User Guide to

SeqMan NGen

DNASTAR, Inc. 2021

Table of Contents

Welcome to SeqMan NGen

Lasergene Genomics provides everything you need for assembly and analysis of genomic, metagenomic, exomes/gene panels and transcriptomic sequencing data, and supports all popular file formats. Most workflows will start with sequence assembly in SeqMan NGen.

SeqMan NGen supports both *de novo* and reference-guided (templated) workflows on all major sequencing platforms (Sanger, Illumina, Ion Torrent, and Pacific Biosciences). SeqMan NGen supports reference-guided assemblies of billions of sequence reads and *de novo* assemblies of up to 30 million sequence reads (genome sizes up to 50 megabases).

About this User Guide:

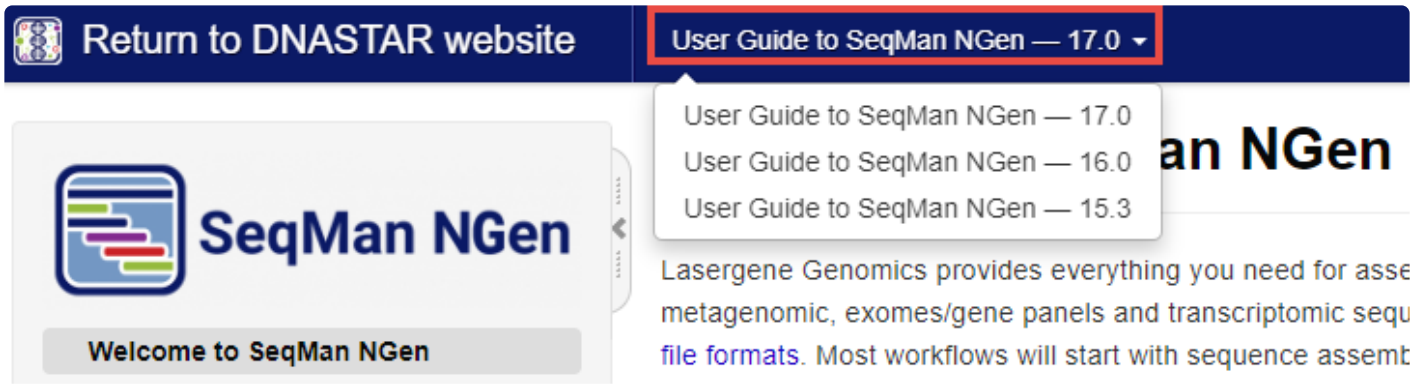
- For help **INSTALLING** Lasergene, see our separate Installation Guide.
- To **PRINT** the current page of the User Guide, click the printer icon in the top right corner ( ).
- To download a **PDF** of the entire User Guide, scroll to the bottom of the table of contents on the left, and press **Download as PDF**.

> **Important:** If you decide to print the PDF, note that the last quarter of the PDF (110+ pages) is a “command-line scripting” manual that applies to only a handful of users. You may wish to omit this when printing.

SeqMan NGen Tutorials

The following tutorials cover several of the most popular workflows in SeqMan NGen. Each tutorial begins with setting up and running an assembly in SeqMan NGen, then proceeds to other Lasergene applications for downstream analysis. Within each tutorial is a link for downloading the corresponding data in archived (*.zip*) format.

SeqMan NGen went through a major update with Lasergene 17.0, which was released in early 2020. Therefore, there are two sets of tutorials and tutorial data. The set included in this version of the User Guide is for Lasergene version 17.0 and later. If you are using Lasergene version 16.0 or earlier, you will want to switch to that version of the User Guide before starting the tutorials. To switch, just use the drop-down menu right above the topic name (“SeqMan NGen Tutorials”) and choose the User Guide that corresponds with your version of Lasergene.



As of Lasergene 17.0 and later, tutorials that feature downstream analysis in SeqMan Ultra are located in the SeqMan Ultra User Guide but can be accessed through the table links below. Tutorials that end with analysis in any other application are located in this SeqMan NGen User Guide.

User Guide	Tutorial Number	Tutorial Name
SeqMan NGen (this guide)	1	RNA-Seq reference-guided workflow with analysis in ArrayStar
	2	ChIP-Seq workflow with analysis in ArrayStar
	3	Copy number variation (CNV) workflow with analysis in ArrayStar and GenVision Pro
	4	Whole-genome reference-guided workflow with analysis in ArrayStar
SeqMan Ultra	1	Whole genome reference-guided workflow
	2	Whole genome *de novo* workflow with mate pair data
	3	Analysis of a whole genome *de novo* assembly

	4	*De novo* assembly using Sanger data
	5	RNA-Seq *de novo* transcriptome workflow
	6	Gap closure workflow

RNA-Seq reference-guided workflow with analysis in ArrayStar

RNA-Seq uses next-gen sequencing to show the presence and quantity of RNA in a genome at a particular moment. DNASTAR's SeqMan NGen application is the starting point for both reference-guided and *de novo* RNA-Seq workflows. Because this tutorial involves a reference-guided workflow, you will then use ArrayStar to analyze the completed RNA-Seq assembly.

In this tutorial, you will compare stationary phase RNA from wild-type *E. coli* cells with that from two different mutant cells, each lacking one of two genes that together encode a transcription factor for flagella and chemotaxis operons. An "operon" is a group of one or more genes that are transcribed as a single RNA unit.

In Part A, you will use SeqMan NGen to create a reference-guided DESeq2-normalized RNA-Seq assembly for two experimental replicates and one wild type control of *E. coli* transcription factors FlhC and FlhD. The data you will assemble is publicly-available single-end Illumina data from *E. coli* (Fitzgerald et al, 2014). In Parts B & C, you will use ArrayStar to perform downstream analysis and will learn to identify flagella-related operons using two different techniques.

IMPORTANT! There are two options for following this tutorial, one significantly longer than the other.

Item	Full tutorial	Abbreviated tutorial
Description	Assemble in SeqMan NGen, then do downstream analysis in ArrayStar	Do only the downstream analysis in ArrayStar, using a provided project
Download size	4.0 GB (unzips to 16 GB)	4.3 MB
Assembly wait time	1-3 hours	none
Where to begin	Perform the steps in Part A: Setting up the RNA-Seq reference-guided assembly in SeqMan NGen	Read the steps in Part A: Setting up the RNA-Seq reference-guided assembly in SeqMan NGen without performing them

Part A: Setting up the RNA-Seq reference-guided assembly in SeqMan NGen

In this part of the tutorial, you will learn how to set up the project in SeqMan NGen and (optionally) run the assembly. To perform the assembly in Part A, you must download a 4 GB zipped data folder that unpacks to 14.3 GB. We expect most to simply read this section, then proceed to Parts B & C, where you can download a 4 MB data set and perform the downstream analysis part in ArrayStar.

1. (optional) If you plan to run the assembly (most readers will not), download *T1_RNA-Seq.zip* (4.0 GB) and extract the contents to any convenient location (e.g., your computer's desktop). The folder contents consist of:

    - The *E. coli* reference sequence: *Escherichia coli str. K-12 substr. MG1655.U00096.gbk*
    - The sample sequences: six folders beginning with *flhC*, *flhD* and *WT*.

2. Launch SeqMan NGen and press **New Assembly**.

3. Select the **RNA-Seq / Transcriptomics** tab on the left and choose **RNA-seq** from the **Quantitative Analysis** section on the right.

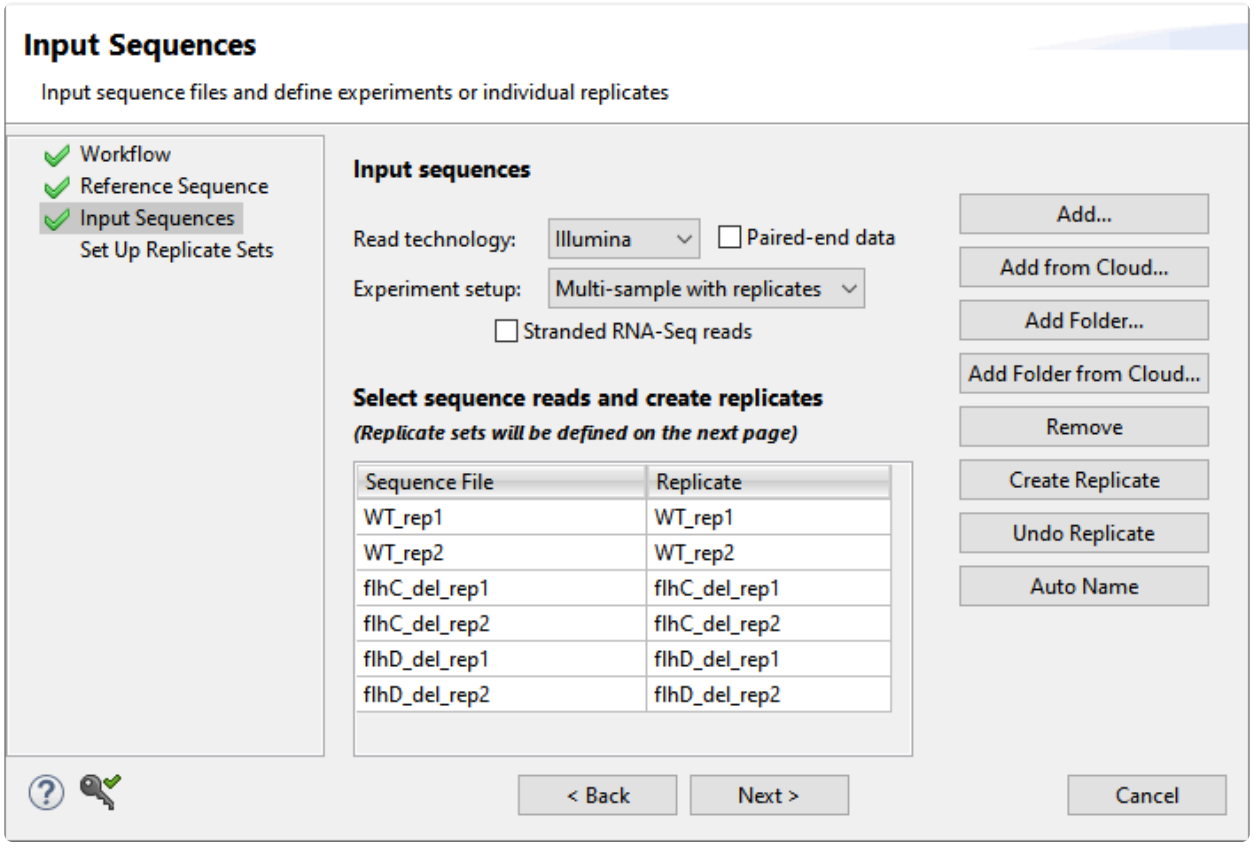
4. In the Reference Sequence screen, press the **Add** button and open the *Escherichia coli str. K-12 substr. MG1655.U00096.gbk*. (Alternatively, drag the file from your file explorer and drop it onto the large white space in the middle of the wizard screen.) Click **Next**.

5. In the Input Sequences screen:

    a. Keep the **Read technology** set at **Illumina**, but uncheck the **paired-end data** box.

    b. Next to **Experiment setup**, select **Multi-sample with replicates**.

    c. Use the **Add Folder** button six times, each time adding one of the six sample folders from the tutorial data folder.



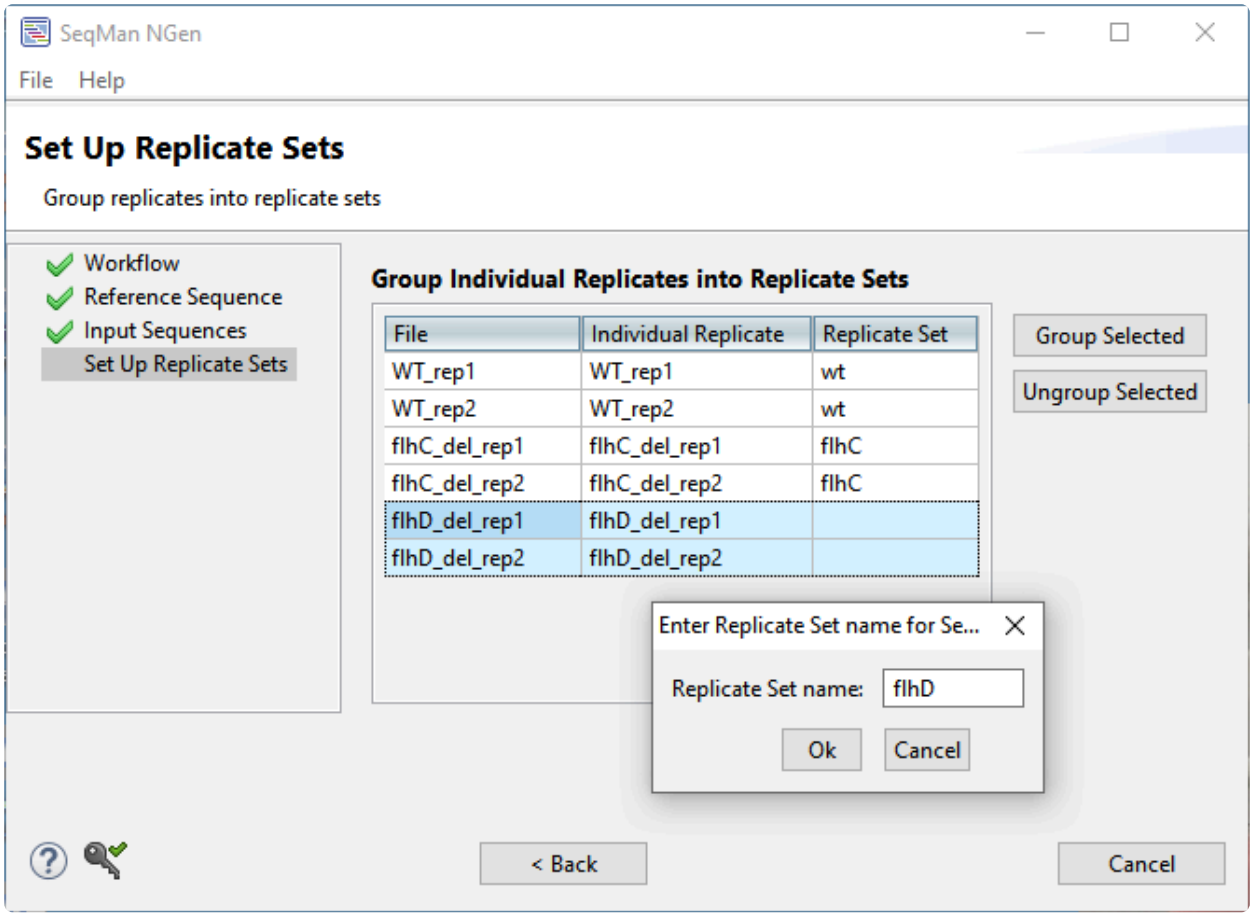
d. Click **Next**.

6. In the Set Up Replicate Sets screen:

   a. Select the two **WT** replicates and click on the **Group Selected** button. In the dialog, name the set “wt” and click **OK**.

   b. Do the same for the two **flhC** replicates, naming the set “flhC.”

   c. Do the same for the two **flhD** replicates, naming the set “flhD.”

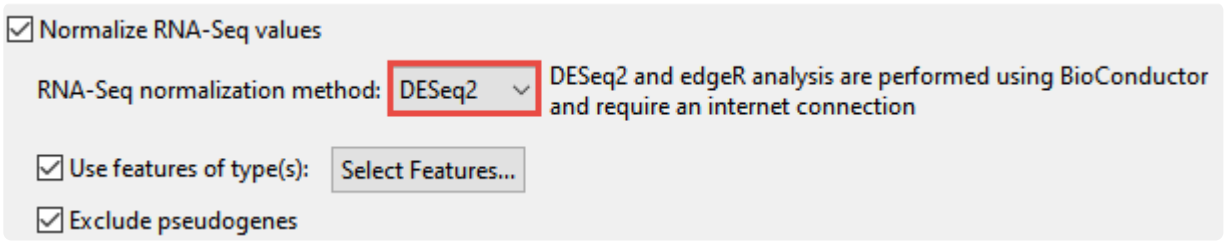


d. Click **Next**.

7. In the Set Up Experiments screen, specify the “wt” group as the control. To do this, check the **Is Control** box to the right of “wt,” then click **Next**.

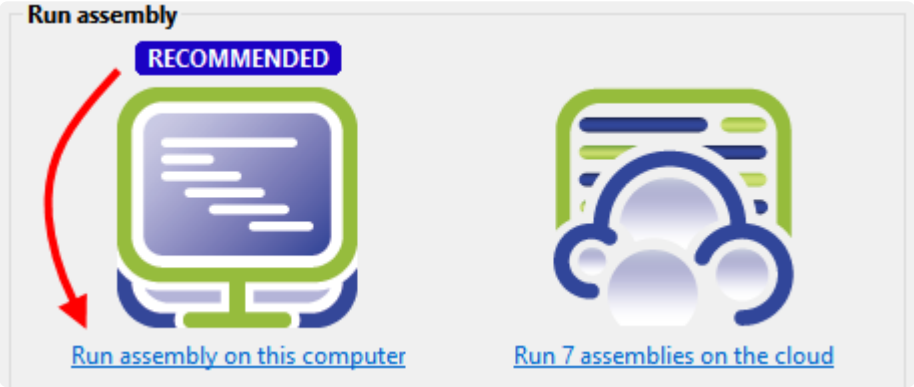
8. In the Assembly Options screen, check the box next to **Maximum total reads** and enter 5000000 (5 followed by 6 zeros) to reduce the assembly time. Click **Next**.

9. In the Analysis Options screen, make no changes, but observe that the **RNA-Seq normalization method** is **DESeq2**. Click **Next**.



10. In the Assembly Output screen, type “Templated RNA-seq” into the **Project Name** text box. This name will be assigned to all output files, including the finished assembly. Use the **Browse** button to specify a **Project Folder** for your assembly output files, then click **Next**.

11. In the Run Assembly Project screen, observe the recommendation for where to run the assembly. Press the link under that recommendation.



12. Wait until being informed that assembly has finished (approximately 10-20 minutes for a local assembly), then click **Next**.

13. From the Assembly Summary screen, click **Analyze differential gene expression**. This launches ArrayStar.

14. (optional) Close SeqMan NGen by clicking the **Finish** button.

15. Within ArrayStar, use **File > Save Project** to save the project as *Templated_RNA-Seq.astar.*

Proceed to Part B: Analyzing the results in ArrayStar using quick gene sets.

Part B: Analyzing the results in ArrayStar using quick gene sets

In Part A of this tutorial, you set up and ran a templated RNA-Seq assembly using SeqMan NGen. In this part of the tutorial, you will analyze assembly results in ArrayStar using a “quick gene set” and the Gene Table to locate a potential operon structure.

1. If you ran the assembly in Part A, skip to Step 3. Otherwise, download *T1_RNA-Seq (abbrev).zip* (4.3 MB) and extract it to any convenient location (e.g., your computer’s desktop). The data set consists of a single file, *Templated RNA-Seq.astar*.

2. Double-click on *Templated RNA-Seq.astar* to open the project in ArrayStar.

3. In ArrayStar, select **Graphs > Venn Diagrams** from the menu, and then press the **Quick gene set creation** button.

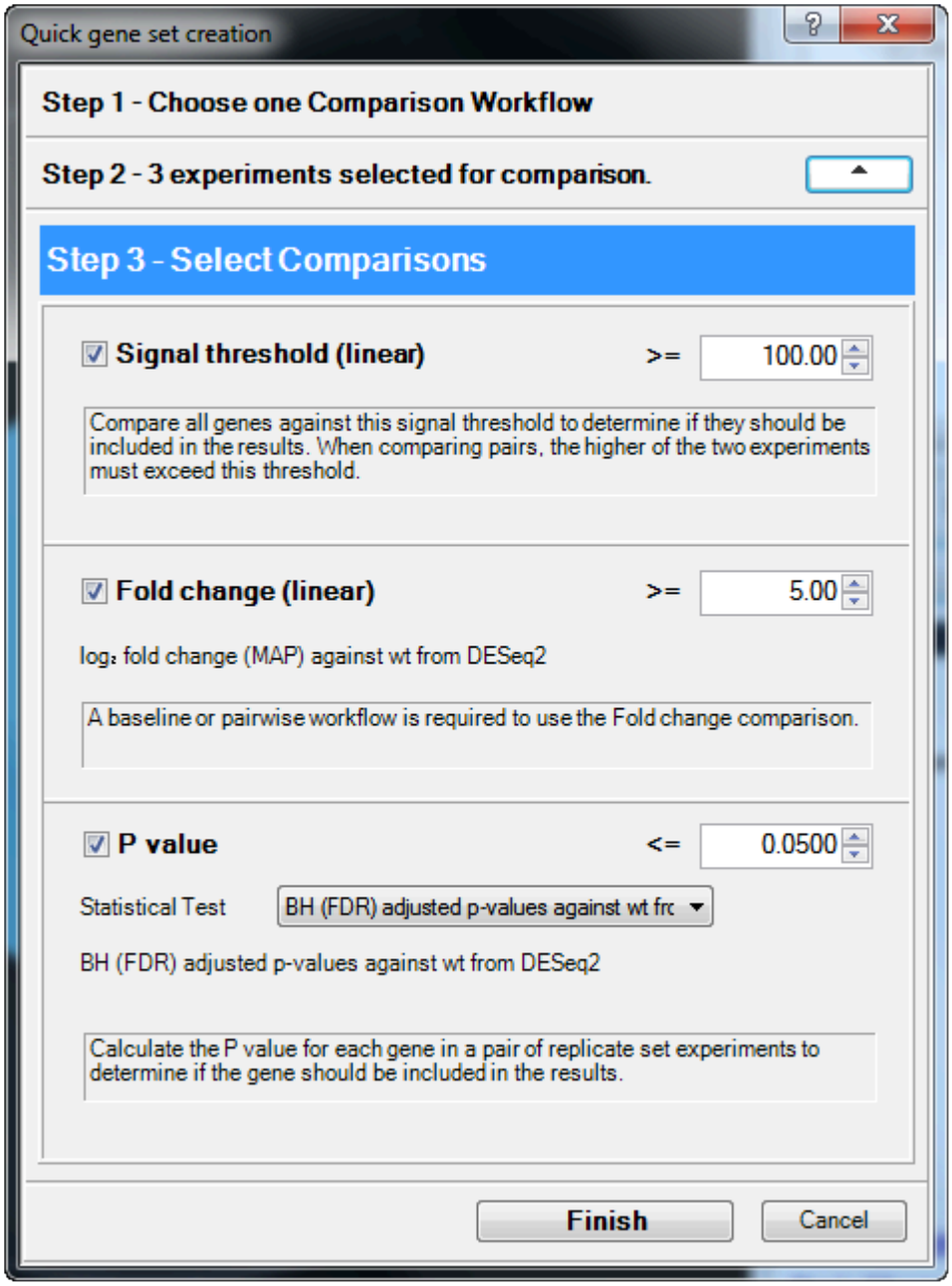
4. Because this project used DESeq2 normalization, Step 1 is not available. Instead, the dialog opens at **Step 2 - Experiments to compare**. Keep the default (all boxes checked) and press **Move to Step 3 (Comparisons)**.

5. In **Step 3 – Select Comparisons**, set up the filter to find genes in the mutant samples that have a ≥ 5-fold change, compared to the wild type, and an rlog signal value ≥ 100. To do this:

    a. Keep the **Signal Threshold (linear)** box checked and change the value from **10.00** to **100.00**.

    Note that signal calculation varies depending on which normalization method was selected in SeqMan NGen. In this tutorial, DESeq2 was used, so the Signal is in terms of rlog values. These values are transformed to the linear scale in this dialog and in ArrayStar’s Gene and Isoform tables. If normalization had instead been edgeR or RPKM, the Signal would have been in terms of the TMM or RPKM values, respectively.

    b. Keep the **Fold change (linear)** box checked and change the value from **2.00** to **5.00**.



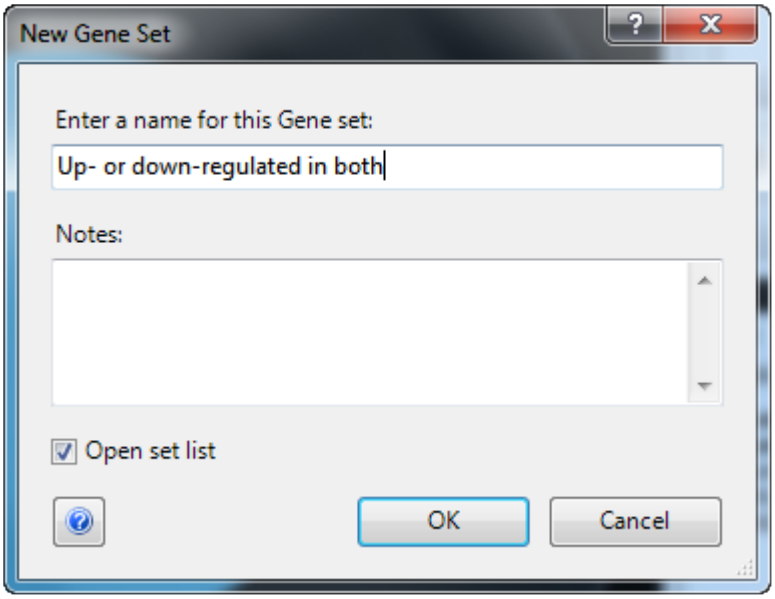
c. Press **Finish**.

6. In the upper right section of the ArrayStar window, note that two sets have been created. Click the box next to **Global Gene Sets** to select both sets. The Venn Diagram becomes populated with data.

   The blue circle labeled “AB” denotes the intersection of the two sets, and represents genes which are up- or down-regulated for both samples: flhC and flhD.

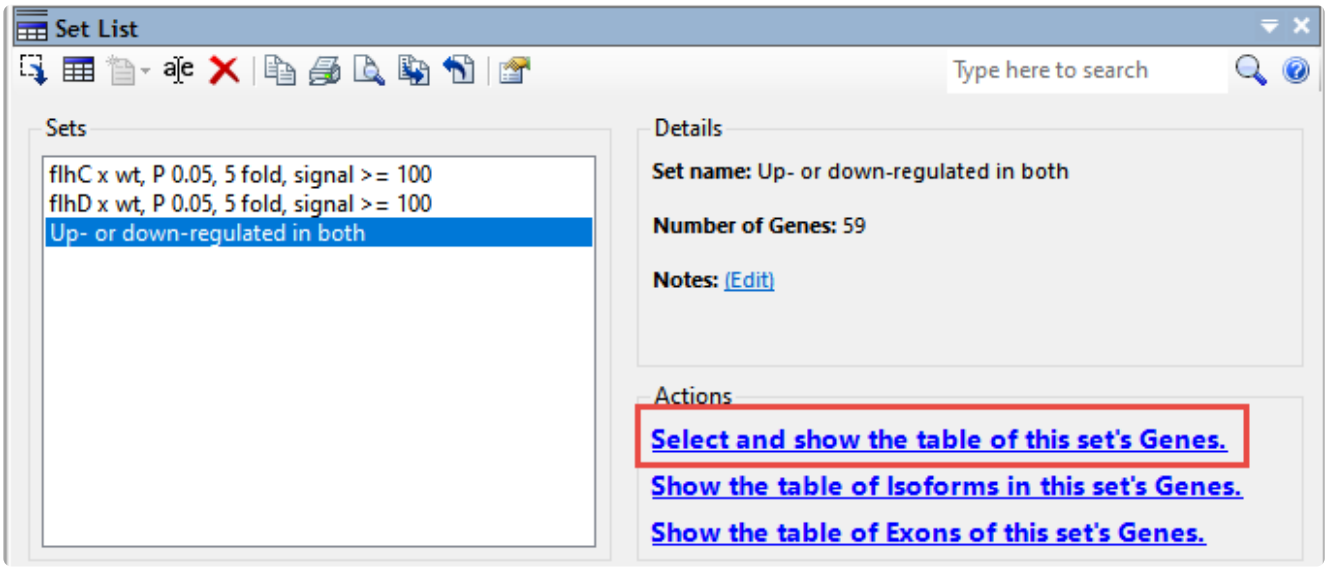
7. Click on **AB** to select it, and then press the link in the bottom right of the window: **Remember the selection as a Gene set**.

8. When prompted to enter a name for the gene set, type “Up- or down-regulated in both” and press **OK**.

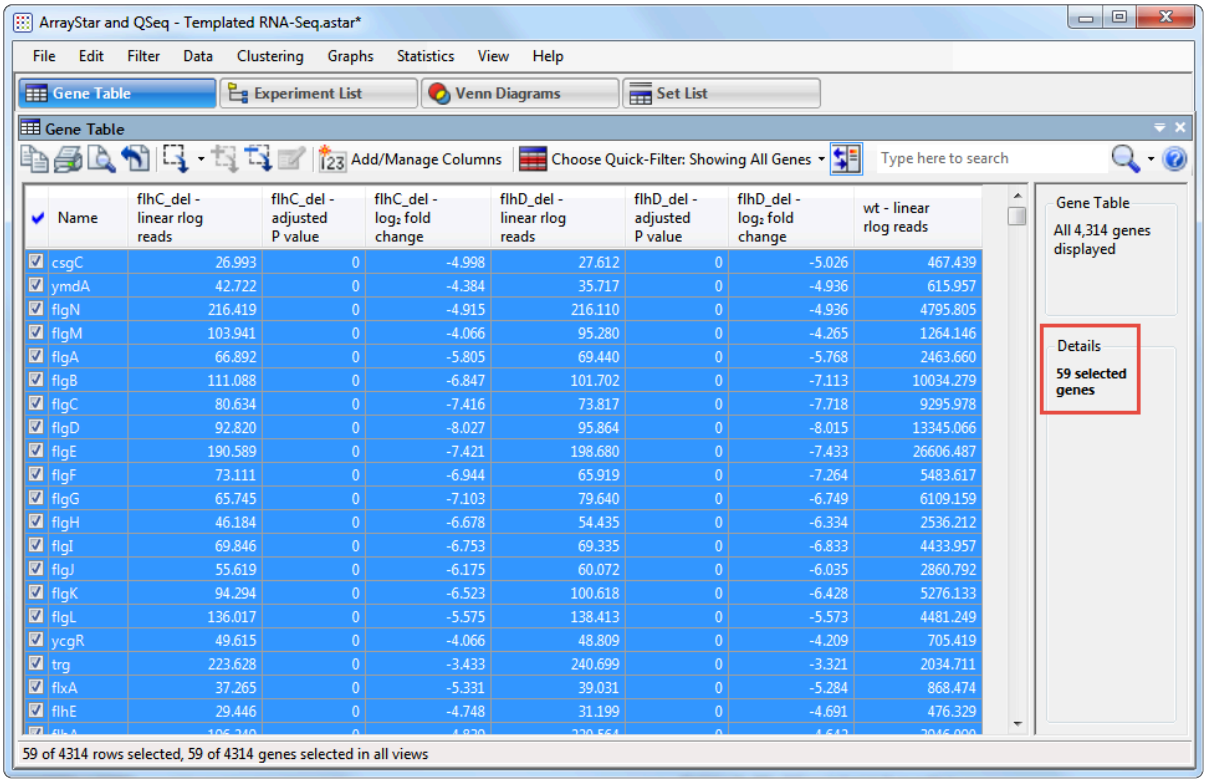


The Set List view opens, with the new set already selected.

9. Click the link **Select and show the table of this set's Genes**.



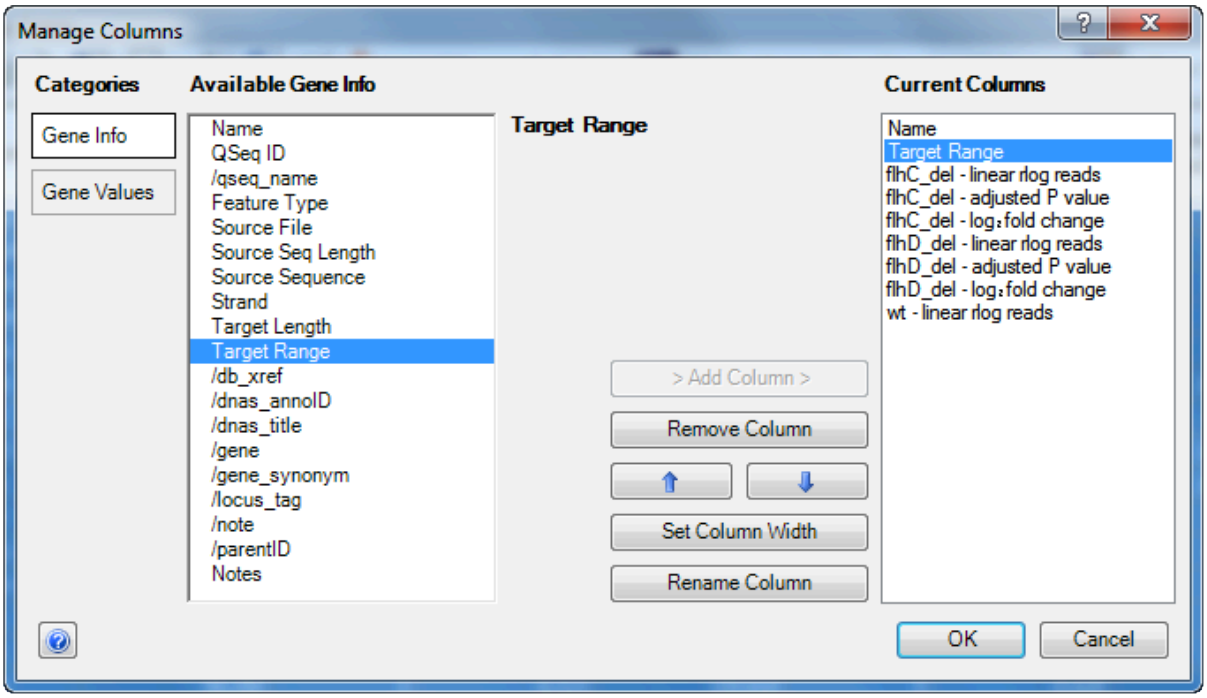
The Gene Table opens with 59 genes selected. Do NOT click anywhere in the table for now, as you need to retain the selection (blue highlighting).



10. Click the **Add/Manage Columns** tool (Add/Manage Columns) to open the Manage Columns dialog:

    a. The **Gene Info** button is active by default. Select **Target Range**, then press the **> Add Column >** button to add the item to the Current Columns list.

    b. Press the up arrow button ( ) repeatedly until **Target Range** appears just after **Name**.

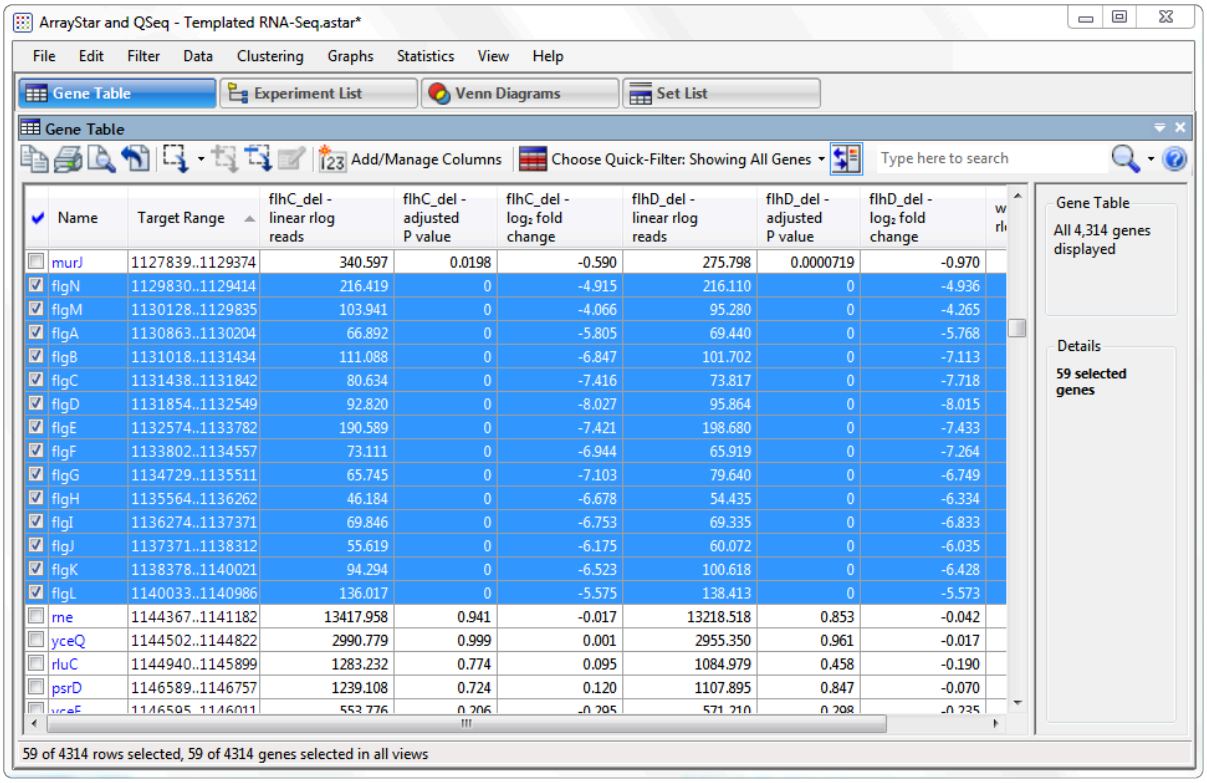


c. Click **OK** to close the Manage Columns dialog and return to the Gene Table.

