. If the genomes fall out of view to the right of the screen, use the horizontal sliders to scroll the image and bring the whole sequence into view, as shown below. You may have to play around with the level of zoom to get the whole genomes shown in the same screen as shown below.

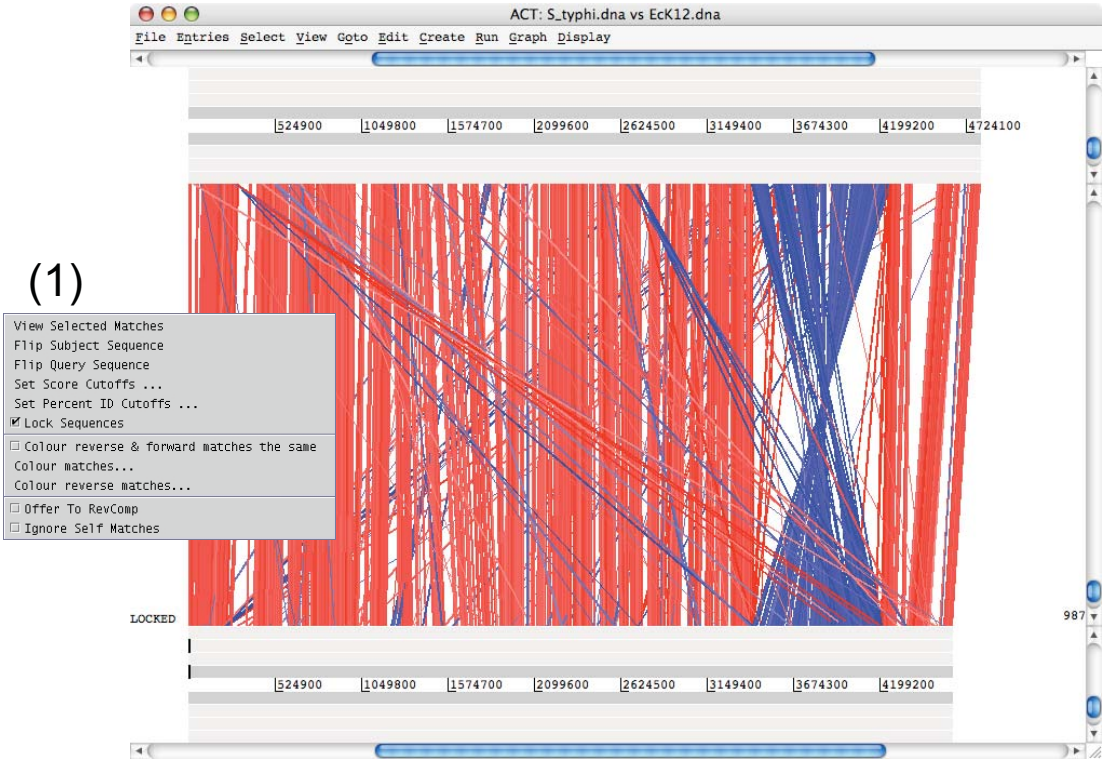


Notice that when you scroll along with either slide both genomes move together. This is because they are 'locked' together. Right click over the middle comparison view panel. A small menu will appear, select Unlock sequences and then scroll one of the horizontal sliders. Notice that 'LOCKED' has disappeared from the comparison view panel and the genomes will now move independently



You can optimise your image by either removing 'low scoring' (or percentage ID) hits from view, as shown below **1-3** or by using the slider on the the comparison view panel **(4).** The slider allows you to filter the regions of similarity based on the length of sequence over which the similarity occurs, sometimes described as the "footprint".





