Finishing Report
Carolyn Cain
Fosmid: XBAA-30G19

Fosmid XBAA-30G19 was challenging to assemble, but ultimately came together in a satisfactory final assembly. The initial problems in my assembly were gaps, low quality regions, and some suspicious areas where the sequences matched elsewhere in the assembly. I resolved these through forced joins, editing and stealing reads.

Unfortunately, plate data was not available to generate an initial low coverage assembly. Therefore, I could not view an assembly with 2x converage. Others' low coverage assemblies showed many small contigs, with frequent low quality; my 2x coverage assembly would likely have looked similar.

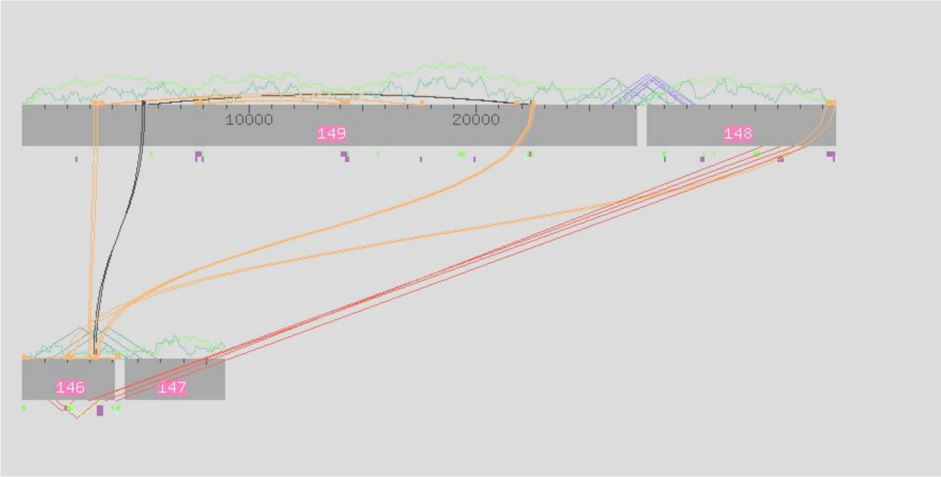
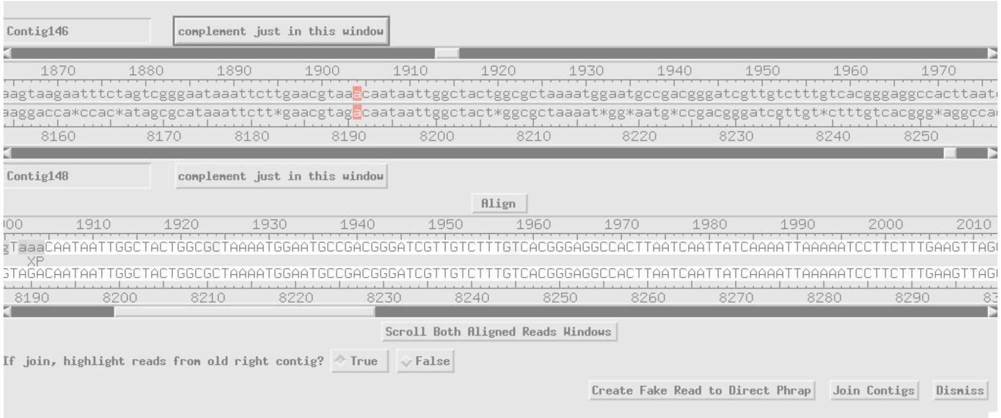


Figure 1, Initial Assembly

Figure 1 shows my initial assembly view with all data included. There were four contigs (three gaps), and many low quality regions. One gap was unspanned. The red lines in Figure 1 show inconsistent forward and reverse pairs. The areas connected by the red lines were farther apart in this assembly than would be expected from our knowledge of the subclones. Orange and black lines connect regions with sequence similarity. There were 83 places in the four contigs where base quality was below threshold. The assembly also had five high quality discrepancies, and many regions that were single-stranded or covered only by one subclone.

