

Drosophila Sequence Improvement Problem Set

Prepared and/or updated by Andrew Nylander, Matt Dothager, William Barshop, Chris Shaffer and Wilson Leung

Prerequisites:

A Guide to Consed

Introduction to Consed

Using Consed Graphically

Files for this Exercise:

The files required for this exercise are compressed into a single archive [finishingHomework.tar.gz].

Path to the Exercise Directory:

This exercise assumes the project package is located in your home directory (e.g. the relative path to the project is `~/finishingHomework`). If you place the exercise package in a different location, you will need to modify the relative path to the project accordingly. For example, if the exercise is inside the Desktop folder of your home directory, the path to the project becomes `~/Desktop/finishingHomework`.

Depending on your computer setup, the absolute path to the project may vary. On a computer running Mac OS X with the user account student4342, the absolute path to the project is `/Users/student4342/finishingHomework`. If you have a different username or use a different operating system (e.g. Linux), you will need to adjust the absolute path to your project. For example, the absolute path on most Linux systems with the account named student4342 and the exercise directory finishingHomework is `/home/student4342/finishingHomework`. You can determine the absolute path to the project directory using the command `pwd` on the xterm.

Purpose:

In this lab exercise, we will apply some of the skills you have learned in the previous Consed tutorials and exercises. We will look at two different Drosophila projects and tackle some of the interesting problems you may encounter in your own sequence improvement projects.

Questions in this exercise will refer to different projects and ace files. You will need to relaunch Consed in order to access these ace files, as there is no “open file” functionality in Consed. When you open an ace file, Consed may ask you if it should apply edits from the edit history file (.wrk file) to the ace file you wish to open. Always choose ‘No’ unless Consed crashed and you did not save your changes (Figure 1).

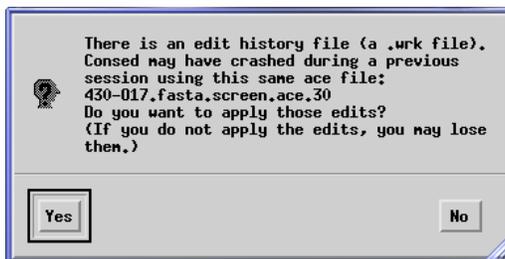


Figure 1. Do not apply edits when opening an ace file unless Consed crashed previously and you have unsaved edits.

Project 485-B16

I. Identifying the Fosmid Clone Ends

Launch X11 and open a new xterm. Navigate to the *edit_dir* of the project 485-B16 inside the exercise package (`cd ~/finishingHomework/485-B16/edit_dir`). Launch Consed and open the ace file '485-B16.ace.28'.

Click on the 'Assembly View' button on the Consed main window to get an overview of the project (Figure 2). The current assembly consists of three large contigs (5, 3, and 4) and the clone has a relatively high repeat density. Furthermore, while we know that contig 3 is connected to contig 4 based on the spanning subclones, we do not know the relationship between contig 5 and the other contigs. To help us make sense of this assembly, we should first identify the fosmid clone ends.

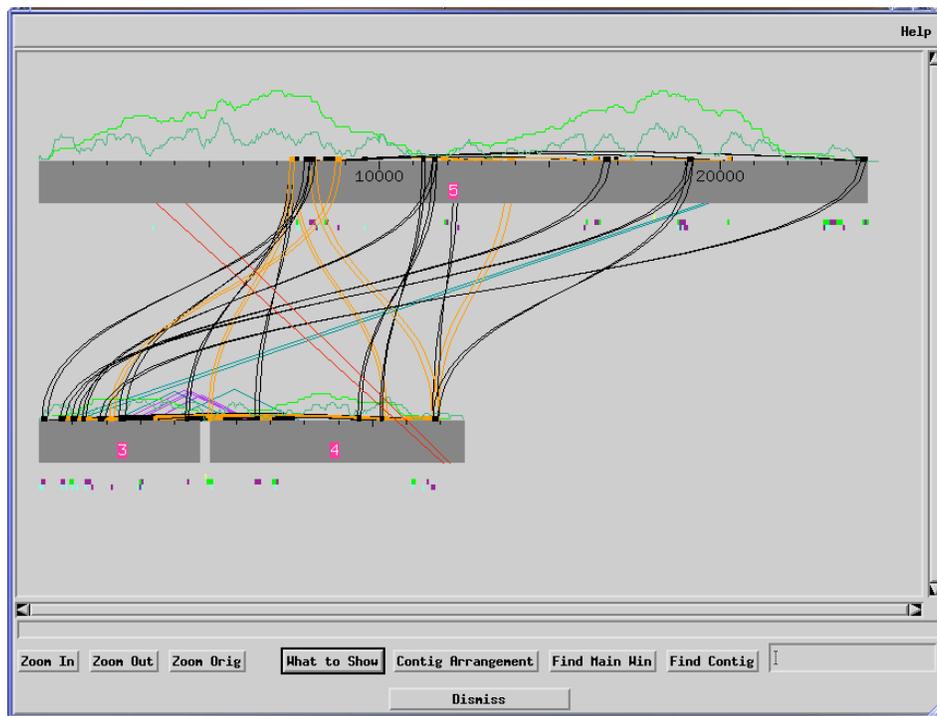


Figure 2. The initial assembly view for the project 485-B16 shows three large contigs

When Consed recognizes the vector sequence within a read, those bases are automatically masked as X's. Hence, in the simplest case where there are high quality reads that consist of sequences from the ends of the fosmid and the vector sequence, identifying the fosmid ends (the junction between the vector and the insert) is trivial. However, because the vector is usually located at the very beginning of a read, Consed often has difficulty identifying the vector correctly due to poor trace quality in the region containing the vector sequence. If you are not confident about the location of the vector or believe there are sequences missing at the beginning or the end of the assembly, you can design oligos near the ends of your clone to generate additional sequences that contain both the insert and the vector. Consed should automatically mask the vector sequences as X's when you incorporate the new reads into the assembly.

Open the Aligned Reads window for contig 5 and navigate to base 1 of the contig. You should see a clear transition between the vector and the beginning of the insert (Figure 3) [The default view used by your version of Consed may be different than that shown in Figure 3. To match the figures, select the option to “Turn Off: sort by quality” found in the Sort menu at the top of the Aligned Reads window]. Since the masked bases are on the left side, the insert is to the right side of this region of the contig. We will add a tag to the consensus sequence to indicate that this is one end of the clone. Select base 1 of the consensus and select ‘Misc’ on the main menu bar. Click on ‘Add Clone End Tag with Insert to the Right’ (Figure 4).



Figure 3. The vector sequences are masked as X's and our clone begins at position 1 of the consensus.

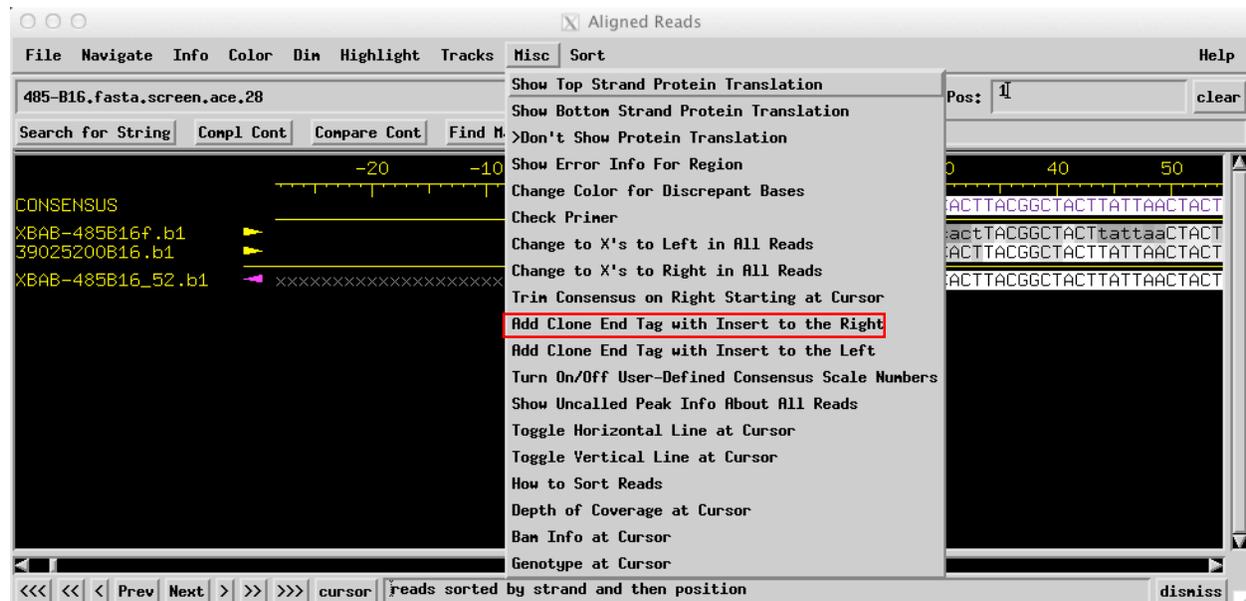


Figure 4. Use the ‘Add Clone End Tag with Insert to the Right’ option to mark the left end of the clone.

1. Open the Aligned reads window for the other contigs and locate the other clone end. Add the corresponding clone end tag. Which contig and what coordinate did you identify as the other clone end?

II. Use Restriction Digests to Estimate Gap Size

Normally, the restriction digests data is used in the finishing process to validate the final assembly. However, in cases where there are difficult gaps or major misassemblies in your project, comparing the real restriction enzyme digests with the *in silico* digests often allows us to devise strategies to resolve these problematic regions. In this part of the exercise, we will use the restriction digests data to estimate the size of the unspanned gap between the end of contig 5 and the beginning of contig 3 (Figure 5).

Close 'Assembly View' and go back to the Consed main window. Click on the 'Assembly View' button again to see an updated figure. The clone ends are now marked by two pink triangles. Each triangle points toward the insert fosmid sequence.

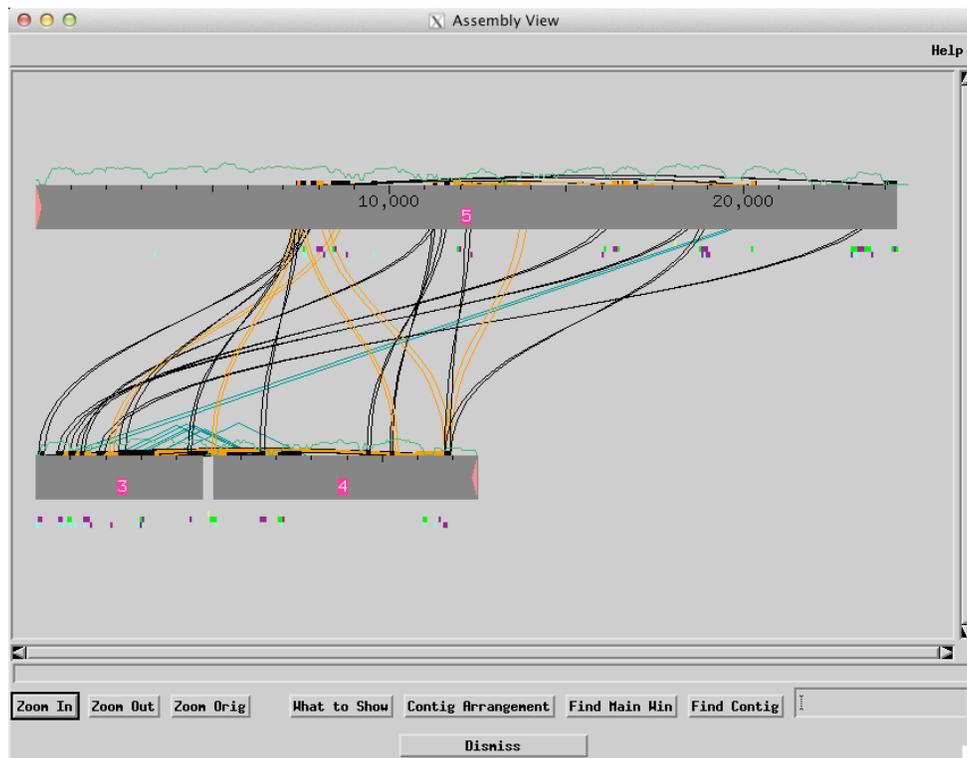


Figure 5. Estimating the size of the unspanned gap between the end of contig 5 and the beginning of contig 3.