11. In the Gene Table, click once on the column header for **Target Range** to sort all of the genes in the project in ascending order of appearance on the assembly.

12. To identify possible operon structures, scroll down the Gene Table. Genes with blue highlighting represent the downregulated genes in the “quick gene set.” Large blue-highlighted sections with overlapping or consecutive genes that show similar trends in expression levels and fold changes are candidates for operons.

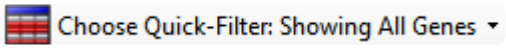
    The first operon candidate begins at **Target Range 1129830..1129414**.



Name	Target Range	flhC_del - linear rlog reads	flhC_del - adjusted P value	flhC_del - log_2 fold change	flhD_del - linear rlog reads	flhD_del - adjusted P value	flhD_del - log_2 fold change
murJ	1127839..1129374	340.597	0.0198	-0.590	275.798	0.0000719	-0.970
flgN	1129830..1129414	216.419	0	-4.915	216.110	0	-4.936
flgM	1130128..1129835	103.941	0	-4.066	95.280	0	-4.265
flgA	1130863..1130204	66.892	0	-5.805	69.440	0	-5.768
flgB	1131018..1131434	111.088	0	-6.847	101.702	0	-7.113
flgC	1131438..1131842	80.634	0	-7.416	73.817	0	-7.718
flgD	1131854..1132549	92.820	0	-8.027	95.864	0	-8.015
flgE	1132574..1133782	190.589	0	-7.421	198.680	0	-7.433
flgF	1133802..1134557	73.111	0	-6.944	65.919	0	-7.264
flgG	1134729..1135511	65.745	0	-7.103	79.640	0	-6.749
flgH	1135564..1136262	46.184	0	-6.678	54.435	0	-6.334
flgI	1136274..1137371	69.846	0	-6.753	69.335	0	-6.833
flgJ	1137371..1138312	55.619	0	-6.175	60.072	0	-6.035
flgK	1138378..1140021	94.294	0	-6.523	100.618	0	-6.428
flgL	1140033..1140986	136.017	0	-5.575	138.413	0	-5.573
rne	1144367..1141182	13417.958	0.941	-0.017	13218.518	0.853	-0.042
yceQ	1144502..1144822	2990.779	0.999	0.001	2955.350	0.961	-0.017
rluC	1144940..1145899	1283.232	0.774	0.095	1084.979	0.458	-0.190
psrD	1146589..1146757	1239.108	0.724	0.120	1107.895	0.847	-0.070

Note that the P-values for both flhC and flhD are 0 in every case, signifying extremely high confidence. Also note that the log_2 fold changes for both flhC and flhD are all between approximately -4 and -8, indicating strong downward regulation for both. By comparison, less than 1.5% of the 4,300+ genes have a log_2 fold change less than -2.

Other candidates for operons begin at: **1962972..1962580**; **2001024..200473**; and **2013014..2012700**. In all cases, all P-values are zero, and log_2 fold changes are similar to those in the first operon.

13. To see only the genes in the ‘quick gene set,’ click on the **Choose Quick-Filter** tool () above the table, and select **Show Only Gene Set**. In the pop-up, select ‘Up- or down-regulated in both’* and press **OK**.

14. Click once on the **Name** column header to sort alphabetically by gene name.

    Observe that most of the down-regulated genes have a name prefix of '*fli*, '*flg'* or '*che*.' The first two gene types encode flagella-related proteins while the third type encodes chemotaxis-related proteins. These findings corroborate that these samples are missing coding genes related to flagellar and chemotaxis functions.

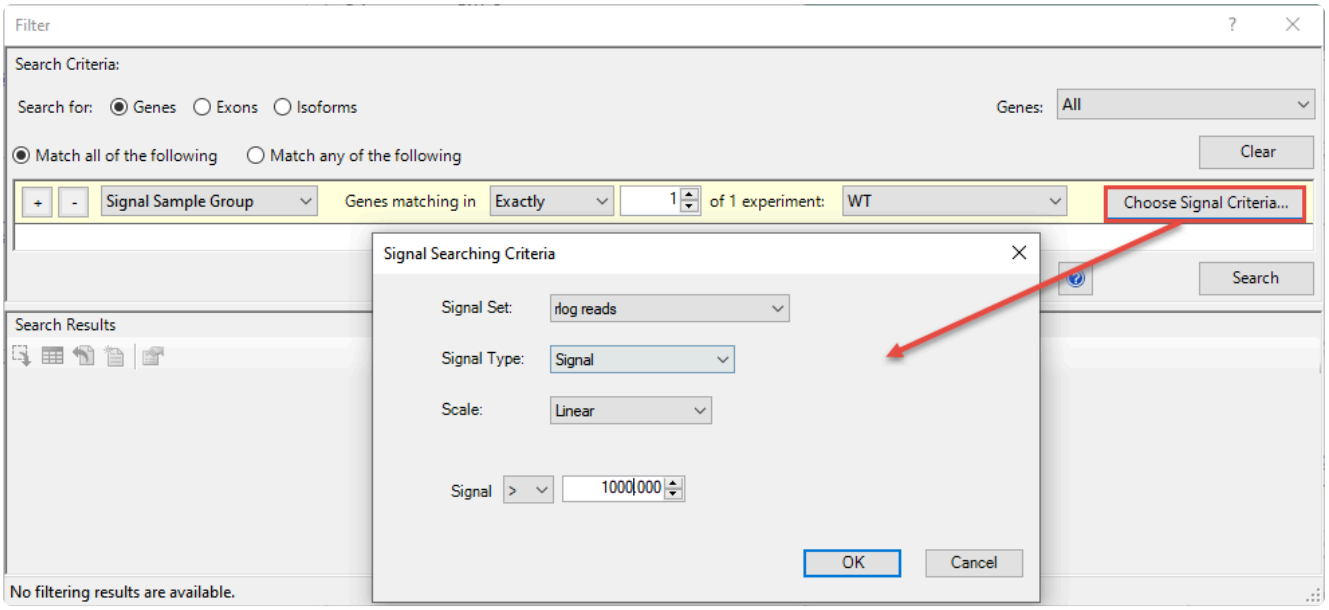
Proceed to Part C: Analyzing the results in ArrayStar using advanced filtering.

Part C: Analyzing the results in ArrayStar using advanced filtering

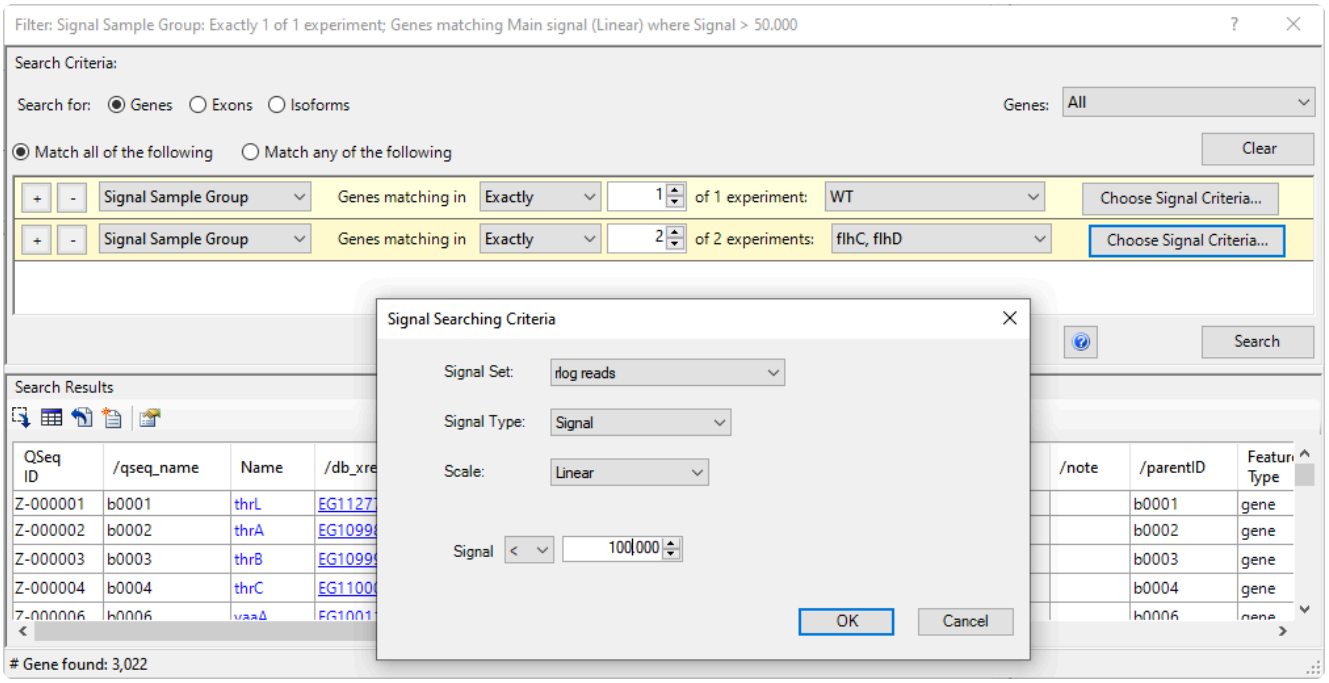
In Part A of this tutorial, you set up and ran a templated RNA-Seq assembly using SeqMan NGen. In this part of the tutorial, you will analyze the assembly results using ArrayStar's advanced filtering functionality to create a set of genes that relate to the flagella structure.

1. Choose **Filter > Filter All** to open the advanced Filter dialog.

2. In the yellow row, select options to match the image below. Note that you must press the **Choose Signal Criteria** button to open the pop-up dialog shown in the image. In the pop-up, change the **Scale** to **Linear** and the **Signal** to **> 1000** (note greater-than sign). Press **OK** to close the pop-up.

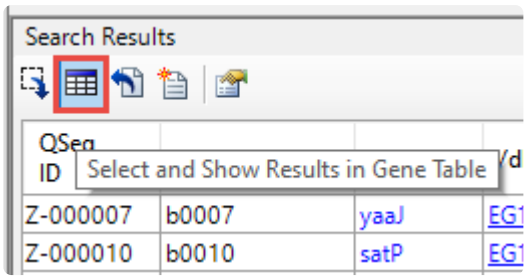


3. Click the plus sign (+) to the left of the yellow row to add a second row. In the new row, set up options to match the image below. In the **Choose Signal Criteria** pop-up, change the **Scale** to **Linear** and the **Signal** to **< 100**. Press **OK** to close the pop-up.



Press **Search**. This two-filter combination results in approximately 25-35 genes.

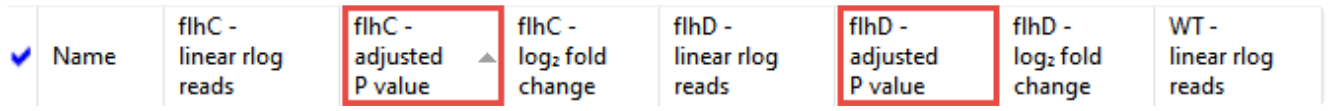
4. Press the **Select and show results in Gene table** tool.



Observe that most of the genes that met the filter thresholds are *fli* and *flg* genes. These genes are involved in producing flagella in *E. coli* and are known to be regulated by the transcription factor encoded by the *flhC* and *flhD* genes. The *che*, *tap* and *tar* genes are involved in chemotaxis and are also known to be regulated by *FlhCD*.

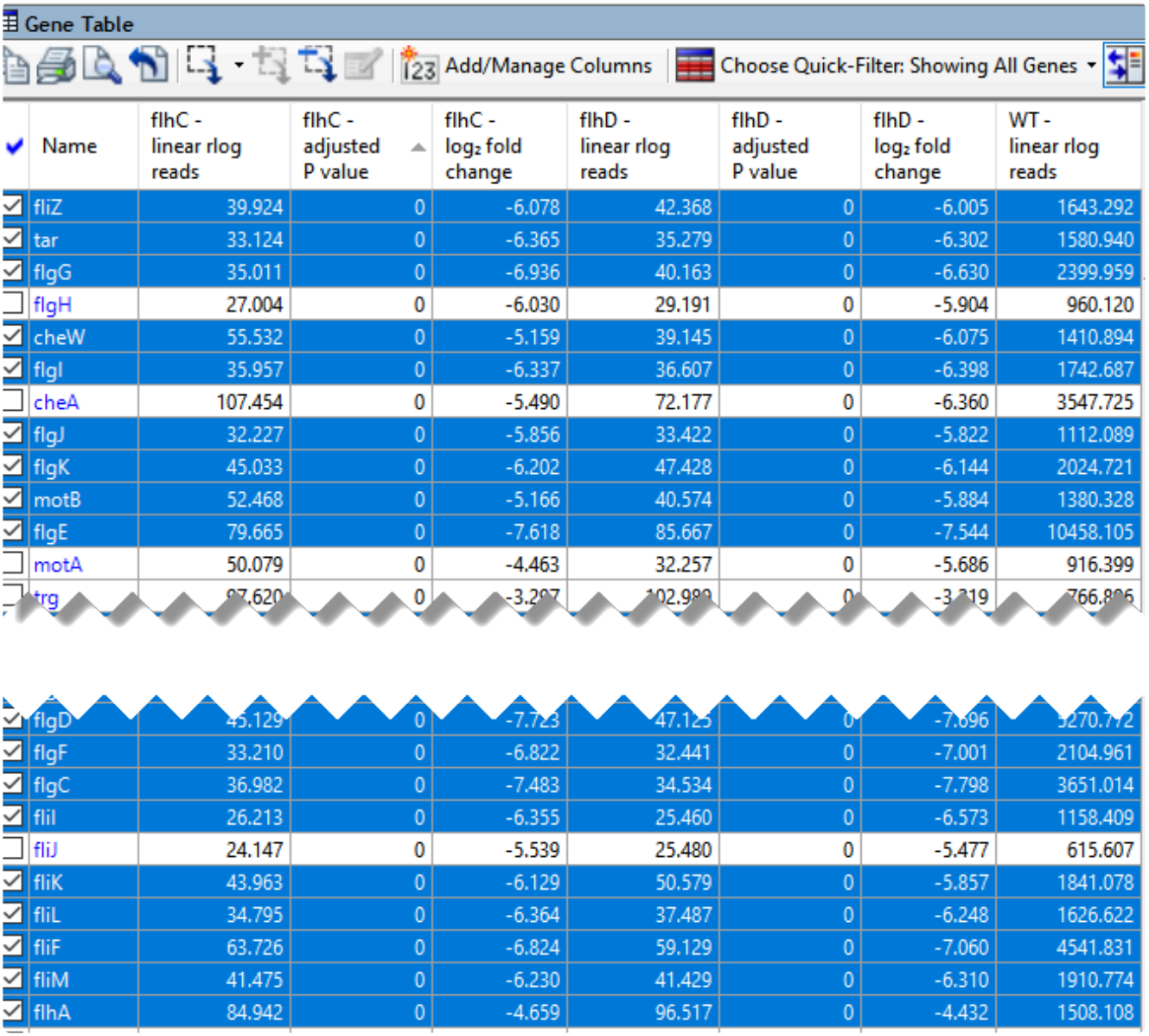
5. Use the **Quick filter** tool in the Gene Table header to **Show all genes**.

6. Sort by the adjusted p-values from smallest to largest by clicking either of the p-value column headers.



Note that there are approximately 40-60 genes where the adjusted p-value is zero for both *flhC* and *flhD*. These include the genes found by filtering (still highlighted in blue), as well as additional flagella

and chemotaxis related genes.



Name	flhC - linear rlog reads	flhC - adjusted P value	flhC - log_2 fold change	flhD - linear rlog reads	flhD - adjusted P value	flhD - log_2 fold change	WT - linear rlog reads
fliZ	39.924	0	-6.078	42.368	0	-6.005	1643.292
tar	33.124	0	-6.365	35.279	0	-6.302	1580.940
flgG	35.011	0	-6.936	40.163	0	-6.630	2399.959
flgH	27.004	0	-6.030	29.191	0	-5.904	960.120
cheW	55.532	0	-5.159	39.145	0	-6.075	1410.894
flgI	35.957	0	-6.337	36.607	0	-6.398	1742.687
cheA	107.454	0	-5.490	72.177	0	-6.360	3547.725
flgJ	32.227	0	-5.856	33.422	0	-5.822	1112.089
flgK	45.033	0	-6.202	47.428	0	-6.144	2024.721
motB	52.468	0	-5.166	40.574	0	-5.884	1380.328
flgE	79.665	0	-7.618	85.667	0	-7.544	10458.105
motA	50.079	0	-4.463	32.257	0	-5.686	916.399
trg	97.620	0	-3.297	102.989	0	-3.319	766.896
flgD	45.129	0	-7.723	47.125	0	-7.696	3270.772
flgF	33.210	0	-6.822	32.441	0	-7.001	2104.961
flgC	36.982	0	-7.483	34.534	0	-7.798	3651.014
fliI	26.213	0	-6.355	25.460	0	-6.573	1158.409
fliJ	24.147	0	-5.539	25.480	0	-5.477	615.607
fliK	43.963	0	-6.129	50.579	0	-5.857	1841.078
fliL	34.795	0	-6.364	37.487	0	-6.248	1626.622
fliF	63.726	0	-6.824	59.129	0	-7.060	4541.831
fliM	41.475	0	-6.230	41.429	0	-6.310	1910.774
flhA	84.942	0	-4.659	96.517	0	-4.432	1508.108

This marks the end of this tutorial.

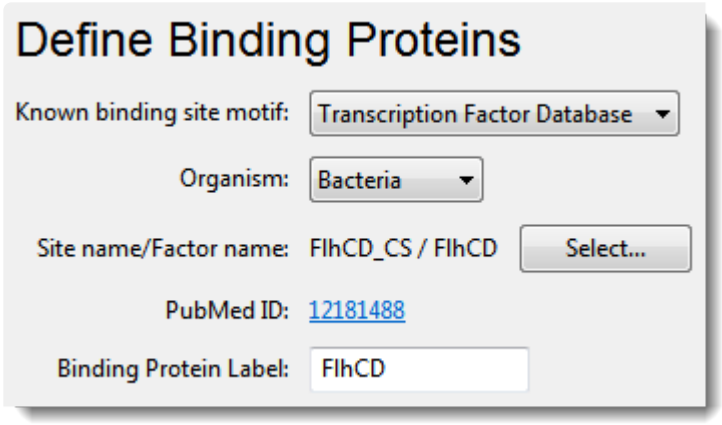
ChIP-Seq workflow with analysis in ArrayStar

Lasergene Genomics' ChIP-Seq analysis workflow enables you to locate the binding sites of DNA-associated proteins and determine how these proteins interact with the DNA to affect expression in nearby genes.

In this tutorial, you will use SeqMan NGen to create a ChIP-Seq assembly for two experimental replicates and one wild type control of *E. coli* transcription factor FlhD. Once the assembly is complete, you will import the data into ArrayStar's ChIP-Seq workflow and filter the results for IP fragments that match in both replicates, and that meet a specified signal threshold. Finally, you will look for genes that the fragments intersect, and see whether they share any common characteristics.

Setting up and running the assembly in SeqMan NGen:

1. Download *T2_ChIP-Seq.zip* (2.4 GB) and extract the contents to any convenient location (e.g., your computer's desktop). The folder contents consist of:
    - The *E. coli* reference sequence: *Escherichia coli str. K-12 substr. MG1655.U00096.gbk*
    - The sample sequences: six folders beginning with *flhC*, *flhD* and *WT*.
2. Launch SeqMan NGen and choose **New Assembly**.
3. Choose **RNA-Seq / Transcriptomics** on the left, then choose **ChIP-Seq** on the right.
4. In the Reference Sequence screen, click **Add** and add the reference sequence *MG1655_U00096.3.gb*. Press **Next**.
5. In the Input Sequences screen, uncheck the box next to **Paired-end reads**. Press the **Add Folder** button to add the folder starting with *WT*. Do the same for each of the next two sample folders. Click **Next**.
6. In the Set Up Experiments screen, check the box to the right of **WT_rep1**, indicating that it is the control. Press **Next**.
7. In the Assembly Options screen, check the box next to **Maximum total reads** and enter 5000000 (5 followed by 6 zeros) to reduce the assembly time. Click **Next**.
8. In the Analysis Options screen, leave the **ChiP-Seq detection method** at **MACS** and press **Next**.
9. In the Define Binding Proteins screen, use the **Known binding site motif** drop-down menu to select **Transcription Factor Database**. Use the **Organism** drop-down menu to select **Bacteria**. Press **Select** and choose **FlhCD_CS / FlhCD**, then press **OK**.



Press **Next**.

10. In the Assembly Output screen, type “ChIP-seq” into the **Project Name** text box. Use the **Browse** button to specify a **Project Folder** for your assembly output files, then press **Next**.

11. In the Run Assembly Project screen, check whether the recommendation is to run the assembly locally or on the cloud. Press the corresponding link to begin assembly. Typical assembly times are 20 minutes (cloud) or 30-45 minutes (local).

12. Wait until being informed that assembly has finished, then click **Next**.

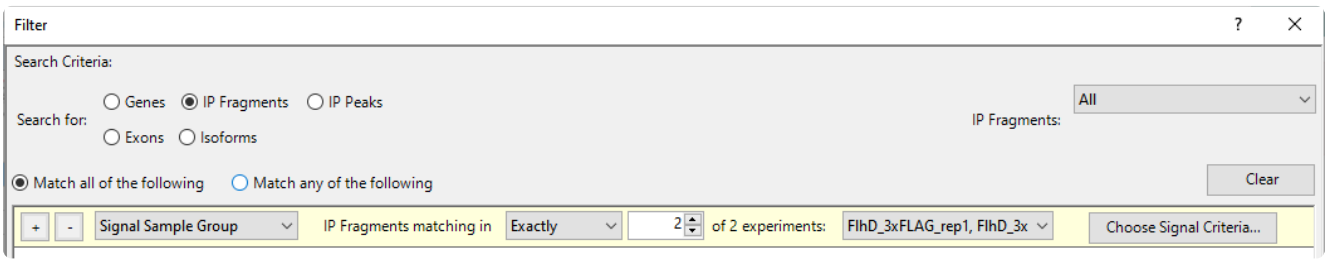
13. From the Assembly Summary screen, click **Analyze peaks** to open the results in ArrayStar.

Analyzing the results in ArrayStar:

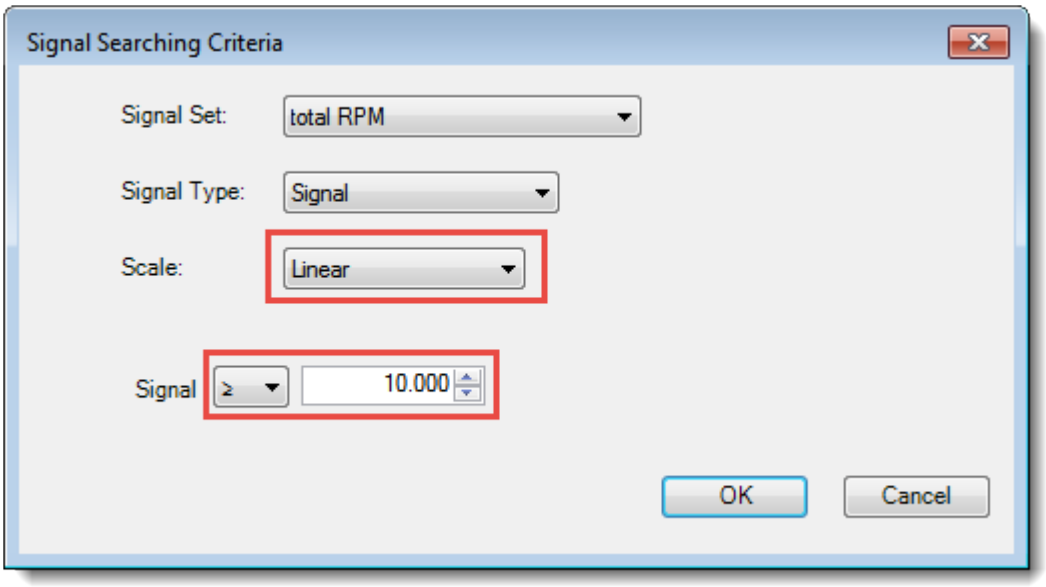
1. In ArrayStar, click on the Fragment Table tab and ensure that approximately 40 IP fragments were found.

2. Similarly, click on the Peak Table tab and ensure that about 50-60 peaks were found.

3. To filter for fragments found in both replicate samples, choose **Filter > Filter All** and make selections to match the image below.



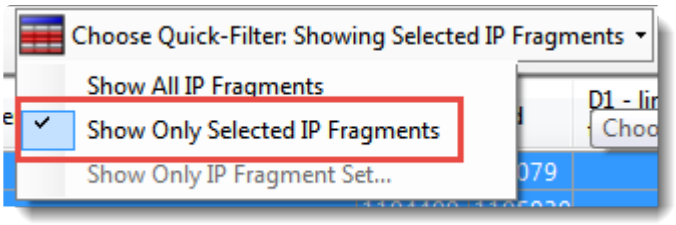
4. Press the **Choose Signal Criteria** button and make selections to match the following image. Press **OK**.



5. Press **Search**. The search should yield about 17 IP fragments.

6. Press the **Select and Show Results in IP Fragment Table** tool (), located above the result table.

7. In the Fragment Table, use the **Quick-Filter** tool to choose **Show Only Selected IP Fragments**.



8. Click on **Add/Manage Columns**. In the IP Fragment Info section, choose **Intersecting Genes**, then **>Add Column>**, then **OK**.

9. Look at the newly-added column and note that many of the fragments intersect genes beginning with “flg”, “fli” and “flh”.

Intersecting Genes
ampH, sbmA
csgA, csgC, ymdA
flgA, flgB
yciK, sohB
flhB, cheZ
azuC, yecR, ftnA
fliA, fliC
fliE, fliF
fliK, fliL
ygdH, sdaC
ygeR
ppiA, tsgA
gntK, gntR
gntR, yhhW
mdtF
yhjV, dppF
gltP, yjcO

These are the major genes for producing flagella in *E. coli* and are known to be regulated by the transcription factor encoded by the *flhC* and *flhD* genes. The *che* gene is involved in chemotaxis and are also known to be regulated by *FlhCD*.

This marks the end of this tutorial.

Copy number variation (CNV) workflow with analysis in ArrayStar and GenVision Pro

Copy-number variation (CNV) is defined as genomic regions that have been repeated one or more times and these variations play an important role in normal genetic variation and in some diseases. DNASTAR's CNV workflow is used to analyze genomic variation by considering changes over a region, as indicated by deleted or duplicated gene copies.

In this tutorial, you will run a set of paired-end Illumina sequence files from MG1655 *E. coli* against the DH10B reference genome. MG1655 has several deleted regions compared to the DH10B reference sequence. In addition, the reference sequence contains a large duplication.

Begin with Part A: Setting up the CNV project in SeqMan NGen.

Part A: Setting up the CNV project in SeqMan NGen

In this part of the tutorial, you will use the SeqMan NGen wizard to import data and run the assembly. You will then press a button to open the results in ArrayStar.

1. Download *T3_CNV.zip* (1.4 GB) and extract the contents to any convenient location (e.g., your computer's desktop). The folder contains the following sequences:

    - Reference sequence *DH10B_NC010473.gbk*
    - Paired end sample sequences *SRR1284938_1.fastq* and *SRR1284938_2.fastq*

2. Launch SeqMan NGen and choose **New Assembly**.
3. Choose **Variant Analysis / Resequencing** on the left, then **Whole genome** under **NGS-Based** on the right.
4. In the Reference Sequence screen, press **Add** and add the file *DH10B_NC010473.gbk*. Click **Next**.
5. In the Input Sequences screen, press **Add** and add the paired reads *SRR1284938_1.fastq* and *SRR1284938_2.fastq*. Use the **Experimental setup** menu to choose **Single sample**, then click **Next**.
6. In the Assembly Options screen, click **Next**.
7. In the Analysis Options screen, choose **Haploid**, since this is a bacterial genome. Check the box next to **Detect CNVs** and keep the default method of **RPK_CN**. Click **Next**.
8. In the Assembly Output screen:

    a. Type "CNV" into the **Project Name** text box. Use the **Browse** button to specify a **Project Folder** for your assembly output files. Click **Next**.

9. In the Run Assembly Project screen, note that the **Estimated coverage** is 375X. A coverage of 50-100X is adequate and additional coverage simply slows down the assembly.
10. To reduce coverage, click **Assembly Options** on the left. Check the box next to **Maximum total reads** and type in 1600000 (16 followed by 5 zeros). Then click **Run Assembly Project** to return to that screen. Note that the **Estimated coverage** is now near 50X.
11. Note the recommendation to assemble on your local computer or on the cloud and click the corresponding link to begin assembly. Local assembly should take approximately 30 minutes.

12. After being informed that assembly has finished, click **Next**.

13. From the Project Report screen, click **Analyze and compare variants** to open the project in ArrayStar.

14. After ArrayStar opens, Use **File > Save Project** to save the project as *CNV.astar*.

15. Close the SeqMan NGen project by clicking the **Finish** button and confirming you would like to close the application.

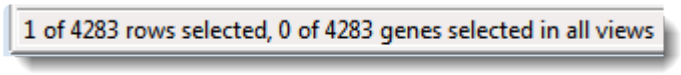
Proceed to Part B: Finding a putative duplication in the reference sequence using ArrayStar.

Part B: Finding a putative duplication in the reference sequence using ArrayStar

In Part A of this tutorial, you ran an assembly and launched the results in ArrayStar. In this part, you will use the ArrayStar Gene Table to locate potential duplications in the reference sequence.

Imagine that you would like to find a region that is repeated in the reference, but present only once in the MG1655 sample. In the Gene Table these regions will have a linear weighted RPKM-CN of approximately 0.5.

1. Click on the Gene Table tab near the top of the ArrayStar window. The footer shows that the Gene Table contains 4,283 genes.

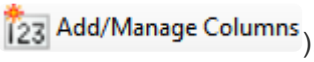
2. Click on the header **CNV (2) – linear weighted RPKM-CN** to sort the column from small to large values

3. Drag the mouse to select all table rows with a linear weighted RPKM-CN from 0.4-0.6, inclusive.

4. Right-click anywhere on the highlighted area and choose **Select and Remember as a Gene Set**.

5. Type in the name **Duplicated in reference**, and then press **OK**.

6. Return to the Gene Table by clicking its tab near the top of the ArrayStar window.

7. Without disturbing the selection, click on the **Add/Manage Columns** tool (

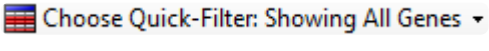
). Select **Target Range** and press the **>Add Column>** button, then click **OK**.

8. Order the genes by ascending location by clicking on the **Target Range** column header.

9. Use the vertical scrollbar to scroll down to the large block of selected rows beginning around Target Range **515000**.

	Name	CNV (2) - linear weighted RPKM-CN	Target Range
☐	ybcN	1.133	511021..511476
☐	ninE	0.760	511476..511646
☐	ybcO	0.779	511639..511929
☐	rusA	0.882	511926..512288
☐	ylcG	0.686	512285..512425
☐	ybcQ	1.207	512511..512894
☐	insH-2	0.715	514308..513292
☑	essD	0.509	515953..516168
☑	ybcS	0.542	516168..516665
☑	rzpD	0.527	516662..517123
☐	rzoD	0.114	516882..517064
☑	borD	0.533	517448..517155
☑	ybcV	0.462	518149..517739
☐	ybcW	0.322	518435..518641
☑	nohB	0.464	519389..519934
☐	appY	0.382	522236..522985
☑	ompT	0.573	524188..523235
☑	envY	0.481	525463..524702
☑	ybcH	0.415	526536..525646
☑	nfrA	0.538	529509..526537
☑	nfrB	0.554	531733..529496
☑	cusS	0.522	533325..531883
☑	cusR	0.414	533998..533315
☑	cusC	0.442	534155..535528
☑	cusF	0.440	535686..536018
☑	cusB	0.474	536034..537257
☑	cusA	0.553	537269..540412
☑	pheP	0.583	540514..541890
☑	ybdG	0.532	543218..541971
☑	nfnB	0.596	543979..543326
☑	ybdF	0.497	544441..544073

This area marks a possible duplication in the DH10B reference sequence.

10. From the **Choose Quick Filter** menu (

) choose **Show Only Gene Set**. Then select **Duplicated in reference** and press **OK**.

    The Gene Table now contains only the ~230 putative duplicated genes.

11. Use **File > Save Project** to save updates to the project.

Proceed to Part C: Confirming the duplication using GenVision Pro.

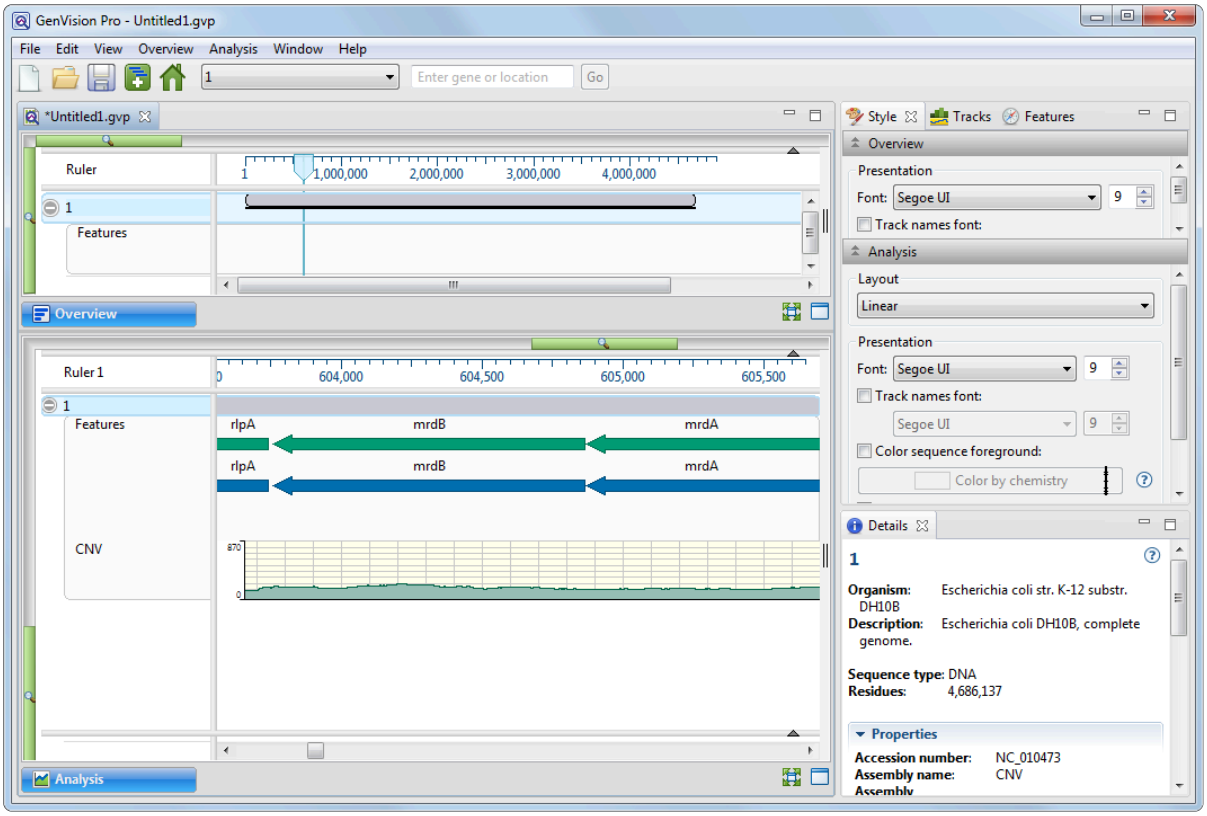
****

Note: The following brief video is not part of this tutorial, but also explores copy-number variation by loading SeqMan NGen assemblies into ArrayStar:

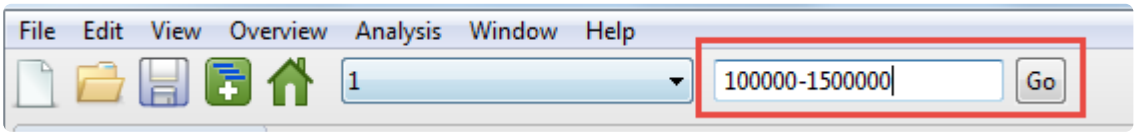
Part C: Confirming the duplication using GenVision Pro

In Part B of this tutorial, you located a putative duplication in the reference sequence using ArrayStar. Deletions or duplications can be confirmed graphically by sending them to SeqMan Pro or GenVision Pro. In this section, you will use GenVision Pro to view a graphical representation confirming the putative duplication found in the previous section.