My first step was to find large assembly errors and places where I could force join (combine contigs). There was an exact sequence match between contigs 148 and 146; Consed had tagged these regions as matching. I aligned the sequences (see Figure 2 for the compare contigs window), and checked for any mismatched base pairs. Discrepencies were in regions of such low enough quality as to not reject the validity of the join. So, I joined the two contigs. The result is shown below in Figure 3.



```
Contig146
  1870      1880      1890      1900      1910      1920      1930      1940      1950      1960      1970
aagtaagaatttctagtcgggaataaattcttgaacgtaaacaataattggctactggcgctaaaatggaatgccgacgggatcgttgtctttgtcacgggaggccacttaat
aaggacca*ccac*atagcgcataaattctt*gaacgtagacaataattggctact*ggcgctaaaat*gg*aatg*ccgacgggatcgttgt*ctttgtcacggg*aggcca
  8160      8170      8180      8190      8200      8210      8220      8230      8240      8250

Contig148
00        1910      1920      1930      1940      1950      1960      1970      1980      1990      2000      2010
gTaaaCAATAATTGGCTACTGGCGCTAAAATGGAATGCCGACGGGATCGTTGTCTTTGTCACGGGAGGCCACTTAATCAATTATCAAAATTAAAAATCCTTCTTTGAAGTTAG
  XP
GTAGACAATAATTGGCTACTGGCGCTAAAATGGAATGCCGACGGGATCGTTGTCTTTGTCACGGGAGGCCACTTAATCAATTATCAAAATTAAAAATCCTTCTTTGAAGTTAG
 8190      8200      8210      8220      8230      8240      8250      8260      8270      8280      8290      83
```

complement just in this window
complement just in this window
Align
Scroll Both Aligned Reads Windows
If join, highlight reads from old right contig?
True
False
Create Fake Read to Direct Phrap
Join Contigs
Dismiss

Figure 2, Compare Contigs Window

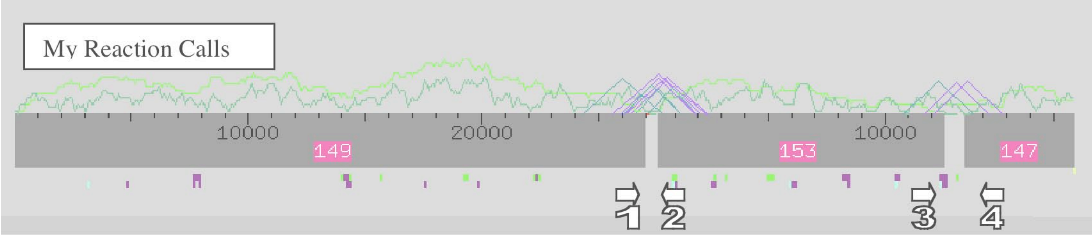


Figure 3, Assembly with my First Round Reaction Calls

The new assembly in Figure 3 looked more manageable. The assembly contained three contigs with two spanned gaps.   At this point, I focused on calling reactions for gaps and not for low quality regions. If I were to do this again, I would have called reactions for all problem areas at the beginning for better use of time. My first round calls are indicated in Figure 3 with numbered arrows pointing in the direction of the called reaction. I used all three chemistries for calls 1 and 2. I requested dGTP for call 3 because many reads were dropping out in that region and direction (see Figure 6), and dGTP works well in areas that are difficult to sequence. I chose Big Dye chemistry for call 4 because the area did not exhibit problems that would require dGTP.

Autofinish derived a similar assembly with the original information. Its assembly view is shown in Figure 4, along with the calls it made. In contrast to my approach, Autofinish tried to resolve low quality discrepancies in the first round. This accounts for most of the differences in calls between Autofinish and me. Also, Autofinish called two reactions (calls 3 and 11) at the end of the second contig for double coverage while I only called one. Autofinish did not call a reaction on the other side of the gap where I did (call 4). Perhaps Autofinish wanted to clear up the low quality region on the right side of the gap with call 12 before trying to span the gap from that side.