Click on the 'Digests' button in the Consed main window to view the digests information (Figure 6). For most fosmid projects, the Washington University Genome Sequencing Center (WUGSC) has generated digest information using four restriction enzymes: EcoRV, HindIII, EcoRI, and SacI. By default, the enzymes EcoRV and HindIII are selected automatically. Under the section 'Huge List of Restriction Enzymes', select 'EcoRI' and 'SacI'.

For some fosmid projects where only a few vector sequences are available, Consed may have trouble identifying the correct junction between the insert and the vector in the *in silico* digest.

To help Consed identify the vector we will change the digest option from ‘Entire Clone’ to ‘Just Part of Clone’. Next, we need to determine the order and relative orientations of the contigs.

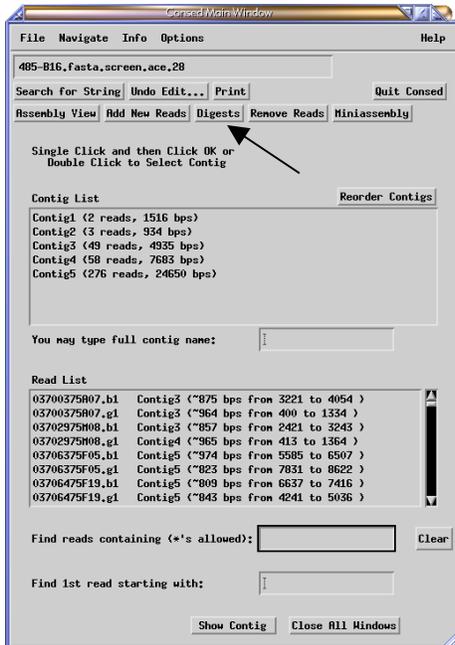


Figure 6. Click on the ‘Digests’ button to compare the real and *in silico* digests.

Based on the clone end tags in the ‘Assembly View’ (Figure 5), the clone starts with contig 5 and ends with contig 4. Based on the consistent forward and reverse read pairs (purple triangles), we know that contig 3 is connected to contig 4. Therefore, even though we do not have read pairs that span contig 5 and contig 3, we can make an educated guess that the order and relative orientation of the contigs is ‘5-3-4’. If a contig were flipped relative to the other contigs, you would need to append a ‘c’ to the contig name (e.g. 1c). In this case, we should type ‘5-3-4’ in the field ‘Enter scaffold of contigs in same format as the contig order and orientation map’. Consed will use this value instead of the order listed in the ‘Contig order and orientation’ field when it creates the *in silico* digest.

In the ‘Leftmost Contig Start Pos:’ field, type 1 to indicate the *in silico* digest should start at position 1 of the contig 5. In the ‘Rightmost Contig End Pos’, enter the last position of contig 4, which is 7683 in this case. The size of contig 4 is available from the Consed main window and the Aligned Reads window.

Since each fosmid library is generated using a different vector, we also need to change the path to the vector sequence. The vector sequence for the fosmid is available inside the *edit_dir* of your project directory in a file called *pcc01.fasta*. We need to tell Consed where to look for this vector sequence in the ‘Full Pathname of File of Vector Sequences:’ field.

In cases where the vector sequence is located in the *edit_dir* of the Consed project, like this one, a full pathname is not required. Instead, you can simply change the ‘Full Pathname of File of

Vector Sequences' field to the name of the vector sequence file (pcc01.fasta in this example). Alternatively, we can go back to the xterm you used to launch Consed to determine the full path to the vector sequence. Type 'pwd' in the xterm and press return to show the full path to the project directory (Figure 7). Append the vector filename 'pcc01.fasta' (without the quotes) to create the full path.

```
~/finishinghomework/485-B16/edit_dir/ >pwd
/Users/student4342/finishingHomework/485-B16/edit_dir
~/finishinghomework/485-B16/edit_dir/ >
```

Figure 7. Determining the absolute path to the directory containing the vector sequence using the *pwd* command

In our directory setup we will change the 'Full Pathname of File of Vector Sequences:' to '/Users/student4342/finishingHomework/485-B16/edit_dir/pcc01.fasta' (Figure 8a). The simpler solution would be to change the 'Full Pathname of File of Vector Sequences:' to 'pcc01.fasta' (Figure 8b). Click "OK" to view the digest information.

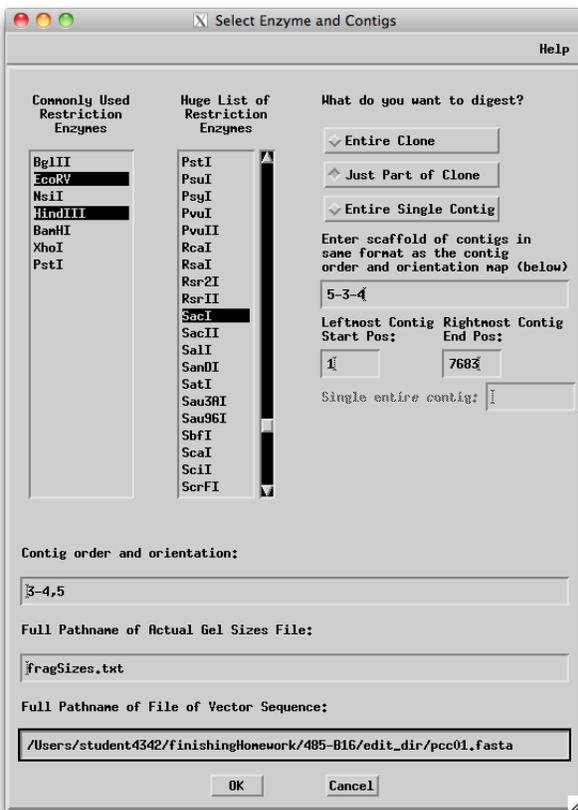


Figure 8a. Select additional enzymes, define contigs to digest and change the path to the full vector path in the digest configuration screen.

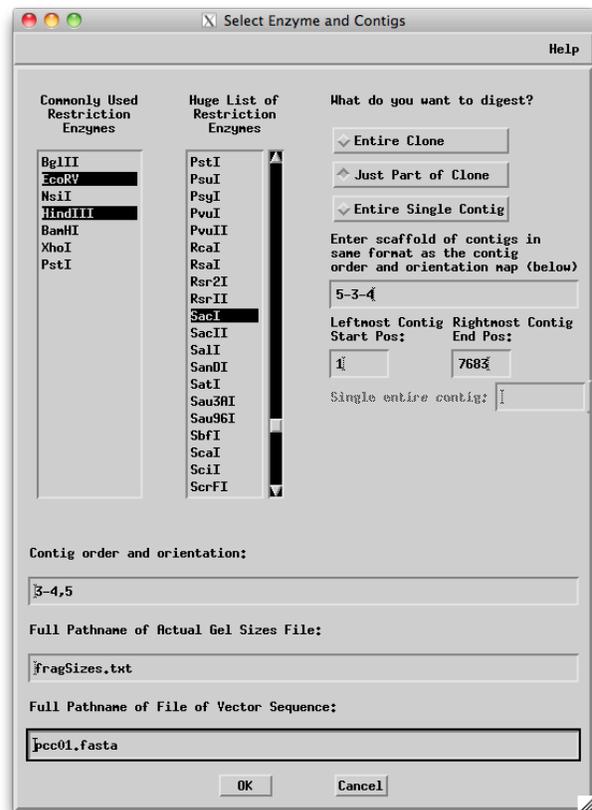


Figure 8b. By default, Consed searches the *edit_dir* directory for the vector sequence file. If the vector sequence file is in *edit_dir*, you can simply specify the name of the vector sequence file (e.g. pcc01.fasta) in the "Full Pathname of File of Vector Sequence:" field.

Since there are no read pairs that span the end of contig 5 and the beginning of contig 3, we need to confirm that Consed should modify the contig map and connect the two contigs (Figure 9).

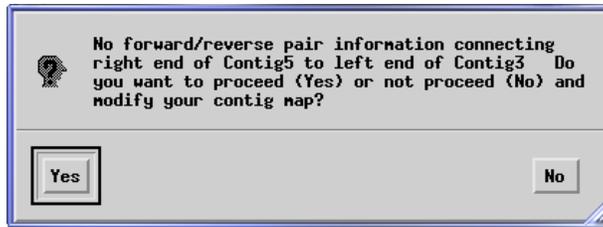


Figure 9. Click 'Yes' to ask Consed to connect the end of contig 5 with the beginning of contig 3.

If all goes well, you should see the digest for EcoRV. You can obtain the digest information for a different restriction enzyme using the 'Select Enzyme' drop-down box. For this problem, we will examine the EcoRI digest so select 'EcoRI' from the drop-down box (Figure 10). The *in silico* fragment sizes for EcoRI are based on the locations of the restriction site sequence 'GAATTC' in your assembly. In addition to the graphical output, you can also see the list of real and *in silico* digest fragments by clicking on the 'Text Output' button in the 'Display Digest' window.

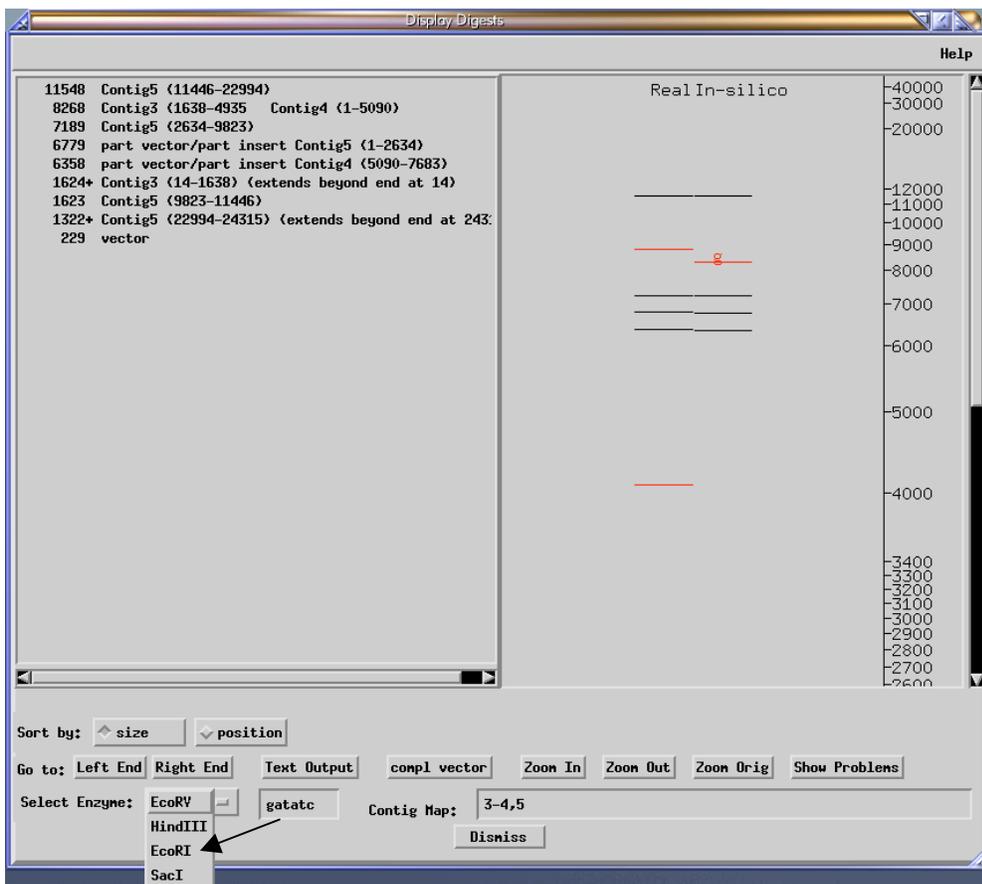


Figure 10. Choose 'EcoRI' from the 'Select Enzyme' drop-down box to evaluate the EcoRI digest.