4. SPI-2 in ACT

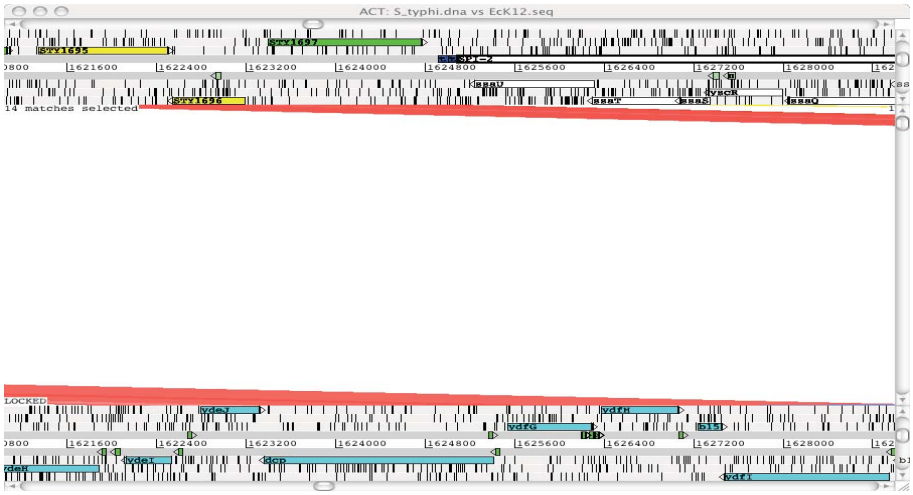
You should now be looking at an image of both the *E. coli* and *S.* Typhi genomes similar to that shown above. It is apparent that there is a backbone sequence shared with *E. coli* K12. Into this various chunks of DNA, specific the *S.* Typhi (with respect to *E. coli* K12) have been inserted.

•Key functions you should now try out in ACT

1.Double click (left mouse button) on the red boxes to centralise them.
2.Zoom right in to view the base pairs and amino acids of each sequence.
3.Also try using some of the other Artemis features e.g. graphs etc.
4.Find an inversion in one genome relative to the other then flip one of the sequences. To do this use the middle window menu shown above (1).

Load into the top sequence (*S.*Typhi) the annotation file 'S_typhi.tab'. You will need to use the 'File' menu and then select the correct genome sequence ('S.typhi.dna') before you can read in the appropriate annotation file or entry. The *E. coli* K12 annotation file (EcK12.tab) is also in the directory for this module so you can load this in too.

Now we are going to go to the region of the *S.* Typhi genome that we looked at earlier, SPI-2. Either by using the sliders or the 'navigator' find the SPI-2 region of the *S.* Typhi genome. If you are unsure of where this region is or how to get to it refer back to the earlier Artemis Module as these functions are the same in ACT.

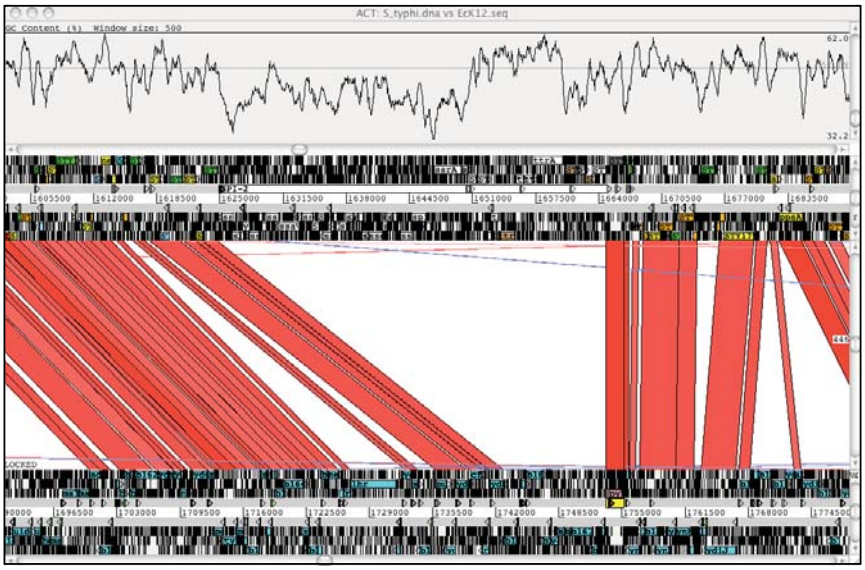


Once you have found SPI-2 in the *S.* Typhi genome double click (left mouse button) on the red boxes and use the sliders to centralise the sequences. Once you have done this it should look similar to the view below. This region is a clear insertion in the *S.* Typhi CT18 genome (see below) and has many of the characteristics of a classical pathogenicity island (PAI):

Jorg Hackers' Definition of a PAI

- Carry mobility functions e.g. integrases
- Inserted next to tRNA
- Anomalous G+C (add G+C plots see below)
- Carry virulence genes
- High number of pseudogenes
- In pathogens absent from non-pathogens

Take this opportunity to explore this region more fully and look for some of these features.