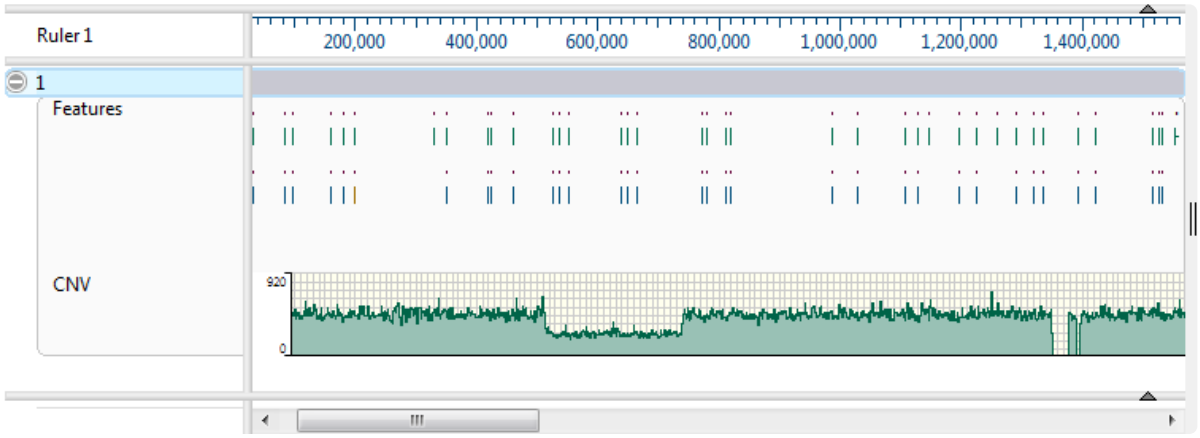
1. In the Gene Table, click on a random row to select it. Then right-click on the row and choose **Send Selection to GenVision Pro**. GenVision Pro launches with the display centered on the target range you selected in Part B.



2. In the navigation area near the top of the GenVision Pro window, enter the range **100000-1500000** and press **Go**.



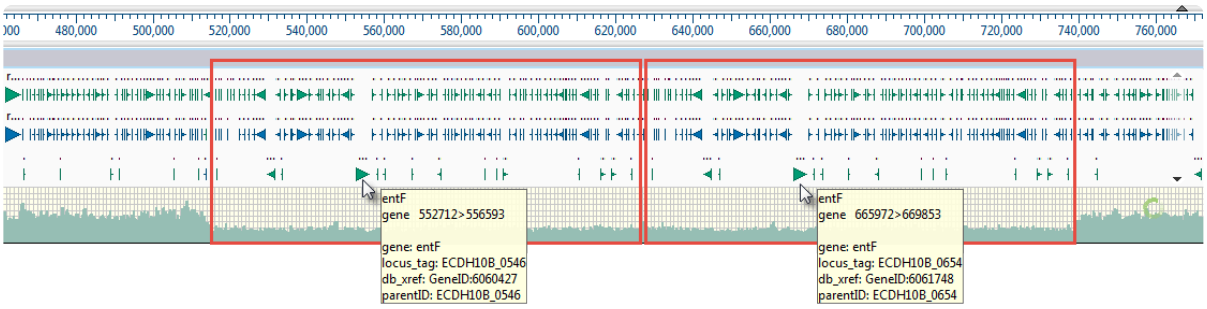
3. Slide the Analysis View's green horizontal slider to the left until you can see the range from approximately 100,000 to 1,500,000.



4. Look at the green coverage graph (“CNV”) at the bottom of the view. Note how the coverage drops in the area from 500,000-700,000. This indicates a likely duplication in the reference sequence. The sample reads have been divided between the two copies, resulting in half the expected coverage in this range. (To the right, you can also see a deletion.)

5. Zoom in on the putative duplication until it fills the entire window. Even though the features are very small at this zoom level, compare the feature sizes and relative positions. It appears there are two duplicate sets of features, side-by-side.

6. Hover over “sibling” features with the mouse, one at a time, to confirm that the feature names match. This further corroborates that the region has been duplicated in the reference sequence.



This marks the end of this tutorial.

Whole genome reference-guided workflow with analysis in ArrayStar

In this tutorial, you will look for deleterious genes in the Caucasian/Utah/Mormon father-mother-daughter trio data from the NIST Genome in a Bottle project. This is commonly referred to as the “CEPH Trio.”

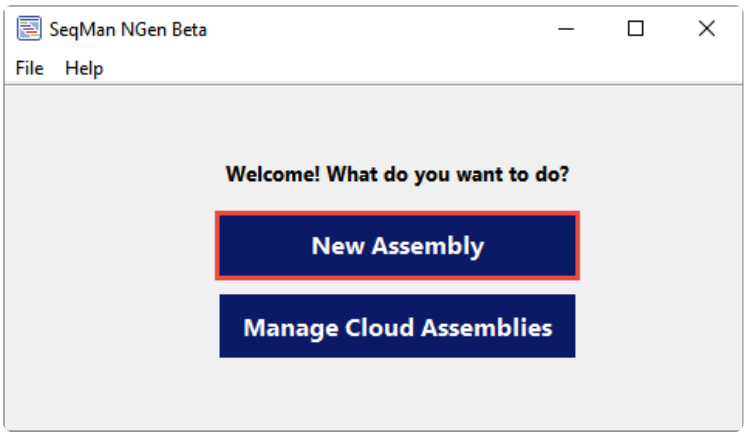
No data are provided for Part A of the tutorial. The samples consist of 2 × 150 bp paired end Illumina reads for NA12891 (father), NA12892 (mother) and NA12878 (daughter). The data set used for assembly is 160 GB in size and requires cloud assembly to perform. Because of this, you will simply read through the steps used to create the whole genome reference-guided assembly in SeqMan NGen. In Part B, you will download the much-smaller finished assembly and follow along with the downstream analysis in ArrayStar.

Begin with Part A: Setting up the assembly in SeqMan NGen.

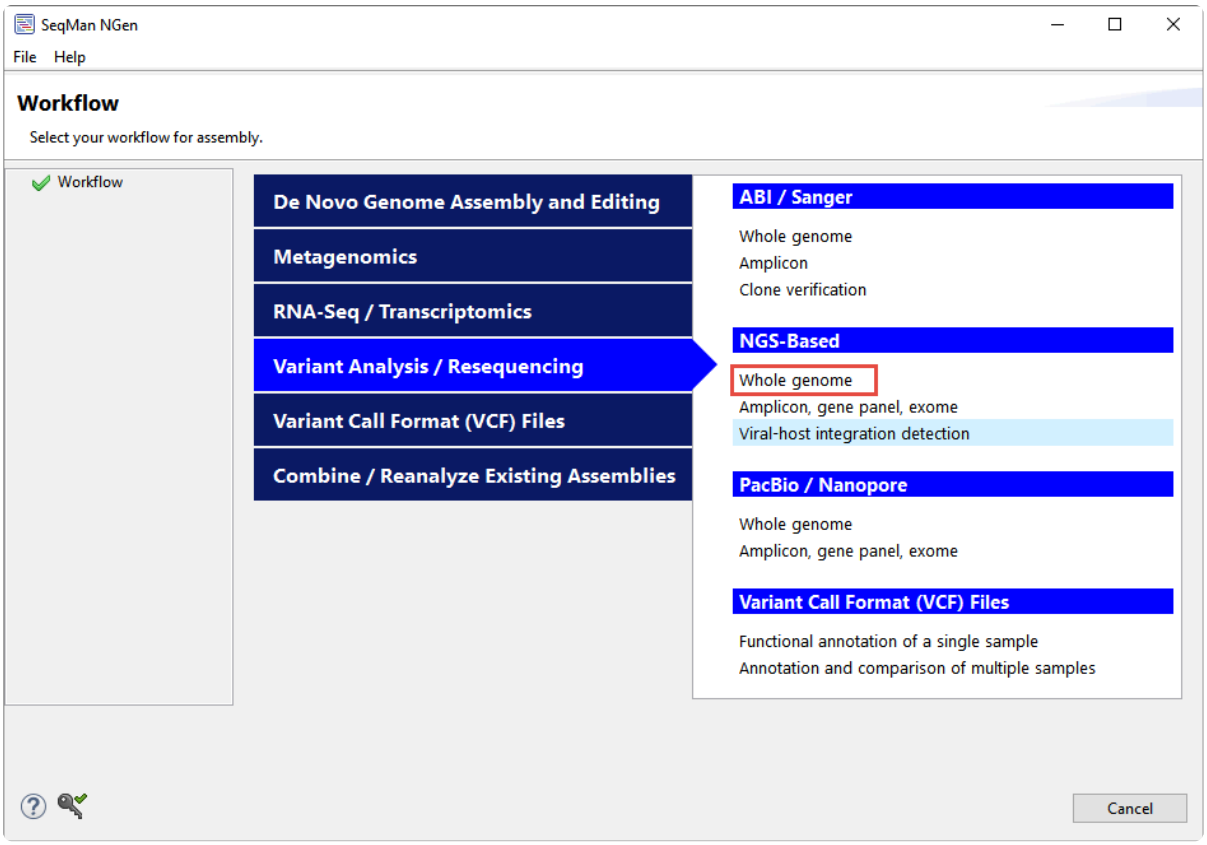
Part A: Setting up the assembly in SeqMan NGen

In this part of the tutorial, you will simply read the steps but won't follow them yourself. No data are provided, as the data set used is 16 GB in size.

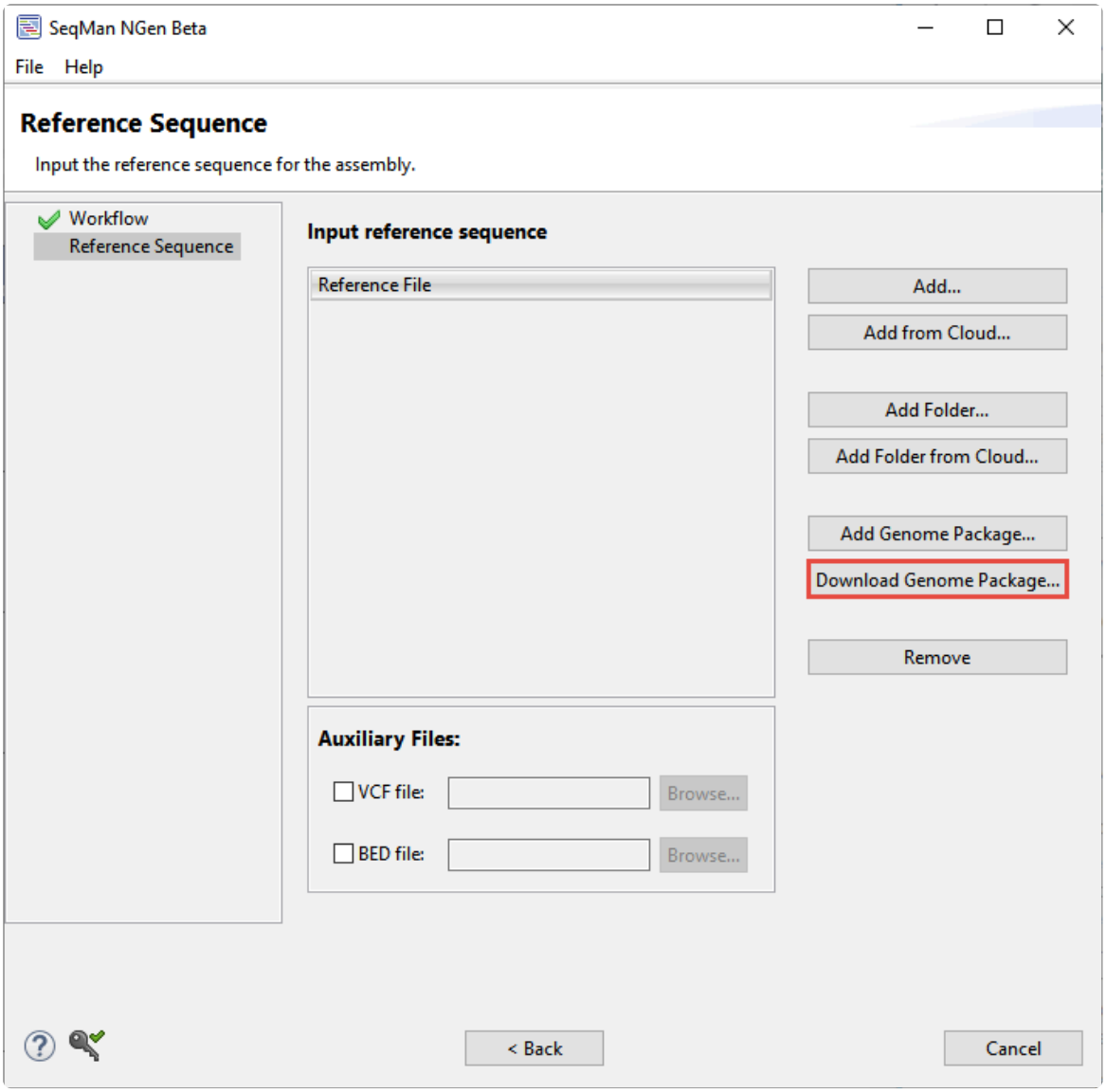
1. Launch SeqMan NGen and choose **New Assembly**.



2. In the Workflow screen, choose **Variant Analysis / Resequencing** on the left. On the right, choose the **NGS-Based** workflow **Whole genome**.



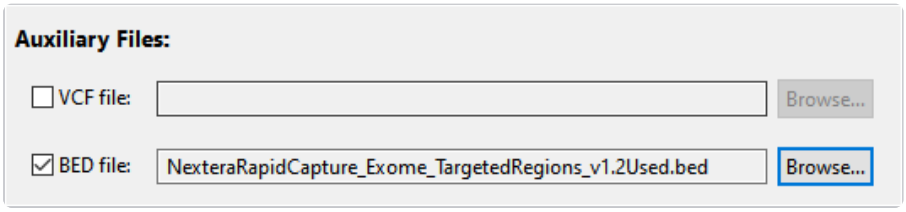
3. In the Reference Sequence screen, click **Download Genome Package**.



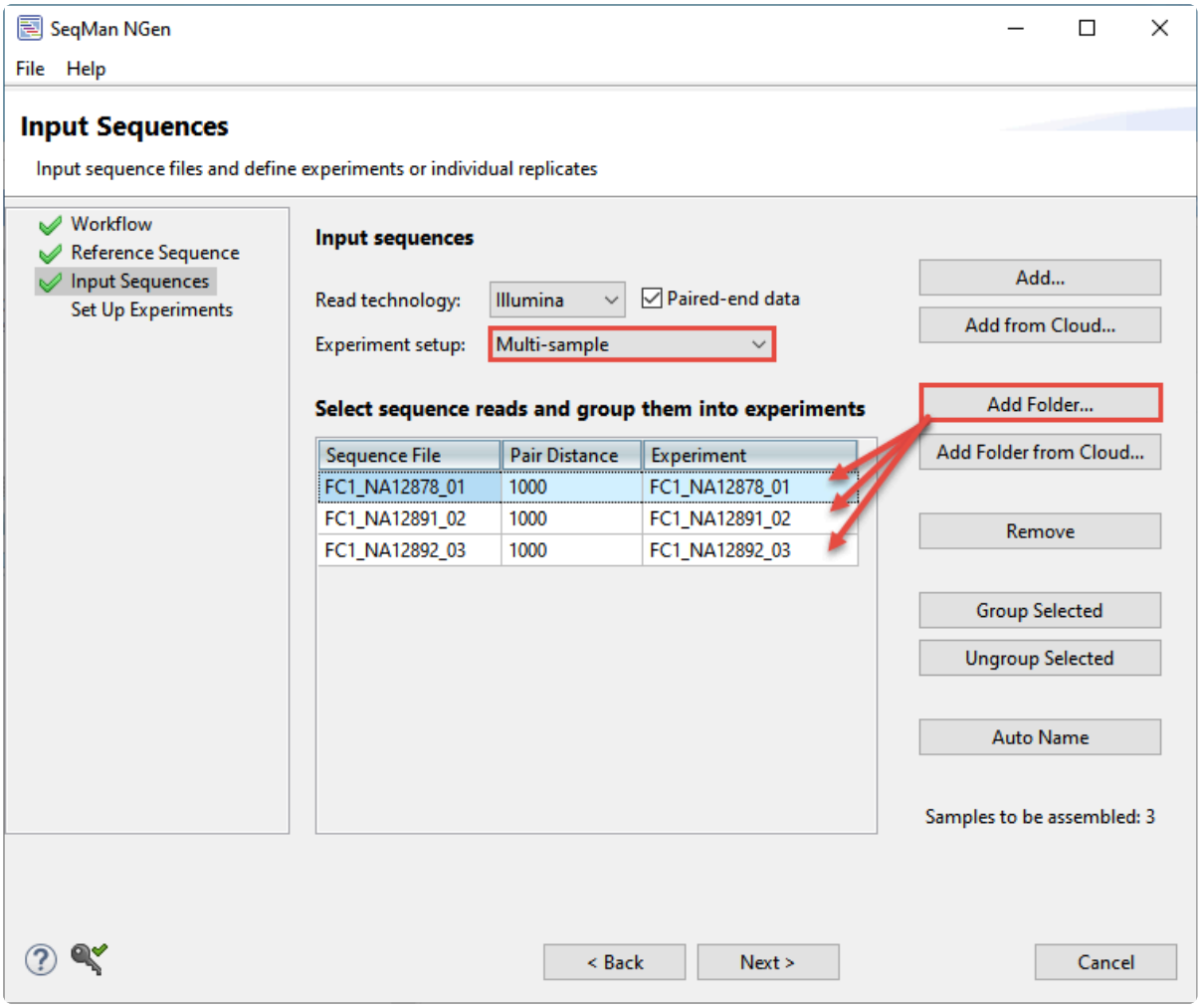
Select **Homo sapiens** build **GRCh38.p7** and press **Select**.

Organism	dbSNP Build	Assembly	Download Size
Apis mellifera	149	Amel_4.5	129.9 MB
Arabidopsis thaliana	138	TAIR10	87.1 MB
Bos taurus	148	UMD_3.1.1	1.8 GB
Escherichia coli K12 MG1655	-	ASM584v2	3.1 MB
Gallus gallus	147	5.0	625.6 MB
Homo sapiens	150	GRCh37.p13	4.9 GB
Homo sapiens	150	GRCh38.p7	3.5 GB
Homo sapiens - Ensembl w/PDB	150	GRCh38.p10 + Ensembl 91	3.5 GB
Homo sapiens mitochondrion	150	GRCh38.p7	33.3 kB
Homo sapiens, CEU reference	142	GRCh37.p10	3.8 GB
Homo sapiens, CHBJPT reference	142	GRCh37.p10	3.8 GB
Homo sapiens, YRI reference	142	GRCh37.p10	3.8 GB
Microbial Genome database	-	-	1.5 GB

4. Check the box next to **BED file** and then use **Browse** to upload *NexteraRapidCapture_Exome_TargetedRegions_v1.2Used.bed*. Click **Next**.



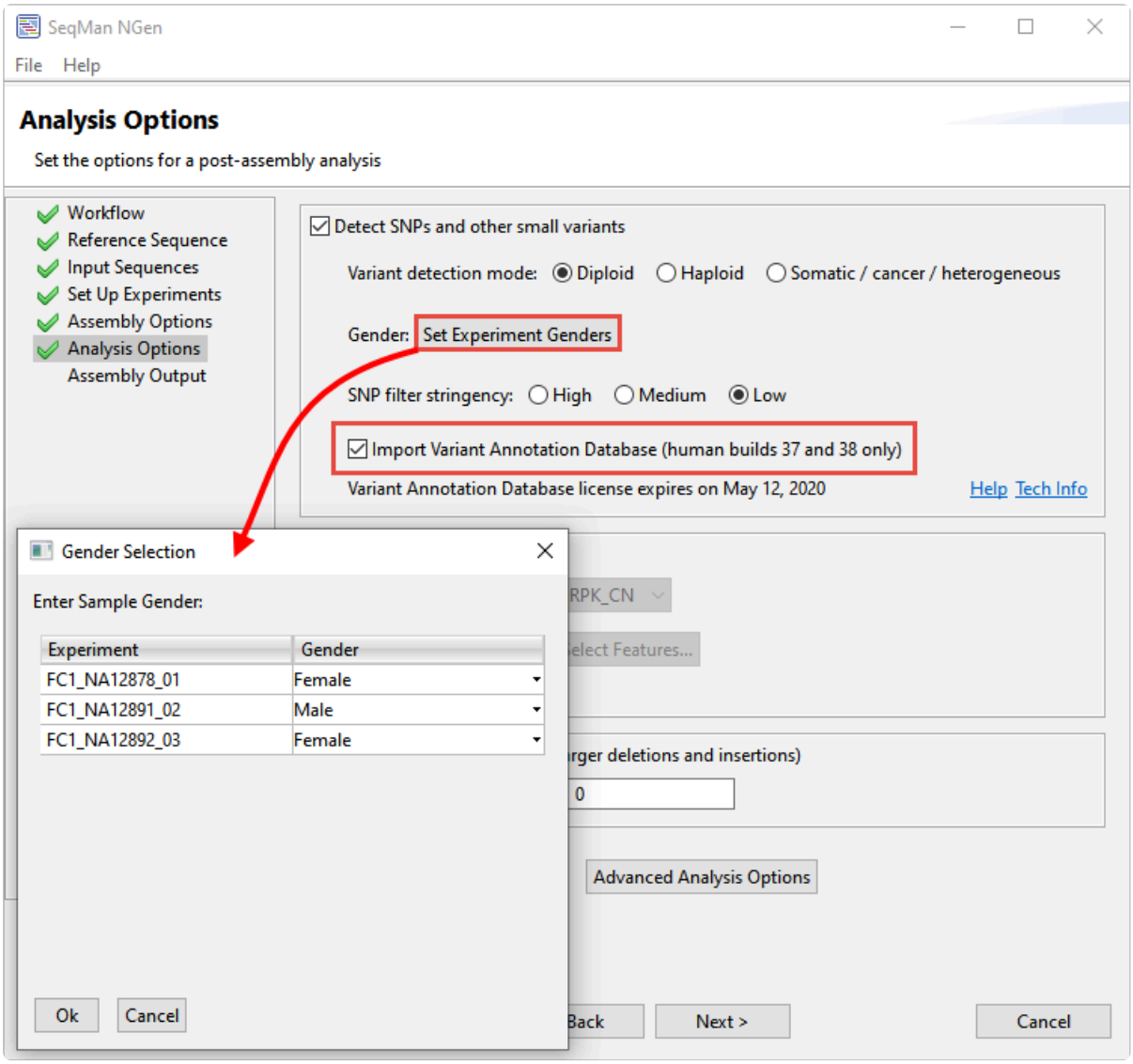
5. In the Input Sequences screen, change the **Experiment setup** to **Multi-sample**. Click **Add Folder** and add *FC1_NA12878_01* (daughter). Repeat twice to add *FC1_NA12891_02* (father) and *FC1_NA12892_03* (mother). Click **Next**.



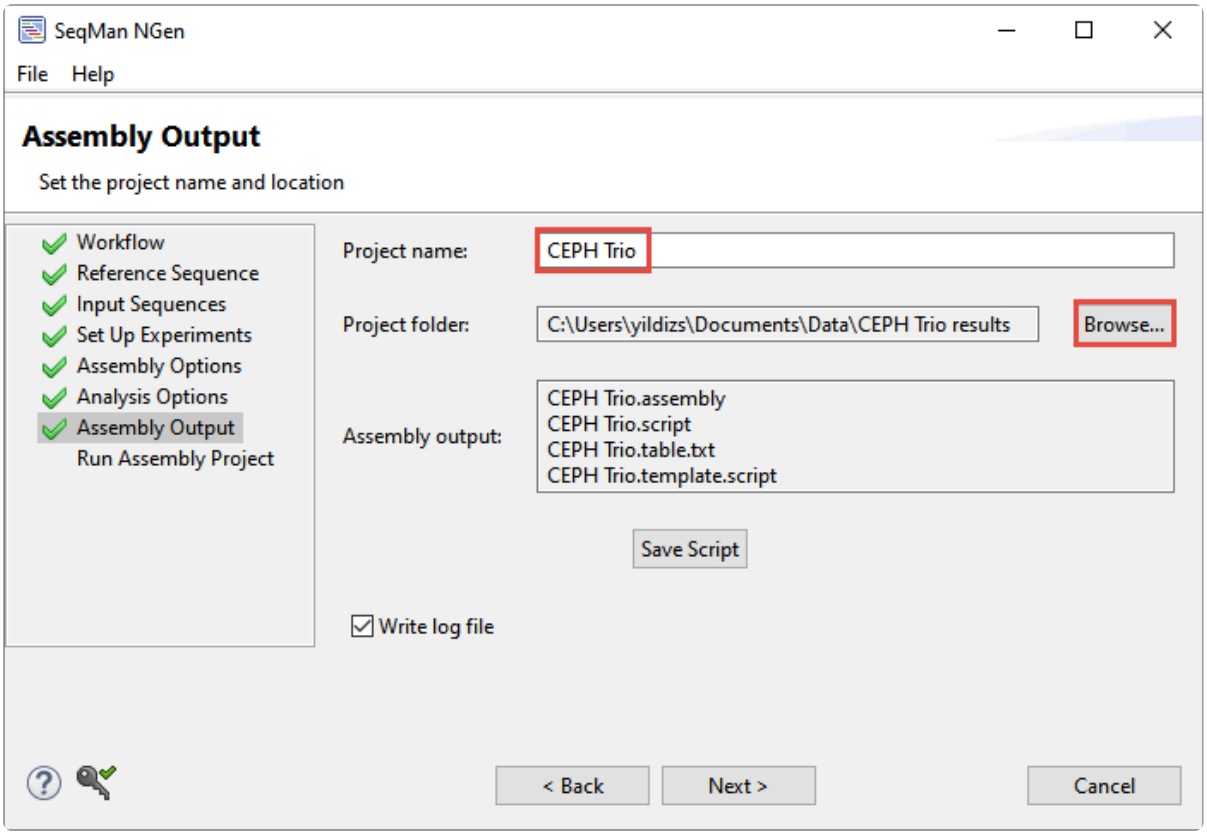
6. In the Set Up Experiments screen, do not check any of the boxes. Click **Next**.

7. In the Assembly Options screen, click **Next**.

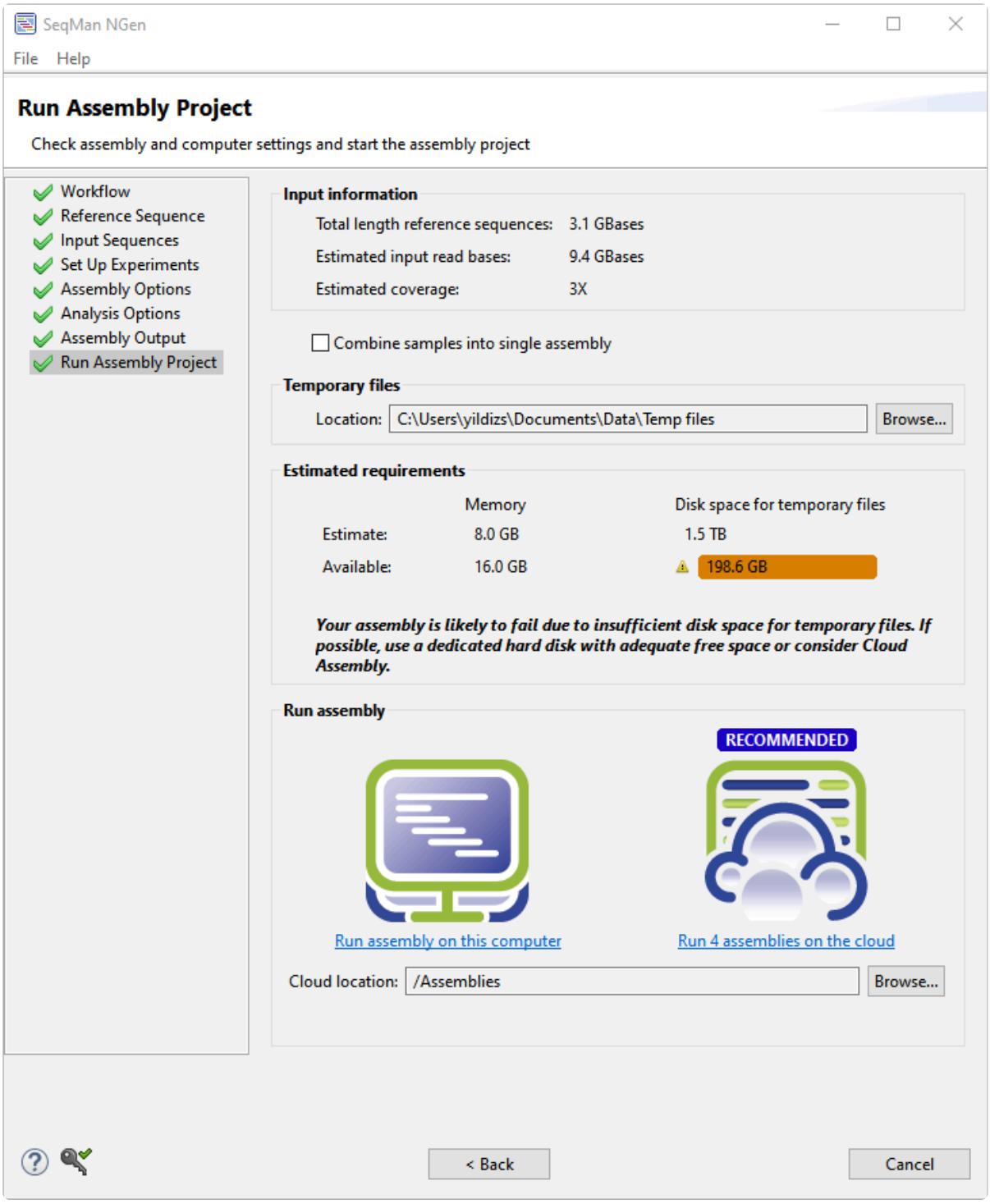
8. In the Analysis Options screen, click **Set Experiment Genders**. Select **Female** for the mother, **Male** for the father and **Female** for the daughter. Click **OK**. Check the box next to **Import Variant Annotation Database**. This will provide a highly enriched source of annotations for genes and other items of interest. Click **Next**.



9. In the Assembly Output screen, type in a **Project name** of **CEPH Trio**. Use **Browse** to select a writable location for the results. Click **Next**.



10. In the Run Assembly Project screen, note that the assembly requires about 1.5TB of disk space for temporary files. The recommendation is to run the assembly on the cloud.



Click the link “Run 4 assemblies on the cloud.” (The number of cloud assemblies needed is usually the number of samples plus one). The assemblies take about 3.5 hours to run on the cloud.

Proceed to Part B. Analyzing the results in ArrayStar.

Part B: Analyzing the results in ArrayStar

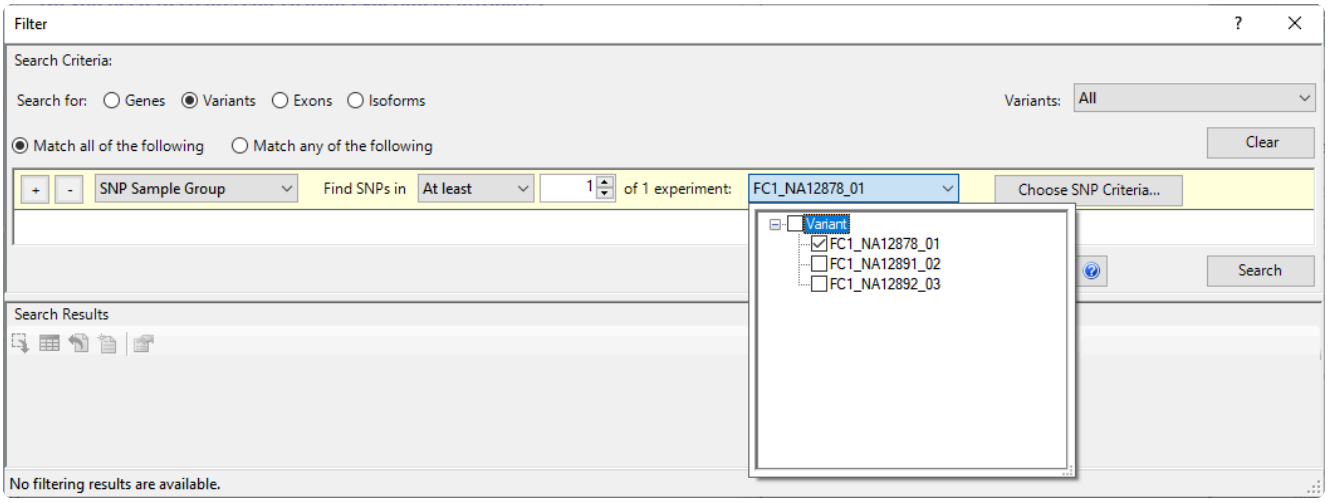
In Part A of this tutorial, you read about how the SeqMan NGen assembly project was set up. In this part, you will download the *CEPH Trio.astar* assembly results file and perform downstream analysis in ArrayStar.

ArrayStar is a discovery tool that provides many different ways to analyze data. The following workflow shows just a sampling of the ways in which you could analyze the CEPH Trio data set. In this case, you will be using advanced filtering to search for potentially deleterious non-synonymous changes in the daughter, and will use the SNP Table to find the source of the SNPs.

1. Download *T4_Whole_Genome.zip* (40 MB) and extract it to any convenient location (i.e. your desktop). The data set consists of a single file, *CEPH Trio.astar*.

2. Double-click on *CEPH Trio.astar* to open it in ArrayStar.

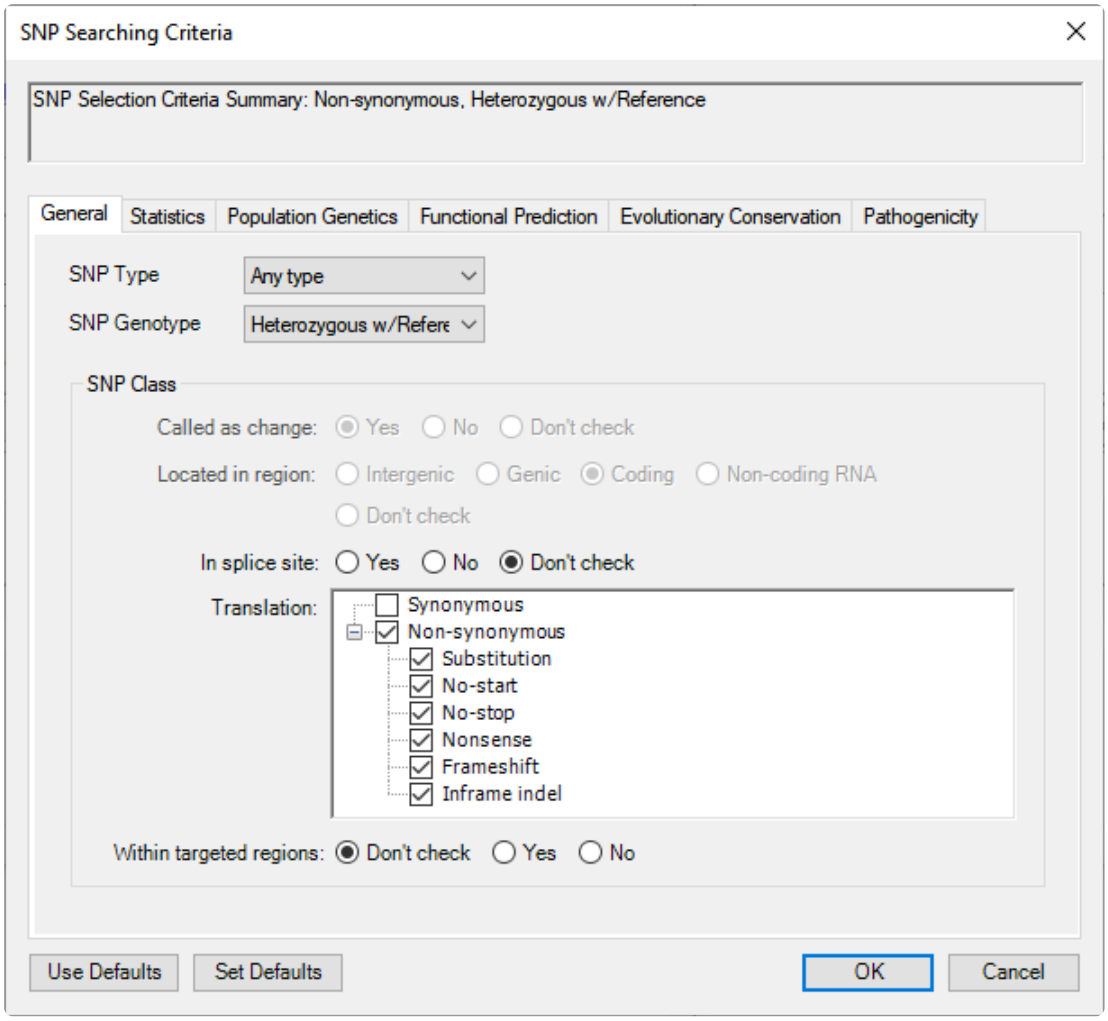
3. Choose **Filter > Filter All**. In the gray heading, keep the default settings **Search for** = **Variants** and **Variants** = **All**. In the yellow filter row, keep the default **SNP Sample Group** and find SNPs in **At least 1** of 1 experiments. To change the second "3" to a "1," click the drop-down menu that begins with **FC1_NA12878** and uncheck all of the boxes except for **FC1_NA12878_01** (the daughter).