Before evaluating a digest, we should verify that the orientation relative to the vector is correct. In some cases where Consed is unable to determine the relative orientation of the insert and the vector, we may need to complement the vector manually (using the 'compl vector' button in the digest window) when viewing the digest results. In this case, we see from the text output (click the "Text output" button to view) that the bands that consist of both vector and insert (description

of these bands begin with 'part vector/part insert' in the digest window) match relatively well with the real digest (Figure 11). Hence we are relatively confident the orientation of the insert relative to the vector is correct.

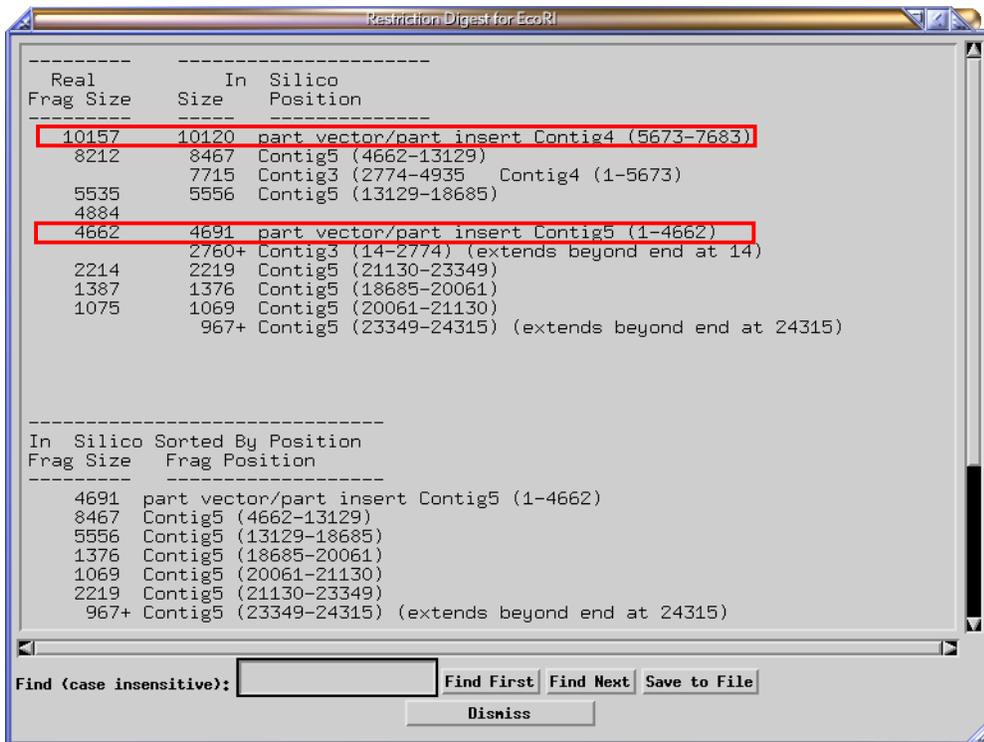


Figure 11. Text output of the EcoRI digest suggests the vector is in the correct orientation relative to the insert.

- Based on the EcoRI digest, what is the real fragment size for the region that spans the end of contig 5 and the beginning of contig 3? What are the fragment sizes of the two corresponding in silico fragments? What is the estimated size of the gap? (Hint: the difference between the real digest fragment size and the sum of the sizes of the two in-silico fragments is the estimated gap size.) Confirm your predictions using the EcoRV digest.

You can also open Assembly View from the Consed main window to get a graphical view of the cut sites in relation to the contigs in your assembly. The red boxes correspond to fragments that show significantly different sizes between the real and the *in silico* digests (Figure 12).

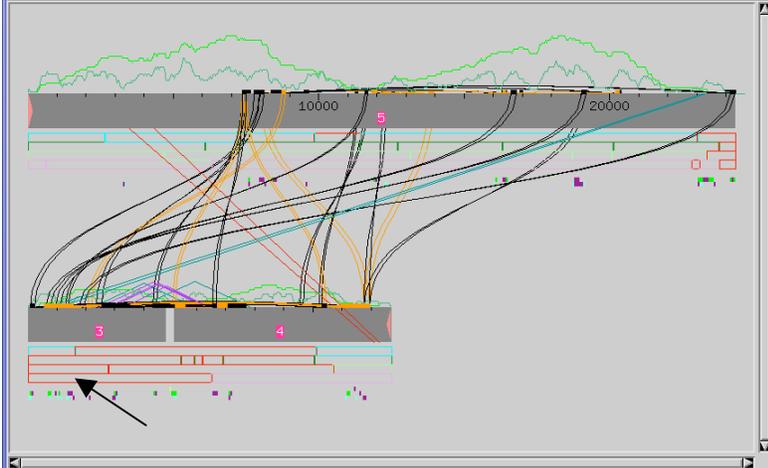


Figure 12. Red boxes demarcate fragment sizes that are significantly different in the real and the *in silico* digests.

Now that we have an estimated size of the gap between contig 5 and contig 3, we can design primers to close this gap, assuming we will get 400-700 high quality bases per read. Since the *Drosophila* projects will be sequenced using the fosmid template directly, we should use the ‘Primer from Clone (BAC, cosmid) template’ options instead of ‘Primer from Subclone’ options when picking the forward and reverse primers.

3. *There are three reaction chemistries that can be used in a sequencing reaction: BigDye, dGTP, and 4:1. What are the scenarios that will lead you to choose each of the three sequencing chemistries? What are the potential problems associated with each reaction chemistry?*
4. *Go to the aligned reads window and examine the sequences at the end of contig 5 and the beginning of contig 3. Which reaction chemistry would most likely produce the high quality sequence data required to resolve this gap? Why?*

III. Design PCR Reactions to Resolve Gaps Surrounded by Repeats

When finishing clones with high repeat density (e.g. red clones), we may be unable to pick a unique oligo for a standard sequencing reaction that is close enough to the gap (within 400 bases) for the read to provide the additional data required to resolve the gap. In these cases, we need to design PCR primers to create templates that are confined to the problematic region (i.e. between the forward and reverse primer pairs). Sequences that are repetitious in the fosmid may be unique within the PCR fragment. This template can then be used for sequencing, allowing you to generate the data required to close the gap.

Examine the Assembly View again with the *crossmatch* results (Figure 12), there is a high density of tandem and inverted repeats near the gap at the right end of contig 3 and the left end of contig 4. We will design a set of PCR primers with Consed to close this gap.

When designing PCR primers, we need to pick both the forward and reverse primers at the same time. Therefore, we should first examine both ends of the gap and determine the ideal locations for Consed to search for primers.

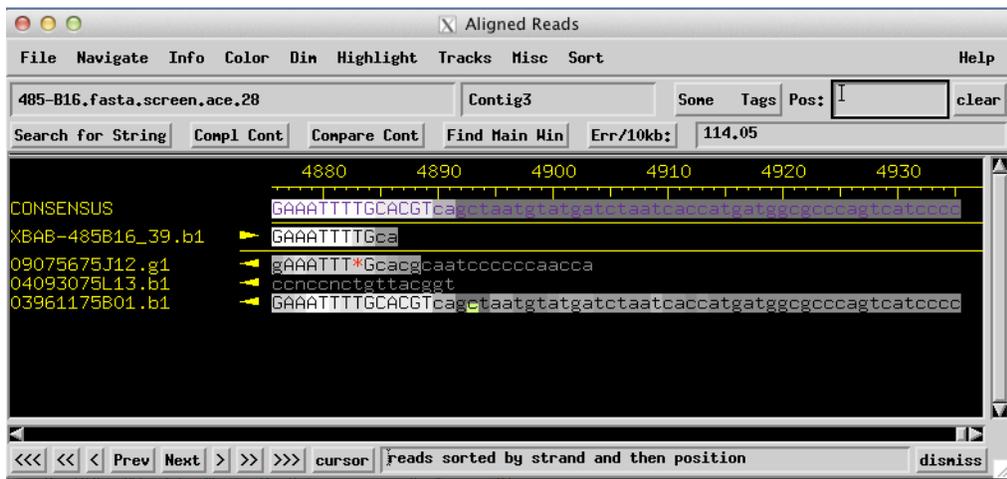


Figure 13. Low consensus quality region at the end of contig 3.

Open the Aligned Reads window for contig 3 and navigate to the right end of the contig (Figure 13). As you may expect, the end of contig 3 has relatively low consensus quality. As is the case when we design a normal primer, we want the PCR primers to anneal perfectly to the fosmid template, so we should avoid the low consensus quality region. We will ask Consed to start searching for a PCR primer starting at base 4860 of contig 3. Since the gap is to the right of contig 3, we will design this PCR primer to go in the forward direction.

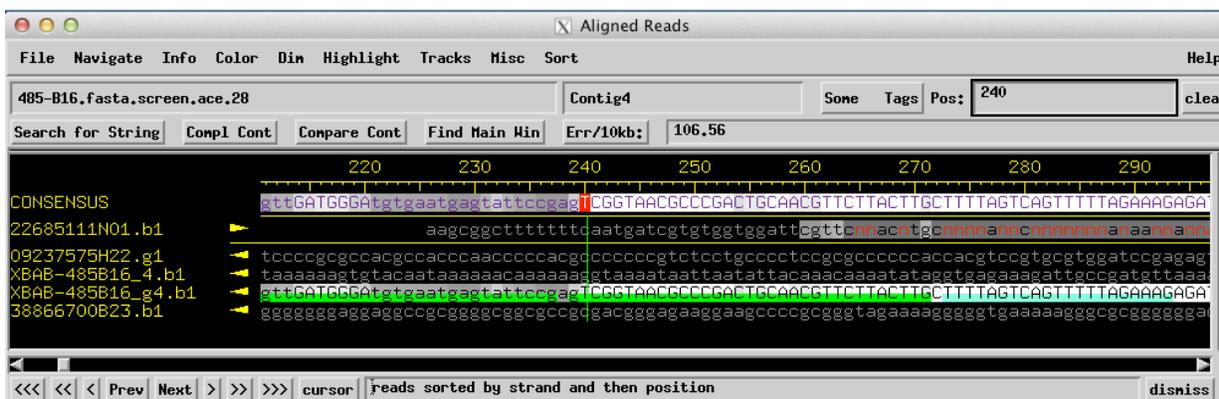


Figure 14. Only a single high quality dGTP read at the beginning of contig 4.

Now that we have determined where Consed should start looking for the forward primer, we need to do a similar evaluation of the region at the beginning of contig 4 to look for the reverse primer. Open the Aligned Reads window for contig 4 and navigate to the beginning of the contig. The first 240 bases of contig 4 are relatively poor in quality. The region from 241 to 374 has high quality data from only a single read (XBAB-485B16_g4.b1). The '_g4' in the reaction name indicates this read came from a custom oligo and used the dGTP chemistry (Figure 14). Due to potential compression problems with the dGTP chemistry, we would like to avoid relying on this read exclusively when designing the primer. Starting at position 375, we have high quality data for reads with BigDye (XBAB-485B16_4.b1) and dGTP chemistries. However, the bases at 375 and 376 have low consensus quality (phred scores of 23 and 27, respectively).