References

Langemead *et al*. (2009) *Genome Biology* 10:R25

Ultrafast and memory efficient alignment of short DNA sequences to the human genome.

Li et al. (2009). *Bioinformatics*, 25:1754-60

Fast and accurate short read alignment with Burrows-Wheeler Transform.

Carver T.J. *et al*. (2010). *Bioinformatics*, (doi:10.1093/bioinformatics/btq010)

Bam View: viewing mapped read alignment data in context of the reference sequence.

Carver T.J. *et al.* (2005) *Bioinformatics*, 21:3422-3

ACT: the Artemis Comparison Tool.

Berriman, M., and K. Rutherford (2003) *Brief Bioinform*, 4 (2) 124-132

Viewing and annotating sequence data with Artemis.

Rutherdord *et al.* (2000) *Bioinformatics* 16 (10) 944-945

Artemis: sequence visualization and annotation.

Teaching manual of open door workshop (2010) *Welcome Trust Sanger Institute*

Working with Pathogen Genomes

Abbot, J. C. *et al.* (2005) *Bioinformatics* 21(18)3665-3666

WebACT – an online companion for the Artemis Comparison Tool.

Appendices

PROCESSES

^c <ctrl>-c kills (definitely stops) current job

^z <ctrl>-z suspends the current job. This can either be moved to the background or resumed in the foreground by using **bg** or **fg**

bg moves the current process to the background

fg moves a process to the foreground. (If there is more than one suspended job, use **jobs** to decide which you want to **fg**)

fg 2 moves process number 2, as listed by jobs, to the foreground

jobs lists background and suspended processes (created with **bg** or **^z**)

jobs -l ("el" not one) includes the pid (process id number)

ps lists all your processes

kill stops a process (use **ps** or **jobs** to find your processes)

kill *2986*
kills off the process with pid *2986*

MISCELLANEOUS

finger tells you who is logged on (see also **w**)

w shows information about logged in users

who produces similar result (see finger)

tar create (or extract) a tarball from (to) a list of files

tar -cvf *tarball.tar subdir/

tar -xvf *tarball.tar*
the option **-z** compacts the files by **gzip**

wc word count

wc *long.file*
prints the number of lines, words and characters in *long.file*. Options include **-l** to count lines only, and **-c** to count characters only

ln create a link or an alias for a file

ln -s *subdir/orig.file alias.file*

history displays last several commands used

!! re-executes the last command

!51 executes command 51 in the history list use also <**up**> - and <**down**> - arrows to navigate in the history

date displays current date and time

passwd invokes a password changing program

exit leaves the current shell (same as **^d** or **<ctrl>-d**) usually = logout

GRAPHIC DISPLAY

To display graphics, most Unix require the configuration of the X-Window server.

Commands on <u>your local</u> computer:

xhost set the list of allowed X-Window clients

xhost +
The "**+**" allows any remote computer to display on your local display

ifconfig gives information about the network configuration (e.g., the current IP_address, usually similar to 123.145.167.189)

Commands on <u>the remote</u> computer:

setenv set up an environment variable (tc-shell)

setenv DISPLAY *IP_address*:0.0
required to tell the remote computer where it should display its graphics

xclock starts a graphic clock (e.g., used to test the X-Window server or to get the current time... ;-)

This document was originally written and designed by Aoife McLysaght and Andrew Lloyd© from the Irish EMBnet node, and modified by Laurent Falquet from the Swiss EMBnet node and distributed by the Publications Committee of EMBnet.