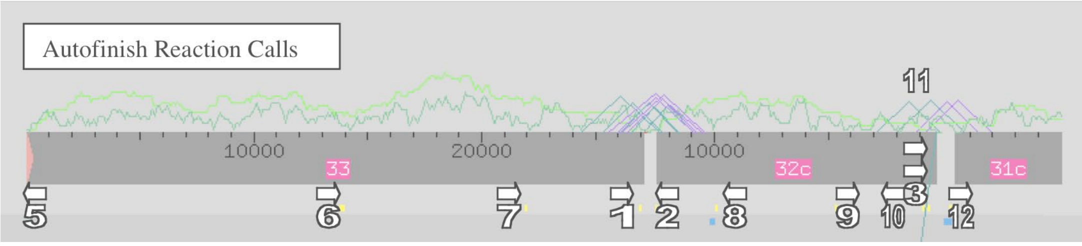


Figure 4, Autonish's Assembly with Reaction Calls

Below is a table (Round 1 Calls) comparing Autofinish's and my first round of reaction calls.

Round 1 Calls

Call	Autofinish Called?	I Called?	Justification for Call
1	Yes	Yes	Close Gap
2	Yes	Yes	Close Gap
3	Yes	Yes	Close Gap
4	No	Yes	Close Gap

5	Yes	No	Low Quality
6	Yes	No	Low Quality
7	Yes	No	Low Quality
8	Yes	No	Low Quality
9	Yes	No	Low Quality
10	Yes	No	Low Quality
11	Yes	No	Close Gap
12	Yes	No	Low Quality

Reads from round 1 closed the first gap. The second gap was persistent though; no reactions I had called to span it had worked. Figure 5 shows my new assembly view.

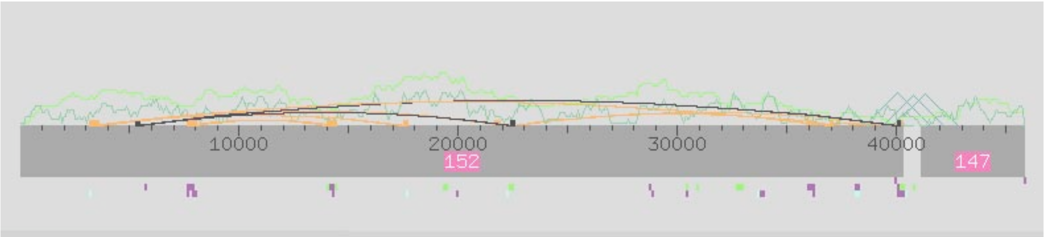
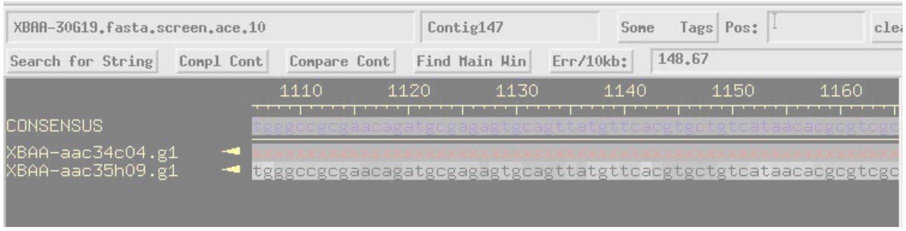


Figure 5, Assembly View

I viewed the ends of the remaining gap to determine why it was not spanned. At the end of contig 152, there was a region where all reads except one dropped off, and the one read that continued perfectly matched sequence far away in the assembly. See Figure 6 for the aligned reads in this region. Because of the region's odd characteristics described above, the sequence after the drop-off is likely an incorrect sequence from a chimera. A chimera is the result of two unrelated pieces of the fosmid sequence fusing in a clone. Sequencing this clone yields the erroneous conclusion that the two pieces belong together in the genome. I instructed Consed to ignore sequence data in this read after the dropping off point.