To determine if the two bases are really low quality, we will turn off the dim feature (click on ‘Dim’ on the menu bar and select ‘Dim nothing’) to see all the available data in the region. Examination of the traces shows that there are three reads (22685111N01.b1, XBAB-485B16_4.b1, and XBAB-485B16_g4.b1) that actually agree with the consensus starting at base 375 (Figure 15). From the evidence we have collected, we can conclude the consensus is probably accurate at positions 375 and 376 even though the consensus at these positions appears to be low quality.

A more detailed investigation into the 22685111N01.b1 trace indicates that phred made many base-calling errors at the beginning of this read. In particular, all the T’s (red) are identified as unknown bases (N’s) by phred. In addition, we see many cases where phred called an A instead of a T when a low ‘A’ peak overlaps with a ‘T’ (Figure 16). Therefore, in principle we can design the reverse primer much closer to the gap. However, just to be absolutely confident of the consensus, we will start looking for the reverse primer at position 375 of contig 4.

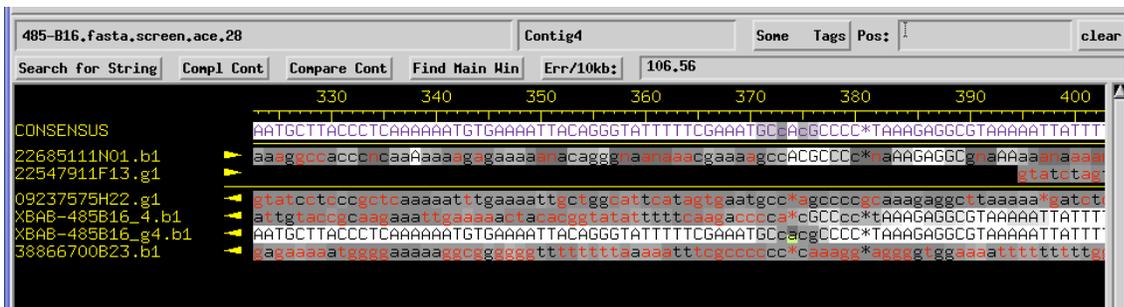


Figure 15. Dim nothing shows multiple reads that agree with the consensus starting at consensus position 375.

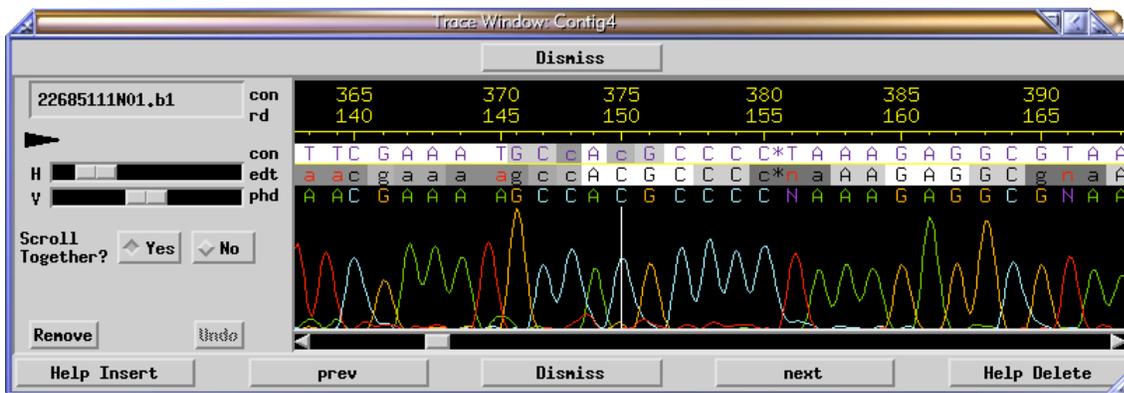
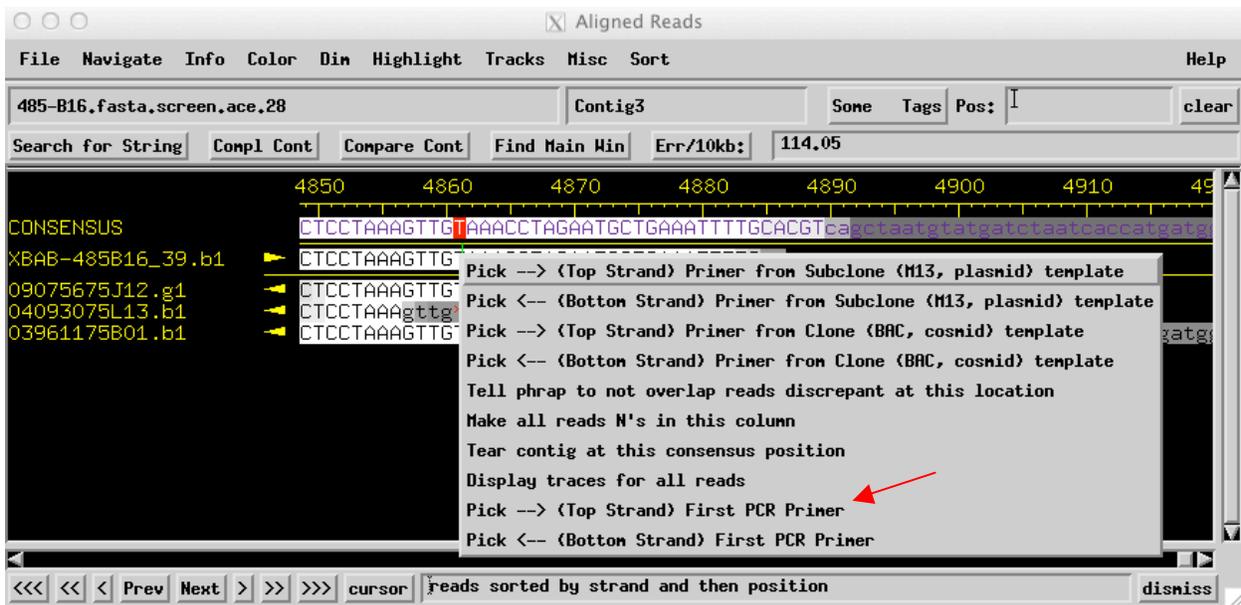


Figure 16. T’s in the trace 22685111N01.b1 were not called correctly by phred.

Since only the fosmid template is available for the Drosophila finishing projects, we will design the primers based on the consensus sequence instead of the individual subclones.

We now have all the information required to design the primers. Go back to the Aligned Reads window for contig 3 and right-click on base 4860 of the consensus to bring up the context-specific menu. Select ‘Pick → (Top Strand) First PCR Primer (Figure 17). A message should appear that asks us to pick the second primer (Figure 18) [The exact number of “Found” oligos may differ from Figure 18, but should be around 360 – 380].



S
Figure 17. Picking the forward PCR primer in contig 3.

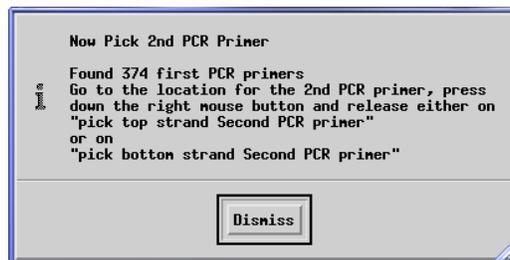


Figure 18. Message from Consed to pick the second primer

Go back to the Aligned Reads window for contig 4 and right click on base 375 of the consensus to bring up the context-specific menu. Select 'Pick <-- (Bottom Strand) Second PCR Primer' (Figure 19). You should see a list of possible PCR primer pairs (Figure 20).

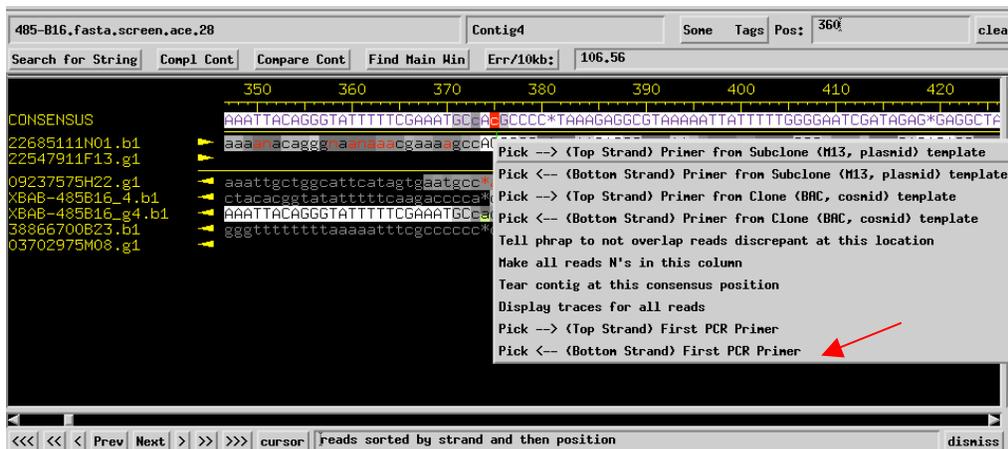


Figure 19. Picking the reverse PCR primer from contig 4

pair #	distance between contig	primer1 left	primer1 right	primer2 left	primer2 right	melting p1	melting p2	primer1	primer2
1	2254	Contig3 4091	4112	Contig4 431	457	55	56	atcgaattttcgctgatataac	gaagcttcaa
2	2254	Contig3 4091	4112	Contig4 431	458	55	56	atcgaattttcgctgatataac	agaagcttcaa
3	2421	Contig3 3924	3945	Contig4 431	457	55	56	tatgctgaaattgtgtcatt	gaagcttcaa
4	2421	Contig3 3924	3945	Contig4 431	458	55	56	tatgctgaaattgtgtcatt	agaagcttcaa
5	2456	Contig3 4091	4112	Contig4 633	659	55	57	atcgaattttcgctgatataac	cgtatttgt
6	2457	Contig3 4091	4112	Contig4 634	659	55	55	atcgaattttcgctgatataac	cgtatttgt

Figure 20. Large list of available PCR primers for the gap between contigs 3 and 4

There are many factors one could consider when designing PCR primers to maximize the probability of a successful reaction. Some strategies for optimizing PCR primers are described by Innis MA and Gelfand DH in the textbook *Optimization of PCRs*¹:

1. The length of primers should be between 17 and 28 bases
2. GC base composition of the primers should be between 50 to 60%
3. Melting temperature should be between 55 and 80 degrees Celsius
4. Both primers should have a similar melting temperature
5. 3' end of the primers should not have a run of more than 2 C's or G's
6. 3' end of the primers should end in G or C (GC clamp)
7. Primer pairs should be as close to the gap of interest as possible

5. *Based on these criteria, which primer pairs would you use to generate a PCR fragment between the end of contig 3 and the beginning of contig 4? Justify your answer. (Note: the primer candidates will not satisfy all the criteria listed above. Hence you should identify primer pairs that satisfy most of the criteria and be able to defend your answer.)*

Project 430-O17

IV. Resolving High Quality Discrepancies

For this part of the exercise, we will look at another project, 430-O17. Quit Consed and navigate to the *edit_dir* of the 430-O17 project in the xterm (e.g. type `cd ../../430-O17/edit_dir`). Open the ace file *430-O17.fasta.screen.ace.30* (about 30th on list).

In order for a fosmid project to be considered finished it must first satisfy all the requirements in the finishing checklist. The checklist is available for download at http://gcp.wustl.edu/repository/course_materials_WU/sequence_improvement/Finishing_Checklist.pdf. One of the items on the checklist is to resolve all high quality discrepancies. Regions tagged by Consed as high quality discrepancies are regions with one or more reads with high quality bases that disagree with the consensus. High quality discrepancies could be attributed to misassemblies (different copies of a repeat piled on top of each other), polymorphisms, or bases miscalled by phred. The finisher's job is to resolve any misassemblies and provide a likely explanation for the remaining high quality discrepancies.