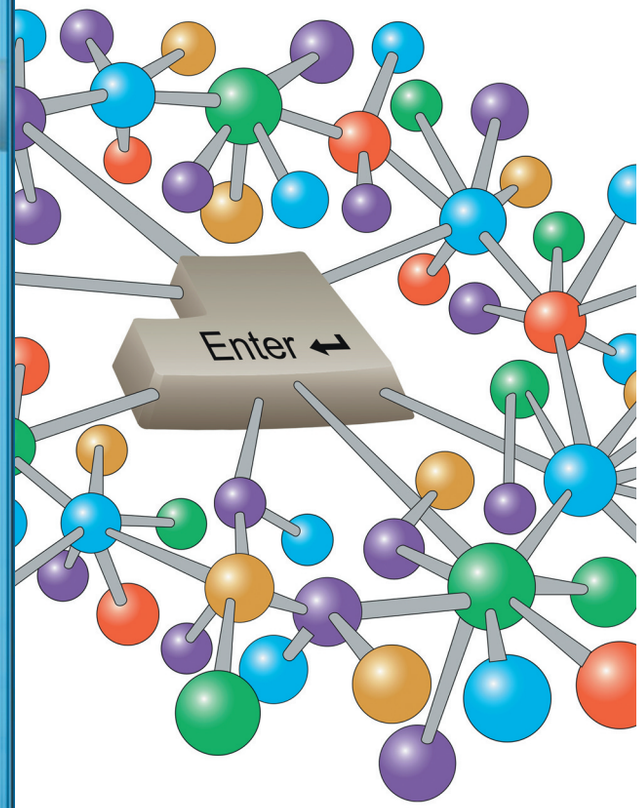
EMBnet - European Molecular Biology network - is a network of bioinformatics support centres situated primarily in Europe. Most countries have a national node which can provide training courses and other forms of help for users of bioinformatics software.
Further information about UNIX is available from your national node. You can find contact information about your national node from the EMBnet web site:

http://www.embnet.org/

If you have found this publication useful, please let us know.
If you have ideas for similar documents we'd like to hear from you: emb-pr@embnet.org

A Quick Guide To UNIX
Revised edition 2003

EMBnet



A Quick Guide
UNIX

A Quick Guide To UNIX

This is an introduction to the UNIX operating system. Unix may seem idiosyncratic, even impenetrable, to begin with but it has the virtue of minimising the number of keystrokes and so speeding up your access to the computer.
The commands listed here are common to different operating systems and shells. They include some of the most useful and frequently used commands in UNIX. The power and utility of most UNIX commands can be enhanced with switches or options preceded by a "**-**" sign.
More information on the options, the effects and how to use the commands is available by using the **man** command:

man gives manual information on a topic
man grep
displays the manual page about grep
apropos lists all the man(ual) entries relating to a topic (same as **man -k**)
apropos print

Another useful source of information is the on-line EMBnet tutorial which includes a page on UNIX

http://www.dk.embnet.org/Embnetut/Universl/unixcmds.html
or equally
http://www.uk.embnet.org/Embnetut/Universl/unixcmds.html

The general format of this document is that anything in **bold** is a command you can enter. Anything in ***italic*** is a fake file or directory name you must change according to yours. Anything preceded by a hyphen "**-**" is an option which will modify the effects of a command. A general description of each command is followed by one or several examples of its use.

FILES

ls lists files in a directory
ls -alF
lists **-a** all files in **-l** long format **-F** identifies directories **/**, executable files *****, and symbolic links **@**, in the current directory

cat concatenates and displays files
cat *my.file*
displays *my.file* on the screen

chmod modifies the read (**r**), write and delete (**w**), and execute (**x**) permissions of specified files and the search permissions of specified directories. The permission can be set for user (**u**), group (**g**) or other (**o**)
chmod go-w *my.file*
stops (**-**) anyone else (**go**) changing or deleting (**w**) *my.file*
chmod g+rwx *my.file*
allows (**+**) anyone of my group (**g**) reading, changing, deleting or executing (**rwx**) *my.file*