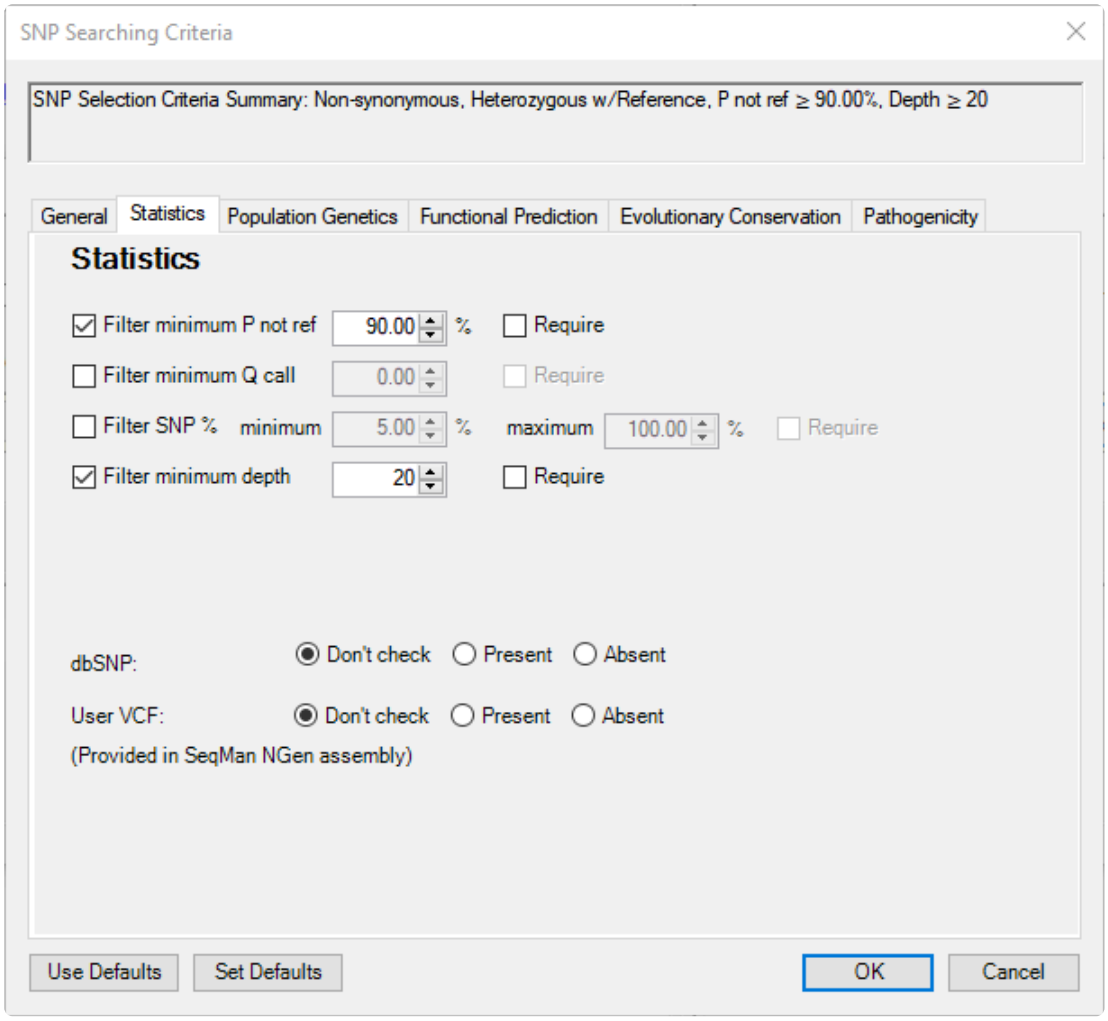
Deleterious changes are unlikely to occur as homozygotes in germline. To reflect this:

4. Click **Choose SNP Criteria**.

    a. In the starting tab (**General**), change **SNP Genotype** to **Heterozygous w/Reference**. In the **Translation** section, check **Non-synonymous**. This will add checkmarks to all subclasses as well.

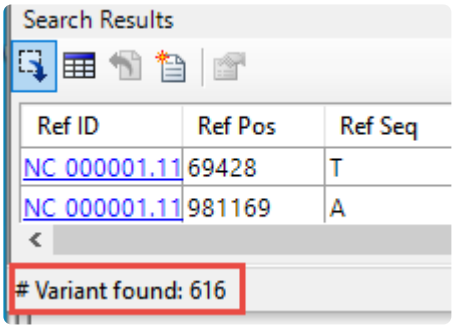


b. Select the **Statistics** tab. Check the box next to **Filter minimum P not ref** and change the number to **90.00** (90%). Check the **Filter minimum depth** box and change the number to **20**.



Press **OK**.

5. Back in the Filter dialog, press **Search**. As shown in the bottom left corner of the dialog, 616 matching variants have been found.

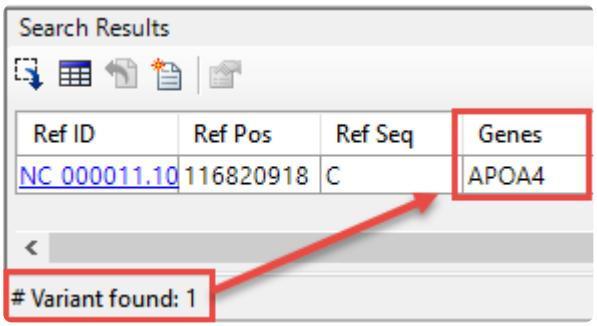


To cut down the number of matches, you will next restrict matches to those variants predicted to be pathogenic in NCBI's ClinVar database.

6. Again press **Choose SNP Criteria**.

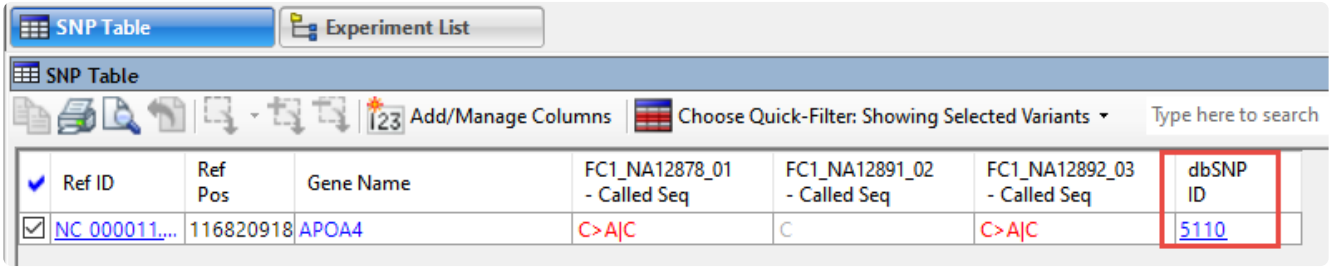
a. Select the **Pathogenicity** tab. On the left, click on **clinvar_clnsig**. On the right, add checkmarks next to **Likely pathogenic** and **Pathogenic**. Press **Add Filter to Set** and then press **OK**.

7. Back in the Filter dialog, press **Search**. Now, a single variant is found that matches all criteria. This variant occurs in the *APOA4* gene.

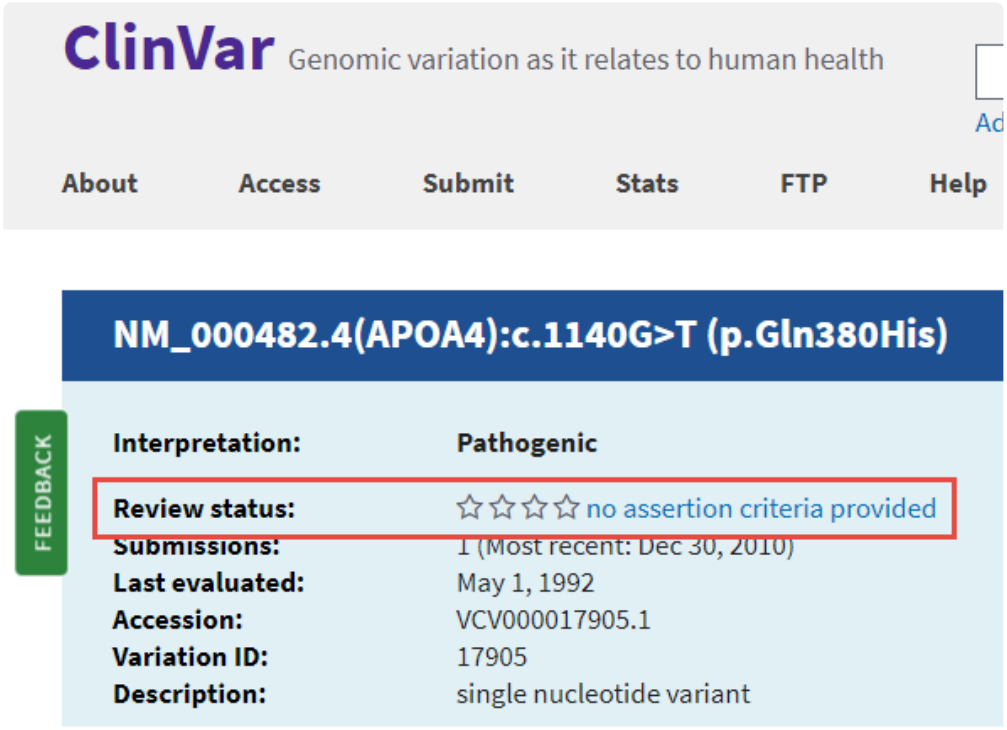


8. Press the **Select and Show Results in Variant Table** tool ( ).

9. In the ensuing SNP Table, click on the link in the **dbSNP ID** column to go to the corresponding web page.



10. On the web page, click the **ClinVar** link to open a page where you can see submitted interpretations, citations and other information about the putative deleterious variant. Note that the Review Status near the top indicates that no assertion criteria have been provided that the variant is pathogenic. This result is therefore inconclusive.



You will next try a different line of inquiry. You will change SNP filtering criteria to filter to those variants predicted to be deleterious rather than pathogenic.

11. Once again, press **Choose SNP Criteria**.

    a. Select the **Pathogenicity** tab and click the **Remove** button to remove the ClinVar criterion.

    b. Select the **Functional Prediction** tab.

        i. On the left, click **LRT_pred**. On the right, check the box next to **Deleterious (D)**. Click **Add Filter to Set**.

        ii. On the left, choose **MutationTaster_pred**. On the right, check both **Disease causing – automatic (A)** and **Disease causing (D)*, then click **Add Filter to Set**.

        iii. On the left, choose **SIFT_pred**. On the right, check **Damaging**, then click **Add Filter to Set**.

        Deleterious mutations are likely to be rare in a given population. So in the next step, you will add criteria to filter to variants that are rare in the European population (MAF <5%).

    c. Select the **Population Genetics** tab. Click on the plus sign next to **1000Gp3_MAF** to expand it, then choose **1000Gp3_EUR_MAF** (the European group). Click **Add Filter to Set**.

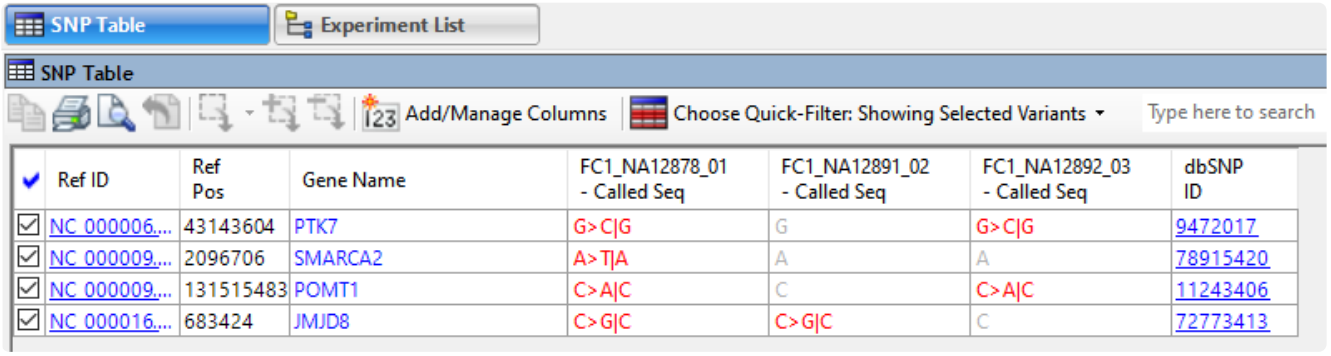
    d. Click **OK**.

12. Back in the Filter dialog, press **Search**. This search yields 4 variants.

    In the final step, you will identify the parent from which each variant was inherited.

13. Click the **Select and Show Results in the Variant Table** tool ( ).

    The SNP Table contains three columns for **Called Seq**. The column ending in 01 pertains to the daughter; 02 to the father; and 03 to the mother. Look at the red text in the **Called Seq** columns. This indicates that two of the four variants were inherited from the mother; one from the father; and one arose *de novo* in the daughter.



Ref ID	Ref Pos	Gene Name	FC1_NA12878_01 - Called Seq	FC1_NA12891_02 - Called Seq	FC1_NA12892_03 - Called Seq	dbSNP ID
NC_000006....	43143604	PTK7	G>C|G	G	G>C|G	9472017
NC_000009....	2096706	SMARCA2	A>T|A	A	A	78915420
NC_000009....	131515483	POMT1	C>A|C	C	C>A|C	11243406
NC_000016....	683424	JMJD8	C>G|C	C>G|C	C	72773413

The variant in this gene...	... is found in:
PTK7	mother and daughter
SMARCA2	daughter only
POMT1	mother and daughter
JMJD8	father and daughter

This marks the end of this tutorial.

Wizard screen descriptions

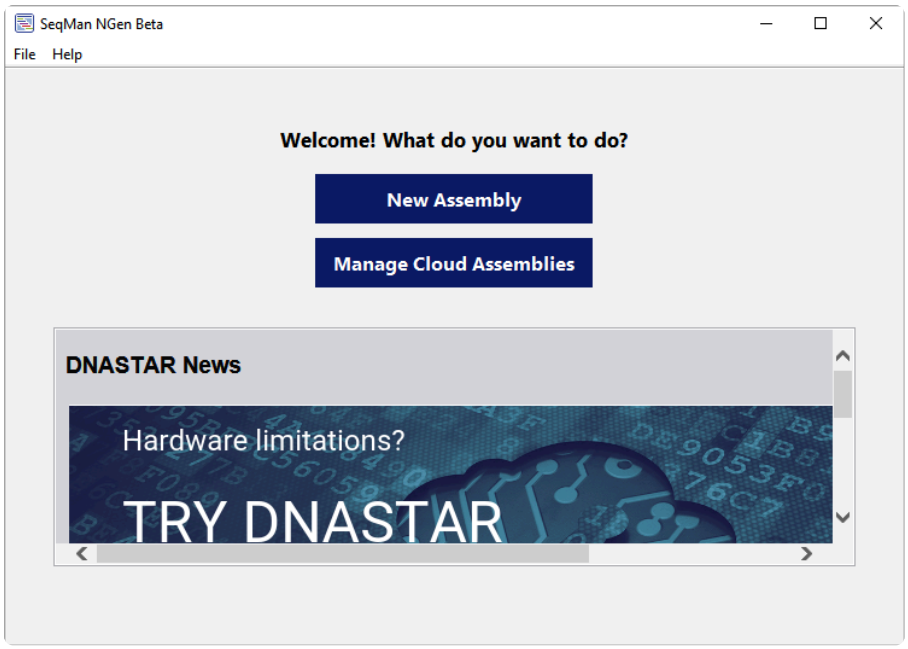
SeqMan NGen's project setup wizard lets you upload files and optimize parameters for your assembly. After choosing your desired workflow in the first ("Workflow") screen, subsequent wizard screens will vary according to the workflow chosen. Most workflows include 5-8 wizard screens. Each screen is described in a separate topic:

- Welcome screen (appears only when you first open SeqMan NGen)
- Workflow (used in all workflows)
- Reference Sequence
- Transcript Annotation Database
- Input VCF Files
- Set Contaminant
- Input Sequences
- Set Up Experiments
- Set Up Replicate Sets
- Assembly Options
- Input Short Read Sequences
- Analysis Options
- Define Binding Proteins
- Refinement Options
- Post Assembly Options
- Assembly Output (used in all workflows)
- Run Assembly Project (used in all workflows)
- Cloud Monitor
- Assembly Log
- Assembly Summary

Additional information pertaining to the wizard can be found in Add and remove files in the wizard and Use editing commands in the wizard.

Welcome

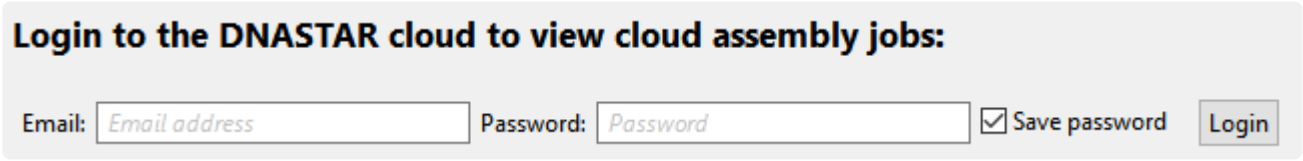
Each time you launch SeqMan NGen, the first screen that appears is the “Welcome” screen.



The top of the screen has two option buttons.

- Press **New Assembly** to set up and run an assembly project using the SeqMan NGen wizard. You will start at the Workflow screen. Subsequent screens depend on which option you choose in the Workflow screen.

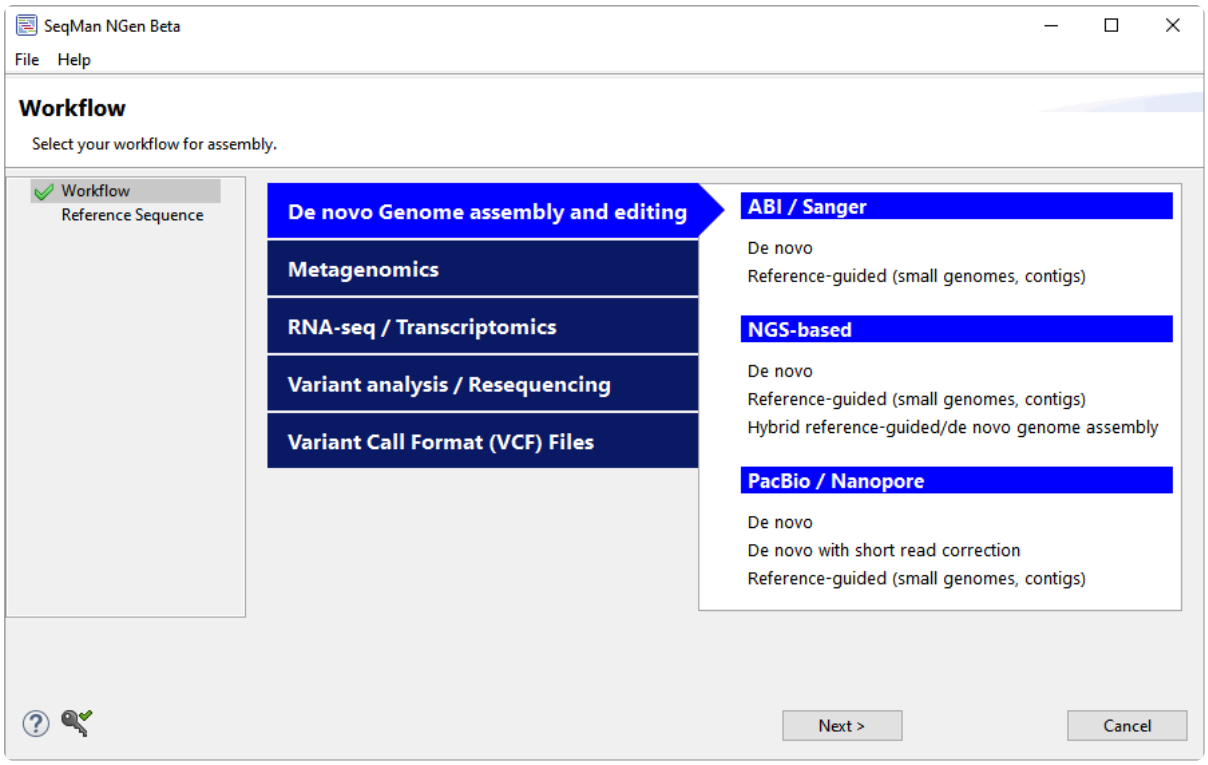
- Press **Manage Cloud Assemblies** to go to the Cloud Monitor screen, where you can monitor current or past cloud assemblies. If you are not currently logged in, you will be prompted to log in using the same email and password you use to access your information on the DNASTAR website.



The bottom of the screen contains news, upcoming events, or new feature announcements from DNASTAR.

Workflow

“Workflow” is the first wizard screen in SeqMan NGen and is where you select the assembly workflow. Each group of workflows is accessed by clicking its dark blue “bar-shaped” tab on the left.



Click a link below for descriptions of each workflow in that group.

- De novo genome assembly and editing
- Metagenomics
- RNA-seq/transcriptomics
- Variant analysis/resequencing
- Variant Call Format (VCF) files

De novo genome assembling and editing workflows

The following table describes each of the workflows available in the **De novo genome assembly and editing** tab of the Workflow screen.

Group	Workflow	Description
ABI / Sanger	De novo	A non templated assembly of up to 30 million sequence reads and up to a 50 Mbase total length for all contigs combined. The capacity is determined by the amount of available RAM. When assembling a data set de novo, we recommend using paired end data, if available.
	Reference-guided (small genomes, contigs)	Assembles/aligns reads onto one or more reference sequences/templates. This workflow is most frequently used for extending off the ends of saved contig consensus sequences. This type of assembly can include up to 10 million reads and up to a 100 Mbase genome. It can be edited at a later time using a utility like SeqMan Ultra or SeqMan Pro.
NGS-based	De novo	A non templated assembly of up to 30 million sequence reads and up to a 50 Mbase total length for all contigs combined. The capacity is determined by the amount of available RAM. When assembling a data set de novo, we recommend using paired end data, if available.
	Reference-guided (small genomes, contigs)	Assembles/aligns reads onto one or more reference sequences/templates. This workflow is most frequently used for extending off the ends of saved contig consensus sequences. This type of assembly, which uses mate-pair data, can include up to 10 million reads and up to a 100 Mbase genome. It can be edited at a later time using a utility like SeqMan Ultra or SeqMan Pro.
	Hybrid reference-guided/de novo genome assembly	This option utilizes both reference-guided and *de novo* assembly steps to resolve three types of structural variation (SV): insertions, deletions and replacements (indels) with minimal user intervention. In this workflow, your data should be from a haploid genome with at least one mate pair data set with read lengths of 100 bases or greater. Your total number of reads should be 10 million or less. If you use a larger data set, only the first 10 million reads will be used. For mate pair data, equal numbers of matching forward and reverse reads are processed. The SQD-formatted assembly can be edited at a later time using SeqMan Ultra or SeqMan Pro. When opened in SeqMan Ultra, contigs will already be organized into scaffolds in the Explorer panel. This workflow replaces the “gap closure workflow” from Lasergene 16.0 and before. This newer version features an additional “refinement” stage before the “gap closure” stage and some additional “finishing” steps after the gap closure portion takes place. During assembly, data is processed in several stages:

		1. Data is mapped and aligned to a user-defined reference genome and then analyzed for characteristic SV motifs. 2. The reference sequence is split at the detected SV sites, forming a series of ordered contigs. 3. Mate pair and split reads from each SV event are collected in site-specific pools and assembled *de novo*. Deletions are detected using three types of data: split reads, spanning paired-end reads, and sequence coverage information. For insertions and replacements, mate pair reads corresponding to the new sequence are collected from the unassembled read pool. Only reads anchored by mates flanking the SV in the main assembly are used at this stage. 4. The *de novo* assembled contigs are then brought into the main assembly and positioned consistently with the mate pair information. 5. For SVs where the gap is not completely covered by the *de novo* assembled contigs (e.g. insertions longer than twice the size of the insert library), additional reads from the unassembled read pool matching and extending the ends of the joining contigs are added in an attempt to “walk” across the gap. This walk is terminated when either no new reads are found or when a repeated element is encountered.
	Short read polishing of a long read draft genome	This workflow is designed to correct assembly errors (single substitutions, indels, mis-assemblies, and coverage gaps) using DNASTAR’s novel algorithm. This workflow is similar to the **Hybrid reference-guided/de novo genome assembly** workflow above, but focuses on taking a draft long-read assembly from an outside application (e.g., CANU) and correcting it with Illumina data.

Create a reference-guided assembly to use in the “SNP to Structure” workflow

If you are working with reference-guided human assemblies, Lasergene’s “SNP to Structure” workflow lets you combine genomic sequencing and variant level data with structure files from the RCSB Protein Data Bank (PDB) to model point mutations on the protein structure and assess the effect on protein stability. By combining structural bioinformatics with sequencing technologies, this integrated workflow can guide genomic and molecular biology researchers to create structure-based hypotheses and to investigate possibilities not evident by sequences alone.

This workflow requires that you be licensed to use several Lasergene applications: SeqMan NGen, SeqMan Pro / SeqMan Ultra and Protean 3D (all required), and ArrayStar (optional).

Only Part A of the workflow involves SeqMan NGen. However, all parts of the workflow are described below.

Part A: Create a reference-guided assembly in SeqMan NGen:

1. In SeqMan NGen’s Workflow screen, choose a reference-guided workflow.

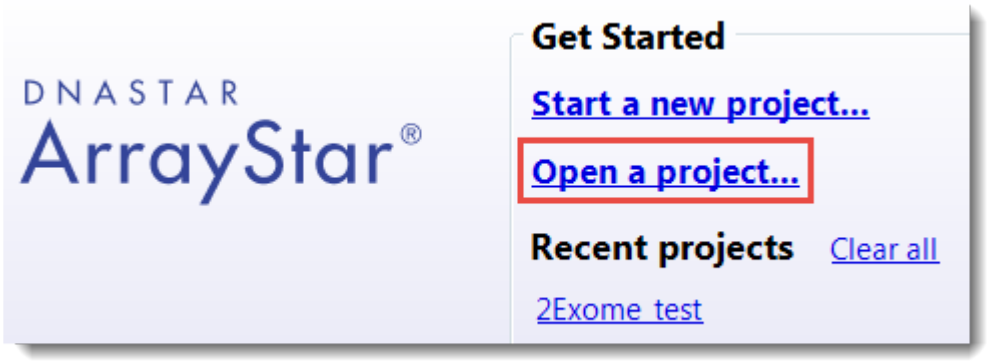
2. In the Reference Sequence screen, add the DNASTAR genome template Homo_sapiens-GRCh38-Ensembl-dbSNP150.zip_. This template contains the mapping of the sequences of PDB structures to the human genomic coordinates, and will later allow SeqMan Pro to communicate with Protean 3D. SeqMan NGen outputs an *.astar* and an *.assembly* package to use in later steps.

3. Follow the rest of the wizard steps and create the assembly.

Part B (optional): Filter for variants of interest in ArrayStar:

If your assembly has a large number of variants, you can use ArrayStar to filter them down to a smaller group of interest before sending them to SeqMan Pro for viewing. This step is highly recommended for all but very small assemblies.

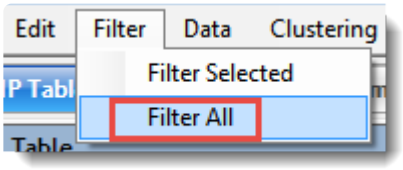
1. Launch ArrayStar and choose **Open a project**.



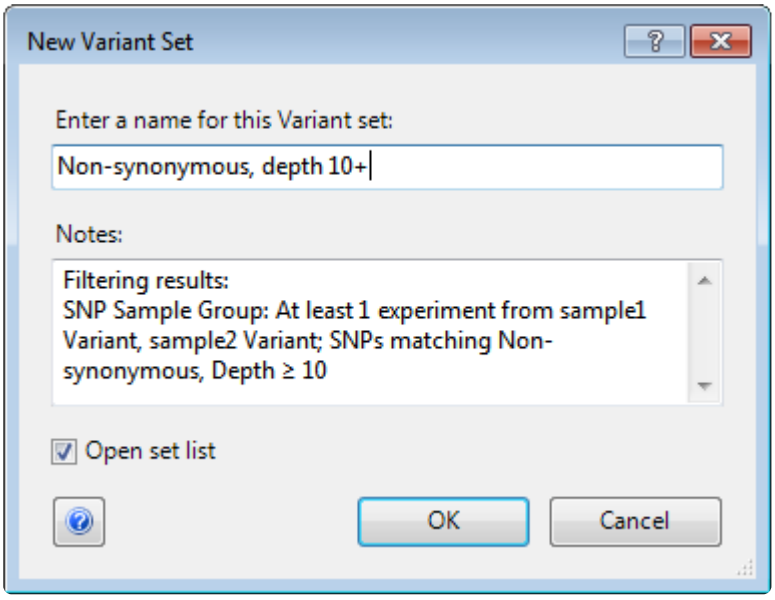
2. Navigate to and select the *.astar* file output by SeqMan NGen.

3. When the file has loaded, click on the **SNP Table** tab.

4. Use **Filter > Filter All** to perform any desired filtering.



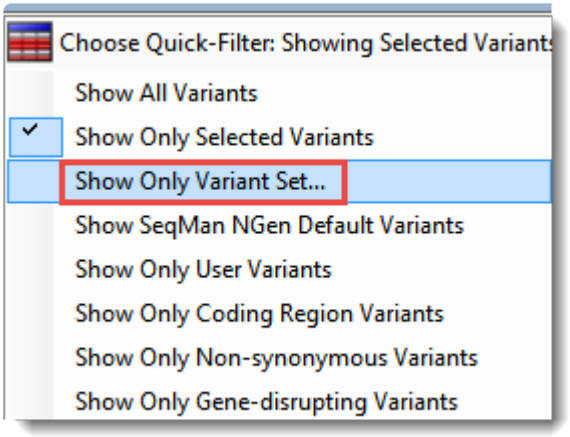
5. At the conclusion of filtering, click the **Remember Results as a Variant Set** tool ( ) above the Search Results table. Type in any desired name and press **OK**.



6. In the Action section in the center-right of the window, click the link **Select and show the table of this set's Variants**.

7. Use the **Choose Quick-Filter** drop-down to select **Show Only Variant Set**. In the ensuing dialog box,

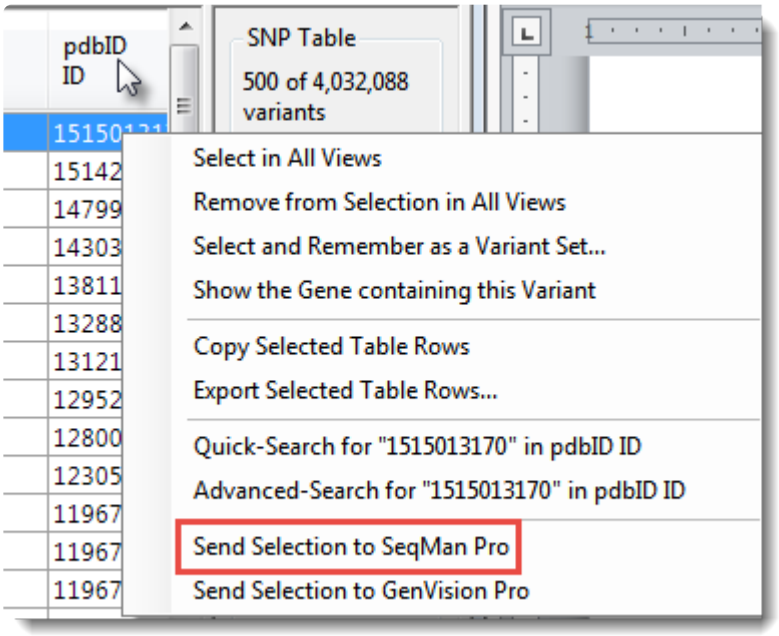
select the named set and click **OK**.



8. Click on **Add/Manage Columns**. Select **SeqMan NGen Assembly Variants**, then **pdbID**. Click **Add Column** and **OK**.

9. In the SNP Table, click on the **pdbID** column header to sort the column and locate rows with PDB entries.

10. Within the subset of rows with PDB entries, select any row of interest. Then right-click on it and choose **Send Selection to SeqMan Pro**.



11. If prompted, select an individual sample of interest and press **OK**.

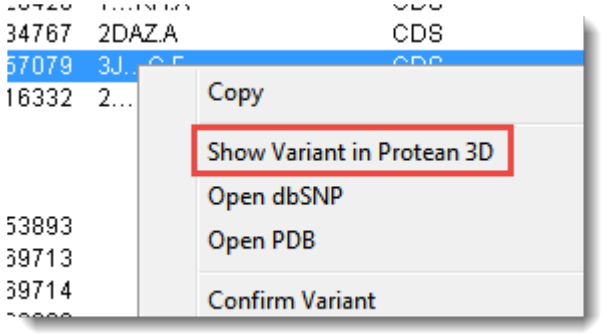
Part C: View variants in SeqMan Pro or SeqMan Ultra:

Instructions below pertain to SeqMan Pro, but are similar in SeqMan Ultra.

1. If you are coming from Part B, above, the sample is automatically selected and its Alignment view is opened. Continue directly to Step 2. If you are coming from Part A, launch SeqMan Pro and use **File > Open** to open the *.assembly* file.

2. Choose **Variant > Variant Report**. (If desired variants are being filtered out, you may need to click on the **Show All** button.)

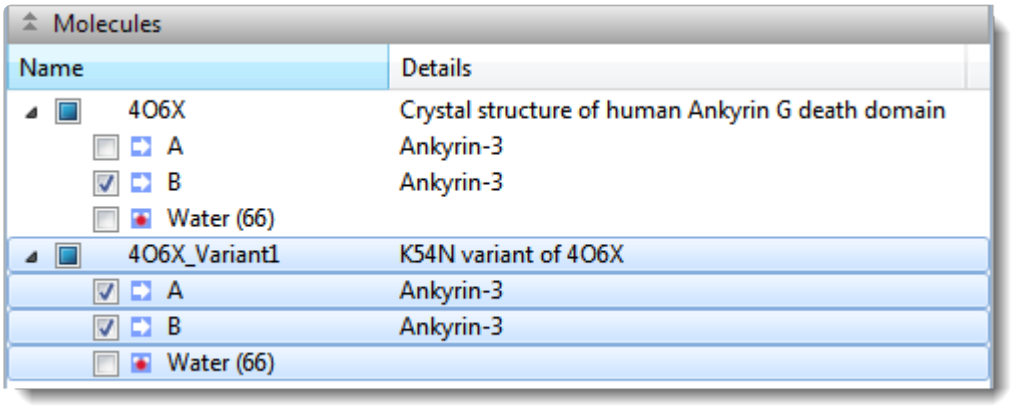
3. Click on the **PDB ID** column header to sort items with PDB IDs to the top. Select one or more rows, then right-click within the selection and choose **Show Variant in Protean 3D** (or use **Variant > Show Variant in Protean 3D**).



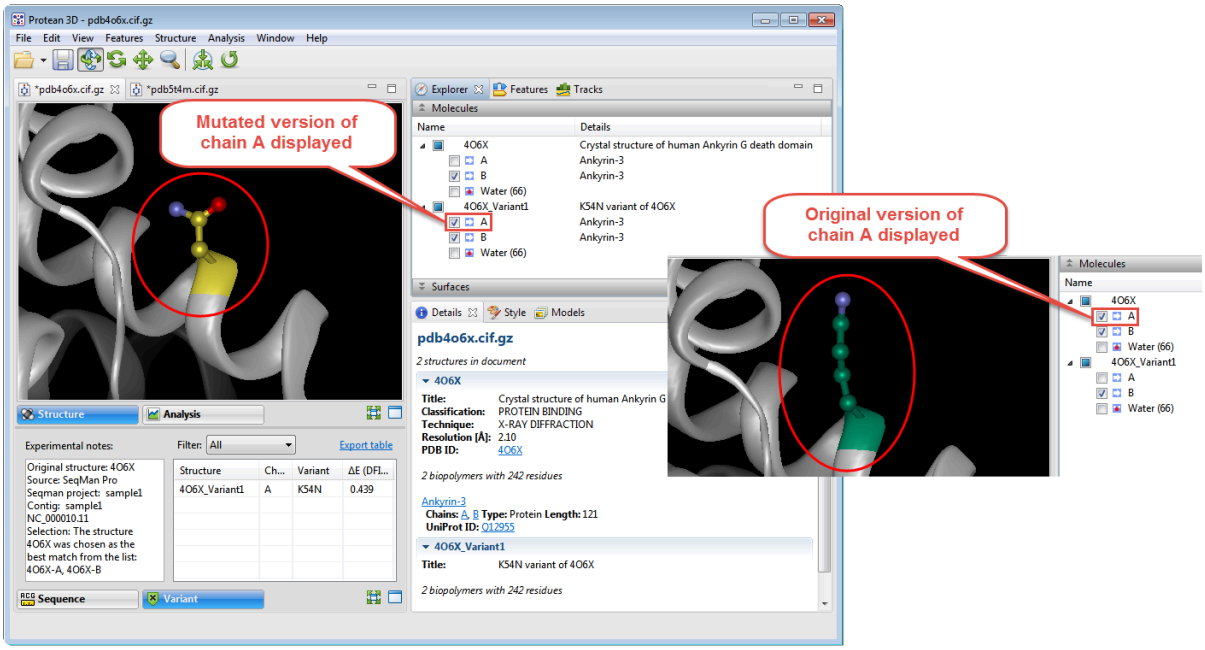
Part D: View the protein structure in Protean 3D:

After finishing Part C, the protein structure with the variant of interest opens in Protean 3D.

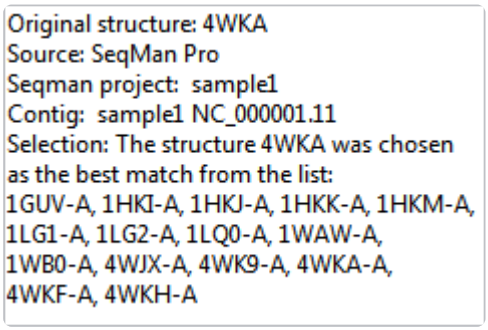
In the Molecules area, two near-identical copies of the structure appear. The upper structure is the original structure from the Protein Data Bank. The lower structure is the variant version calculated by Protean 3D.