¹ Innis MA, Gelfand DH, Sninsky JJ, White TJ (eds.): **PCR Protocols: A Guide to Methods and Applications**. Academic Press, Inc., San Diego, 1990.

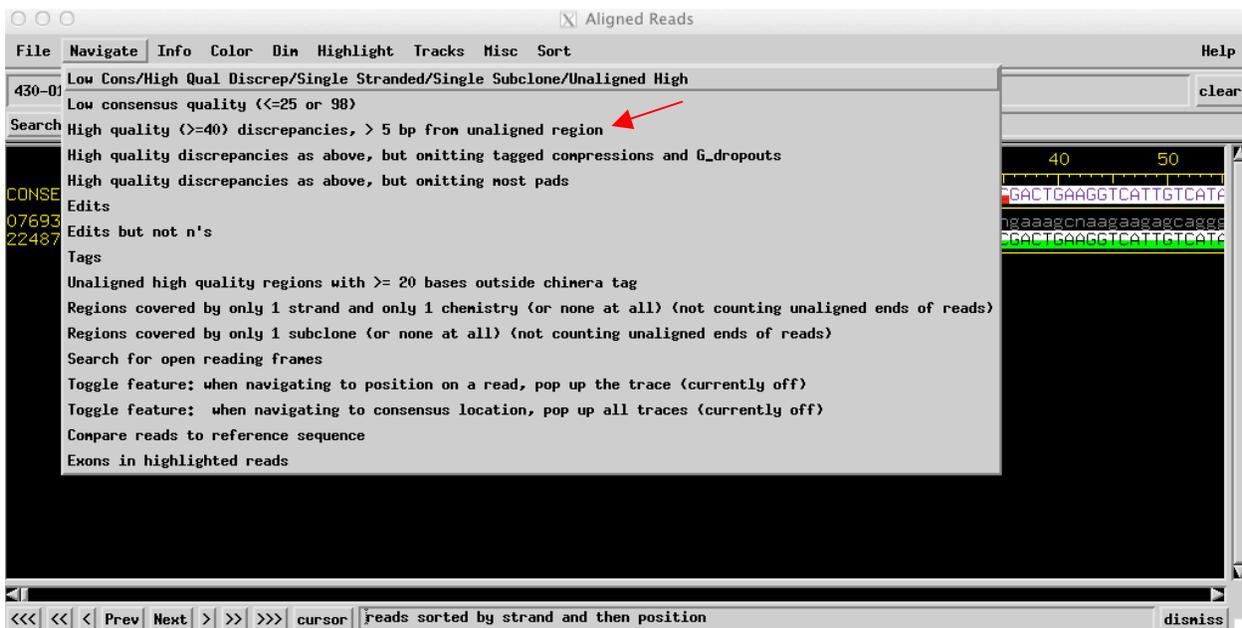


Figure 21. Use the 'High quality (≥ 40) discrepancies, > 5 bp from unaligned region' option in the aligned reads window to navigate to all of the regions with high quality discrepancies in the contig.

6. Double click on 'contig 6' from the Consed main window. In the aligned reads window for contig 6, click on 'Navigate' and select the 'High quality (≥ 40) discrepancies, > 5 bp from unaligned region' option to view the list of high quality discrepancies on contig 6. How many high quality discrepancies are there in contig 6? Where are these high quality discrepancies in contig 6? Repeat the same procedure for contig 7 and list the locations of all high quality discrepancies in contig 7.

You can jump to a specific region of the assembly by double clicking on an item in the custom navigation window. Double-clicking on the first item in the list will open the "Aligned Reads" window with the cursor placed at the discrepant base (position 4457 in contig 6). To investigate this first discrepancy in contig 6, click on the base with the middle mouse button to open the trace window. This will show the electropherogram generated by the sequencing machines and the corresponding bases that were called by phred. By default, you can open a maximum of four traces simultaneously. Middle-click on two other high-quality reads that agree with the consensus sequence to incorporate them into the trace window.

For the discrepant base at 4457 of contig 6, we see from the trace window that the read "03802575D08.b1" has an extra "T" compared to all other reads in this region and the consensus sequence. Clicking on the base in the "Aligned Reads" window will show the quality of the base in the status bar (Figure 22). This extra T is very high quality, with a phred score of 68. The surrounding bases also are high quality (in particular, the other 5 T's in this read also have a phred score of 68).

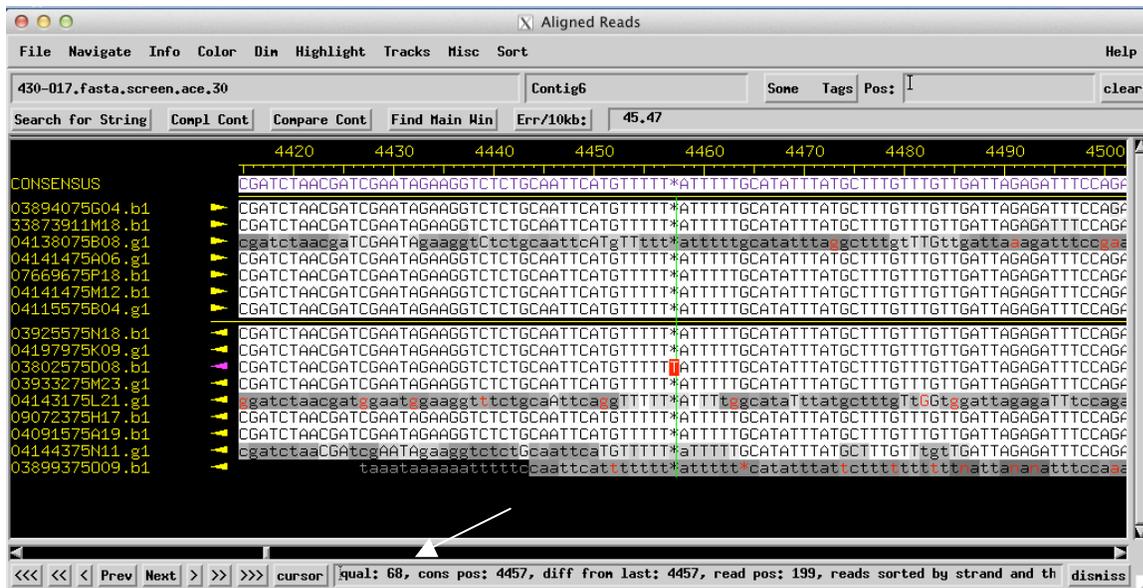


Figure 22. The discrepant T has a phred score of 68

The phred scores are consistent with the data in the trace window, where we see strong and distinct peaks in the 03802575D08.b1 read (Figure 23). The spacing between base calls in this read shows a regular pattern and is consistent with the spacing of the other high quality regions of this read. Scanning left and right, we observe a good match with the consensus sequence throughout the entire length of this read. Hence the evidence we have collected suggests that this “T” is a real high quality discrepancy and not a low quality base that was mistakenly given a high quality score by phred.

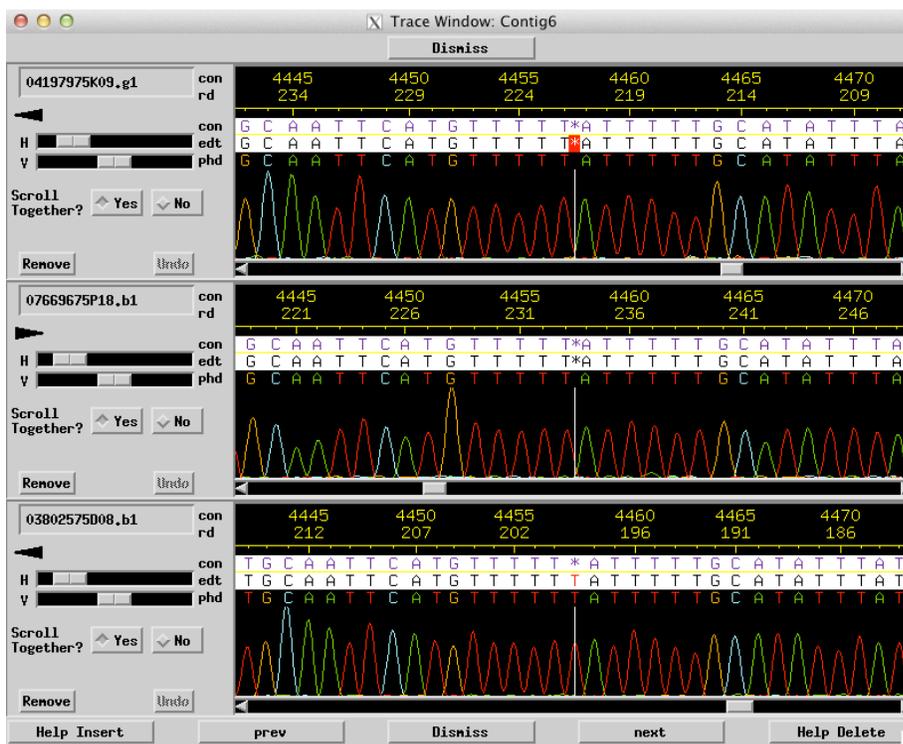


Figure 23. Trace with the discrepant T (03802575O08.b1) shows strong distinct signal.

These types of discrepancies can be created in several ways. When sequencing using the whole genome shotgun strategy, this type of discrepancy can arise from differences (polymorphisms) between the two chromosomal copies found in diploid organisms. For base discrepancies caused by polymorphisms, one often finds about half the reads with one allele and the other half of the reads with the alternative allele. However, in this case there is only a single read that does not agree with the other sequences and the consensus sequence so it is unlikely that this discrepancy is a genuine polymorphism. The most likely explanation is that a base pair change is introduced into the template sometime during the process of going from whole organism to final sequence. This type of mutation is known as a “growth difference” since changes of this type historically happened during growth of the template in *E. coli*. Modern sequencing techniques typically skip the step of cloning and growing in *E. coli* but have other steps in which changes like this can be introduced.

Because our goal is to create the best consensus sequence we can with the evidence available, we should not modify the consensus sequence, as all the reads except for one do not have the extra “T” in the sequence. However, since the “T” appears to be a high quality base according to the phred score and the trace, we should add a comment tag to this base to denote the fact that we have examined the high quality discrepancy and believed the discrepancy is caused by a growth difference.

To add the tag, middle click on the discrepant T in the read 03802575D08.b1 in the “Aligned Reads” window to open the trace. Then middle-click on the discrepant T inside the trace window and select “Add Tag” inside the “What to Do with Selection” window. Select “comment” and enter a comment in the textbox such as “High quality discrepancy possibly caused by growth differences in *E. coli*” (Figure 24).