cp copies files
cp *orig.file copy.file*
cp *orig.file subdir/new.file*
copies *orig.file* to *new.file* in *subdir* directory
cp *subdir/orig.file .*
copies *orig.file* from *subdir* to the current directory (**.**) without changing its name

mv moves/renames a file (or directory)
mv *oldname newname*
mv *my.file subdir/my.file*
a move (**mv**) is equivalent to a copy (**cp**) followed by a remove (**rm**)

rm removes/deletes a file.
rm *oldfile*
rm -i **.file*
option **-i** (interactive) advised if wildcards (*) in use

diff compares two files and prints how they differ
diff *file1 file2*
prints differences to screen options include **-b** to ignore differences in blank space, and **-i** to ignore case

find searches the directory tree for a file
find . -name *lostfile* -print
will search your current directory (**.**) (and any subdirectories) for *lostfile*

grep searches a file for a string
grep *word my.file*
grep "two words" my.file
options include **-i** to ignore case and **-n** to print line numbers

vi simple screen oriented text editor

pico simple display oriented text editor
pico *myfile.txt*

head prints the first few (default = 10) lines of a file
head *oddfile*
head -20 *oddfile*
displays first twenty lines of *oddfile*

tail displays last few lines of a file (see head)

more displays a file one screenful at a time
more *longfile*
hit <**spacebar**> to see the next screen
Note: some people prefer **less**

OUTPUT REDIRECTION

> redirects output of a command to a file
diff *file1 file2* > *new.file*
puts differences into *new.file*
cat *one.file two.file* > *both.file*
writes the output of the cat command into *both.file* (overwrites *both.file*)

>> appends a file to the bottom of another
cat *three.file* >> *both.file*
appends *three.file* to the bottom of *both.file*

| "pipe" - uses the output of the first command as the input of the second
grep *string my.file* | wc -l
finds how many lines on which "*string*" occurs (see **grep** and **wc**)

DIRECTORIES

cd changes current directory
cd */etc*
go to */etc* directory
cd ..
go up one level in directory tree
cd *../subdir2*
go "sideways" to *subdir2*

mkdir creates a new subdirectory
mkdir *subdir*

rmdir removes a directory - you must delete all the files in it first
rmdir *subdir*

pwd print working directory, tells your current location (path)

EMBOSS

A Quick Guide

European Molecular Biology Open Software Suite

History
Since 1988, the sequence analysis package EGCG has provided extensions to the market leading commercial sequence analysis package GCG. EGCG development was a collaboration of groups within EMBnet and elsewhere.
That project has reached the limits of what we can achieve using the GCG package. Specifically, it is no longer possible to distribute academic software source code which uses the GCG libraries and has become difficult even to distribute binaries.
As a result, the former EGCG developers have been designing a totally new generation of academic sequence analysis software. This has resulted in the present EMBOSS project.
EMBOSS is a new suite of freely available programs and libraries for sequence analysis. It incorporates and integrates a range of currently available public packages and tools into a general, publicly available, suite specially developed for the needs of the Sanger Centre and the EMBnet user community.

Licensing
The EMBOSS core application suite is licensed under the General Public License (GPL) allowing free copying, modification and distribution of the package.
The EMBOSS Libraries are licensed under the the Library General Public License.
Associated packages may be licensed under different terms, all of which permit free redistribution of the software.

Obtaining EMBOSS
EMBOSS and the associated packages can be obtained via FTP from the Sanger Centre, UK at ftp.sanger.ac.uk/pub/EMBOSS

EMBOSS home page
http://www.sanger.ac.uk/Software/EMBOSS

Running EMBOSS
All EMBOSS programs are designed to be run from the comamnd line. Each program has a specific description file (ACD file) that describes the input and output parameters. All the parameters can be specified on the command line, allowing modular integration into graphical interfaces.
To run an EMBOSS program, just type its name. Your system administrator should ensure that the programs are available in your $PATH.