Figure 6, Chimeric Region

I discovered that my two fosmid projects overlapped when I accidentally searched for the same string of bases in both fosmids. After confirming the overlap by searching for more strings of sequence, I promptly stole reads from my other project, XBAA-47I6, to cover the remaining gap and all low quality regions in this project. Stealing reads involved copying a specific read file from the XBAA-47I6 chromat_dir into the XBAA-30G19 chromat_dir, and re-running phredphrap. Figure 7 shows one very low quality region from contig 147 for which I stole reads. For each problem region, I generally stole several reads. The locations where the stolen reads were incorporated are indicated in Figure 8 by arrows pointing in both directions. The table below (Round 2 Calls) summarizes the reads stolen.



```
XBAA-30G19.fasta.screen.ace.10    Contig147    Some  Tags  Pos:      clea
Search for String   Compl Cont   Compare Cont   Find Main Win   Err/10kb:  148.67
                     1110      1120      1130      1140      1150      1160
CONSENSUS            tgggccgcgaacagatgcgagagtgcagttatgttcacgtgctgtcataacacgcgtcgc
XBAA-aac34c04.g1  ◄  xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
XBAA-aac35h09.g1  ◄  tgggccgcgaacagatgcgagagtgcagttatgttcacgtgctgtcataacacgcgtcgc
```

Figure 7, Low Quality Region

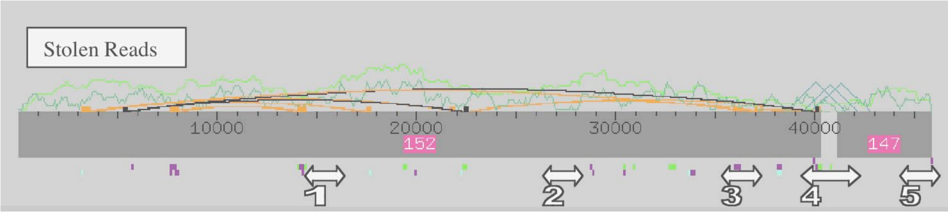


Figure 8, Assembly View with Second Round Stolen Reads

Round 2 Calls

Call	Justification for Call
1	Low Quality
2	Low Quality
3	Low Quality
4	Close Gap
5	Low Quality

After round 2 reads were incorporated, the assembly was one continuous contig with no regions of low quality. See Figure 9 for a view of my final assembly.

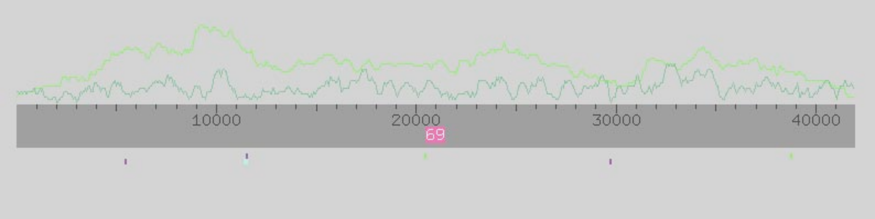


Figure 9, Final Assembly View

I then resolved remaining discrepancy problems by editing. Figure 10 shows an edit I made to resolve an incorrect base call. In Figure 10, the high 'A' background was obscuring the true 'G' peak in the trace.

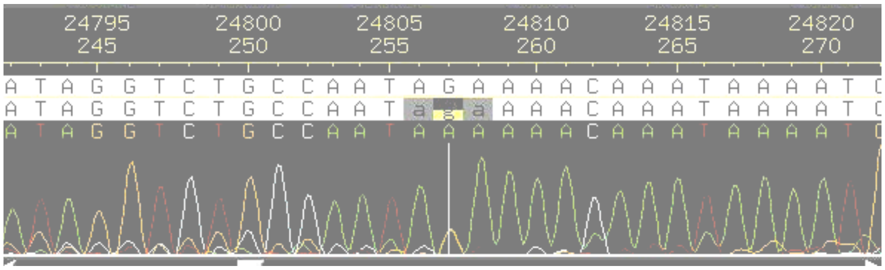

24795
245
24800
250
24805
255
24810
260
24815
265
24820
270
A T A G G T C T G C C A A T A G A A A A C A A A T A A A A T C
A T A G G T C T G C C A A T a g a A A A C A A A T A A A A T C
A T A G G T C T G C C A A T A A A A A A C A A A T A A A A T C

Figure 10, Ambiguous Call

Sometimes low quality data masqueraded as high quality data and caused such problems as high quality discrepancies, or apparent base pair runs. These problems were easily resolved once I looked at the traces and saw the true quality of the data. Figure 11 shows an example of low quality data being called as high quality. In this instance, because the entirety of the trace was poor quality, I removed the offending trace into its own contig. This action fixed the problem in the area. In another region, I simply marked part of a read as low quality because there was helpful information elsewhere on the trace.