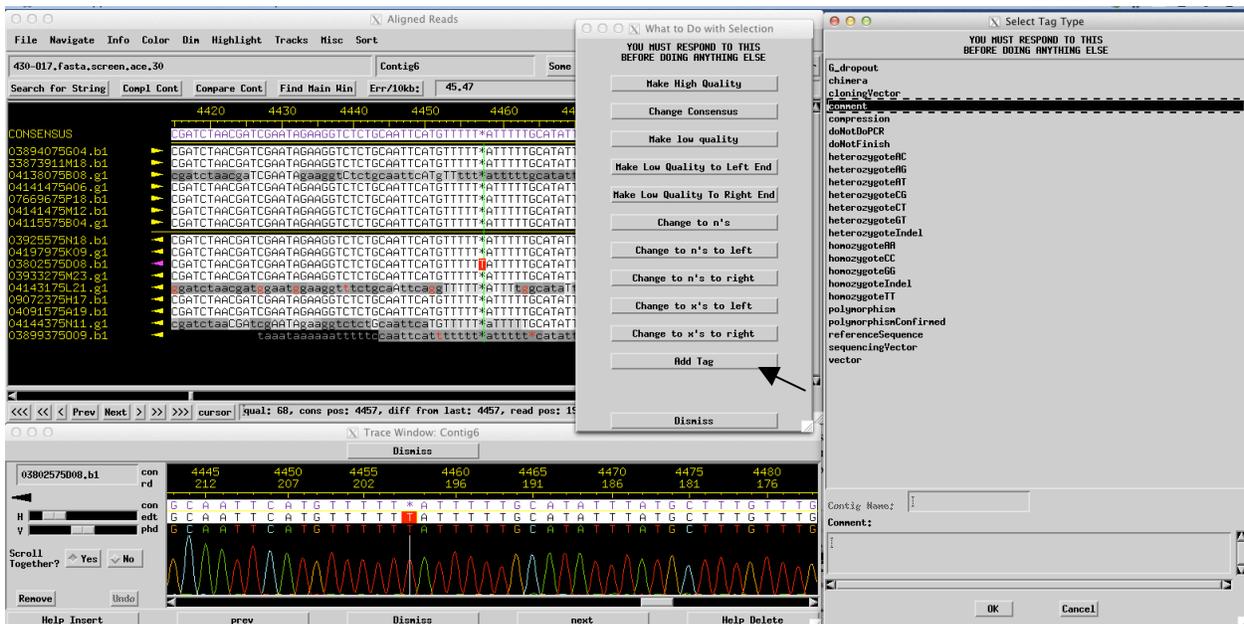


Figure 24. Add a comment tag to the discrepant T in the read 03802575008.b1 at position 4457

In cases where you disagree with Consed and would like to correct an error in the consensus sequence, you can change the consensus sequence from the “Trace Window”. Middle-click on the base in the Trace Window (you can also hold on to the middle mouse button and move the mouse to select multiple bases) to bring up the “What to Do with Selection” window. You can click on ‘Change Consensus’ to replace the consensus sequence in the selected region with the bases called in a particular read (Figure 25). Besides changing the consensus, you can also make the selected bases low quality, change bases to n’s (unknown bases), mask the bases in a read as x’s (vector) or add tags to a read through this selection menu.

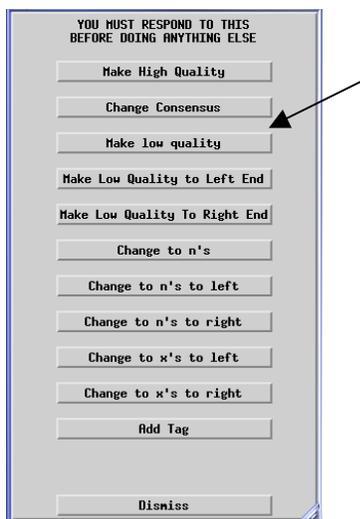


Figure 25. Bring up the ‘Change Consensus’ option by middle clicking on a base in the trace window.

Another common cause of high quality discrepancies is bases that were miscalled by phred. For examples of miscalled bases, return to the navigation list of high quality discrepancies and navigate to base 14,072 of contig 6 (Figure 26). Looking at the trace for the discrepant read (03923075C08.g1) concurrently with two other high quality reads in this region, we see that the quality of the discrepant read is much lower than the other reads that agree with the consensus (Figure 27). Furthermore, if you measure the distance between the three A’s from position 640-642 and compare it with the distance between the bases immediately downstream of the run of poly-A (CGT from positions 631-633), we see that the three A’s are much closer together. Therefore, the evidence we have collected thus far indicates we should be skeptical of the number of A’s called in the discrepant read. Hence we will make the high quality A that disagrees with the consensus a low quality base instead.

V. Resolving Regions with Low Consensus Quality

Another problem you may find in your finishing project is regions with low consensus quality. These are typically regions in the assembly that are difficult to sequence or have insufficient sequencing coverage (e.g. not enough high quality reads in a region) for Consed to be confident of the consensus sequence. The sequence improvement standard used for the mouse genome requires every base in the consensus to have a quality score of phred 25 or above for regions with sequencing reads from both strands. Bases in regions that are single stranded must have a phred score of 30 (error in 1 base per 1000 bases) or above. We will change the Consed settings to classify all regions with phred 30 or below as low quality. If manual inspection shows that a low quality region has a phred score of 25 or above with sequencing reads from both strand, we can then reclassify the region as high quality according to the sequence improvement standard.

From the Consed Main Window, click on the “Options” tab and select “General Preferences” (Figure 28). Change the first field “Threshold for Low Consensus Quality” to 30 and click ‘Apply & Dismiss’ (Figure 29).

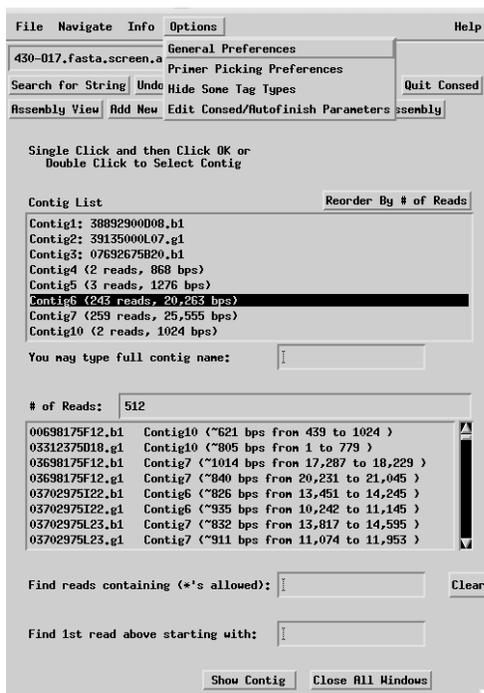


Figure 28. Use the ‘General Preferences’ option to change the quality score considered as low quality.

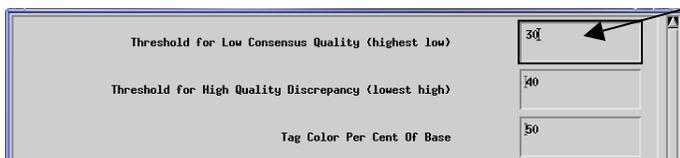


Figure 29. Change the ‘Threshold for Low Consensus Quality’ from 25 to 30.

Go back to the Consed main window and open the Aligned Reads window for contig 7. Select ‘Navigate’ and click on the option ‘Low consensus quality (<=30 or 98)’ (Figure 30). This will generate a list of positions with consensus quality 30 or below (Figure 31a).

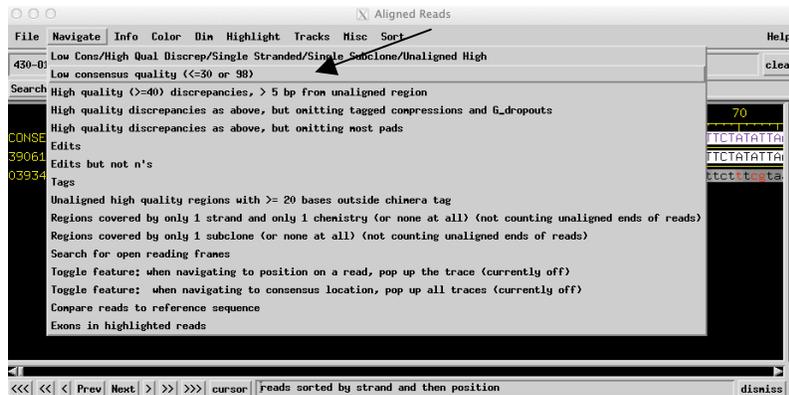


Figure 30. Use the 'Low consensus quality' navigation option to see the list of low quality bases in contig 7.

Contig Name	Read Name	Consensus Positions	
Contig7	(consensus)	1-30	base quality below threshold
Contig7	(consensus)	51	base quality below threshold
Contig7	(consensus)	53-59	base quality below threshold
Contig7	(consensus)	7662-7664	base quality below threshold
Contig7	(consensus)	7666-7668	base quality below threshold
Contig7	(consensus)	7672	base quality below threshold
Contig7	(consensus)	7681	base quality below threshold
Contig7	(consensus)	7683	base quality below threshold

Figure 31a. Numerous regions in contig 7 are classified as low quality.

In this case, we see there are numerous regions that do not meet our standard of phred 30 (Figure 31a). Double-clicking on each item in the 'Low consensus quality' window will open the 'Aligned Reads' window for the corresponding problem area. While the list of low consensus regions is quite long, many of these regions can be clustered together, which will shorten the list of the problem areas considerably. For example, the three problematic regions from 19424-19429, 19435-19441, and 19453-19459 can be grouped together (19424-19459) and could be resolved by a single read.

Low quality regions can be resolved by editing the consensus (in cases where sufficient data is available) or by generating additional sequence data.

8. *How many of these low quality clusters did you define for contig 7? For each of these low quality clusters, design oligos (from both sides of the problematic region if possible) to generate additional sequencing data. Which reaction chemistries would you use for each of your oligos? Explain why the reaction chemistry you chose would likely work better for each oligo you designed.*

Designing oligos to resolve low quality regions can sometimes be a non-trivial process. In cases where Consed cannot pick a primer, you would see an error dialog with the message "could not figure out what to relax" and a window that explains why Consed could not design a primer. In highly repetitive regions, Consed may be unable to pick a primer because all of the possible primer candidates could anneal to multiple locations in the assembly. For example, the low quality sequences between 7643-7833 in contig7 are in the middle of a repeat. In order to design a primer, we would need to identify a unique region that is close to this problem area. We can visualize the repeat structure in this area using *crossmatch*. Open 'Assembly View' from the Consed main window, then click on "What to Show" -> "Sequence Matches" -> "run

crossmatch.” Click on the “Zoom in” button at the bottom left corner of the Assembly View window and navigate to the region around 7643 in contig 7. If you hover your mouse over the lines connecting the blocks they will turn yellow (Figure 31b), clicking on the yellow line will open a window with more detailed information about the sequence match (e.g. size, location, similarity; see below for more details).

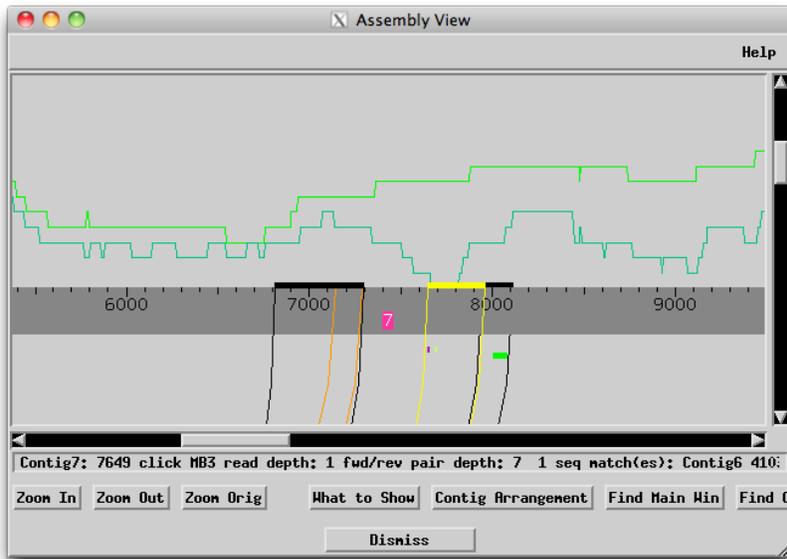


Figure 31b. Sequence matches by crossmatch show repeats that span the region with low consensus quality.

The *crossmatch* result suggests there is a small unique region (~300bp) upstream of the region with low consensus quality. Consed may be able to pick primers within this unique region.