- Use the Structure view to observe the mutated side chain along the backbone. To show/hide the different versions of each chain, check/uncheck boxes in the Molecules area.



- To see notes about the structure chosen as the best match, look in the Experimental notes box at the bottom of the Variant view. If the Variant view is not visible, click on the **Variant** tab at the bottom of the Protean 3D window.



> **Note:** If the residue has no atoms, you will see an error message in this box. In this case, return to SeqMan Pro and choose a different row in the Variants table.

Protean 3D uses several metrics to determine the “best” PDB file to display when a variant is located in a CDS that is associated with more than one PDB file. Quality is the first consideration, with high-resolution crystal structures > NMR structures > low-resolution crystal structures > other techniques. This ordering is refined by alignment to the corresponding Uniprot sequences. This refinement considers the percent match and the number of gaps before the variant position. If two structures are still tied as the “best,” the largest structure is chosen.

- To predict whether the mutation is stabilizing or destabilizing to the protein structure, use the table on the right of the Variants view. The **delta-E (DFIRE-A)** column displays the change in energy value based on the DFIRE calculation (reference). This number can be used to predict whether the mutation is stabilizing or destabilizing to the protein structure. A positive number is considered destabilizing to

the structure when compared to the original amino acid; a negative number is considered non-destabilizing.

> **Note:** If multiple entries appear in the table, you can sort them by clicking on their column headers, or filter them using the **Filter** drop-down menu. If desired, you can save the table by clicking the **Export table** link.

- To further explore potential impact of mutation on protein stability and function, apply a solvent-accessible surface.
- Analyze secondary structure characteristics to interrogate the mutation's effect on protein flexibility, amphiphilicity, charge density, hydropathy, and more.

Remove PhiX control reads from Illumina data prior to import

During *de novo* assembly, contamination of Illumina data with PhiX control sequence may result in the generation of spurious contigs. For background information, see Mukherjee et al., 2015.

Note that:

- Not all Illumina data are contaminated with PhiX.
- PhiX contamination is not a major concern for reference-guided assemblies.
- In informal tests at DNASTAR, the amount of contamination in most data sets was so low that the spurious contigs were automatically discarded for being “under the minimum coverage” per the SeqMan NGen defaults. And when coverage was higher, contaminated contigs contained *only* PhiX174 reads and could be readily recognized post-assembly.

If you are following a *de novo* workflow, you can easily remove PhiX contamination prior to assembly by following these steps:

1. Download the sequence NC_001422.1 (*Enterobacteria phage phiX174 sensu lato, complete genome*) from the NCBI website.
2. Launch SeqMan NGen.
3. In the Workflow screen, choose a *de novo* assembly option.
4. In the Assembly Options screen, check the **Contaminant Scan** box. Use the associated **Add** button to add the PhiX174 sequence downloaded in the previous step.
5. Proceed through the rest of the SeqMan NGen wizard screens and assemble as usual.

Any PhiX174 sequence will be removed prior to assembly.

Metagenomics workflows

The following table describes each of the workflows available in the **Metagenomics** tab of the Workflow screen.

Group	Workflow	Description
ABI / Sanger	De novo	This workflow allows optional removal of specified host DNA. The default parameters for this workflow have been optimized to take into account the short read lengths and presence of repetitive DNA sequences common to metagenomic and 16S rRNA data.
	Reference-guided	This workflow allows optional removal of host DNA before assembling/aligning the remaining reads onto one or more reference sequences/templates.. The default parameters for this workflow have been optimized to take into account the short read lengths and presence of repetitive DNA sequences common to metagenomic and 16S rRNA data.
NGS-based	De novo	This workflow allows optional removal of host DNA. The default parameters for this workflow have been optimized to take into account the short read lengths and presence of repetitive DNA sequences common to metagenomic and 16S rRNA data.
	Reference-guided	This workflow allows optional removal of host DNA before assembling/aligning the remaining reads onto one or more reference sequences/templates.. The default parameters for this workflow have been optimized to take into account the short read lengths and presence of repetitive DNA sequences common to metagenomic and 16S rRNA data.

RNA-seq/transcriptomics workflows

The following table describes each of the workflows available in the **RNA-seq/transcriptomics** tab of the Workflow screen.

Group	Workflow	Description
Quantitative analysis	RNA-seq	Quantify RPKM gene expression and differential gene expression using DESeq2 and EdgeR statistics from Bioconductor. First-pass assembly is done using the usual XNG assembler. Second-pass assembly utilizes the QNG analysis module to determine expression level statistics. For each sample in a project, two new files for each contig/chromosome are put into the .assembly package. These contain the QNG calculated expression values for each gene and its isoforms: **-[contig number].genes-features** and **-[contig number].isoforms-features**. To learn how to use the output of an RNA-Seq *de novo* transcriptome assembly as input for the RNA-Seq reference-guided workflow, see Use RNA-Seq *de novo* transcriptome output as a reference. After performing an RNA-Seq reference-guided assembly, you can view the results in any of three applications: • **SeqMan Pro** – Use SeqMan Pro's 3-tabbed Feature Table for downstream analysis. Each tab (**All Features**, **Gene Features**, **CDS Features**) displays expression values in a column entitled "RPKM". If the sample is part of a replicate set, a second column entitled "RPKM – Replicate" displays the expression value for the feature determined from the replicate set. • **GenVision Pro** – Display a Sashimi plot for the assembly. Sashimi plots are designed to display data indicative of mRNA splicing, and are generated automatically during RNA-Seq assembly. See RNA-Seq reference-guided workflow output for a list of output files resulting from this type of assembly. • **ArrayStar** – Use ArrayStar's Gene and Isoform tables to filter for differentially-expressed genes of interest. ArrayStar tables can also display any DESeq2 or edgeR statistics included in the assembly.
	ChIP-seq	Choose from several different normalization and peak detection methods, including ERANGE and MACS.
	miRNA	To quantify miRNA gene expression and to discover new miRNAs.
De novo	De novo	Also called the "transcript annotation" (or "StarBlast") workflow. In the past, *de*

assembly	transcriptome	*novo* assembly of RNA-Seq data could result in thousands of contigs representing the expressed transcripts, without any context or labels. For Lasergene 13.0 and later, SeqMan NGen automatically attempts to group contigs from the same gene, and then name and annotate them based on the best match to a collection of annotated reference sequences. Two different SeqMan NGen assembly engines are used to optimize your results. Note that results from this workflow are non-quantitative. Result files for this workflow are described in detail in RNA-Seq *de novo* transcriptome workflow output.
	De novo miRNA	Allows you to *de novo* assemble miRNA data.

Include DESeq2 or edgeR statistics

To view statistics from DESeq2 or edgeR in ArrayStar, you first need to create one or more SeqMan NGen assembles in which one of these statistical packages was specified as the normalization method. That specification also triggers the use of the package for differential expression analysis.

1. In SeqMan NGen:

    a. Choose a reference-guided RNA-Seq workflow.

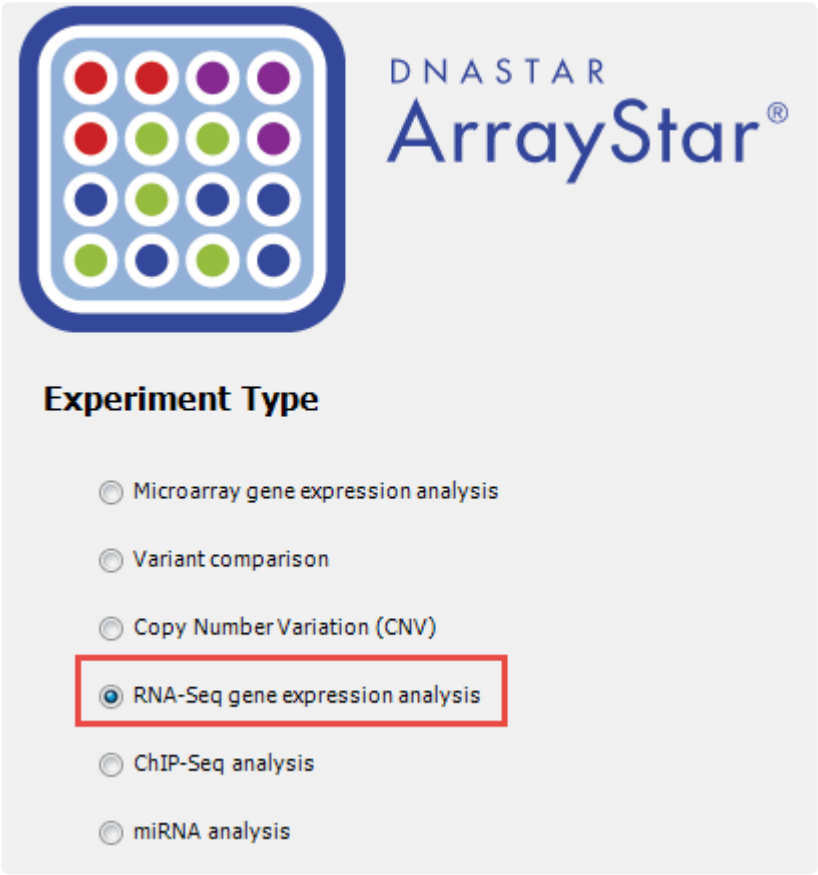
    b. In the Analysis Options screen, specify that **DESeq2** or **edgeR** be used as the **Normalization method**.

    c. Run the assembly.

    d. If you performed the assembly on a Windows machine and will be doing the ArrayStar analysis on the same machine, click the **Compare variants / differential gene expression between samples** button in the Assembly Summary screen. Then skip ahead to Step 3. Otherwise, continue to Step 2.

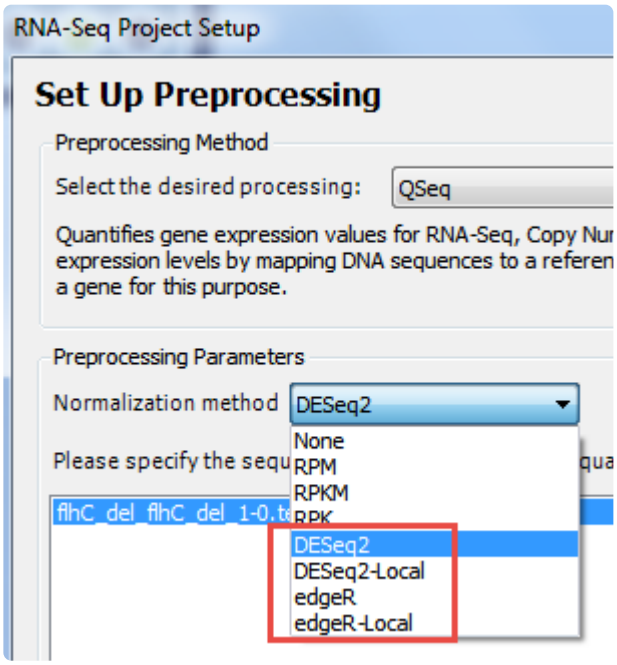
2. If you did not press the **Compare variants / differential gene expression between samples** button in SeqMan NGen, you can use an alternative method to open the assembly in ArrayStar:

    a. Launch ArrayStar and begin an RNA-Seq project.



b. Use ArrayStar's Project Setup wizard to import one or more *.assembly* packages created in Step 1, above.

c. In the Set Up Preprocessing page, choose **DESeq2**, or **DESeq2-Local**, **edgeR**, or **edgeR-Local**.



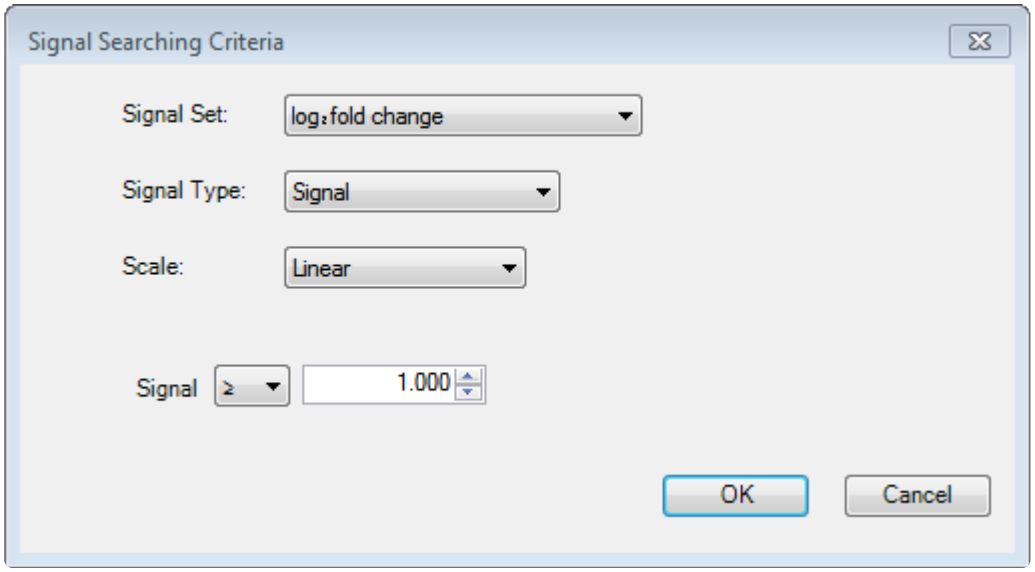
The Genomics Cloud-based options (without **"-Local"**) are suitable for most users. Windows users who have already installed R, Bioconductor and DESeq2 can elect to run locally by selecting the version ending in **-Local**.

3. Access DESeq2 or edgeR statistics in ArrayStar using either of these methods:

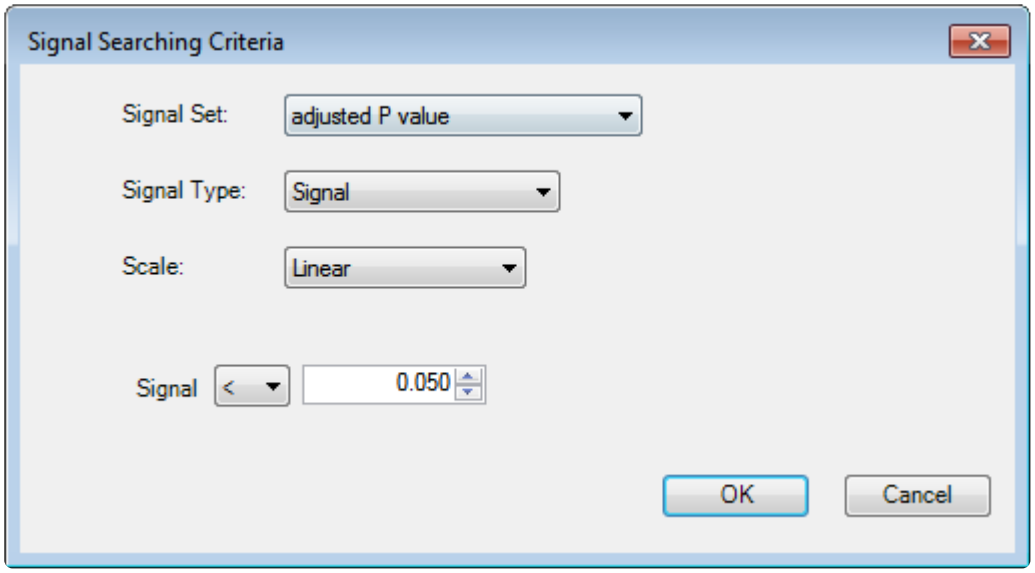
   - Open the Gene or Isoform tables and use the Add/Manage Columns tool to add DESeq2-related columns from the **Gene Values** or **Isoform Values** tabs.

   - Use **Filter > Filter All** to open the Advanced Filtering dialog. In the header, elect to search for **Genes** or **Isoforms**. Set up one or more filter rows. In each row, use the left-most drop-down menu to select **Signal Sample Group**, and use the middle section of each row to set up a comparison of interest. Finally, click the **Choose Signal Criteria** button in each row and choose the desired settings.

     For example, in the first row you might search for reads with a **$\log_2$ fold change** greater than or equal to **1**.



In the second row, you could search for items with an **adjusted P-value** less than **0.05**.



Use the **Search** button and then select all the filtered isoforms. Click on the **Remember Results as an Isoform Set** tool, choose a name and press **OK**. In the Set List, click the link **Show the** table of Genes containing this set's Isoforms. In the Gene table, use the Add Fold Change tool to add the **Fold Change** column.

4. (optional) To view or copy a Bioconductor script/log which you can use for QC/QA purposes or include in publications, open the Experiment List, select an experiment, and use the Show log links in the Info Pane on the right.

Variant analysis/resequencing workflows

The following table describes each of the workflows available in the **Variant analysis/resequencing** tab of the Workflow screen.

Group	Workflow	Description
ABI / Sanger	Whole genome	Assembles/aligns reads onto one or more reference sequences/templates and allows you to compare variations between multiple samples. This type of assembly can include billions of reads and large eukaryotic genomes. After assembly, compare results in ArrayStar using the SNP Report.
	Amplicon	Assembles a region of interest produced by PCR amplification.
	Clone verification	Align reads to confirm clone integrity and insert orientation. Note that a dedicated clone verification workflow exists in the SeqBuilder Pro application.
NGS-based	Whole genome	Assembles/aligns reads onto one or more reference sequences/templates. This type of assembly can include billions of reads and large eukaryotic genomes.
	Amplicon, gene panel, exome	Gene panels look at specific gene regions, usually those corresponding to known defects. Exome assembly saves assembly time and resources by specifically targeting only exons and coding regions, but do require you to have the corresponding *.bed* file from the capture kit. For instance, if you used Human Genome build 38 as the reference, for example, the corresponding *.bed* file might be called *Human genome build38.bed*. If using this workflow with cancer samples, check the box next to **Somatic/Cancer/Heterogeneous** in the Analysis Options screen. In most cases, downstream analysis of these finished assembly will place in ArrayStar.
	Viral-host integration detection	Used to locate putative viral insertion sites or to predict the location of other inserted sequences, such as transposable elements. When you select this workflow, SeqMan NGen automatically sets up a templated assembly that is optimized for locating viral insertion sites. To explore possible viral insertion sites post-assembly, launch SeqMan Ultra and use **Contig > Contig Coverage** to view tabular data for the individual contigs. During both reference-guided assembly steps, SeqMan NGen “masks” (trims) whichever half of the chimeric read does not match the template for that step. Use the Coverage view to navigate to positions with multiple reads, as evidenced in the depth column. The reads at these positions should be trimmed to the same base indicating the insertion site. You may “untrim” the reads to verify that they also contain viral sequence.

Variant calling accuracy workflow

While SeqMan NGen 17 no longer has a workflow named “Variant calling accuracy” (AKA “validation control accuracy,” “reference SNP accuracy”), you can still perform this workflow as follows:

1. In the Workflow screen, select the **Variant analysis/Resequencing tab** and choose the “NGS-based” **Amplicon, gene panel, exome** workflow.

2. In the Reference Sequence screen, add the relevant DNASTAR genome package; in most cases, this will be “human”. Check the **VCF file** and browse to a VCF file of the true variants. If you have a *.bed* file, you can also check the **BED file** box (optional) and browse to that file.

3. In the Input Sequences screen, add read data for the Validation Control.

4. In Assembly Options, check the box next to the desired **Variant detection mode**.

5. Follow the rest of the wizard screens and initiate the assembly.

6. Variant calling accuracy is evaluated within ArrayStar. Once assembly is complete, see the ArrayStar help topic Validation Control Accuracy for further instructions.

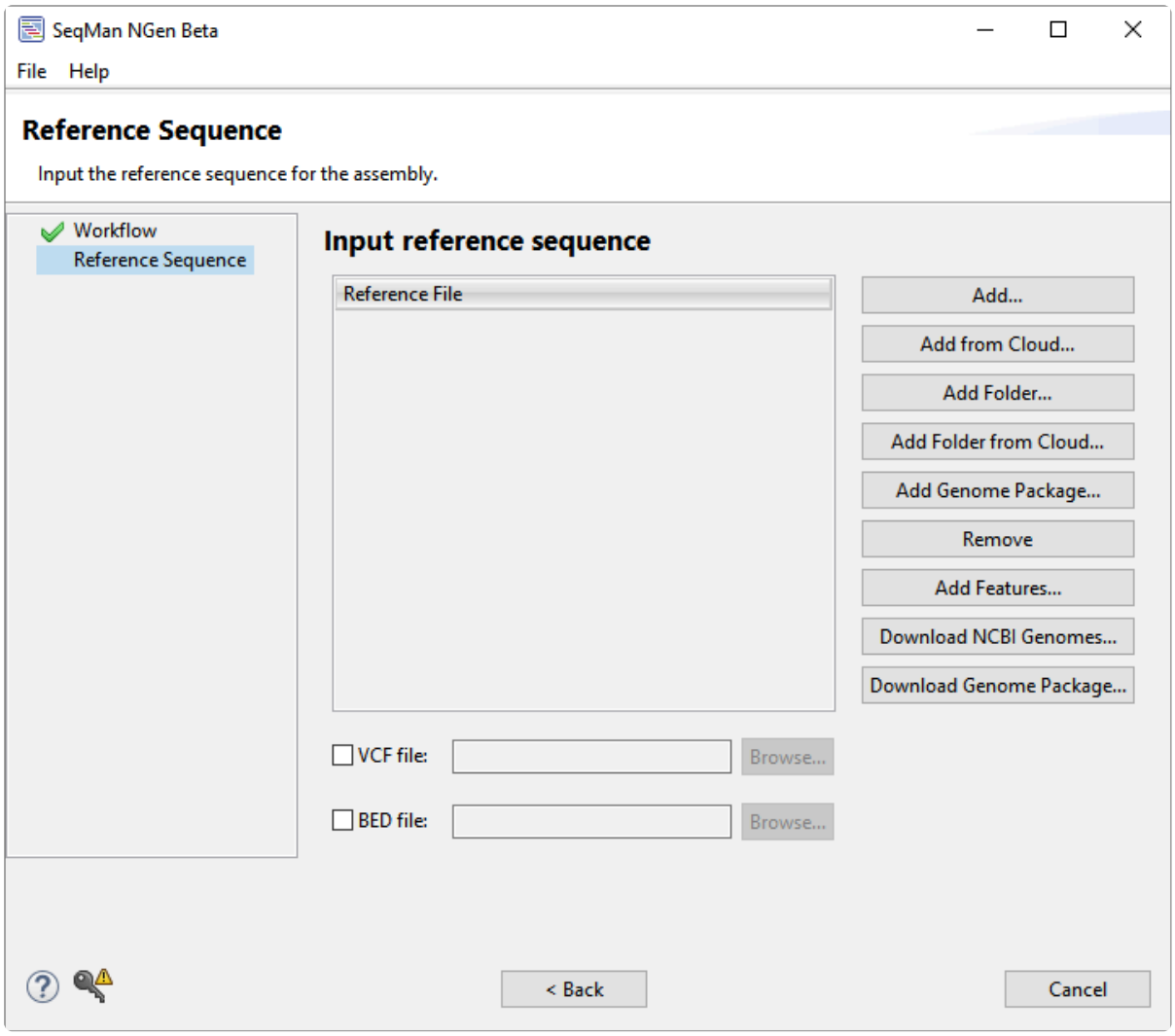
Variant Call Format (VCF) files workflows

The following table describes each of the workflows available in the **Variant Call Format (VCF) files** tab of the Workflow screen. These workflows are used if you have assembled your data and done SNP calling in another application and have VCF files with a *.vcf* or *.abi* file extension that you wish to compare to a database or to one another. These workflows both output an *.astar* file that can be opened in ArrayStar for downstream analysis.

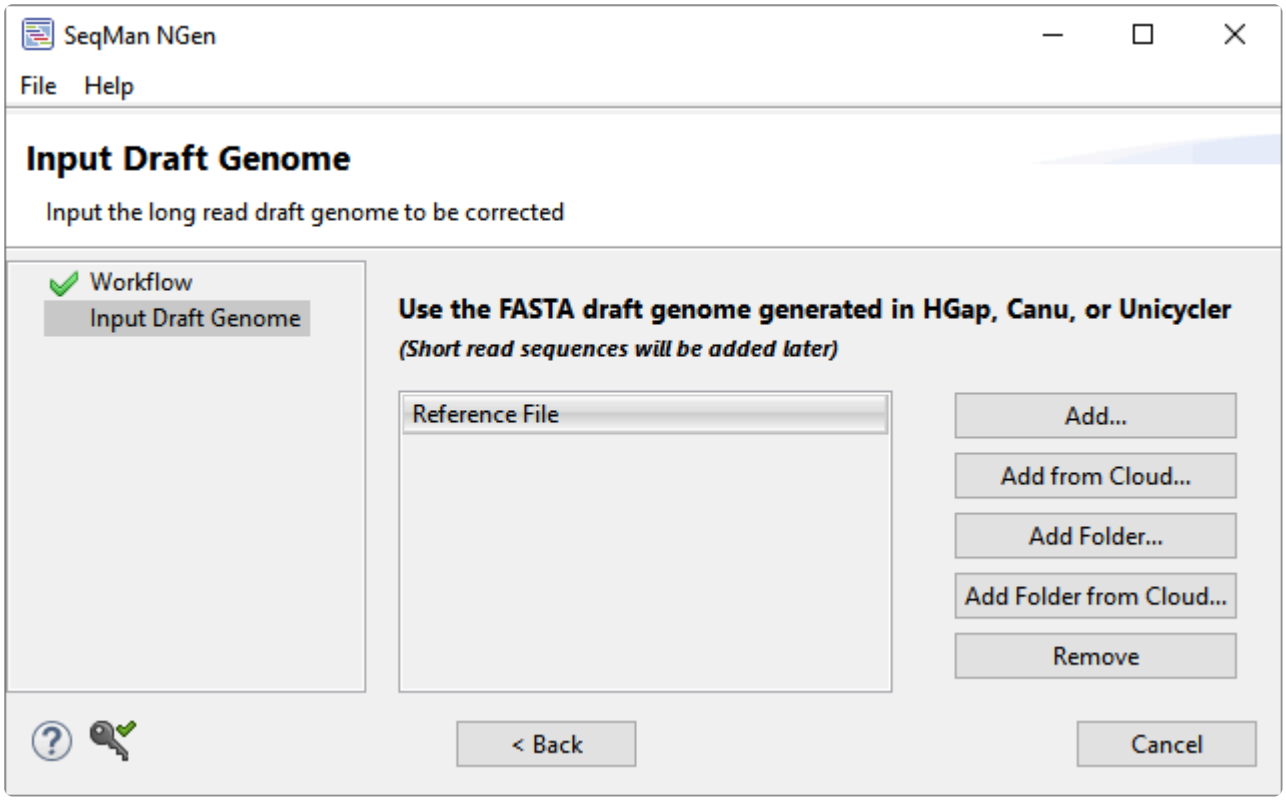
Group	Workflow	Description
VCF file analysis	Functional annotation of a single sample	Annotates the variant positions with functional information from a database, including affected genes and impact on protein encoding regions and/or splice sites.
	Annotation and comparison of multiple samples	Allows multiple samples in VCF format to be annotated and then compared to identify genes and/or variants of interest in ArrayStar. This workflow is designed to use with assemblies created outside SeqMan NGen (e.g., using BWA + GATK). Such assemblies often have *.vcf* files as their only output.

Reference Sequence / Input Draft Genome

If the Reference Sequence screen appears, you must input some type of reference sequence or genome template package before proceeding further in the wizard.



A modified version of the screen is called Input Draft Genome and appears only in the “Short read polishing of a long read draft genome” workflow.



Optional pre-import steps for the Reference Sequence screen:

- Annotate reference sequences prior to import
- Manually specify an isoform prior to import
- Make a custom VCF file
- Make a custom BED file
- Troubleshoot a Manifest file

Add and remove reference sequence files or draft genomes:

SeqMan NGen can read and produce output using a variety of common chromosome naming conventions, including “chr1” and “ch1,” as well as Arabic and Roman numerals. See our Supported File Types page for allowable file extensions.

- Add a reference sequence or draft genome from your computer or the Cloud
- Add a genome template from DNASTAR – (RECOMMENDED) Genome template packages include dbSNP information that is very useful for downstream analysis.
- Add a genome template from NCBI
- Use RNA-Seq de novo transcriptome output as a reference
- Remove a sequence from the list

Options pertaining only to the Reference Sequence screen:

- Specify a VCF, BED or Manifest file

Click **Next >** to proceed to the next wizard screen or **< Back** to return to the previous screen.

Annotate reference sequences prior to import

Using annotated reference sequences in SeqMan NGen may enable you to better analyze the identified putative SNPs when viewing your assembled project in SeqMan Pro or SeqMan Ultra. If desired, annotate your reference sequence in SeqBuilder Pro (the Lasergene application for sequence editing and visualization) prior to adding it to the Reference Sequence dialog.

1. Launch SeqBuilder Pro.
2. Go to **File > Open** and select the reference sequence.
3. Select the range of sequence where a feature will be added. (Use **Edit > Go to Position** to navigate quickly up and down your sequence.)
4. Go to **Features > New Feature.** A new “misc_feature” will be added to your sequence and displayed in the Feature List.
5. Click on “misc_feature” from within the Feature List and select the appropriate feature type from the list provided*.* For example:
    - For SNPs, choose **Variation > variation**.
    - For exons, choose **Gene > exon**.
    - For CDS features, choose **Transcript > CDS**.
    - For origin of replication, choose **Structure > rep_origin**.

    Note: The next feature you create will automatically be of the same feature type you just selected, enabling you to create all the features of one type more quickly.
6. Repeat steps 3-5 until all of your features have been added. Then go to **File > Save As** and save your sequence in *.sbd*, *.seq* or *.gbk* format. Your annotated reference sequence is now ready for assembly in SeqMan NGen and subsequent analysis in SeqMan Pro.

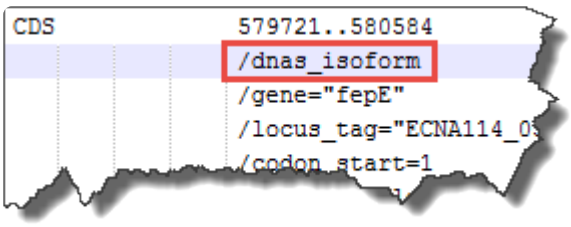
Manually specify an isoform prior to import

By default, SeqMan NGen chooses the longest CDS as the isoform for SNP calling. If desired, you may override the automated choice by specifying the preferred isoform manually in the reference sequence.

To do so, follow these steps prior to importing the reference sequence into SeqMan NGen's Reference Sequence screen:

1. Open the reference sequence in a text editor.

2. Locate the feature of interest. Just below its location coordinates, type in **/dnas_isoform**.



3. Save the edited template sequence.

4. Input the saved version of the template sequence into SeqMan NGen's Reference Sequence screen.

Use RNA-Seq de novo transcriptome output as a reference

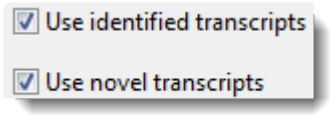
You may use the contigs output from an RNA-seq de novo workflow as reference sequences in the templated RNA-Seq workflow. Doing this may allow you to quantify the relative abundances of transcripts using ArrayStar. Note that this use case assumes the same reads are used for both rounds of assembly.

1. Follow the RNA-Seq de novo transcriptome workflow. After assembly is complete, close the SeqMan NGen wizard.

2. Launch SeqMan NGen again and follow the RNA-seq templated workflow.

3. In the Reference Sequence screen do one of the following:

    - Press **Add Folder**, navigate to the Transcriptome package that was output in Step 1, and press **OK**. Then choose which types of transcripts to input by checking one or both boxes at the bottom of the screen.

    - If you exported a subset of transcripts from SeqMan Pro and wish to use those, rather than the full Transcriptome package, press **Add** and add a single *.fas* file.

    Note: SeqMan NGen does not support the addition of more than one *.fas* file from the *.Transcriptome* folder.

4. In the Input Sequences, load the desired reads. These can be the same reads that were used as input in the original transcript annotation workflow.

5. Set other options, as desired, and run the assembly.

Specify a VCF, BED or Manifest file

Certain workflows allow or require you to import a BED file, Manifest file, or a custom VCF SNP file with data from one or more assemblies. These options, if available, will be offered at the bottom of the Reference Sequence screen.

To add a BED, Manifest, or targeted regions file:

These files can be used in many types of assemblies, and should always be uploaded when doing exome assemblies.

To add a file of these types, check the box next to **BED file** and then use the corresponding **Browse** button to navigate to the file. If you are doing a local assembly, select the file and click **Open**. If you are doing a Cloud assembly, select the file and click the green check mark (). Note that:

- You must upload the *.bed* file from the capture kit, not any random *.bed* file. For instance, if you used Human Genome build 38 as the reference, for example, the corresponding *.bed* file might be called *Human genome build38.bed*. Sequencing services do not always send the *.bed* file to customers, but can often provide it on request.
- BED files must have the extension *.bed*. For information on making a BED file, see Make a custom BED file.
- Manifest files are typically used to represent coordinates of regions that were captured in procedures, such as exon capture performed prior to sequencing. They can have various extensions (usually *.txt*), but must be in the correct format.

To add a VCF file:

Most users will not have a VCF file to upload, but are looking forward to producing a VCF file as part of the assembly output. However, researchers working with standard reference data sets may have a VCF file to upload to the wizard at this stage.

To upload a VCF file, check the box next to **VCF file** and then use the corresponding **Browse** button to navigate to the file. If you are doing a local assembly, select the file and click **Open**. If you are doing a Cloud assembly, select the file and click the green check mark (). Positions within the VCF file will be given a VCF SNP ID during the assembly process. After assembly, information about each position can be viewed in the SeqMan Pro SNP Report (**SNP > SNP Report**) or the SeqMan Ultra Variants view. SeqMan NGen only supports one VCF file per assembly project. If you have multiple VCF files (e.g., one per chromosome), you must merge the information into a single VCF file before browsing to the file. For more information, see Make a custom VCF file.

Make a custom VCF file

Variant Call Format (VCF) files have multiple uses. For instance, they can provide a way to flag previously known SNPs and to filter them in SNP tables. In DNASTAR's SeqMan NGen, these SNPs are called "annotated SNPs"; in ArrayStar, they are referred to as "user variants." VCF files can also be used to keep track of previously identified variants so that they can be verified in a new assembly or experiment.

The following brief video is an overview of how VCF files can be used in an assembly and in downstream analysis:

VCF files can be custom-made or automatically generated by sequencing software. For instance, you can create a VCF file using software such as SeqMan Pro, SeqMan Ultra or ArrayStar. Certain SeqMan NGen assemblies also output a VCF file called [assembly_name}.sample.vcf. VCF files are also available from other sources, such as the UCSC Genome Browser, and the Genome In a Bottle project. For a description of various VCF version specifications, see the Sourceforge VCF Specification page.

These two columns are REQUIRED, and must be in the order shown. All cells in these columns must be filled.		**These four columns are OPTIONAL. If optional columns are present, the assembler will check the length of the string and compare against the length of the called variant. The base identities will not be checked.**				**Columns 7 and beyond are allowed, but will be ignored.**
#CHROM	**POS**	**ID**	**REF**	**ALT**	**INFO**	**(Misc.)**
Chromosome identifier. Numbers are preferred, but **chr** or **ch** prefixes	Position in the reference sequence.	For known dbSNP entries, the rs ID. The valid format is rs followed by a series of	The reference base(s). For	The variant base(s). For	User ID and source assembly information.	These columns may contain data, but they will be

are allowed. All cells in this column must be filled.	All cells in this column must be filled.	digits. For unknown or nonexistent IDs, a period (.)	unknown bases, a period (.)	unknown bases, a period (.)	For unknown bases, a period (.)	ignored by the SeqMan NGen assembler.

- The table portion of the file must be sorted numerically, first by #CHROM, and then by POS. Make sure to sort the columns numerically (1, 2, 3…) and not alphabetically (1, 11, 12…). If you attempt to run the assembly after loading an improperly-sorted VCF file, multiple red error messages will be displayed during the assembly.

- When you try to open extremely large VCF files in a spreadsheet program or text editor for sorting purposes, you may receive an “insufficient memory” warning. If you need to sort a VCF file that is too big to open on your machine, we recommend using Sourceforge’s VCFTools.

- If quotation marks are used anywhere in the VCF file, they must be straight quotes, not curly or “smart” quotes. In addition, quotation marks should not be used in lines beginning with ##contig, ##UnifiedGenotyper, or ##INFO. If these rules are not followed, an error message will appear during assembly stating that “the VCF file has an incorrect or missing header.” Though the assembly will continue, the VCF SNP file that is output will be empty.

- Chromosome names are captured from genome template packages and used to assign contig IDs to entries from BED, VCF and manifest files. SeqMan NGen can read and produce output using common naming conventions (i.e., “chr” and “ch”) and Arabic numerals. It understands that chr1, ch1, or 1 can all be used to represent “the first template in the index,” and so on. In addition, Genome Template Packages sometimes internally define “short names” for particular chromosomes. For example, the *C. elegans* template package names its chromosomes using the standard convention for that organism: “I”, “II”, “III”, “IV”, “V”, “X”, “M.” SeqMan NGen does not normally recognize Roman numerals, but can in this case, because the numbers are “short names” that have been mapped to specific chromosomes.

Make a custom BED file

Certain workflows allow or require you to import a targeted regions file, also known as a BED file, within the Reference Sequence screen.

BED files are used to define capture regions in the assembly, and can be generated by the sequence provider or made by hand. These files are basically tab-separated text files whose extension has been changed to *.bed*. See the UCSC Genome Bioinformatics BED file page for detailed information.