The Uniform Sequence Address (USA)
The USA is a method of specifying the location of a sequence and its format. The general form is:
Format::database:sequencename
eg. `embl::em:scact`
EMBOSS is normally very good at identifying sequence *formats* automatically but occasionally needs a hint. *Database* will be one of the databases already set up at your site. The command
% `showdb`
lists the databases available on your system.
The *sequencename* can be either its name, accession number, the filename in which the sequence is found, or the sequence itself if `asis::` format is specified. If you are taking one sequence from a multiple sequence file, put the sequence number in braces after the filename, eg:
`allmyseqs.fasta{32}`

EMBOSS programs
You can obtain a list of EMBOSS programs with the command `wossname`. Useful qualifiers for `wossname` are :

`-alphabet`	List all programs in alphabetical order
`-auto`	List all programs without asking for a keyword.

% `wossname -alphabet -auto`
will list all the available emboss programs with a short description of the function of each program

EMBOSS will by default only prompt you for the minimal input it needs to run the program. The default behaviour can be changed using command line qualifiers.

Important qualifiers
The behaviour of EMBOSS programs can be modified by using a large number of qualifiers. This is a list of the more useful ones.

`-help`	Prints a summary of the options the program can take. With `-verbose` it gives a more detailed list.
`-options`	Prompt the user for the optional parameters
`-auto`	Accept all the default settings and run without prompting the user.
`-sask`	Ask for the start, end and reverse of the sequence input
`-stdout`	Print output to `stdout` (the screen) instead of to a file.
`-filter`	Take input from `stdin` (keyboard) and output to `stdout`

What `-help` tells you
The `-help` option lists the inputs to the program along with the input type (sequence, integer etc). There are additional qualifiers associated with many types. `-verbose` will list all the additional qualifiers related to the input types for the program.

The qualifiers are listed in three sections:

Mandatory Qualifiers	These are the minimum inputs the program needs to run. Some of these have default values which can be selected using `-auto`
Optional Qualifiers	These are qualifiers for which you will be prompted if you use the `-option` qualifier. All these qualifiers have default values.
Advanced Qualifiers	You will never be prompted for these. If you wish to use them you must specify them on the command line.

EMBOSS parameter types

Type	Allowed values
`bool`	yes: `-param` no: `-noparam`
`integer`	Whole numbers `-param=5`
`float`	decimal numbers `-param=23.9`
`range`	sequence ranges. eg. `-param=1-12,35-99`
`regexp`	a regular expression pattern
`string`	ordinary text. `-param='text with *'`
`infile`	path of a file
`matrix`	integer scoring matrix for alignments
`matrixf`	floating point scoring matrix
`codon`	codon usage table
`sequence`	Uniform sequence address (USA) for the
`seqset`	sequence or set of sequences.
`seqall`	
`features`	Feature table
`list`	list of options
`selection`	selection list of options
`outfile`	path to a file for nonsequence output
`seqout`	output sequence USA
`seqoutset`	multiple sequence file for output
`seqoutall`	multiple or single sequence output files
`featout`	output feature table
`graph`	output device for graphics images
`xygraph`	output device for XY graphs

See the descriptions below for many of these.

Associated qualifiers: sequence, seqset, seqall

-sbegin	integer	first base used [start]
-send	integer	last base used [end]
-sreverse	bool	reverse sequence [N]
-sask	bool	prompt for begin/end/reverse [N]
-snucleotide	bool	Sequence is nucleotide [N]
-sprotein	bool	Sequence is protein [N]
-slower	bool	Make sequence lowercase[N]
-supper	bool	Make sequence uppercase[N]
-sformat	string	input sequence format
-sopenfile	string	input filename
-sdbname	string	database name
-sentry	string	entry name/accession number
-ufo	string	Feature table (UFO)
-fformat	string	features format

Associated qualifiers: seqout, seqoutset, seqoutall

-osformat	string	output sequence format
-osextension	string	filename extension
-osname	string	base filename
-osdbname	string	database name to add
-ossingle	bool	seperate file for each entry[N]
-oufo	string	features UFO
-offormat	string	features format
-ofname	string	features filename