VI. Use ‘Add New Reads’ to Incorporate New Sequence Data into an Assembly

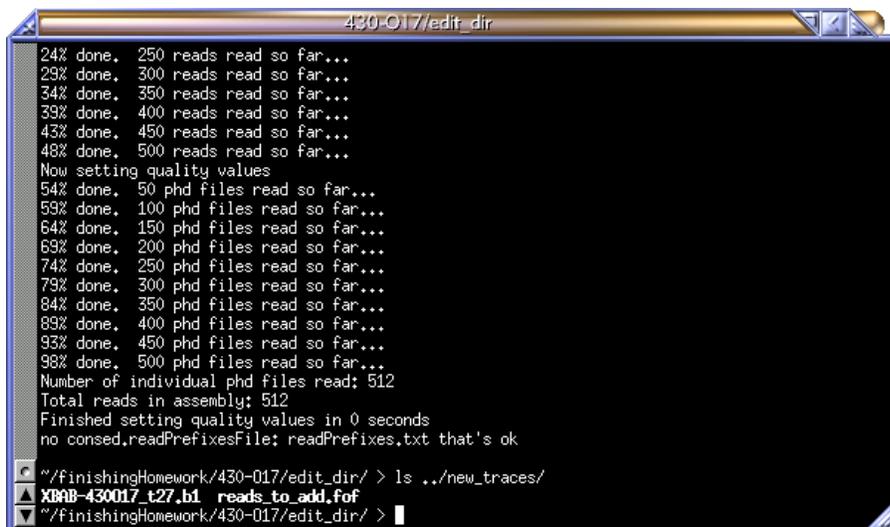
We have previously designed a custom oligo (430-O17.27) to resolve the low quality cluster from 7662 to 7833 in Contig7. The new sequencing read is available in the *new_traces* folder inside your project directory. In this part of the exercise, we will try to incorporate this new read into our assembly.

In order to incorporate new data into an assembly we need to put the new trace file(s) into the *chromat_dir* directory so that Consed can find the new trace. If you have access to a graphical environment (e.g. on Mac OS X or Linux with a window manager), you can simply drag and drop the trace file (*XBAB-430O17_t27.b1*) from the *new_traces* folder into the *chromat_dir* folder. Similarly, move the fof file (*reads_to_add.fof*) from the *new_traces* folder to the *edit_dir* folder.

However, if you only have access to the command-line, then we need to move these files using the xterm. Go back to the xterm you used to launch Consed and press return, you should get the command prompt back. We can use the *ls* command to see the contents of the *new_traces* directory. Type the following into the xterm command prompt (Figure 32):

```
ls ../new_traces/
```

Press return and you should see two files in the directory: *XBAB-430O17_t27.b1* and *reads_to_add.fof*. The *XBAB-430O17_t27.b1* file is the new read we wish to incorporate into the assembly. The file *reads_to_add.fof* contains a list of filenames that corresponds to the new reads we wish to add to the assembly.



```
430-O17/edit_dir
24% done. 250 reads read so far...
29% done. 300 reads read so far...
34% done. 350 reads read so far...
39% done. 400 reads read so far...
43% done. 450 reads read so far...
48% done. 500 reads read so far...
Now setting quality values
54% done. 50 phd files read so far...
59% done. 100 phd files read so far...
64% done. 150 phd files read so far...
69% done. 200 phd files read so far...
74% done. 250 phd files read so far...
79% done. 300 phd files read so far...
84% done. 350 phd files read so far...
89% done. 400 phd files read so far...
93% done. 450 phd files read so far...
98% done. 500 phd files read so far...
Number of individual phd files read: 512
Total reads in assembly: 512
Finished setting quality values in 0 seconds
no consed.readPrefixesFile: readPrefixes.txt that's ok
~/finishingHomework/430-017/edit_dir/ > ls ../new_traces/
▲ XBAB-430017_t27.b1 reads_to_add.fof
▼ ~/finishingHomework/430-017/edit_dir/ > |
```

Figure 32. The *new_traces* directory contains a new read *XBAB-430O17_t27.b1* and the *reads_to_add.fof* file

To move the trace from the *new_traces* directory to the *chromat_dir* directory, we will issue the following command in the xterm:

```
mv ../new_traces/*.b* ../chromat_dir/
```

Press return and after a brief pause you should get the prompt back – which indicates the files have been successfully moved to the new location.

Now that we have moved the new trace to the correct location, we need to create a new assembly (ace file) that includes the new trace. There are two ways to incorporate new reads into an assembly: run the *phredPhrap* script or use the ‘Add New Reads’ function within Consed. The *phredPhrap* script uses all the reads inside the *chromat_dir* directory to create a completely new assembly. The ‘Add New Reads’ function, on the other hand, attempts to align each of the new reads with the current assembly and incorporating them into the region with the best alignment. Both techniques of incorporating new reads into an assembly have advantages and disadvantages.

Ideally, we would like to generate a completely new assembly using all the available data. Therefore in most cases we should reassemble the project by typing *phredPhrap* inside the *edit_dir* of your project directory to create an assembly with the new reads. However, for more complicated clones (e.g. clones with repetitious sequences within the insert) *phrap* (the assembler) often makes mistakes assembling the clone. You or a previous finisher may have done some tears or force joins to manually resolve the problematic regions. If one were to run the *phredPhrap* script to generate a new assembly, *phrap* will likely repeat the same mistakes and you would be forced to do the tear and force joins again.

You can avoid redoing tears and force joins by using the “Add New Reads” function within Consed to incorporate new reads into an assembly. However, because “Add New Reads” simply take the new reads and align them against the current assembly, the new reads are more likely to be incorporated into the wrong place within the assembly compared to running the *phredPhrap* script. In addition, “Add New Reads” may also introduce artifacts into the assembly (such as insert extra pads, fail to recalculate consensus scores or fail to update the consensus sequence). Hence the ‘Add New Reads’ function should be used with caution and only when it is judged worthwhile.

In this case, the previous finisher has already done some tears and force joins. To preserve those tears and force joins we will use the “Add New Reads” function to incorporate the new read into the assembly. “Add New Reads” requires a .fof (file of files) file which contains the list of filenames that correspond to the new reads we wish to add to the assembly. Because Consed will by default look for the .fof file inside *edit_dir* in your project directory, we should move the *reads_to_add.fof* file from the *new_traces* folder to *edit_dir* by typing the following command into the xterm:

```
mv ../new_traces/*.fof ../edit_dir/
```

Press return and after a brief pause you should get the prompt back (Figure 33).

```
430-017/edit_dir
29% done. 300 reads read so far...
34% done. 350 reads read so far...
39% done. 400 reads read so far...
43% done. 450 reads read so far...
48% done. 500 reads read so far...
Now setting quality values
54% done. 50 phd files read so far...
59% done. 100 phd files read so far...
64% done. 150 phd files read so far...
69% done. 200 phd files read so far...
74% done. 250 phd files read so far...
79% done. 300 phd files read so far...
84% done. 350 phd files read so far...
89% done. 400 phd files read so far...
93% done. 450 phd files read so far...
98% done. 500 phd files read so far...
Number of individual phd files read: 512
Total reads in assembly: 512
Finished setting quality values in 0 seconds
no consed.readPrefixesFile: readPrefixes.txt that's ok

~/FinishingHomework/430-017/edit_dir/ > ls ../new_traces/
XRAB-430017_t27.b1_reads_to_add.fof
~/FinishingHomework/430-017/edit_dir/ > mv ../new_traces/*.b* ../chromat_dir/
~/FinishingHomework/430-017/edit_dir/ > mv ../new_traces/*.fof ../edit_dir/
~/FinishingHomework/430-017/edit_dir/ > █
```

Figure 33. Move the traces to the *chromat_dir* and the .fof file to the *edit_dir*

After moving the files to the appropriate folders, we can incorporate the new reads by following these steps:

1. Click on the ‘Add New Reads’ button in the Consed main window
2. Under ‘Files’ select ‘reads_to_add.fof’ (you may need to scroll to the bottom of the list.)
3. Make sure that “Yes” is selected for both questions at the top of the window (this option should be selected by default).
4. Click “OK”

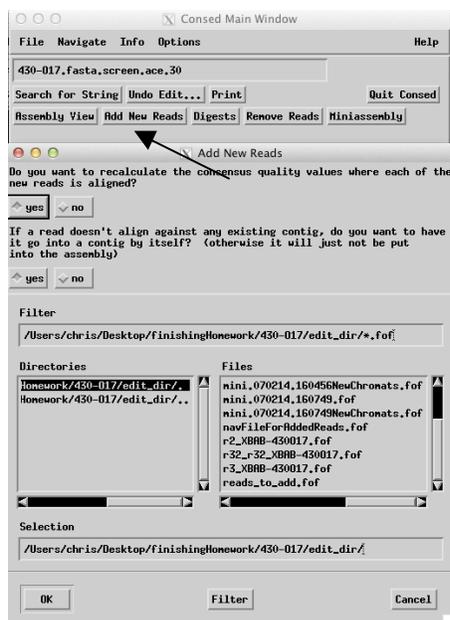


Figure 34. Select the *reads_to_add.fof* file in the “Add New Reads” dialog

If all goes well, you should see two windows: the “New Reads in Assembly” navigation window allow us to directly jump to the new read and the “Reads Added” window shows the list of reads that were successfully added (Figure 35).

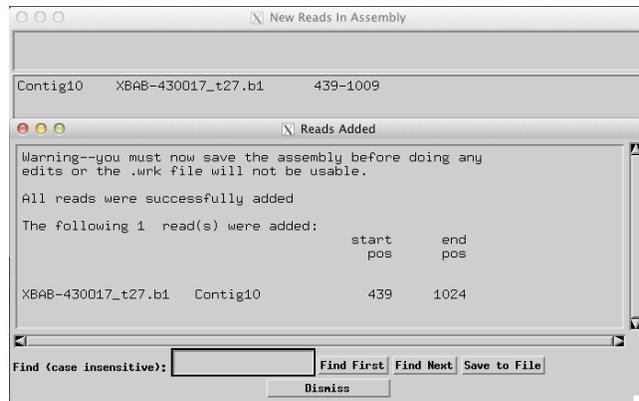


Figure 35. The “New Reads in Assembly” and “Reads Added” window shows that the new read XBAB-430017_t27.b1 was added to contig 10 (439-1024).

After incorporating the new reads into an assembly, the first question we should ask is whether the new reads were added to the locations we expect. The “Reads Added” window shows the read XBAB-430017_t27.b1 was added to contig 10. From the name of this read, we can see that this read is generated from a custom oligo since it has the prefix XBAB. The next part of the read name is 430017, which corresponds to the name of our project. The *_t* in the read name indicates that the 4:1 chemistry is used to generate this read. The 27 in the read name indicate this read is generated by the custom oligo 27. For additional information on the naming conventions for custom oligos, refer to the “Calling Additional Reads” document available on the GEP website under “Sequence Improvement”.

Naturally, we expect the read generated by oligo 27 to be incorporated downstream of the 3' end of oligo 27 in the assembly. Therefore, to determine if Consed incorporated the read correctly, first we need to find oligo 27 (the full name of the oligo is 430-O17.27). Go back to the Consed main window and select 'Navigate' from the main menu, then select 'Tags in all contigs'. Select 'oligo' from the list and click 'OK' to see all the oligo tags in the assembly.

9. *Where is oligo 430-O17.27 in the assembly? Did Consed correctly incorporate the XBAB-430O17_t27.b1 read into the assembly? If so, justify your answer. If not, where should the read go in the assembly? If the read is in the wrong position, tear it out of its current incorrect position and join the read in the proper contig downstream of the oligo that was used to create the read. Did the read successfully resolve the low quality regions it was designed to cover? (Hint: Refer to the document "Using Consed Graphically" for instructions on how to tear and join contigs).*

If you do tear and force joins, be aware that the contig numbers will change. The final contig numbers after you have completed question 9 depends on the number of tears and force joins you have done. Therefore, to keep the names of the contigs consistent for the rest of the exercise, we will work with the original ace file *430-O17.fasta.screen.ace.30*. To do this, you should save your current assembly and exit Consed. Relaunch Consed and open the ace file *430-O17.fasta.screen.ace.30*.