The following brief video shows how BED files can be incorporated into an assembly for later downstream analysis:

SeqMan NGen can read and produce output using a variety of common chromosome naming conventions, including “chr1” and “ch1,” as well as Arabic and Roman numerals. Chromosome names are captured from genome template packages and used to assign contig IDs to entries from BED, VCF and manifest files.

The BED file can consist of multiple sections, each with a different track name. Text is allowed between the tables without restriction.

These three columns are REQUIRED, and must be in the order shown. All cells in these columns must be filled.			**Columns 4 and beyond are allowed, but will be ignored.**
chrom	**chromStart**	**chromEnd**	**(Misc.)**
The name of the	Starting position for the feature. The coordinates for	Ending	Data in these

chromosome or scaffold. Numbers are preferred, but **chr** or **ch** prefixes are allowed.	BED intervals are in 0-based coordinates as follows: (1-0) .. (100+1-1). Therefore, base 1-100 of a chromosome is represented in a BED file as 0-100.	position for the feature.	columns are ignored.

- A header row is optional and can contain any text; text need not match that shown in the table header row above.

- **IMPORTANT**: Each table in the file must be primarily sorted by the first column, and secondarily sorted by the second column. The columns must be sorted numerically (1, 2, 3…) and not alphabetically (1, 11, 12…). If only chromosome 1 (and possibly 11) appears in SeqMan Pro's "Coverage of Targeted Regions" report (**Project > Show coverage of target regions**), this is indicative of incorrect sorting.

Troubleshoot a Manifest file

In certain workflows, you may be able to specify a Manifest file in the Reference Sequence screen.

Manifest files are tab-separated files used to define the chromosomal coordinates of gene targets in the assembly, and are normally generated automatically , e.g., by Illumina. (See Illumina's manifest file PDF for a description.) Manifest files can have various file extensions, though *.txt* is commonly used.

SeqMan NGen can read and produce output using a variety of common chromosome naming conventions, including "chr1" and "ch1," as well as Arabic and Roman numerals. Chromosome names are captured from genome template packages and used to assign contig IDs to entries from BED, VCF and manifest files.

The following examples show the most basic required columns for the manifest file, as well as two formats that are used by Illumina. These examples are provided in case you need to trouble-shoot problems with existing manifest files. If you want to create your own targeted regions file, we recommend making a BED file rather than a manifest file. See Make a custom BED file for detailed instructions.

The columns below can appear in any order. Columns with **blue** headers are required, while columns with **orange** headers are optional.

User-made file (most basic version):

[Regions]			
Name	**Chromosome**	**Start**	**End**
28324371	chrM	577	647

Illumina manifest file – format 1:

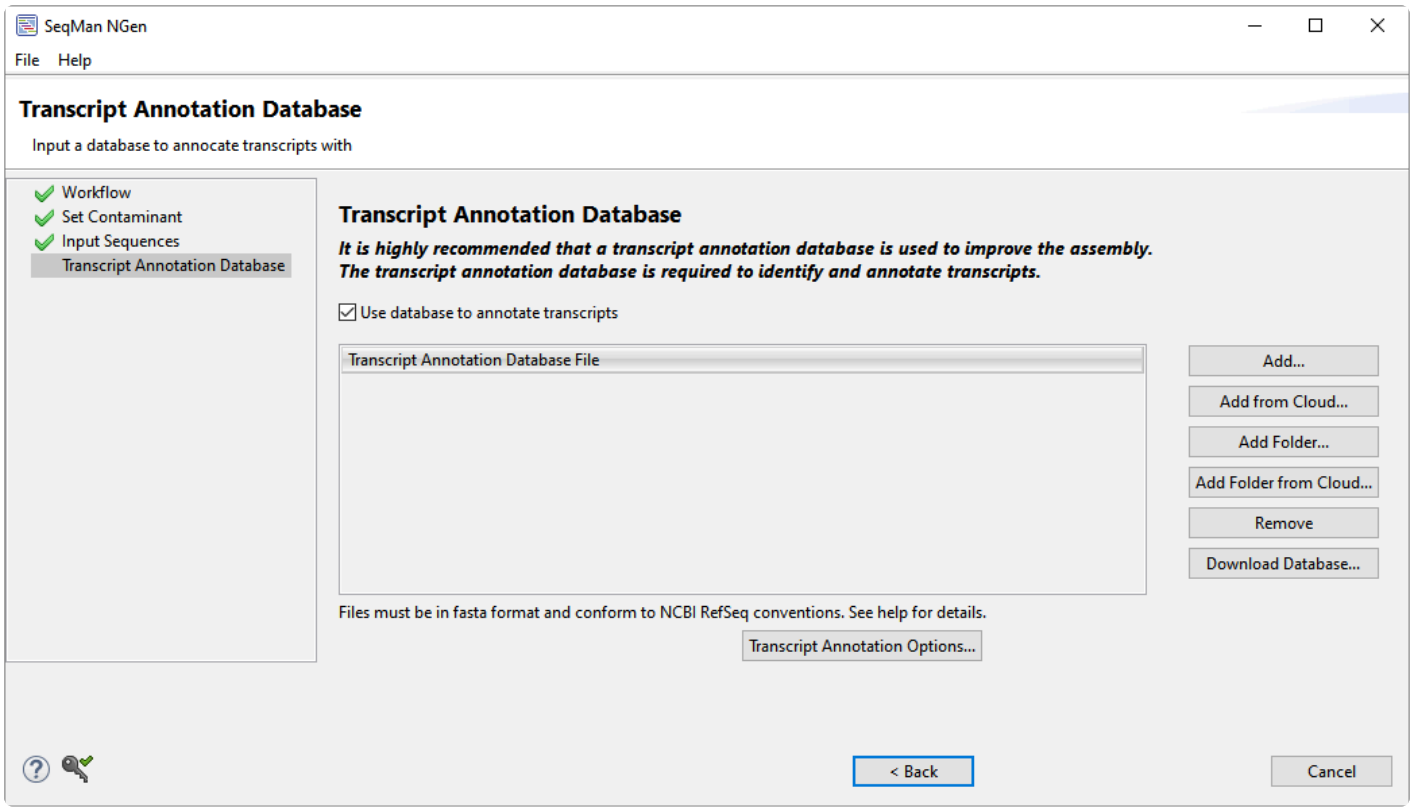
[Targets]						
TargetA	**TargetB**	**Target Number**	**Chromosome**	**Start Position**	**End Position**	**Probe Strand**
chr1.43815008.43815009	chr1.43815008.43815009	1	chr1	43814982	43815163	+
chr1.115256528.115256531	chr1.115256528.115256531	1	chr1	115256500	115256680	+

Illumina manifest file – format 2:

[Regions]					
Name	**Chromosome**	**Start**	**End**	**Upstream Probe Length**	**Downstream Probe Length**
28324371	chrM	577	647	0	0

Transcript Annotation Database

Selecting the RNA-seq/Transcriptome workflow **De novo transcriptome** on the Workflow screen, causes the wizard to include the Transcript Annotation Database screen.



We highly recommend that you use a transcript annotation database when doing this workflow.

If you want to annotate transcripts during assembly using information from a specified database, leave **Use database to annotate transcripts** checked . You will need to specify a transcript annotation database, as described below. If you do **not** want to annotate transcripts during assembly, uncheck the box. If you do this, note that all other buttons in the dialog will be disabled.

To add or remove transcript annotation databases:

If you check the box as described above, you must add a transcript annotation database using any of the methods below.

- Add a DNASTAR transcript package
- Create a custom transcript annotation database and add it to the list. Any files added added must conform to strict formatting specifications and must be in *.fasta* format.
- Use a local copy of RefSeq as a database
- Remove a sequence from the list

To access and edit advanced options:

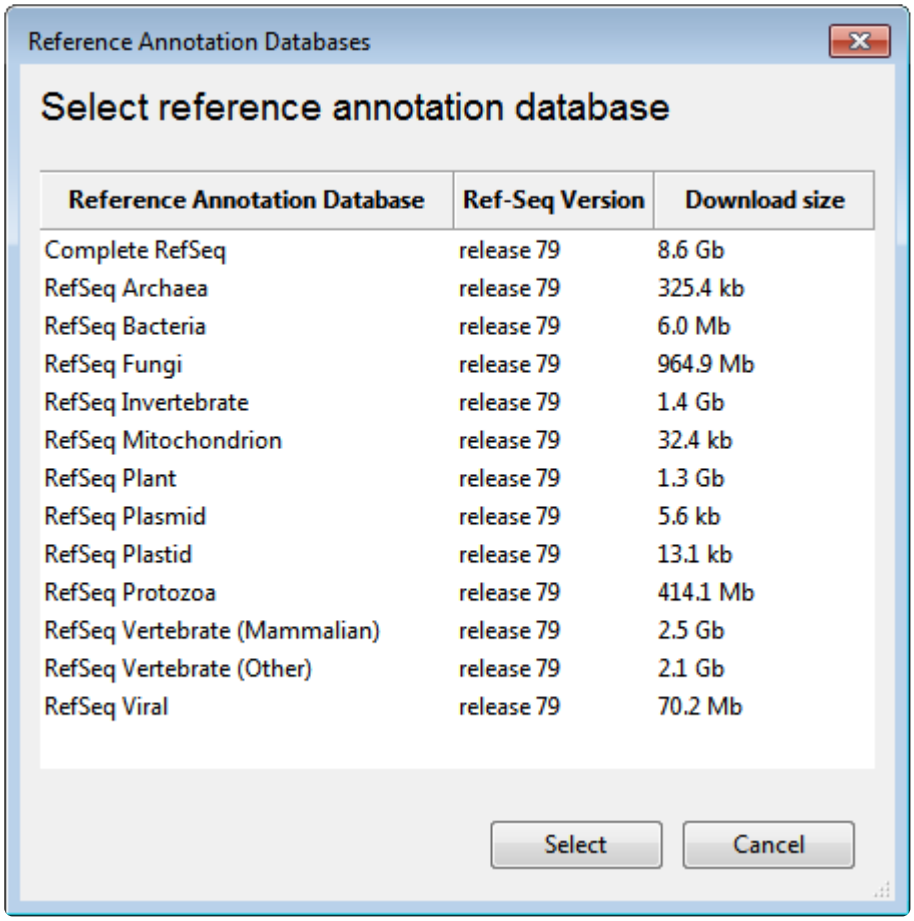
Press the **Transcript Annotation Options** button to launch the Annotation Options dialog.

Click **Next >** to proceed to the next wizard screen or **< Back** to return to the previous screen.

Add a DNASTAR transcriptome package

The Transcript Annotation Database screen provides the option for licensed users to add a DNASTAR database of transcript annotations extracted from data on NCBI's RefSeq website.

To do this, click the **Download Database** button, select one or more items from the list and then press **Select**.



Create a custom transcript annotation database

The Transcript Annotation Database screen allows you upload a *.fasta_-formatted database for use in _de novo* RNA-seq workflows.

A custom database must meet the same formatting specifications as NCBI RefSeq files. They must:

- Be in *.fasta* format (either single or multi-sequence files are supported)
- Use the field delimiter '|' (without quotes) between fields
- Have a header line for each entry, written in the format:

  ref | [Accession] | [Organism Name] [Description] ([Gene Name])

  … where:

  - **Accession** – All characters between third and fourth field delimiters
  - **Organism Name** – The first two words after fourth field delimiter
  - **Description** – All words after **Organism Name** up to the end of the line, or up to a comma or parentheses, if the gene name exists
  - **Gene Name** – All characters in parentheses after **Description**

Example:

ref | XM _ 005842486.1 | *Chlorella variabilis* hypothetical protein (CHLNCDRAFT_144668) mRNA, partial cds

Use a local copy of RefSeq as a transcript annotation database

To use a local copy of RefSeq as a transcript annotation database:

1. Download the latest version of the RefSeq package from this NCBI ftp site.

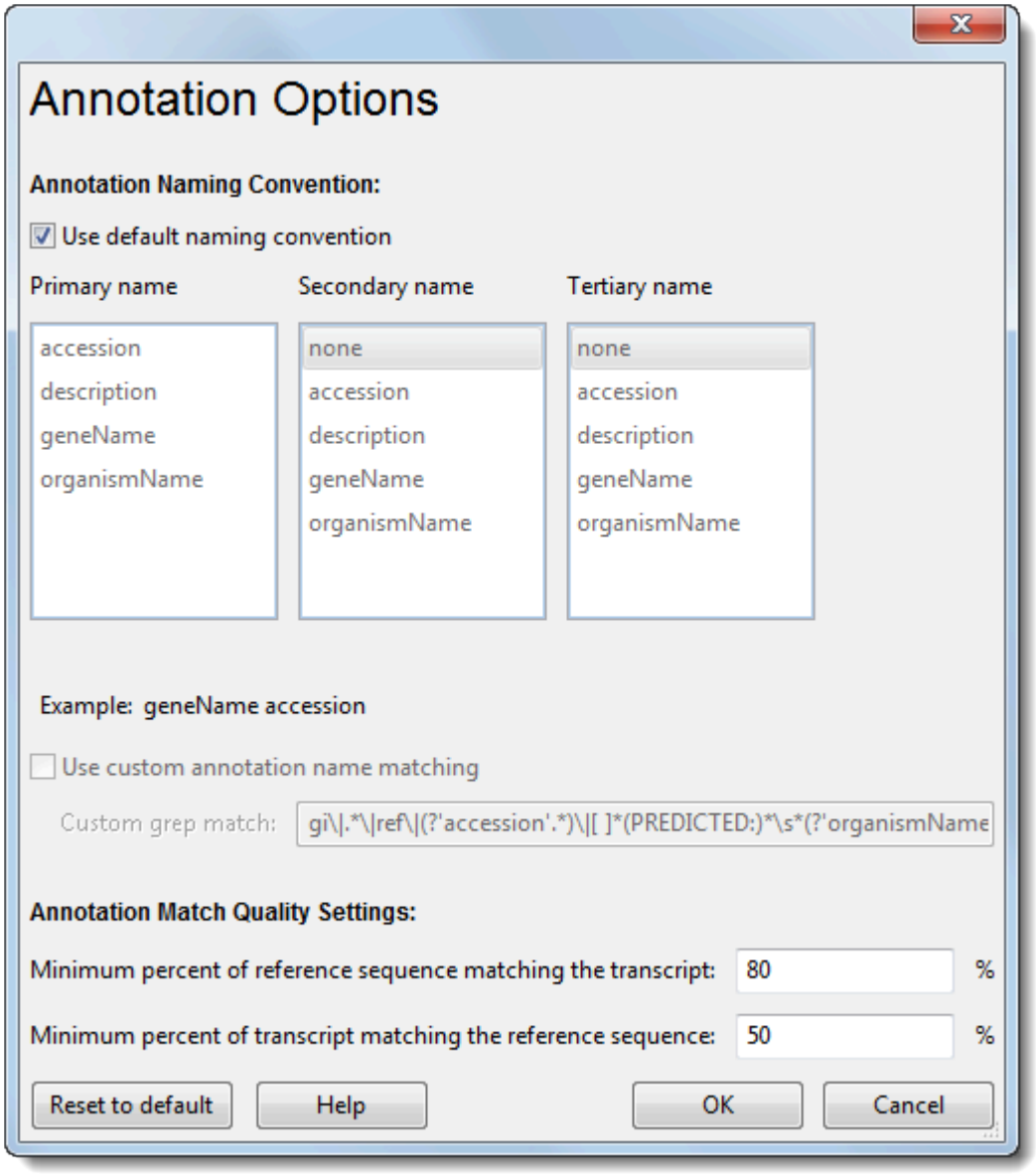
2. Choose a de novo transcriptome RNA-seq workflow.

3. On the Transcript Annotation Database screen, select **Add** or **Add Folder** and select the file or folder for the RefSeq package that you downloaded in Step 1. Ignore the **Transcript Annotation Options** button on this screen. Since you are still using a RefSeq package, the default grep string and naming convention will be used automatically.

4. Finish setting up and running the assembly as usual for this workflow type.

Annotation Options dialog

To open the Annotation Options dialog, press the **Transcript Annotation Options** button in the Transcript Annotation Database screen. The Annotation Options dialog can be used to change the default naming convention used for RefSeq packages, or to specify a custom GREP and naming convention for non-RefSeq packages.

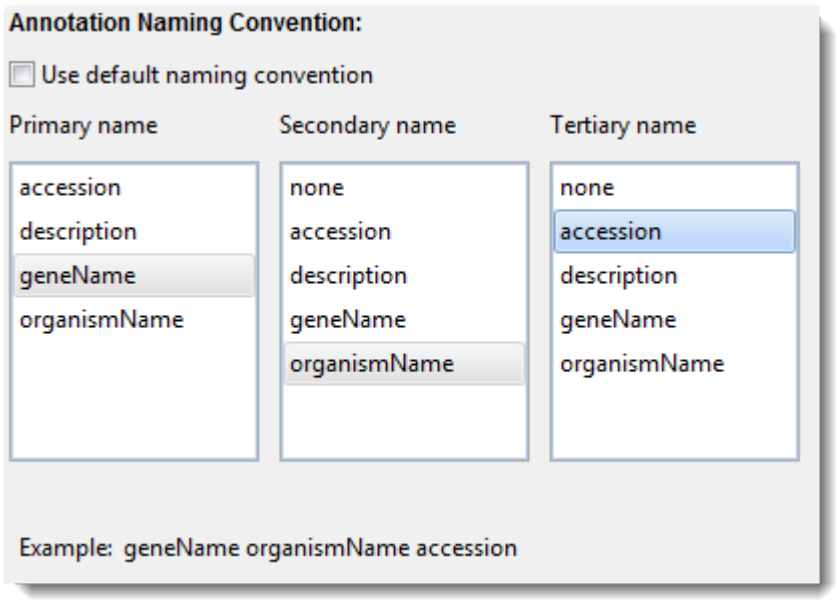


Annotation Naming Convention:

Check the **Use default naming convention** box if you want to keep or to return to the default naming convention (geneName accession). Uncheck the box to enable dialog options allowing you to customize the naming convention.

If you uncheck the **Use default naming convention** box, the dialog provides two ways to customize names.

- Manual selection of naming components – Using the **Primary**, **Secondary** and **Tertiary** name boxes, you can select up to three “compound name” components in any order. Click in each box you want to use to enable it, then make a selection from the list: **accession**, **description**, **geneName**, **organismName** or **none**. The Example below the boxes shows your current selection(s).



  If you made the selections as shown in the preceding image, for example, then a sample *E. coli* transcript would automatically be assigned a name like: *thrB - Escherichia coli str. K-12 substr. DH10B - 6058639*

- Annotation name matching using grep syntax – If you uploaded a transcript annotation database, it automatically describes a GREP string that uses the FASTA headers to define rules for the naming convention. The same string will define the “default naming convention”. If you want to edit this GREP string, thus changing the rule regarding the extraction of contig name fragments from the FASTA headers, check the **Use custom annotation name matching** check box. After checking the box, you may edit the regular expression/GREP by typing in the **Custom grep match** textbox. An example is provided in the text box: gi\|.*\|ref\|(?‘accession’.*)\|[ ]*(PREDICTED:)*\s*(?‘organismName’\w* \w*)\s*((?‘description’.*)([^(]*)(\((?‘geneName’\w*)\)[^(]*,.*)|((?‘description’.*),.*)|(?‘description’.*))

Annotation Match Quality Settings:

This section of the dialog supplies two metrics for specifying what constitutes a valid match between the reference sequence (the “query”) and the database entry.

- **Minimum percent of reference sequence matching the transcript** – Enter the minimum percentage of the query that must match the database entry. The default is 80%.

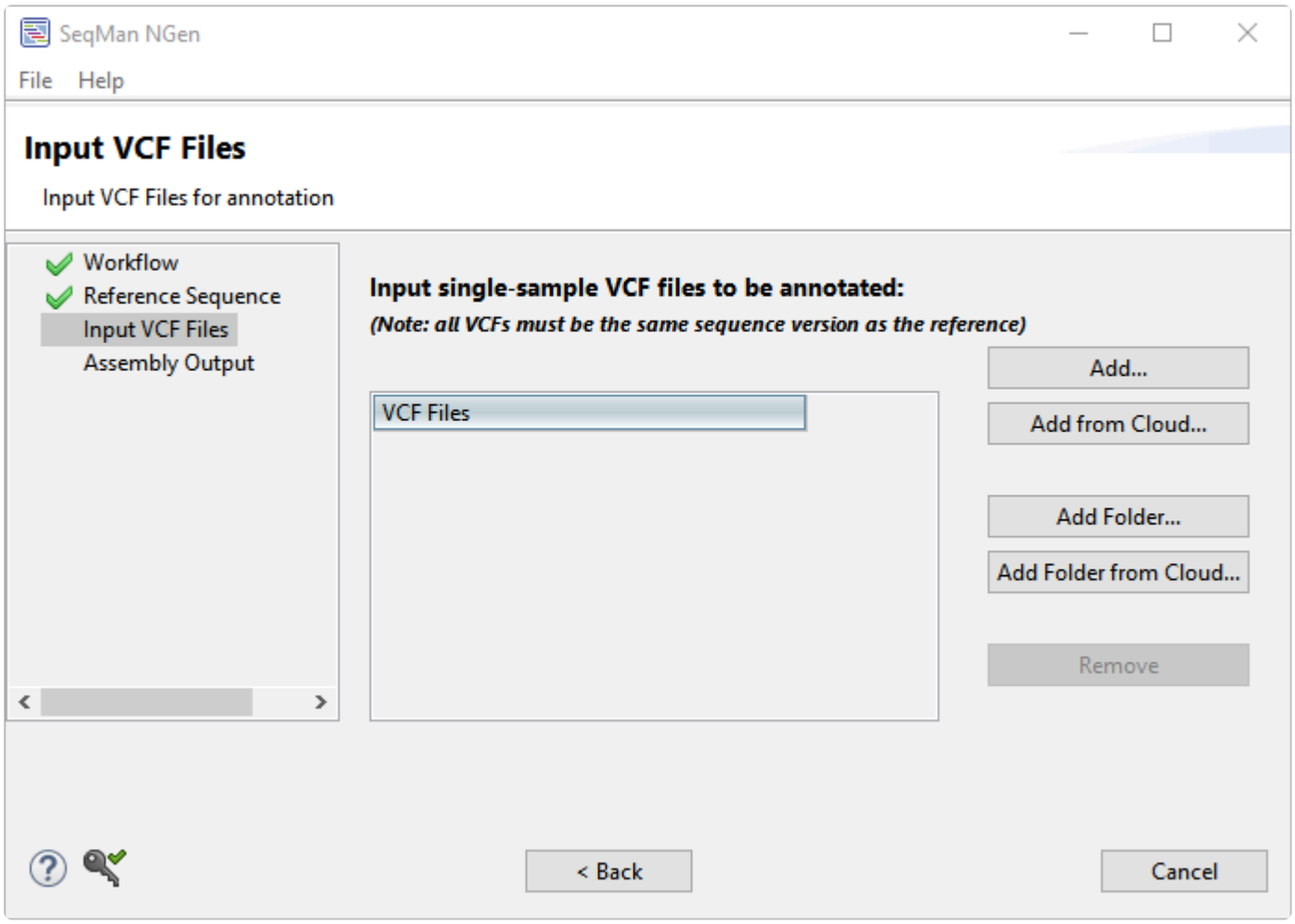
- **Minimum percent of transcript matching the reference sequence** - Enter the minimum percentage of the database entry that must match the query. The default is 50%.

****

To save the current selections and close the dialog, click **Save**. To return to the default settings, press **Reset to Default**. To close the dialog without saving changes, click **Cancel**.

Input VCF Files

In certain workflows, the Input VCF Files screen prompts you to input the VCF files to be annotated. All VCF files must be the same sequence version as the reference sequence added in the Reference Sequence screen.



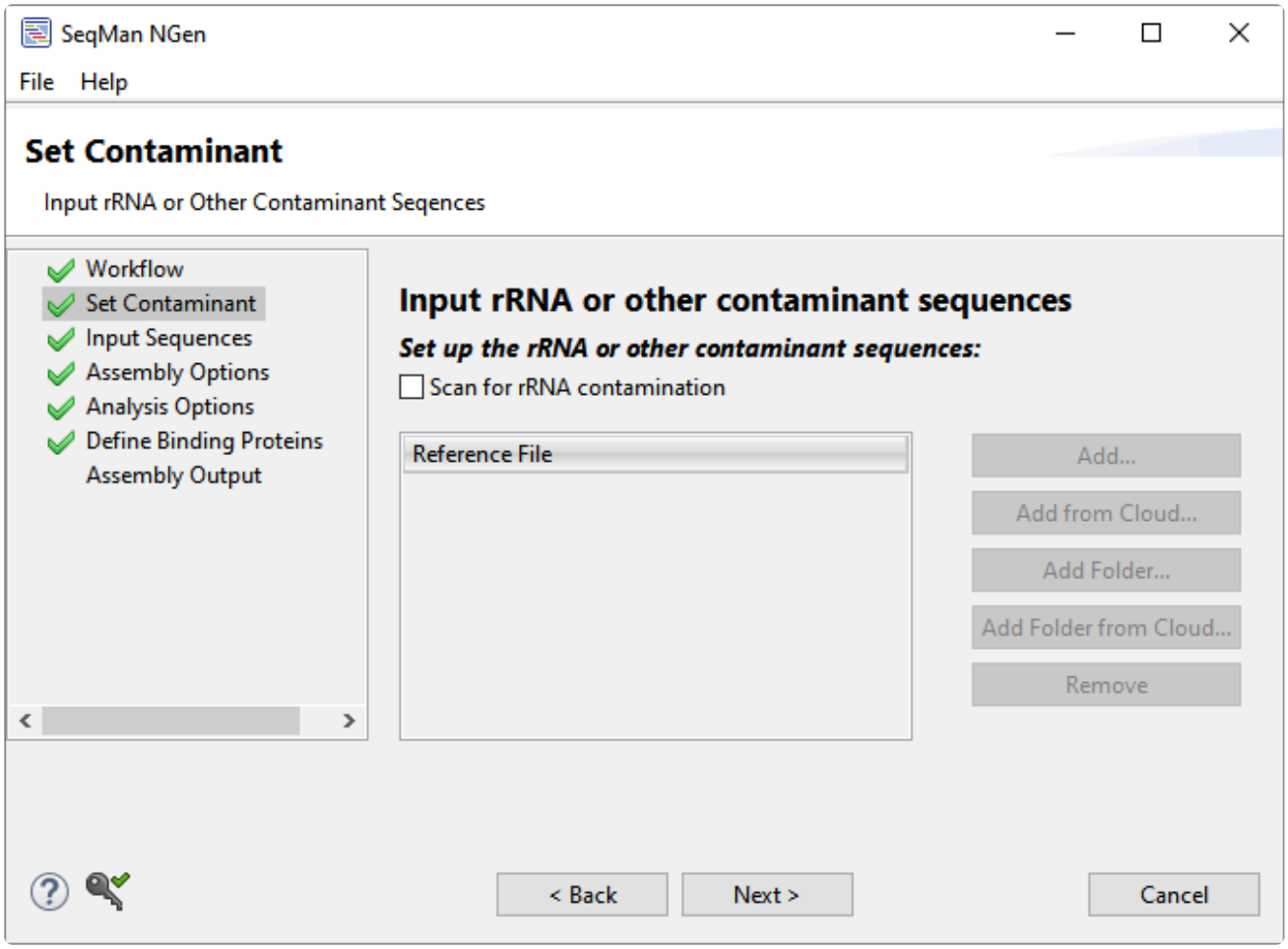
See the following topics to learn how to:

- Add sequences from your computer or the Cloud
- Remove a sequence from the list
- Make a custom VCF file

Click **Next >** to proceed to the next wizard screen or **< Back** to return to the previous screen.

Set Contaminant

If you are following certain RNA-Seq workflows, the wizard includes the Set Contaminant screen. The screen prompts you to remove rRNA sequences if they are present. We highly encourage doing so, as these sequences can represent up to 80% of the reads in a de novo transcriptome data set and can swamp the assembly due to their large numbers.



Check **Scan for rRNA contamination** to enable the three add/remove sequence buttons on the right. You will then need to specify rRNA sequences here, as described below. Typically, you will add reference sequence(s) from a 16S rRNA database (e.g., Silva, Greengenes Ribosomal Database Project, etc.

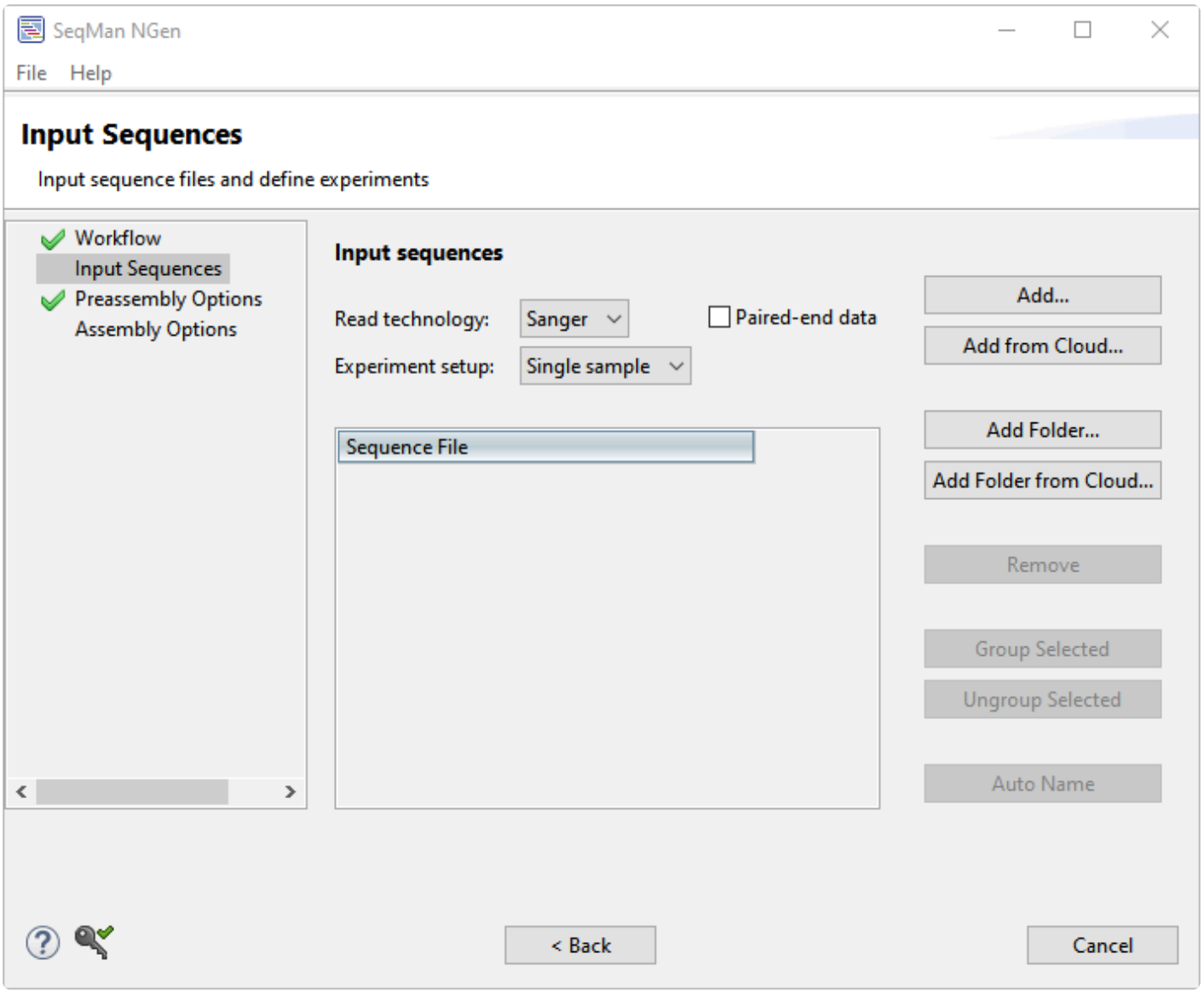
See the following topics to learn how to:

- Add rRNA sequence files or folders of files from your computer or the Cloud
- Remove a biome genome from the list

Click **Next >** to proceed to the next wizard screen or **< Back** to return to the previous screen.

Input Sequences

If the Input Sequences screen appears, you must specify the required options and enter one or more read files in this dialog before proceeding to the next screen. If you are following a long-read workflow, this screen will direct you to “Input long read sequences.”



The upper part of the dialog consists of a drop-down menu for specifying the read technology of your sequence files. Other options in this dialog will vary depending on the workflow and the selection made in the **Read technology** drop-down menu.

See the following topics to learn how to specify options and add data in this screen:

Pre-import steps (optional):

- Check paired-end data naming specifications

Specify data options:

- Specify read technology
- Specify paired-end data
- Specify RNA-Seq options
- Specify single sample, multi-sample or replicate data

Add and remove sequences:

- Add sequences from your computer or the Cloud
- Remove a sequence from the list

Click **Next >** to proceed to the next wizard screen or **< Back** to return to the previous screen.

Specify read technology

To specify read technology in the Input Sequences screen, make a selection from the **Read technology** drop-down menu. Default values for parameters and other assembly options in subsequent panels will be based on this selection.

Considerations when choosing a read technology:

- If you are following a long-read workflow, the only options are **PacBio** (Pacific Biosciences) or **ONT** (Oxford Nanopore Technologies).

- If you choose **Illumina**, SeqMan NGen assumes that you have paired-end data > 50 bp in length, and with a 500 bp insert distance. For all other technologies, SeqMan NGen presumes single-end data. If the read length is shorter than 50 bp, you may wish to specify a shorter **Mer size** in the Assembly Options screen. When using very short reads, you may also consider optimizing the **Minimum aligned length** and **Maximum gap size** in the Alignment tab.

- For de novo workflows, if you select **Illumina** and enter an insert size of 150 bp or less in the **Set Pair Information** dialog, the assembler will assume the reads overlap and will attempt to create a single “super-read” from each pair. Read pairs that cannot be merged, either because they do not overlap or have numerous errors in the overlapping region, will not be included in the assembly. See Remove PhiX control reads from Illumina data prior to import for a description of how to use **Contaminant scan** to remove PhiX174 control sequence from Illumina data prior to assembly.

- Because the technology does not support paired reads, **PacBio** is not available when you select the Hybrid reference-guided/de novo genome assembly workflow, as the gap closure step is dependent on the presence of paired reads.

- Both types of **Ion Torrent** paired reads—”mate pairs” and “paired ends”—are supported.

- If using both Sanger and Illumina data, choose **Illumina** for all reads.

Specify paired-end data

Depending on the workflow and the read technology selected, the Input Sequences screen may allow you to specify paired reads.

To specify paired reads, check the **Paired-end data** box. This causes the Pair Distance dialog to pop up. Type in the pair distance and press **OK**. The default pair distance of **500** bp is suitable for most projects.

Preparing paired-end reads:

Paired end reads are typically in two files with the forward reads in one file and the reverse reads in the other. SeqMan NGen assumes the pair will be from opposite ends of the same DNA fragment, and sequenced from the end of the fragment inwards.

To enable SeqMan NGen to identify pairs, a sequence naming convention must systematically distinguish between different pair reads while specifying which pair reads are associated. Forward and reverse sequences must have identical names except for the unique portion that determines the direction of the clone. Expressions for these naming conventions are created using a subset of *regular expressions*, which utilize elements of the Grep language. The following rules apply:

- Two parallel files must use standard naming convention (e.g. s_7_1_sequence and s_7_2_sequence).
- “Forward” and “reverse” reads must be in *exactly* the same order in the two files.
- Both forward and reverse reads must be present for every pair, including pairs where one of the reads failed or is of very low quality.

As an example, forward and reverse Sanger pair files are named as follows: 01f.abi and 01r.abi, where “01” distinguishes that they are members of the same pair. The “f” and “r” at the end of each sequence name distinguishes the orientation.

In Grep, the naming convention would be written as follows:

- Forward convention: (.*)f\..*$
- Reverse convention: (.*)r\..*$

For more information on Grep name patterns, see Example regular expressions.

SeqMan NGen considers paired-end reads whose fore and reverse reads start at the same position in two reads to be clonal. In these cases, the reads with highest scores are retained, while the other reads are ignored.

Conventions for Sanger pairs:

Paired end Sanger reads are typically all in multiple files with the forward pairs having an “f” or “forward” in the name and the reverse pairs having “r” or “reverse” in the name.

Conventions for Illumina pairs:

Paired end Illumina reads are typically in two files, or a small number of files if they are from multiple runs or lanes. These pairs are specified by a naming convention used in the *.fasta* file comment line.

For *de novo* assemblies with paired end reads, SeqMan NGen automatically adds the following information to the script:

```
setPairSpecifier pairs:
 { {
   forward: “(.*)/1”
   reverse: “(.*)/2”
   min: 0
   max: 750
   key: Illumina
 } }
```

If reads do not match one of the pair specifiers, or if the forward and reverse specifiers are represented by empty strings (“”), the assembler will attempt to match using the whole name of the sequence. If exactly two reads have the same name, they will be considered a match.