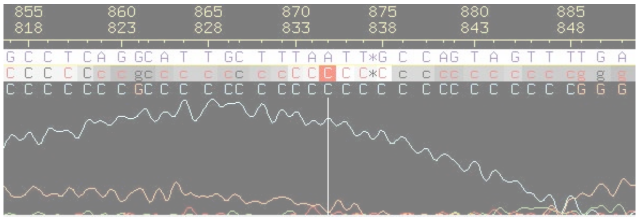


Figure 11, Low Quality Read

I next went through the pre-submit checklist. I verified the presence of cloning sites at both ends of my contig. However, the final GATC sequence was labeled as vector sequence, and I had to go into the traces to find it and change the consensus sequence (see Figure 12).

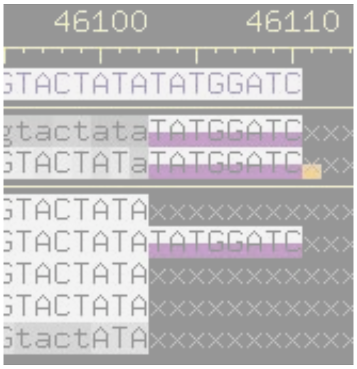


Figure 12, Cloning Site

Calling reactions and stealing reads had resolved all low quality regions. High quality discrepancies were corrected through editing. Any single strand/single chemistry/ single subclone regions with quality above Phred30 were tagged as being acceptable. There were no inconsistent forward and reverse pairs. I also found no problems in the consensus while searching for 'X,'s 'N,'s and strings of 15 or more 'C's.

I did, however, find a potential problem when I searched for strings of 'A's. Figure 13 shows a string of 17 'A's that I have confidence is correct, because there are many high quality reads (only two are shown) in both forward and reverse orientations.

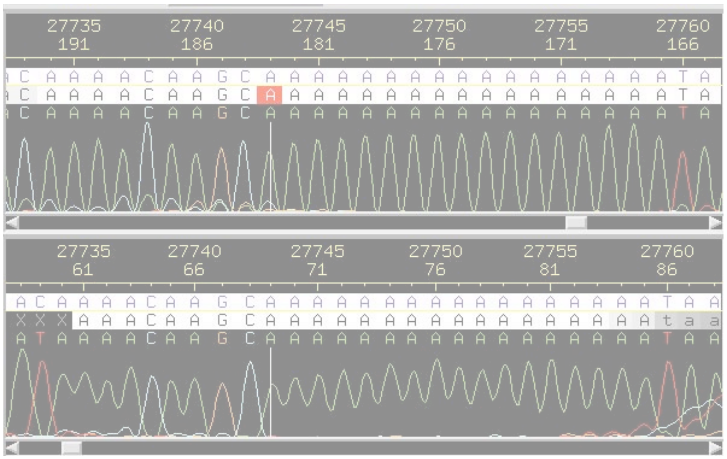

27735 191
27740 186
27745 181
27750 176
27755 171
27760 166
C A A A A C A A G C A A A A A A A A A A A A A A A A A T A
C A A A A C A A G C A A A A A A A A A A A A A A A A A T A
C A A A A C A A G C A A A A A A A A A A A A A A A A A T A
27735 61
27740 66
27745 71
27750 76
27755 81
27760 86
A C A A A A C A A G C A A A A A A A A A A A A A A A A A T A A
X X X A A A C A A G C A A A A A A A A A A A A A A A A A t a a
A T A A A A C A A G C A A A A A A A A A A A A A A A A A T A A

Figure 13, Run of 'A's

Next, I browsed my extraneous contigs looking for any over 2kb. I did find one (Figure 14), but as the picture shows, it was of low quality. None of the traces in the contig contained useful information, so I am confident that it does not belong in my main assembly.