VII. Using Restriction Digests to Resolve an Anomaly in the Assembly

We have previously used the restriction digest to estimate gap size. In this part of the exercise, we will use the digest information to detect anomalies in the assembly. Open Assembly View to get an overview of the assembly with repeats identified by *crossmatch*. There are two major contigs (contigs 6 and 7) in the assembly (Figure 36).

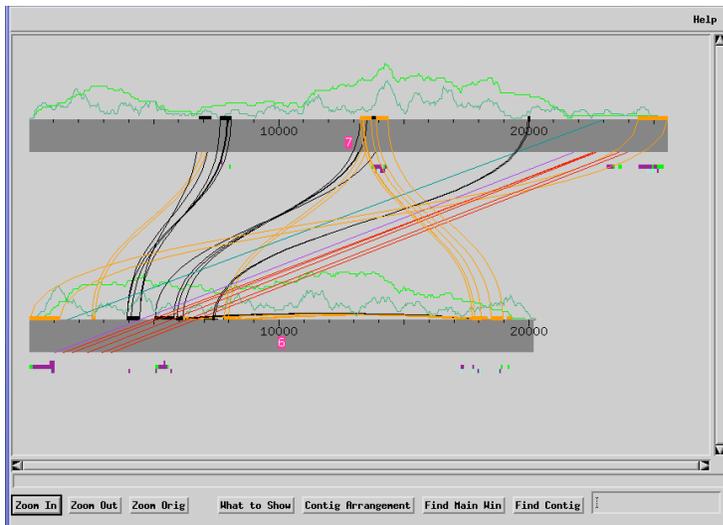


Figure 36. Two large unspanned contigs in the assembly for project 430-O17

The result from *crossmatch* indicates there is a large (1180 bases) segment of putative tandem repeats (highlighted in orange) at the end of contig 7 and the beginning of contig 6. Clicking on the hit in Assembly View shows that the two repeats have 99.5% sequence similarity (Figure 37). However, we should also remember that the fosmid reads came from a whole genome shotgun assembly and may contain polymorphisms. In cases of polymorphisms, phrap (the assembler) may tear a single region apart into two contigs, separated according to the allele that is present in the read.

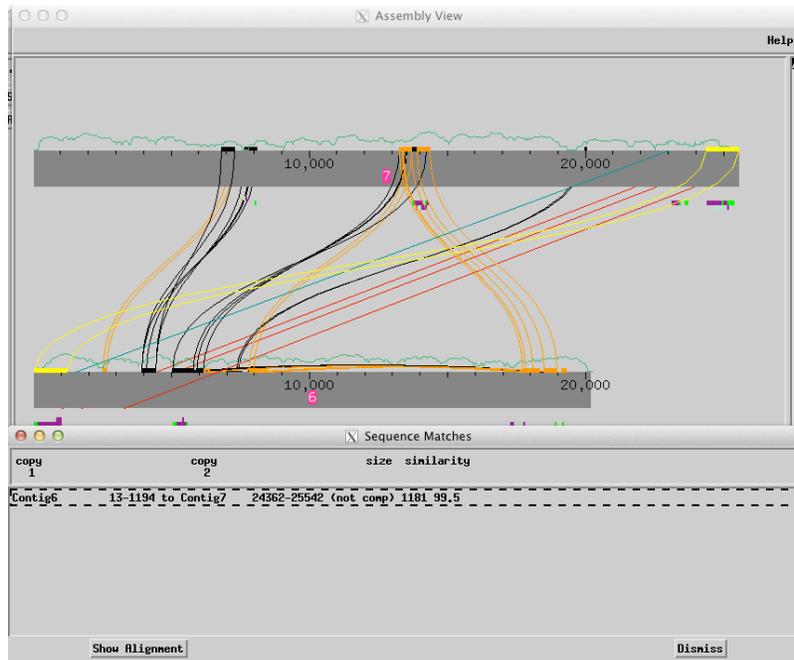


Figure 37. Large copy of tandem repeats found by *crossmatch* at the left end of contig 7 and right end of contig 6.

We will use the restriction digest data to determine which of the two hypotheses (distinct tandem repeats or polymorphism) is the more likely explanation for this region. We will follow the same steps as described previously to set up the display for our restriction digest, modifying the path to the vector sequence and the contigs to digest.

1. Click on the 'Digests' button in the Consed main window
2. Under the section 'Huge List of Restriction Enzymes', select 'EcoRI' and 'SacI'
3. Under 'What do you want to digest?' select 'Just Part of Clone'.
4. In the 'Enter scaffold of contigs' section, type '7-6' to denote the fact that the end of contig 7 should be connected to the beginning of contig 6.
5. In the 'Leftmost Contig start Pos' field, enter '1', and in the 'Rightmost Contig End Pos' field, we will enter the coordinates corresponding to the end of contig 6 – '20263'
6. Change the 'Full Pathname of File of Vector Sequences:' to: *pcc01.fasta* (Figure 38). Click "OK" to view the digest information.
7. Click 'Yes' to connect the end of contig6 with the beginning of contig 7 (Figure 39).

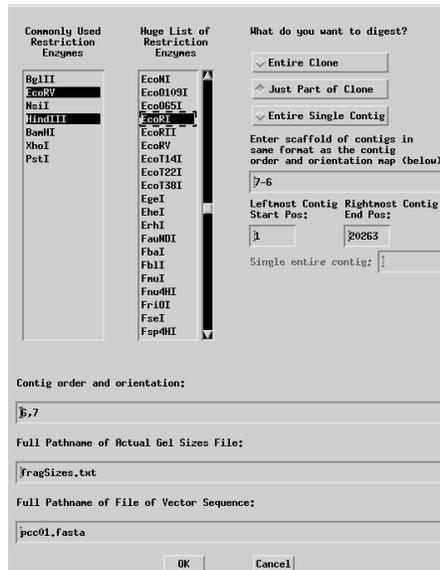


Figure 38. Configuration of the real and *in silico* digest for project 430-O17

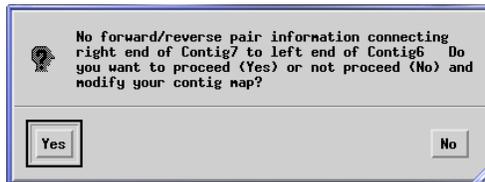


Figure 39. Click yes to confirm we would like Consed to connect contig 7 with contig 6.

Again, be aware of the potential problems with the orientation of the inserts relative to the vector (in this case, for both contigs 6 and 7). Based on the 'Text Output', it again appears that the insert is in the correct orientation relative to the vector.

For this project, we do not have restriction digest data for two of the enzymes (EcoRI and SacI). In the digest window for these two restriction enzymes, we see all the *in silico* bands do not have corresponding bands in the real digest (Figure 40). Hence our analysis will be based only on the digest data from EcoRV and HindIII.



Figure 40. Restriction digest information for EcoRI and SacI are unavailable for this project

10. Based on the comparison of the real and *in silico* digests for *EcoRV* and *HindIII*, do you believe there is a gap between the end of contig 7 and the beginning of contig 6? Justify your answer using the restriction digests results.

VIII. Using Force Joins to Merge Contigs

One way to determine if contig 7 and contig 6 actually overlap is to force join the two contigs and see if the *in silico* digests of the merged contig are consistent with the real restriction digests.

Go to the Consed main window and open Assembly View. Click on the large putative tandem repeats identified by *crossmatch* at the end of contig 7 and the beginning of contig 6 to bring up the 'Sequence Matches' window again. Select the first entry in 'Sequence Matches' and click on the 'Show Alignment' button to align the end of contig 7 with the beginning of contig 6 (Figure 41).



Figure 41. Click on 'Show Alignment' to display the alignment for the tandem repeat identified by *crossmatch*

As we expect from the 99.5% similarity reported by *crossmatch*, the alignment looks relatively good between contig 6 and contig 7. Besides the low quality regions, there are four high quality discrepancies that may indicate that these are two distinct copies of a repeat (Figure 42). If we decide to join these two contigs together, these mismatches may show up as high quality discrepancies. Our hypothesis is that these mismatches are either base miscalls or polymorphisms.

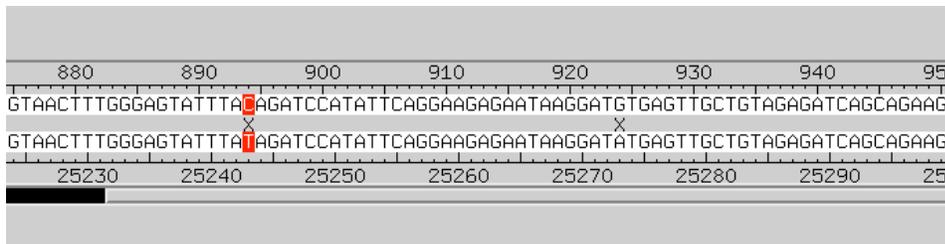


Figure 42. Example of a high quality discrepancy between the putative tandem repeats in contig 6 and contig 7.

Click 'Join Contigs' to force join the right end of contig 7 with the left end of contig 6. Following the force join, a new contig 8 should be created that consists of 44612 bases. Go back to the Consed main window, **save the assembly**. Click on the 'Close All Windows' button before you proceed to the next step. Open 'Assembly View' and run *crossmatch* ('What to Show' -> 'Sequence Matches' -> 'run crossmatch') to identify all the repeats in the contig.

We will now use the digest information to confirm the force join. The configuration to view the digest is similar to the ones we have set up earlier. The only change is that instead of contig 6 and 7, we should digest contig 8 instead.

In the 'What do you want to digest?' field select the 'Entire Single Contig' option. In the 'Single entire contig:' field, type '11' to indicate that we would like to analyze only contig 11. Verify that the path to the vector sequence is set to 'pcc01.fasta'. Alternatively, you can specify the full path to the vector sequence file at '/Users/student4342/finishingHomework/430-017/edit_dir/pcc01.fasta' (Figure 43). You may see a warning regarding the possible need to complement the contig.

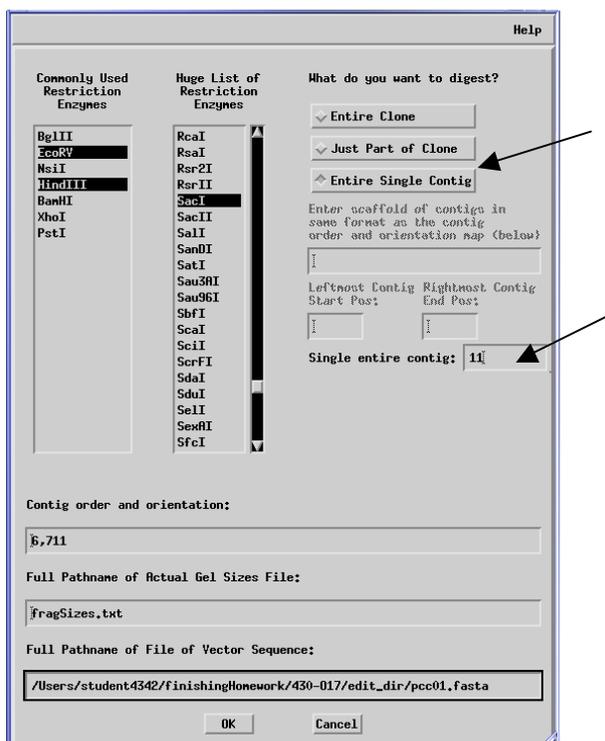


Figure 43. Digest entire contig 11 to see if the force join is correct.

11. *Using both the read pair information from Assembly View and the comparison between the in silico and real digest, should contigs 6 and 7 be force joined into a single contig? Justify your answer.*

Last Update: 07/29/2013