For reference-guided assemblies, SeqMan NGen adds the following information:

```
 {
   is Pair: true
   file: “****”
   SeqTech: “Illumina”
   minDist: 0
   maxDist: 750
 }
```

For reference-guided assemblies with paired-end reads, SeqMan NGen recognizes the pairs by their file names. The following examples demonstrate some of the filename formats that SeqMan NGen supports for reference-guided pairs. Large-bold text in the examples is used to highlight the region of each filename that specifies the forward and reverse reads:

“R_2011_11_21_11_06_08_user_C29-100_PE_DH10B_11_Auto_C29-100_PE_DH10B_11_4120_reverse_**pe2**.fastq
“R_2011_11_21_11_06_08_user_C29-100_PE_DH10B_11_Auto_C29-100_PE_DH10B_11_4120_forward_**pe1**.fastq

“Strain1234_L7_*R1*_ATCACG_Index1.fastq”,
“Strain1234_L7_*R2*_ATCACG_Index1.fastq”,

“K12-1-B_TGACCA_L006_**R1**.fastq”,
“K12-1-B_TGACCA_L006_**R2**.fastq”,

“GBBC920_GGCTAC_L008_**R1**.filt.50bp.fastq”,
“GBBC920_GGCTAC_L008_**R2**.filt.50bp.fastq”

“tiny*_1*.txt”,
“tiny*_2*.txt”,

“tiny*_1*_sequence.txt”,
“tiny*_2*_sequence.txt”,

“**tiny1**._qseq”,
“**tiny2**._qseq”,

“s_1*_1*_sequence.txt”
“s_1*_2*_sequence.txt”

“C29-129_forward_**pe1**.fastq”
“C29-129_forward_**pe2**.fastq”

The Grep used to match the **pairFileNames** is shown below:

“(?‘name’.*?)_R1_(?‘ext’.*)\\.fastq”,
“(?‘name’.*?)_R2_(?‘ext’.*)\\.fastq”,

“(?‘name’.*?)_R1\\.(?‘ext’.*)\\.fastq”,
“(?‘name’.*?)_R2\\.(?‘ext’.*)\\.fastq”,

“(?‘name’.*?)_forward_pe1(?‘ext_p’\\.fastq)”,
“(?‘name’.*?)_reverse_pe2(?‘ext_p’\\.fastq)”,

“(?‘name’.*?)_{0,1}1\\.fastq”,
“(?‘name’.*?)_{0,1}2\\.fastq”,

“(?‘name’.*?)1\\.fastq”,
“(?‘name’.*?)2\\.fastq”,

“(?‘name’.*?)1_sequence\\.txt”,
“(?‘name’.*?)2_sequence\\.txt”,

“(?‘name’.*?)1\\.txt”,
“(?‘name’.*?)2\\.txt”,

“(?‘name’.*?)1\\._qseq”,
“(?‘name’.*?)2\\._qseq”,

“(?‘name’.*?)1\\.fq”,
“(?‘name’.*?)2\\.fq”,

The following script command can be used to add support for a new filename format. The command must be executed before assembly. The pattern will be used for all subsequent **assembleTemplate** commands for that run of the reference-guided assembler.

pairFilePattern forward: “(?‘name’.*?)_R1_(?‘ext’.*)\.fastq” reverse: “(?‘name’.*?)_R2_(?‘ext’.*)\.fastq”

Example regular expressions

Examples of expressions you may find useful regarding paired end naming specifications follow. Please note this is not a complete list of regular expressions, and the definitions of the terms used are limited to their application to SeqMan NGen paired end naming specifications.

Special Characters	
[ ]	Character class used to enclose a list of alternatives. For example: [Aa]bc matches abc and Abc. If the first character is a carat (^), it means anything but the characters on the list. Thus: [^a]bc matches xbc but not abc.
****	A switch that makes special characters literal and literal characters special.
( )	Grouping--used to delimit a string comprising a “phrase.” Phrases are necessary in paired end specification so you can match a pair of forward and reverse reads while still distinguishing their orientation. In SeqMan NGen, phrases in parentheses must match for two reads to qualify as a pair; phrases outside the parentheses are used to distinguish members of the same pair.
\d	Any digit (0-9)
\D	Any non-digit character.
\w	Any alphanumeric “word” character (including “_”)
.	Any character
|	Alternate--either the term before “|” or after “|”
^	Match at the beginning of the line only.
$	Match at the end of the line only.
Numerical Modifiers	
*****	0 or more
+	1 or more
?	1 or 0
{n}	Exactly n
{n,}	At least n
{n,m}	At least n but not more than m
Example Expressions and Their Meanings	

d	Literally the letter d
\d	Any digit (0-9)
\d*	Zero or more digits
\d+	One or more digits
(\d+)	A phrase comprising one or more digits--same as “\d+”, but causes SeqMan NGen to match the names from the string inside the phrase when other characters in the name may not match.
\.	Literally the period symbol (.)
.	Any character
.+	One or more of any characters
.*	Zero or more of any characters
a|b	a OR b
ab[i1]	abi or ab1
abi$	Ends with abi
[\.\d]	A period OR a digit
[abc]	a OR b OR c
[abc]+	One or more characters from the set a, b, c
.*f	Any number of any characters followed by the letter “f”
(.*)f*	A phrase comprising any number of any characters, followed by the letter “f”--same as “.*f”, but causes SeqMan NGen to match the phrase in parentheses without matching the “f” in a read name
(\D+)r(\d+)	One or more non-digit characters followed by “r” followed by one or more digits.
(\d{2,4})f(\.abi)	Two, three or four digits followed by “f” followed by “.*abi*”

Specify single sample, multi-sample or replicate data

Depending on the workflow, the Input Sequences screen may include an **Experiment setup** drop-down menu. You must make a selection from this menu before proceeding to the next screen.

To run each sample individually:

Choose **Single sample**.

To run multi-sample data as separate assemblies:

Choose **Multi-sample**. When multi-sample assemblies are run separately, each data set is run against the reference sequence independently and an .assembly package is created for each sample. If you choose this option, an **Experiment** column is added to the sequence file table. Add sequences to the table as usual. Next, replace each blank **Experiment** cell with a name; this must be done before you can proceed to the next screen. Data files that share identical names in this column will be assembled together.

- To assign experiment names manually, click on each individual cell and type in a name. All rows with identical **Experiment** names will be treated as a group.
- To assign experiment names automatically based on their file names minus the file extensions, select one or more rows using click, **Ctrl+click**, **Cmd+click** or **Shift+click** and then press the **Auto Name** button.
- To group and name experiments, select one or more rows and press the **Group Selected** button. A popup will appear prompting you to type in a name for the selected file(s). Any names assigned to the row before using **Group Selected** will be overwritten.
- To remove experiments from a group, select one or more rows and press the **Ungroup Selected** button. In the selected rows, the **Experiment** column will return to its original (blank) state.

If you choose this option, a Set Up Experiments screen will be included among subsequence wizard screens.

To run multi-sample data as a single assembly:

Choose **Multi-sample with replicates**. When multi-sample data is run as a single assembly process, the data from all the samples are processed together and a single .assembly package is produced. In that case, there is a single alignment view with the data separated into the corresponding sample groups, each with a pseudo-consensus. Note that in assembling multi-sample data as a single assembly, SeqMan NGen

considers all samples together. This can affect the final gapped alignment and therefore potentially yield slightly different results than assembling each sample individually. If you choose this option, a **Replicate** column is added to the sequence file table. Add sequences to the table as usual. Next, replace each blank **Replicate** cell with a name; this must be done before you can proceed to the next screen. Data files that share identical names in this column will be treated as replicates.

- To assign replicate names manually, click on each individual cell and type in a name.
- To assign replicate names automatically based on their file names minus the file extensions, select one or more rows using click, **Ctrl+click**, **Cmd+click** or **Shift+click** and then press the **Auto Name** button.
- To group and name replicates, select one or more rows and press the **Create Replicate** button. A popup will appear prompting you to type in a name for the selected replicates. Any names assigned to the row before using **Create Replicate** will be overwritten.
- To remove items from a replicate group, select one or more rows and press the **Undo Replicate** button. In the selected rows, the **Replicate** column will return to its original (blank) state.

If you choose this option, a Set Up Replicate Sets screen will be included among subsequence wizard screens.

Specify RNA-Seq options

When following the reference-guided RNA-Seq workflow, the Input Sequences screen has an additional option: **Stranded RNA-Seq reads**.

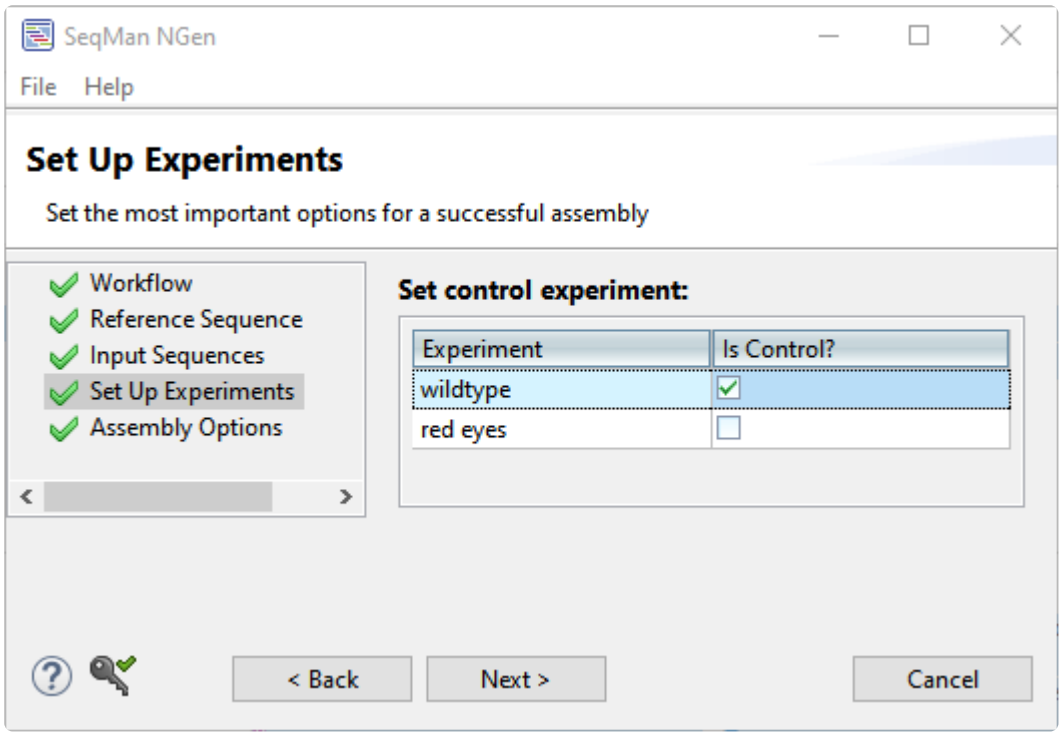
Some library preparation methods preserve the directionality of reads, i.e., reverse reads always point 5' to 3' in the direction of transcription, while forward reads point from 3' to 5. If you selected the RNA-seq/ Transcriptomics > Quantitative analysis > RNA-seq workflow, the Input Sequences screen provides a **Stranded RNA-seq data** checkbox. Check the box if you want SeqMan NGen to determine whether reads come from the top or bottom strand of the genome. Strandedness information is used in two ways:

- To disambiguate transcription from overlapping genes on opposite strands.
- To allow the assembler to properly allocate reads from overlapping genes.

If you check the box, you can open the finished assembly in SeqMan Pro or SeqMan Ultra and view results in the form of color-coded reads.

Set Up Experiments

If, in the Input Sequences screen, you choose the **Experiment setup** option **Multi-sample**, Set Up Experiments will appear as the next wizard screen.



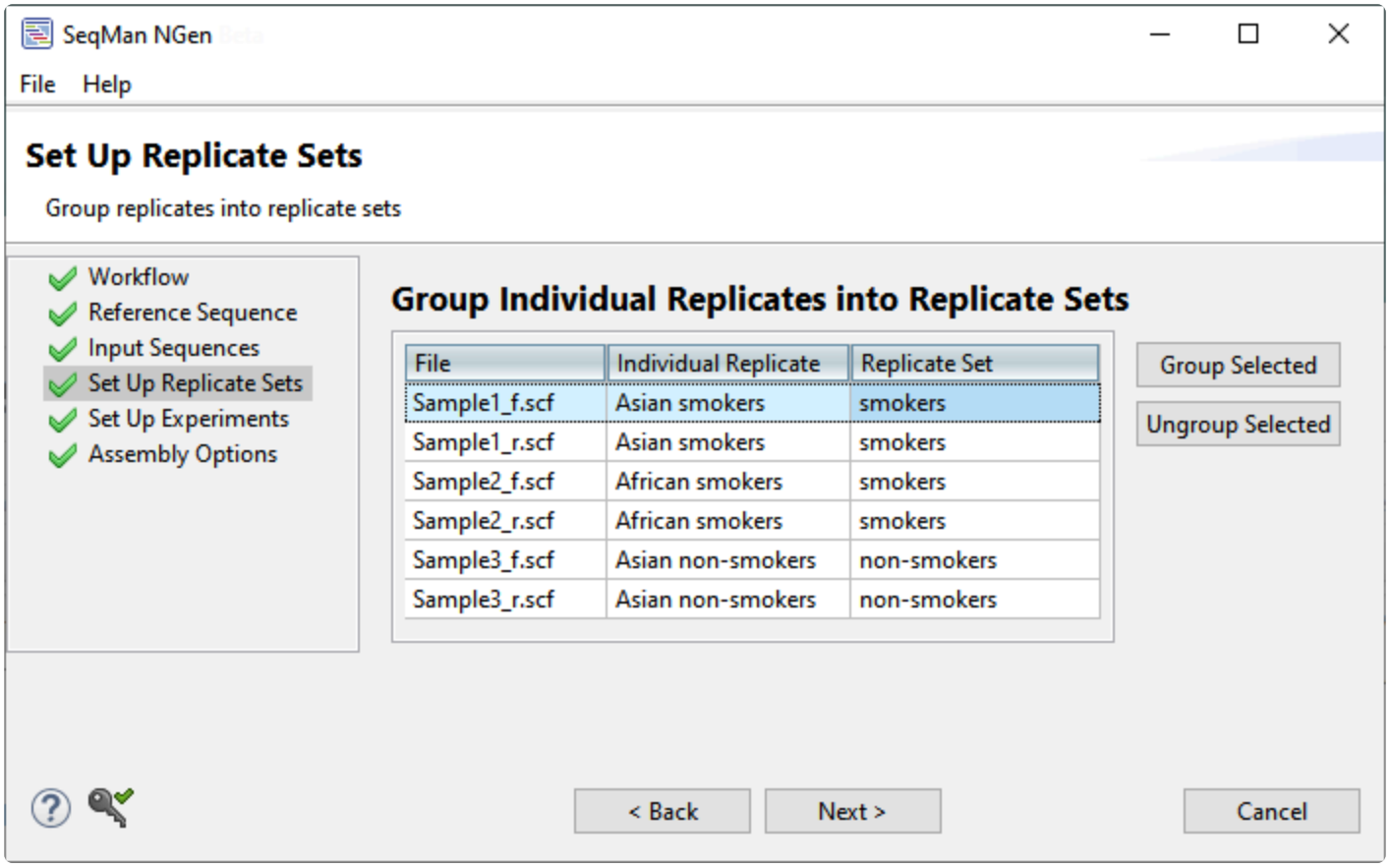
The **Experiment** column is pre-loaded with the “Experiment” names specified in the Input Sequences screen. If you wish to edit an experiment name, you must use the **< Back** button and edit them in one of those previous screens.

In the **Is Control** column, check one or more boxes to designate which experiments should be used as the baseline control for variant or CNV analysis.

Click **Next >** to proceed to the next wizard screen or **< Back** to return to the previous screen.

Set Up Replicate Sets

If, in the Input Sequences screen, you choose the **Experiment setup** option **Multi-sample with replicates**, Set Up Replicate Sets will appear as the next wizard screen. Replicates are defined in Input Sequences, while replicate sets are specified in the Set Up Replicate Sets screen. The image below shows an example screen after the information has been filled in.



- To group and name replicate sets automatically, select one or more replicates using click, **Ctrl+click**, **Cmd+click** or **Shift+click** and then press the **Group Selected** button to group the replicates and give them a single shared name. Type the name when prompted and click **OK**. All selected rows will now share the same “Replicate Set” name.

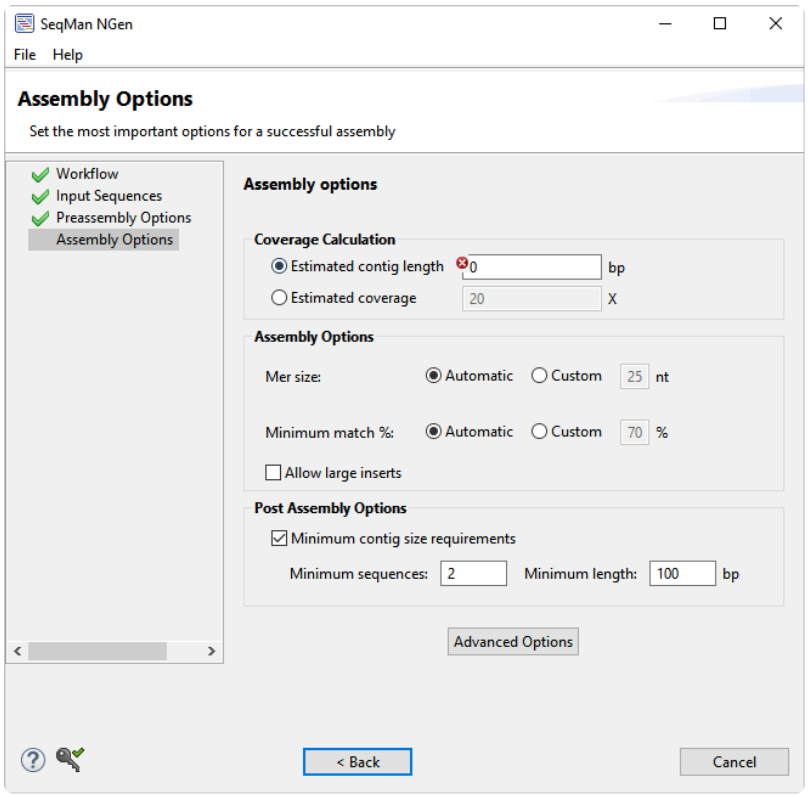
- To enter a single replicate set name manually, click on an individual “Replicate Set” cell and type in a name.

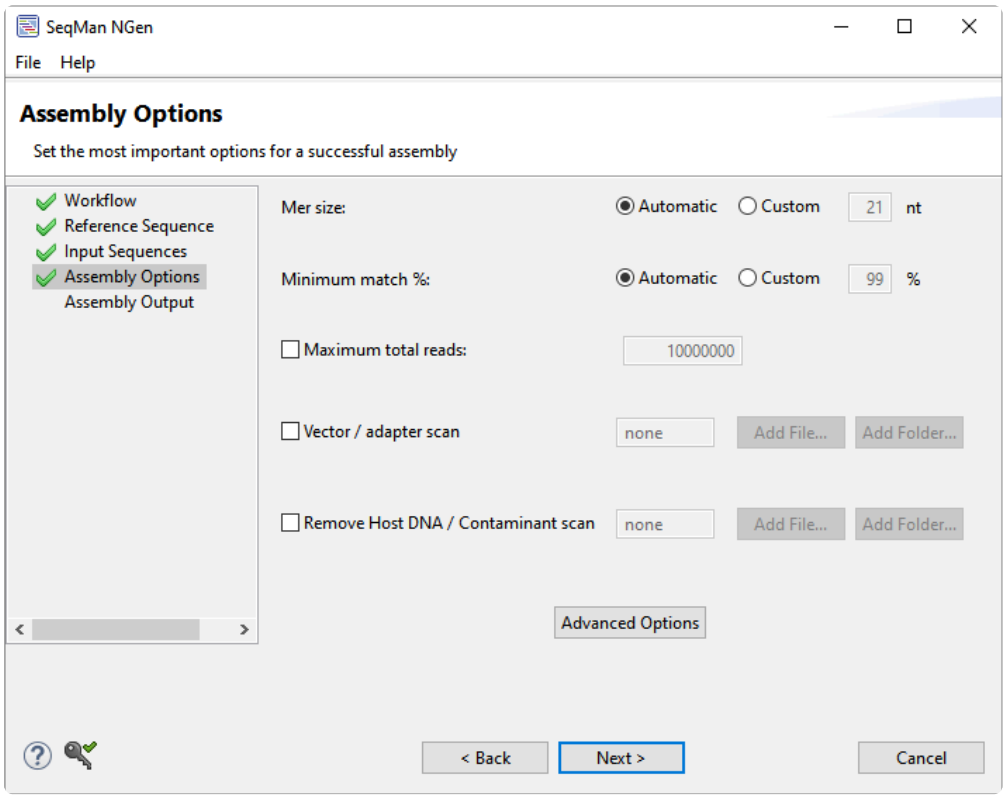
- To remove replicate sets, select one or more rows using click, **Ctrl+click**, **Cmd+click** or **Shift+click** and then press the **Ungroup Selected** button. In the selected rows, the “Replicate Set” column will return to its original (blank) state.

Click **Next >** to proceed to the next wizard screen or **< Back** to return to the previous screen.

Assembly Options

The Assembly Options wizard screen allows you to specify assembly parameters. This screen comes in several variations, two of which are shown below (click on either to see full-size):





Depending on the workflow, only a subset of the following options will be available:

Category	Options and Descriptions
Coverage calculation	If you know the approximate length of the genome or fragment being assembled, select **Estimated genome/contig length** and specify a length. SeqMan NGen will then calculate the expected average coverage empirically from the amount of data. This, in turn, allows repeat regions to be identified and handled more accurately, resulting in a better assembly. If the approximate genome length is not known, use the **Expected coverage** option. If you do not know the length of the genome/fragment, select **Estimated coverage** and provide an estimate of the depth of the sequencing. The default value for this field is 20, and the maximum allowable value is 65,535. If you enter a value larger than the maximum, you may receive an error message and be prevented from continuing until you choose a value less than or equal to the maximum. Use caution when estimating the value for **Expected coverage**. If the value you use is significantly lower than the actual depth, the assembly may take a much longer time to complete and may have too many mers flagged as repeats. We recommend using **Expected genome length** whenever possible.
Mer size	The minimum length of a mer (overlapping region of a fragment read), in bases, required to be considered a match when arranging reads into contigs. Mer size information is used to identify matches during the assembly layout phase. The default mer size is determined by the selected read technology and is shown in the window. For more information, see How mer tags are chosen. • **Automatic** – Select this button to automatically set the size based on assembly type and sequencing technology. • **Custom** – Select this button to choose the size yourself. You must enter the desired number of base pairs in the field at right. Lowering the mer size increases the sensitivity of finding matches, but also increases the likelihood of finding spurious matches in addition to the correct match. Lowering the mer size can also greatly increase the requirements for storing intermediate and temporary files with large projects.
Minimum match %	Specifies the minimum percentage of matches in an overlap required to join two sequences in the same contig. SeqMan NGen determines the percentage to use based on the sequencing technology you specified in the Assembly Options dialog. For more information, see Calculation of. • **Automatic** – Select this button to automatically set the percentage based on assembly type and sequencing technology. • **Custom** – Select this button to designate the percentage yourself. You must enter a number in the field at right.

Post assembly options	Check **Minimum contig size requirements** and type values for one or both of the following to remove assembled, untemplated contigs that do not meet minimum thresholds. This can lead to a desirable decrease in project size. • **Minimum sequences** – Disassembles any untemplated contigs with fewer than the specified number of sequences. This option affects only untemplated contigs. No templated contigs are removed. • **Minimum length** – Using this option disassembles any untemplated contigs shorter than the specified length.This option affects only untemplated contigs. No templated contigs are removed.
Maximum total reads	Check the box and enter a value if you wish to limit the read depth. Utilizing this option can make the assembly proceed faster. **TIP**: To see the effect of changes to this parameter, look at the **Estimated coverage** in the Run Assembly screen. In general, a coverage (AKA “depth”) of 50-100 is ideal and additional depth does nothing but slow the assembly. If your depth differs from the ideal, return to this screen and change **Maximum total reads** until the **Estimated coverage** is satisfactory.
Vector/ adapter scan	Adapter sequences are added to the ends of fragments during sequencing library preparations, and can interfere with downstream processing, if not removed. Check **Vector/ adapter scan** to add either one single- or multiple-sequence *.fasta* file or one folder of *.fasta* files containing known or suspected adapter sequences. The file(s) must be in *.fasta* format. During assembly, sequencing reads will be scanned for the presence of each of the specified adapters and when detected, trimmed off of that read. The trimmed read will then be used in any downstream processing. There is no specific header formatting. There is a minimum exact match length of 11 bases, and a minimum overall match of 15 bases, that allows for some mismatching. Both ends are searched within a specified range (default = 130), and all bases from an identified match to that end of the read are trimmed off.
Remove Host DNA / Contaminant scan	If you want SeqMan NGen to ignore host or contaminant DNA during assembly, check the box and navigate to the file or folder containing the host DNA or contaminant sequence(s). See Remove PhiX control reads from Illumina data.
De novo PacBio/ Nanopore options	Choose **Assemble all reads** or **Use subset of reads**. If you choose the latter, you may (or must) enter: • **Expected genome length** – If you know the approximate length of the genome/ fragment being assembled, select this button and specify a length. SeqMan NGen will then calculate the expected average coverage empirically from the amount of data. This, in turn, allows repeat regions to be identified and handled more accurately, resulting in a better assembly.

	• **Desired depth of coverage of final assembly** – For optimal results, use a subset of reads to achieve 100x depth of coverage. • **Use the longest reads in data set to achieve depth** • **Use the first n reads in data set to achieve depth**

To set additional assembly options, press the button named **Advanced Options** or **Advanced Assembly Options**.

Click **Next >** to proceed to the next wizard screen or **< Back** to return to the previous screen.

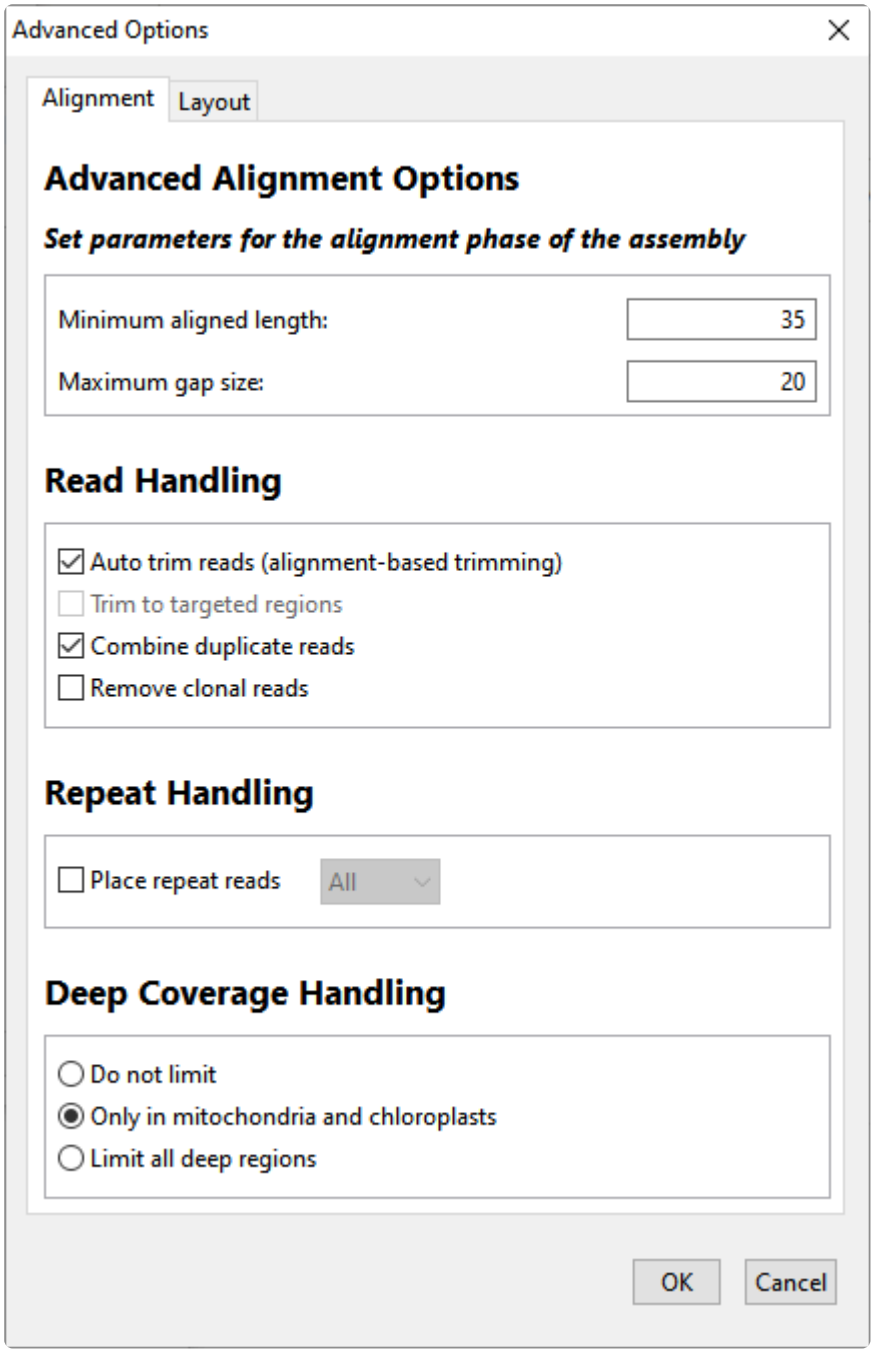
Advanced Assembly Options

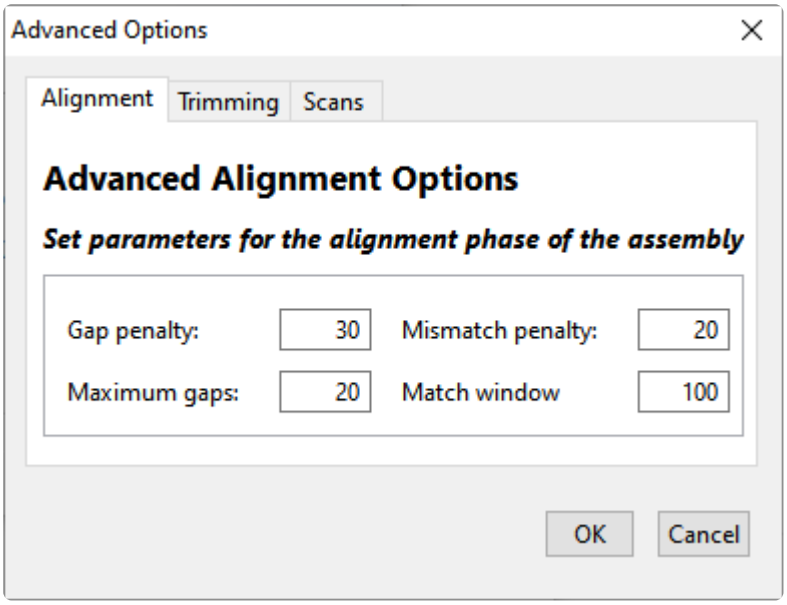
Advanced Assembly Options is a multi-tabbed dialog that opens when you click the **Advanced Options** button in the Assembly Options screen. Each tab has changeable parameters for different parts of the assembly process. Different workflows have different subsets of tabs. In addition, tabs with the same names may contain different options depending on the workflow.

- Alignment – Parameters for the alignment phase of the assembly.
- Layout – Parameters for the layout phase of the assembly. .
- Trimming – Parameters for the trimming phase of the assembly.
- Scans – Scanning parameters.

Alignment tab

The Alignment tab of the Assembly Options dialog is used to set parameters for the alignment phase of the assembly. To access the tab from the Assembly Options screen, click the **Advanced Options** button then click on the **Alignment** tab. The options available in this tab vary depending on the workflow, and two different versions of the tab are shown below.





The table below shows editable options in alphabetical order; each workflow includes a subset of these options. Default parameters vary according to the sequencing technology and project type specified elsewhere in the wizard, and values seldom need to be changed.

Parameter	Description
Auto trim reads (alignment-based trimming	If this box is checked, the ends of reads are trimmed to best match alignment to the reference. SeqMan NGen will mark the portion of the read that aligns well to the reference and will set the trimming to skip any of the poorly aligning parts of the read. Checking this option optimizes the end trimming of reads to maintain as much of the read as possible, while still meeting the minimum match percentage threshold. However, checking the box can also lead to the removal of true variant bases located near the ends of reads. The box is checked by default.
Combine duplicate reads	Duplicate reads are those which share the same starting position and the same sequence. Check this box if you wish to combine the reads and only enter one of them into the alignment. Any duplicates will be scored but not aligned. Combining duplicate reads collapses reads with identical sequences with the same start and stops and replaces them with a single entry with a suffix “[dup #]” where # is the number of collapsed reads. However, this option does not take the location of a paired end read into consideration. It is used primarily to reduce issues with alignment and visualization of very deep sequence regions, typical of RNA-Seq data for highly expression genes.
Deep coverage handling	This section of the Alignment tab dialog lets you specify whether and when to filter deep coverage regions. The default selection is made automatically by SeqMan NGen based on the current workflow. However, you may make any desired selection. The way in which very deep coverage is handled can greatly affect assembly time. For example, if unlimited deep coverage is allowed, it could take upwards of eight hours to align human NA24385 sequences (Genome in a Bottle “Ashkenazim Trio Son”) to the mitochondrial (MT) genome. That is because most of that human sample has a coverage depth of 35,000. In such a case, limiting deep coverage regions

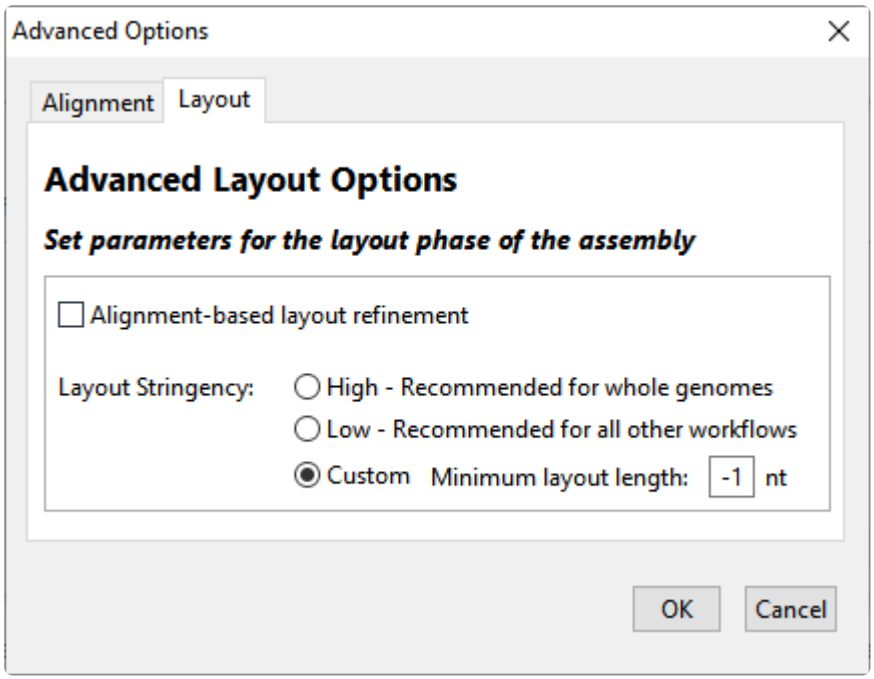
	can allow the assembly to proceed much more quickly. Choose between: • **Do not limit deep coverage regions** – Use all reads, regardless of depth. With the exceptions described below, this is the default for most reference-guided workflows. • **Only limit deep coverage regions for Mitochondria and Chloroplasts** – Limit the depth of coverage only in the specified areas.The default for all assemblies using references that include mitochondrial and chloroplast genomes. • **Limit all deep coverage regions** – Limit the depth of coverage along the entire length of the sequence. The default for all miRNA and microbial genome workflows.
Gap penalty	The penalty for opening or extending a gap during an alignment. This penalty is deducted from the pairwise score used to calculate match percentage. A high gap penalty suppresses gapping, while a low value promotes gapping.
Layout align	In cases where a read has an identical, or nearly identical, overlap score to more than one location on the reference, indicative of a repeated sequence, the read can be evaluated by attempting a fully gapped alignment to each potential mapping position and selecting the position with the best score. In case of ties, the read is placed in one of the locations at random. Checking this box will further lower the false discovery rate (FDR), but may substantially increase the assembly time. The default is for this box to be unchecked.
Layout stringency	To specify the non-permanent “soft” filters for SNP data. SNPs that do not meet thresholds specified in this section are removed from certain displays (e.g., tables) but are still retained in the final project and may be displayed in downstream analysis, if desired. Specify **Low**, **High** or **Custom** stringency. Choosing **Custom** enables additional options.
Match window	The size of the window used to calculate match percentage.
Maximum gap size	The theoretical maximum length of a gap that could be inserted. In practice, the maximum gap size will usually be about half of this value. The maximum allowable value is 99.
Maximum gaps	The theoretical maximum length of a gap that could be inserted. In practice, the maximum gap size will usually be about half of this value. The maximum allowable value is 99.
Minimum aligned length	The minimum length of at least one aligned segment of a read after trimming. The default value varies depending on the read technology you selected. Allowed values are 0-999.
Mismatch penalty	The penalty for a base mismatch during an alignment. This penalty is deducted from the pairwise score used to calculate match percentage.
Place repeat reads	Choose to place repeated reads **Once**, **All** or **Never**. The default is **All** for metagenomics workflow and **Once** for all other workflows. The way in which very deep coverage is handled can greatly affect assembly time. For example,

	if unlimited deep coverage is allowed, it could take upwards of eight hours to align human NA24385 sequences (Genome in a Bottle "Ashkenazim Trio Son") to the mitochondrial (MT) genome. That is because most of that human sample has a coverage depth of 35,000. In such a case, limiting deep coverage regions can allow the assembly to proceed much more quickly.
Remove clonal reads	Clonal reads, where the sequence and endpoints of both reads in a pair match those in another pair, are usually the result of PCR artifacts. Check this box if you wish to retain one of the pairs in the assembly, but completely remove the clones (duplicate pairs) after the alignment phase of assembly. If the box is checked, cloned reads will not be scored, and will not be included in SNP calculation or gene quantification. This option can be useful in genome/exome/gene panel sequencing workflows where clonal reads can skew variant calculations. Checking the box may add substantially to the time required for assembly. Checking this option does not remove a pair if its two reads are duplicates of different pairs. It only removes duplicate pairs if the entire pair is completely identical to another pair. For example, SeqMan NGen will not remove a pair whose forward read is a duplicate of a read from pair A, but whose reverse read is a duplicate of a read from pair B. **Note:** Do not check both **Combine duplicate reads** and **Remove clonal reads**, as this will lead to unpredictable results due to the order in which SeqMan NGen removes clones and combines duplicates.
Trim to targeted regions	This box is only enabled for workflows that offer the ability to add a *.bed* file, and where a *.bed* file was specified in the Reference Sequence screen If this box is checked, reads extending beyond the 5' or 3' end of a targeted region will be trimmed to the target boundary. The box is unchecked by default.