Associated qualifiers: features

-fformat	string	features format
-fopenfile	string	features filename
-fask	bool	prompt for **fbegin, fend,** and **freverse**
-fbegin	integer	features starting position
-fend	integer	features end position
-freverse	boon	features on the reverse strand [N]

Associated qualiifers: featout

-offormat	string	feature format
-ofopenfile	string	output filename
-ofextension	string	filename extension
-ofname	string	filename
-ofsingle	bool	write one feature per file

Associated qualifiers: graph, xygraph

-gprompt	bool	graph prompting
-gtitle	string	graph title
-gsubtitle	string	graph subtitle
-gxtitle	string	x axis title
-gytitle	string	y axis title
-grtitle	string	right axis (y2) title
-gpages	integer	number of pages
-goutfile	string	output filename

EMBOSS and Graphics

EMBOSS can support a number of different graphics output types depending on the features available on your system. It will prompt for a graphics device:

Graphics device [x11]:

Typing rubbish here then pressing return will give a lengthy list of devices, many of which are equivalent.

The main graphics options are:

[X]	x11	Output to an X-window
	postscript	Output to a postscript file (good for printing on a laser printer)
	cps	Output to a colour postscript file
	text	Output to a text file
	data	Output XY data points to a file. (good for importing into a graphing package)
[P]	png	Output to a PNG image file (good for web pages)
[X]	Tek	Output to tektronics terminal
[X]	xterm	Output to an Xterm window

[X]- requires X-windows [P] – requires PNG support

The default filename is *prog.format* eg. **octanol.ps**

Some useful programs

General

wossname	lists all EMBOSS programs
showdb	Shows the available databases

Sequence retreival

seqret	retreives and/or changes format of a sequence
seqretset	retrieve and or change formats of a number
seqretall	of sequences at once
transeq	translate a DNA sequence to protein
backtranseq	translate a protein sequence to DNA
extractseq	extract regions from a sequence
cutseq	remove a region from a sequence
pasteseq	inserts a sequence into another sequence
infoseq	display information about a sequence
splitter	split a sequence into smaller sequences

Sequence comparison

needle	Needleman-Wusch sequence alignment
water	Smith-Waterman sequence alignment
stretcher	Myers and Miller global alignment
matcher	Huang and Miller local alignment
dottup	dotplot comparisons of two sequences.
dotmatcher	
prettyplot	plots multiple sequence alignments
polydot	dotplot comparisons of multiple sequences.
supermatcher	

Sequence parameters

cusp	generates a codon usage table
syco	synonymous codon usage plot
dan	calculates DNA/RNA melting temperature
compseq	sequence composition tables

DNA Sequence features

remap	restriction map of the sequence
cpgplot	CpG island detection
cpgreport	
etandem	finds tandem and inverted repeats
einverted	
plotorf	plots potential ORFs
showorf	pretty display of potential ORFs
fuzznuc	DNA pattern search
tfscan	scans sequence for TF binding sites

Protein Sequence features

ief	Isoelectric point calculation
antigenic	Finds potential antigenic sites
digest	protein digestion map
findkm	Vmax and Km calculations
fuzzpro	protein pattern search
garnier	protein 2D structure prediction
helixturnhelix	finds nucleic acid binding motifs
octanol	displays protein hydropathy
pepwindow	
patmatdb	searching with motifs vs protein sequences
patmatmotifs	
pepcoil	predicts coiled coil regions
pepinfo	Protein information
pepstats	
pepwheel	shows protein sequences as a helix.

File formats supported by EMBOSS

IntelliGenetics, Genbank, NBRF, EMBL, GCG, DNAStrider, Fitch, FASTA, Phylip, PIR, MSF, ASN.1, PAUP, Clustalw

This Quick Guide was written by and is copytight Dr David Martin at the Norwegian EMBnet node.
Comments and suggestions for improving this guide should be addressed to him at david.martin@biotek.uio.no

EMBnet is a network of academic and commercial bioinformatics institutes, supporting bioinformatics research and collaboration in more than countries worldwide.

More information about EMBnet and details of your local node can be found at http://www.embnet.org