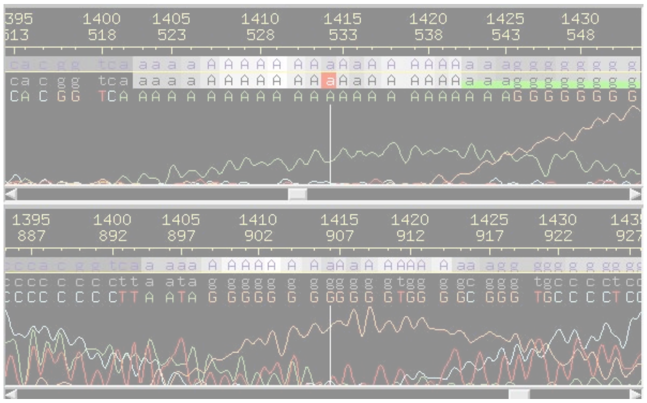


Figure 14, Low Quality Contig

My restriction digest information looks promising. Figure 15 shows the *SacI* digest. The only discrepancies are in bands of fragments smaller than 1kb, which is acceptable; bands in that range are difficult to score in the real digest. Figure 16 shows the *HindIII* restriction digest information. Here too, all discrepancies are in the smaller bands. One discrepancy is simply a difference between a double band in the real digest and a single band in the in-silico digest. The difference between double and single bands is often difficult to distinguish on a gel, so such discrepancies are not considered a problem. Perhaps some of the discrepancies would be resolved by analyzing a picture of the original gel.

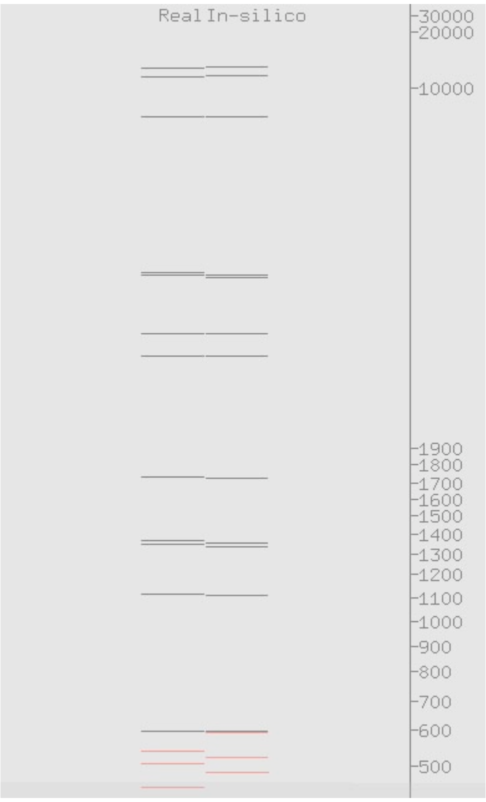


Figure 15, *SacI* Digest

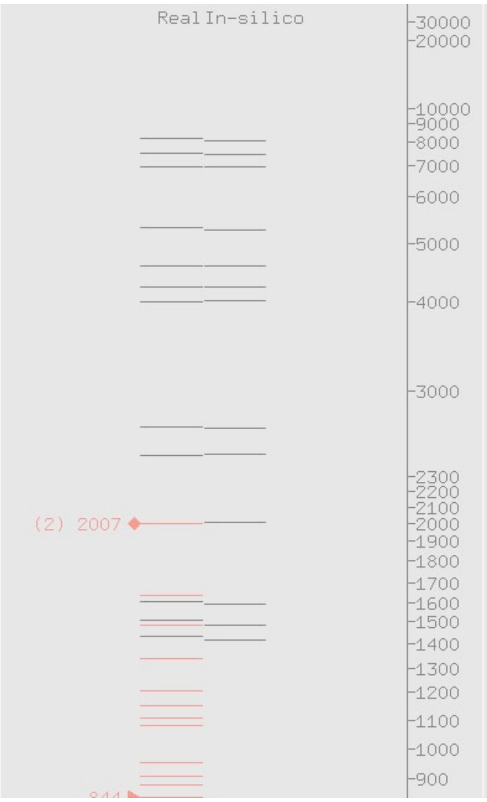


Figure 16, *HindIII* Digest

In conclusion, finishing my fosmid required many methods (joining, stealing reads, editing, etc.) to generate a final assembly that I can present with confidence. My confidence is based on meeting the requirements of the Genome Sequencing Center finishing pre-submit checklist. The most compelling evidence for correct assembly is the restriction digests showing consistency between my assembly and the original fosmid.