Once you are finished, click **OK** to save changes and return to the Assembly Options screen, or **Cancel** to return without saving changes.

Layout tab

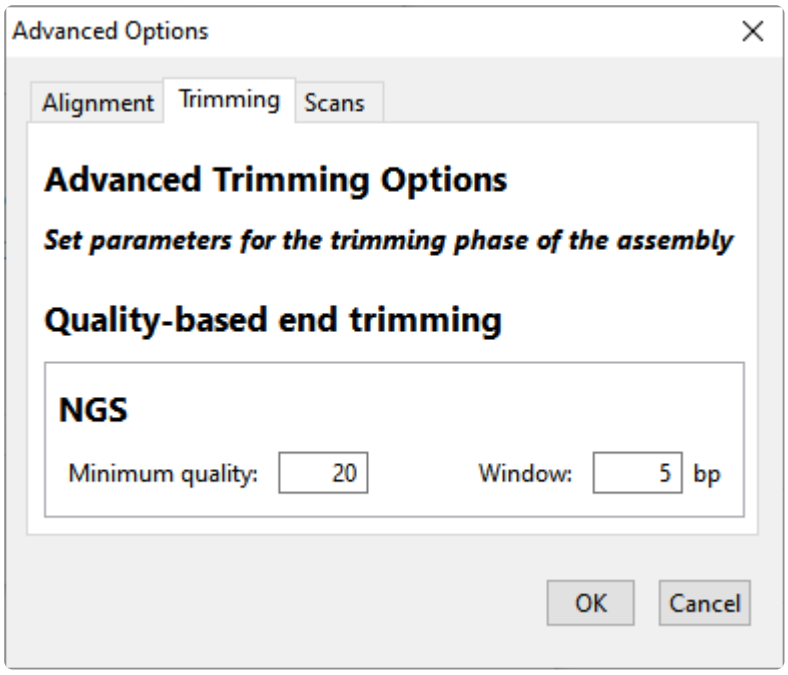
The Layout tab of the Assembly Options dialog is used to set parameters for the layout phase of the assembly. To access the tab from the Assembly Options screen, click the **Advanced Options** button then click on the **Layout** tab. The options available in this tab may vary depending on the workflow.



Parameter	Description
Alignment-based layout refinement	If you are following a whole-genome workflow, checking this button may improve the accuracy of the assembly.
Layout Stringency	Choose from the following: • **High** – Recommended for whole genomes. • **Low** – Recommended for all other workflows. • **Custom** – If you wish to enter the **Minimum layout length** manually. This is the minimum number of identical matching bases (from the mer analysis only) for a read to be included in the layout. It is specified by an integer, with a default of 50 nucleotides. For reads shorter than 100 bases, the setting is automatically adjusted to the mer size. Increasing this number decreases the false discovery rate (FDR) and true positive rate (TPR). Allowed values are 0-999. The default is -1 nt (nucleotides).

Trimming tab

The Trimming tab of the Assembly Options dialog is used to set parameters for the trimming phase of the assembly. To access the tab from the Assembly Options screen, click the **Advanced Options** button then click on the **Trimming** tab.



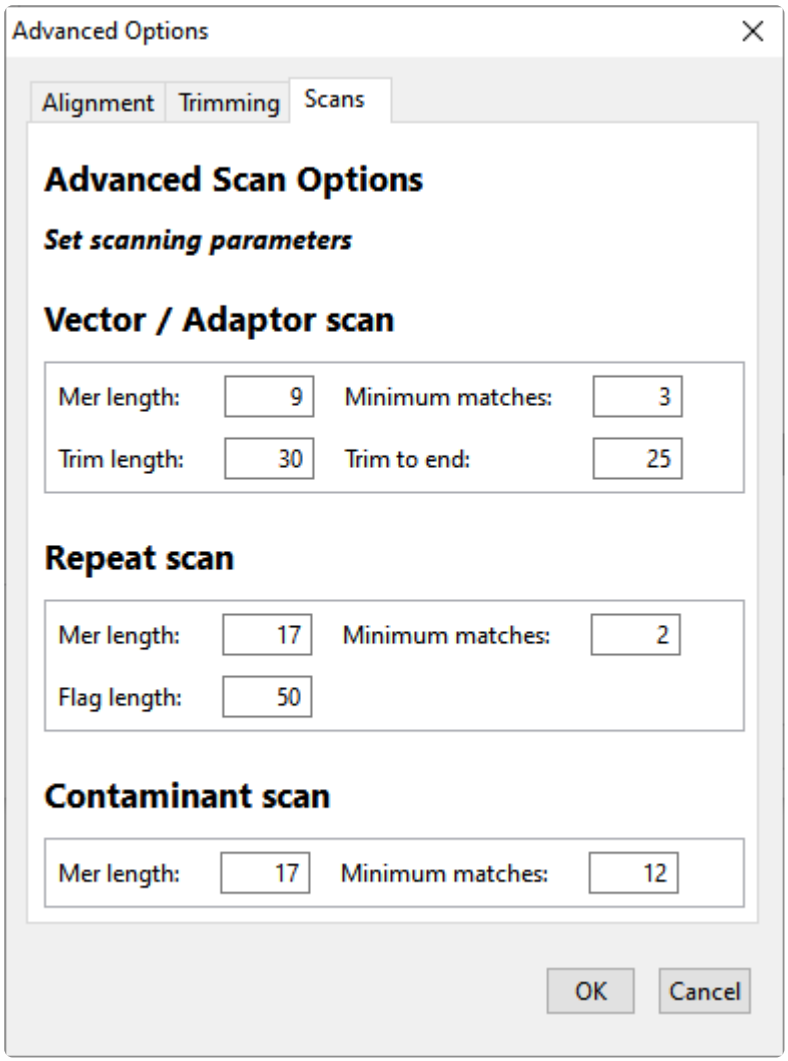
Default parameters vary according to the sequencing technology and project type specified elsewhere in the wizard, and values seldom need to be changed.

Parameter	Description
Minimum quality	The minimum averaged quality score of the evaluated window that is required in order to be considered low-quality.
Window	The length of the window to be used for averaging quality scores.

Once you are finished, click **OK** to save changes and return to the Assembly Options screen, or **Cancel** to return without saving changes.

Scans tab

The Scans tab of the Assembly Options dialog is used to set scanning parameters. To access the tab from the Assembly Options screen, click the **Advanced Options** button then click on the **Scans** tab.



Default parameters vary according to the sequencing technology and project type specified elsewhere in the wizard, and values seldom need to be changed.

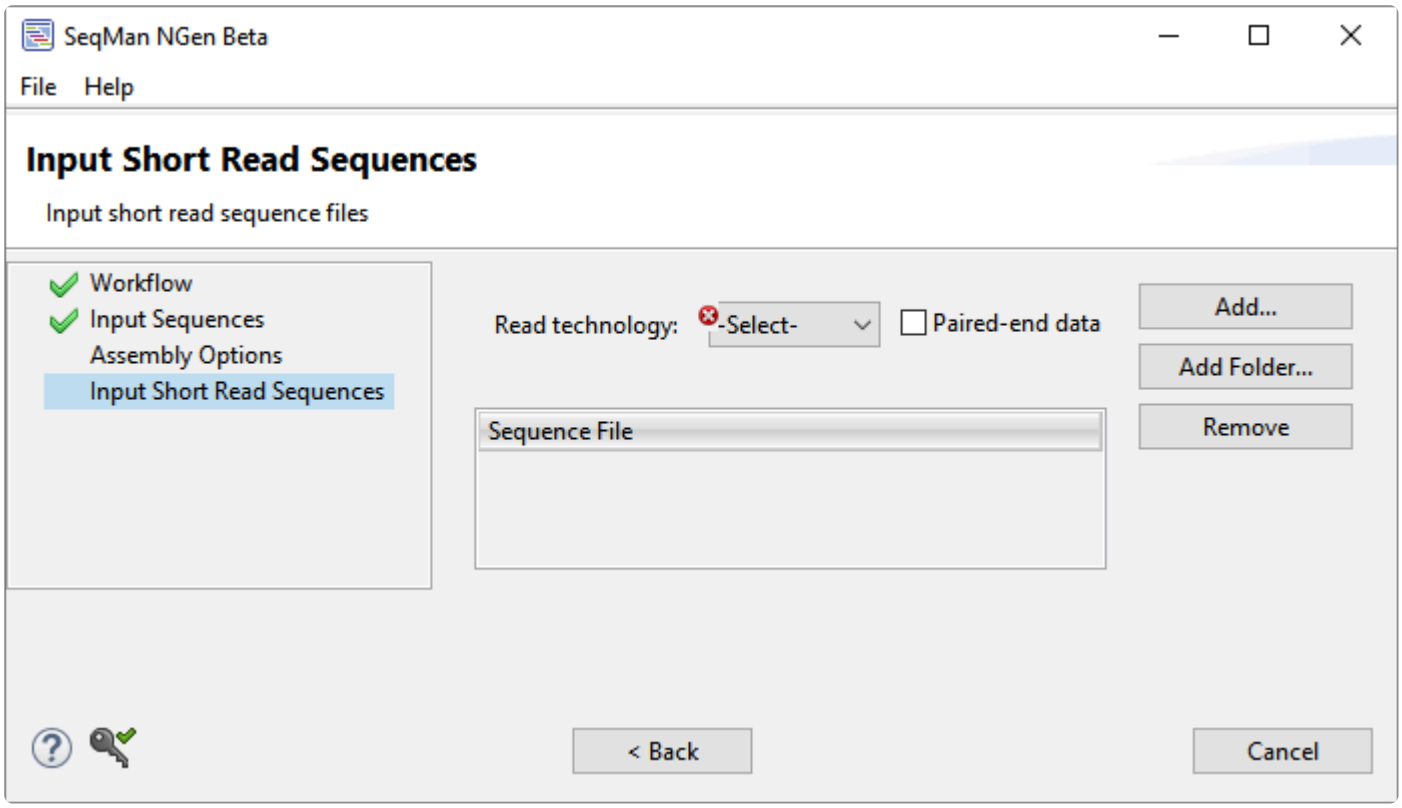
Section	Parameter	Description
Vector/Adaptor scan section	**Mer length**	The minimum length of a mer required to be considered an exact match when searching for vector.
	Trim length	The minimum length required for a mer to be considered as a match for vector trimming.
	Minimum matches	The minimum number of matching mers required to start an alignment.

	Trim to end	The distance to the endpoint where trimming will go all the way to the end of the sequence.
Repeat scan section	**Mer length**	The minimum length of a mer required to be considered an exact match when scanning for repeats.
	Flag length	The minimum length required for a mer to be flagged as a repeat.
	Minimum matches	The minimum number of matching mers required to be considered a repeat.
Contaminant scansection	**Mer length**	The minimum length of a mer required to be considered an exact match when scanning for contaminants.
	Minimum matches	The minimum number of matching mers required to mark the sequence as a contaminant.

Once you are finished, click **OK** to save changes and return to the Assembly Options screen, or **Cancel** to return without saving changes.

Input Short Read Sequences

If the Input Short Read Sequences screen appears, you must specify the required options and enter one or more read files in this dialog before proceeding to the next screen.



See the following topics to learn how to specify options and add data in this screen:

Pre-import steps (optional):

- Check paired-end data naming specifications
- Remove PhiX control reads from Illumina data

Specify data options:

- Specify read technology
- Specify paired-end data
- Specify RNA-Seq options
- Specify single sample, multi-sample or replicate data

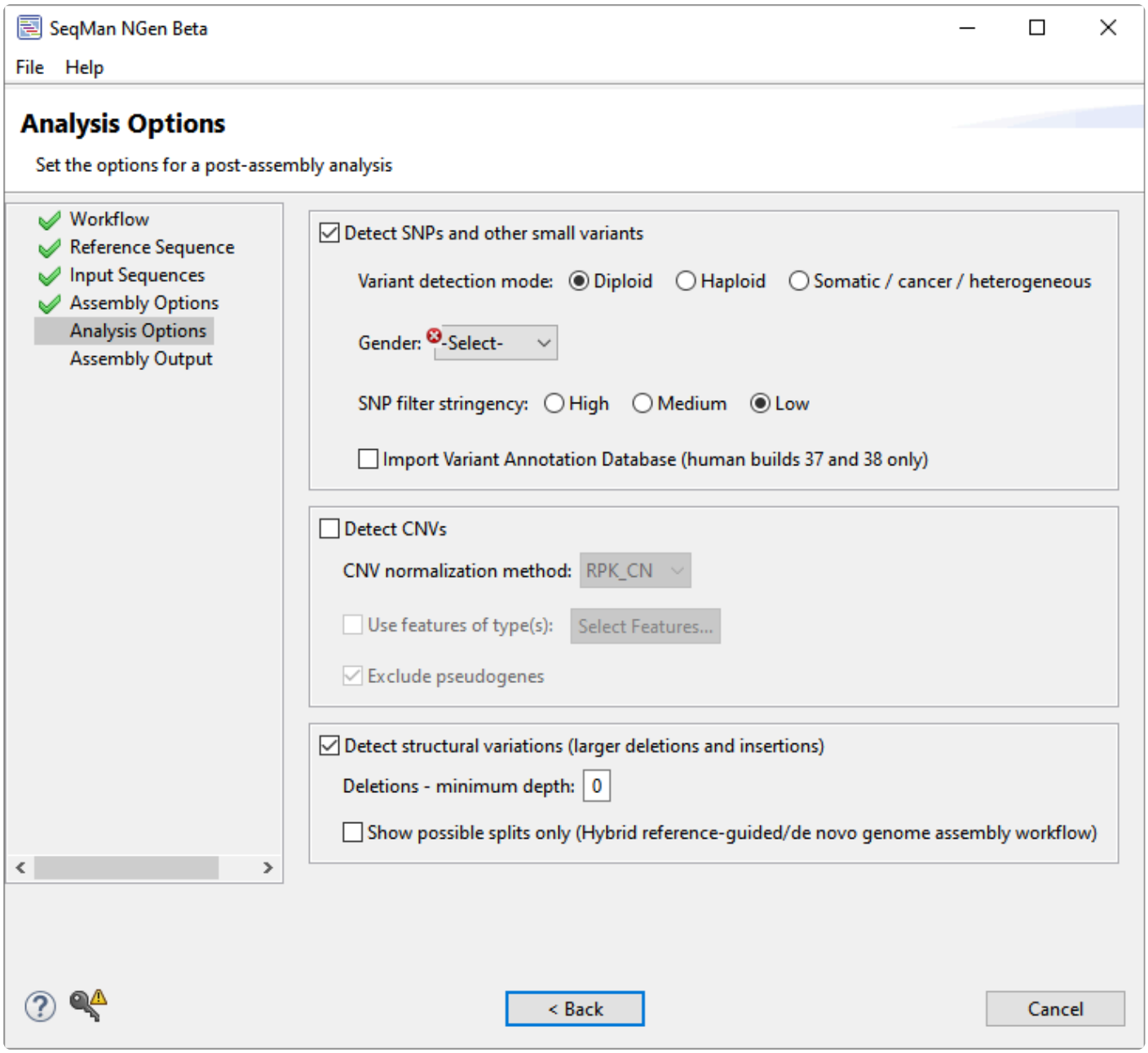
Add and remove sequences:

- Add sequences from your computer or the Cloud
- Remove a sequence from the list

Click **Next >** to proceed to the next wizard screen or **< Back** to return to the previous screen.

Analysis Options

The Analysis Options wizard screen allows you to specify the analysis parameters to use for your assembly. This screen comes in several variations, one of which is shown below:



Depending on the workflow, only a subset of the following options will be available:

Category	Options and Descriptions
SNP detection	To enable SNP detection, start by checking **Detect SNPs and other small variants**. If this box is checked, you can later open the assembly in ArrayStar or SeqMan Pro to view the SNPs.

Use **Variant detection mode** to specify genome ploidy for SNP detection purposes. Choosing **Haploid** or **Diploid** establishes the statistical model SeqMan NGen will use in estimating the probability that a given called variant is real (i.e., that the sequence really differs from the reference). Selecting **Somatic/cancer/heterogeneous** (e.g. for a polyploid genome, cancer panel, etc.) prevents SeqMan from calculating probabilities.

If the **Gender** checkbox is present, specify the gender of the subject (**Male/Female**), if known. Otherwise, select **Unknown**. This checkbox appears only if you are using a DNASTAR genome template package and have chosen a genome ploidy other than **Haploid**.

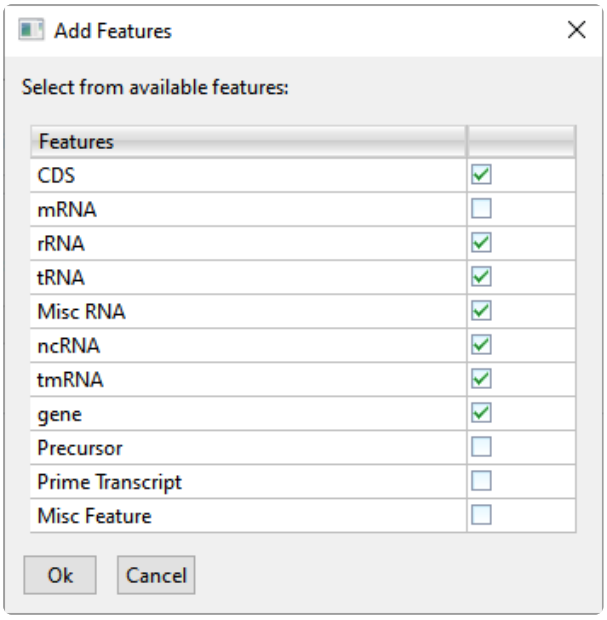
SNP filter stringency specifies two key settings for placing a read in the layout. When building an assembly, SeqMan NGen uses a three stage strategy: overlap, layout, and alignment. In the overlap stage of a reference-guided assembly, for example, each read and the reference are broken up into an overlapping set of substrings or “mers” of a specified length (“mer length” or “mer size”). Identical mer matches are an indication that the read matches the reference at that position. The more overlapping mers between two sequences, the stronger the indication that the match is real. The layout stage uses that overlap information and attempts to place each read in its true position on the reference. The final layout of all the reads is then sent to an aligner that produces the final fully gapped alignment. Layout stringency settings can be used to adjust the extent of overlap data required to include a putative match in the final layout.

The radio buttons specify stringency levels for “soft” filtering of SNPs. Soft filtering means that SNPs of the least interest to you will be automatically hidden when SNP reports/tables are viewed in SeqMan Pro, SeqMan Ultra or ArrayStar. Your selection in this screen controls the three assembly parameters shown in the table below. For more information on PnotRef, see Filter based on P not Ref.

- **High** has Min SNP%=15, PnotRef%=99.9, and Depth=20. This option has a lower false discovery rate (FDR) for SNPs and is recommended for whole genome workflows.
- **Medium** (where available) has Min SNP%=15, PnotRef%=99, and Depth=20.
- **Low** has Min SNP%=15, PnotRef%=90, and Depth=20. This option has a higher true positive rate (TPR) for SNPs and is recommended for all workflows other than whole genome.

Check **Import Variant Annotation Database (human builds 37 and 38 only)** if you are working with human samples and would like to import variant annotations from a specific portion of the NCBI RefSeq database maintained on the DNASTAR website. This checkbox is only available for human samples assembled against builds 37 or 38.

Note 1: If a BED, manifest and/or VCF file was specified during project setup and a SNP table is opened in SeqMan Pro or ArrayStar, then only the variants in the targeted regions and at the

	positions specified in the VCF within those targeted regions will be shown by default. Note 2: These “soft filtered” SNPs are *not* removed from the assembly, and can be made visible again by changing the SNP filtering parameters in SeqMan Pro, SeqMan Ultra or ArrayStar. This is in contrast to “hard filtering” of SNPs, which is done through the Layout tab of the Advanced Assembly Options dialog.
CNV detection	Check **Detect CNVs** if you wish to calculate copy number variants (CNV) as part of the assembly. If the box is checked, you may choose between two **CNV Normalization** method options: **RPK-CN** and **None** (i.e., no data normalization). If you also select a **Variant detection mode** other than **Do not calculate variants**, then CNVs, SNPs and small indels will be calculated from the assembly. After assembly, you can then use ArrayStar to view all three types, or SeqMan Pro or SeqMan Ultra to view only the SNPs and small indels. Check **Use features of type(s)** to only report results when a specific type of feature is used as the target for mapping reads. Note that mapping occurs regardless of the type of feature annotation. However, when you check this option, the mapping results for unwanted feature types will not be reported. Put checkmarks next to the feature types you wish to use, then press **OK**.  If you leave **Use features of type(s)** unchecked or if the reference sequence has no feature annotations, each individual sequence in the reference set will be used as a separate transcript (i.e., a single gene feature). Check **Exclude pseudogenes** to *not* report mapping results for features with **/pseudo** in their annotations. As with the previous option, mapping occurs regardless of the type of feature annotation.
RNA-Seq normalization	Check **Normalize RNA-Seq values** if you want to apply a normalization method to the data on a per-isoform basis. If you check this option, use the drop-down menu to choose the desired **RNA-Seq normalization method**. In order to enable this option, some workflows require you

	to check the **Calculate Copy Number Variation** option. See **CNV detection**, above, to learn about the **Use features of type(s)** and **Exclude pseudogenes** options.
ChIP-Seq normalization	Check **Set ChIP-seq peak detection method** if you want to apply a normalization method to the data. If you check this option, use the drop-down menu to choose the desired **ChIP-seq peak detection method**. See **CNV detection**, above, to learn about the **Use features of type(s)** and **Exclude pseudogenes** options

Click **Next >** to proceed to the next wizard screen or **< Back** to return to the previous screen.

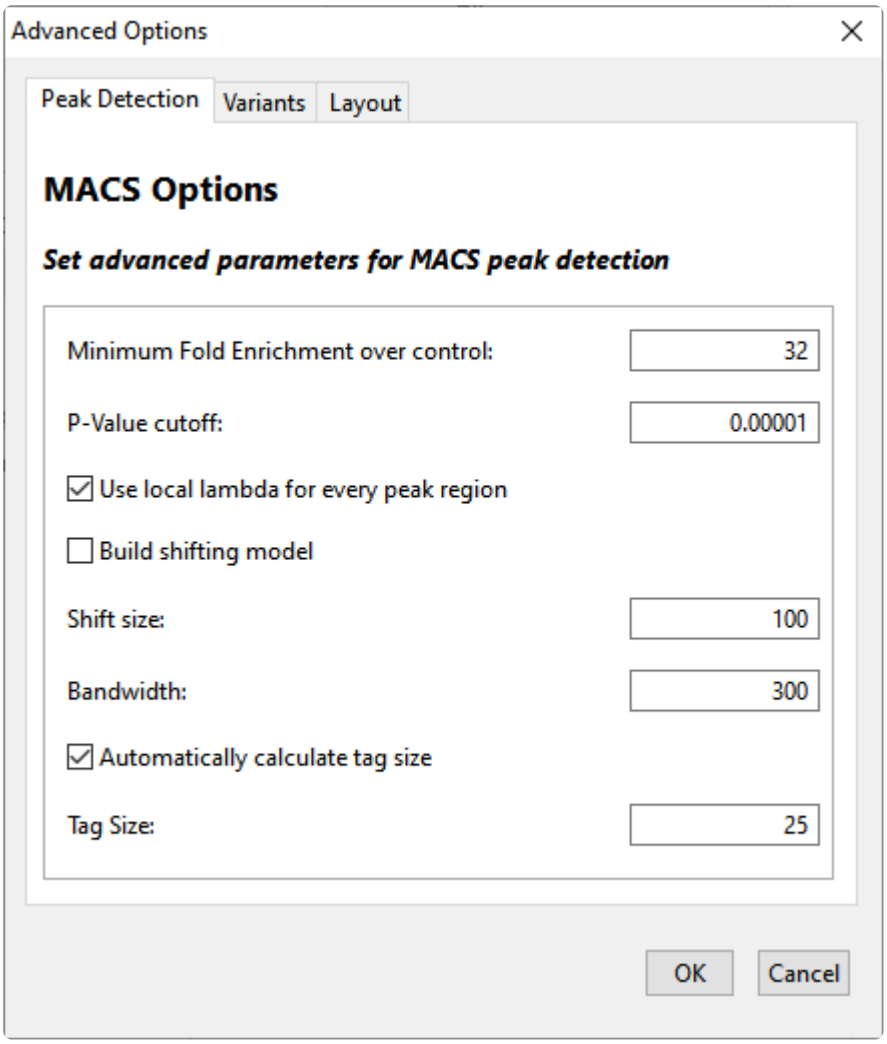
Advanced Analysis Options

Advanced Analysis Options is a multi-tabbed dialog that opens when you click the **Advanced Analysis Options** button in the Analysis Options screen. Each tab has changeable parameters for different parts of the analysis process. Different workflows have different subsets of tabs. In addition, tabs with the same names may contain different options depending on the workflow.

- Peak Detection
- Variants
- Layout

Peak Detection tab

The Peak Detection tab of the Analysis Options dialog is used to set parameters for MACS peak detection. To access the tab from the Analysis Options screen, click the **Advanced Analysis Options** button then click on the **Peak Detection** tab. The options available in this tab may vary depending on the workflow.

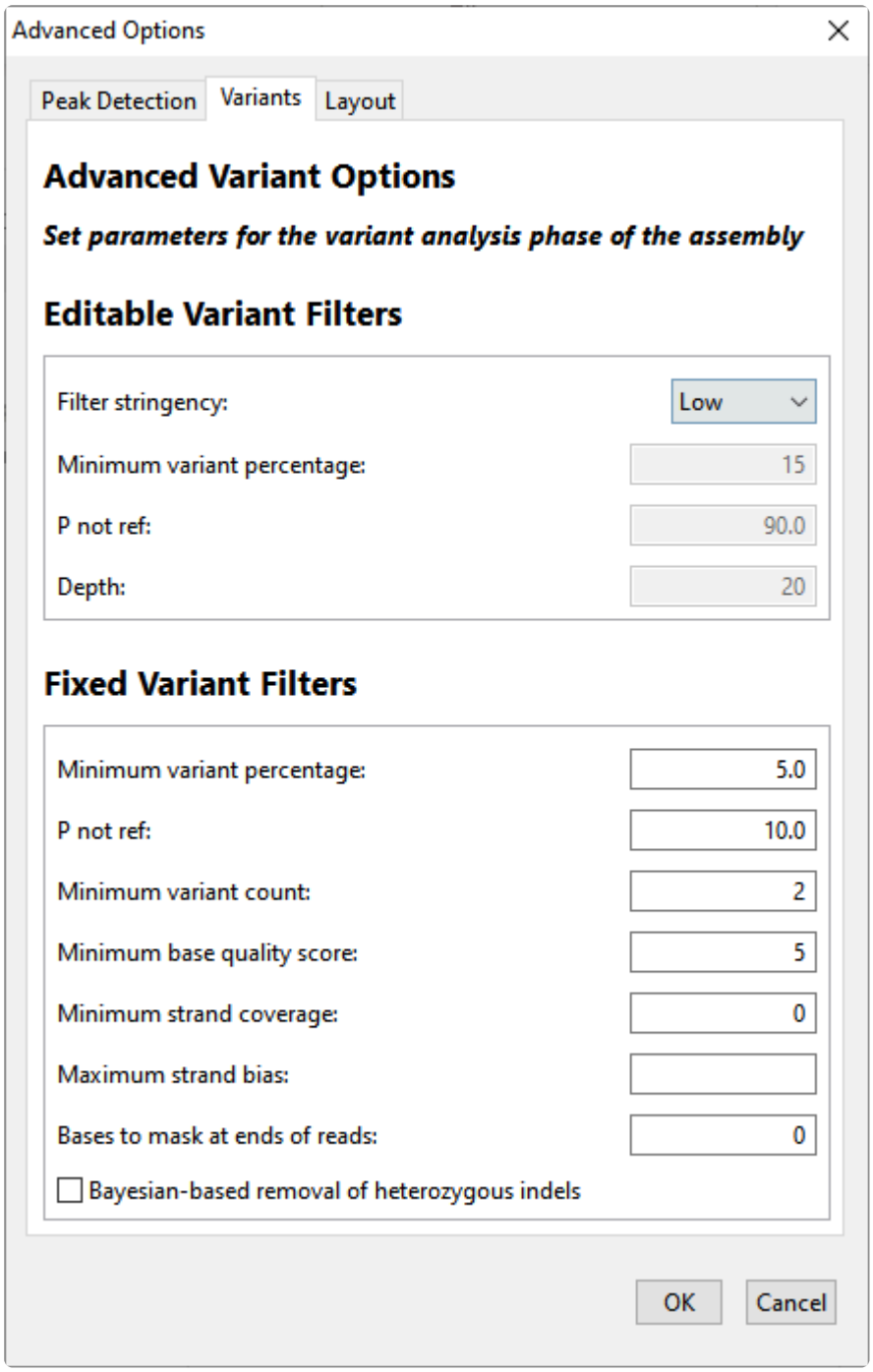


Parameter	Description
Minimum Fold Enrichment over control	This parameter controls how enriched a peak must be, compared to the background read distribution, in order to be considered in building the peak model. If a previous assembly attempt returned the error message “too few paired peaks to build a model,” we recommend using a lower number in this box.
P-Value cutoff	Local read distribution is compared to the threshold value entered here in order to calculate whether a peak should be counted as “real.” SeqMan NGen calculates the likelihood that a detected peak is actually a peak based on the local read distribution and only returns peaks with values below the **P-Value cutoff**. The default P-value cutoff is 0.0001. This doesn’t mean that values greater than 10^{-5} are filtered out, but rather that they are not included in the count

	of “real” peaks.
Use local lambda for every peak region	Lambda is a parameter used to define a Poisson distribution which MACS uses to determine the expected number of reads in a given region. When this option is checked, the Poisson distribution is calculated for the peak region and for three regions surrounding the peak. MACS can use this information to determine the expected number of reads in a given region. If the option is unchecked, then the local distributions are not calculated. Instead the expected distribution is based on the total number of reads and the effective size of the genome.
Build shifting model	Check this option to have MACS build a model based on the data to determine the width of and the distance between the "paired peaks.” Alternatively, leave this option unchecked to set **Shift Size** and **Bandwidth** values manually. The **Shift Size** is the distance each of the paired peaks will be shifted to try to center them over the actual binding site. The **Bandwidth** value defines the expected width of peaks. SeqMan NGen will search for peaks using a window twice as long as the bandwidth. **Note:** The creators of the MACS algorithm advise disabling model building when dealing with broad peaks (i.e. binding sites); for example, when studying histone binding.
Shift size	This parameter is described in Zhang et al., 2008.
Bandwidth	This parameter is described in Zhang et al., 2008.
Automatically calculate tag size / Tag Size	MACS treats all reads as though they have equal length. To explicitly specify that length, check **Automatically calculate tag size** and enter a length in the **Tag Size** text box.

Variants tab

The Variants tab of the Analysis Options dialog is used to set parameters for the variant analysis phase of the assembly. To access the tab from the Analysis Options screen, click the **Advanced Analysis Options** button then click on the **Variants** tab. The options available in this tab may vary depending on the workflow.



The table below shows editable options in alphabetical order; each workflow includes a subset of these options. Default parameters vary according to the sequencing technology and project type specified elsewhere in the wizard, and values seldom need to be changed.

Parameter	Description
Heterozygous peak threshold	This option is only available if you are using Sanger trace data. It is designed to identify positions in a read that contain two different bases that are both real. This can occur, for example, when you sequence a PCR product from a diploid genome at sites that are heterozygous. The percentage threshold is the minimum height of the secondary peak relative to the primary peak that's required to call the second base. Increasing the percentage increases the stringency at the cost of potentially increasing false negatives; decreasing the percentage calls more positions at the cost of potentially increasing false positives.
Editable Variant Filters section	
This section lets you specify the non-permanent "soft" filters for SNP data. SNPs that do not meet thresholds specified in this section are removed from certain displays (e.g., tables) but are still retained in the final project and may be displayed in downstream analysis, if desired.	
Filter stringency	Use the down menu to specify **low**, **medium**, **high** or **custom** stringency. Choosing **custom** enables the next three options in the dialog, Otherwise, these options are disabled and instead populated with unchangeable default values based on your stringency selection.
Minimum variant percentage	The minimum percent of non-reference bases required to call a SNP. When it performs SNP passes, SeqMan NGen will include regions in an assembly that have coverage less than or equal to the specified value. The default value is 5. A non-zero value is recommended when using Ion Torrent data, or working with larger genomes or doing population studies. Very low values will lead to larger files, but do not necessarily result in better SNP calls. This is only enabled when **Custom** is chosen as the **Filter stringency**.
P not ref	The minimum SNP quality score (Q_{call}) required to include a position as a putative SNP. For more information on the several ways to set P not Ref, see the topic Filter based on P not ref. This is only enabled when **Custom** is chosen as the **Filter stringency**.
Depth	The minimum depth of coverage required to include a position as a putative SNP. This is only enabled when **Custom** is chosen as the **Filter stringency**.
Fixed Variant Filters section	
This section lets you specify permanent "hard" filters for SNP data. SNPs that do not meet thresholds specified in this section are permanently deleted without saving and will not be displayed at any point downstream.	
Minimum variant percentage	The minimum percent of non-reference bases required to call a SNP. When it performs SNP passes, SeqMan NGen will include regions in an assembly that have coverage less than or equal to the specified value. The default value is 5. A non-zero value is recommended when using Ion Torrent data, or working with larger genomes or doing population studies. Very low values will lead to larger files, but do not necessarily result in better SNP calls. **Minimum variant percentage** and **Minimum variant count** can be used in tandem to control the number of reportable SNPs, and by extension, the size of the SNP table.
P not ref	**The minimum SNP quality score (Q_{call}) required to include a position as a putative SNP. For more information on the several ways to set P not Ref, see Filter based on.**

Minimum variant count	The minimum number of non-reference bases required to call a SNP. When it performs SNP passes, SeqMan NGen will include regions in an assembly that have coverage less than or equal to the specified value. Minimum SNP percentage* and **Minimum variant count** can be used in tandem to control the number of reportable SNPs, and by extension, the size of the SNP table.
Minimum base quality score	The minimum quality score below which a base will not be considered.
Minimum strand coverage	The minimum number of reads from each strand required to call a variant at a given position.
Maximum strand bias	Strand Bias (SB) for a SNP is the bias for the SNP appearing on one strand versus the other. It is measured relative to the strand bias in the assembly at the location of the SNP. For example, in a column with 60 forward reads and 40 backward reads, 6 SNP bases on the forward strands, and 4 on the reverse strands would be unbiased. SB is given by the formula: SB = $\|SNP\%_f - SNP\%_r\| / Total\ SNP\%$ …where $SNP\%_f$ and $SNP\%_r$ are the percentage of reads containing the variant on the forward (top) and reverse (bottom) strands, respectively; and SNP% is the total percentage of reads containing the variant. SB is calculated based on an “absolute value,” and will therefore be a positive number. • 0 – Perfectly balanced (unbiased) strands. Reads with variants are present on both strands, and variants appear equally on both stands. . • 0-1, not inclusive – As the number ‘1’ is approached, more variants are called with unbalanced variants containing reads at that position. . • 1 – All variant-containing reads are on a single strand. **Note:** If **Maximum strand bias** is blank or absent in the wizard, this indicates that the corresponding scripting parameter has been turned off in the script. For more information and an example, search this help document for the scripting parameter **snp_maxStrandBias**.
Bases to mask at ends of reads	The specified number of bases from both the 5’ and 3’ ends of each read will be masked from the SNP caller and will not be considered during variant calling.
Bayesian-based removal of	Check this box to turn on *H-factor*, a Bayesian-based model that excludes heterozygous calls. If you want to view the MID column in the ArrayStar SNP Report, you must check this box. By default, the box is unchecked.

heterozygous indels	

Once you are finished, click **OK** to save changes and return to the Assembly Options screen, or **Cancel** to return without saving changes.

Filter based on “P not Ref”

In reference-guided workflows, “P not Ref” is the probability that the base does not match the reference. The P not Ref cutoff can be set using “hard” and/or “soft” filters. The following table describes the wizard parameters and their corresponding scripting commands that relate to P not Ref filtering.

To specify a “hard” filter:

In a “hard” filter, data not matching the criterion are permanently removed from the assembly. To set a hard filter, use the **P not ref** parameter located in the Variants tab of the Advanced Analysis Options dialog.

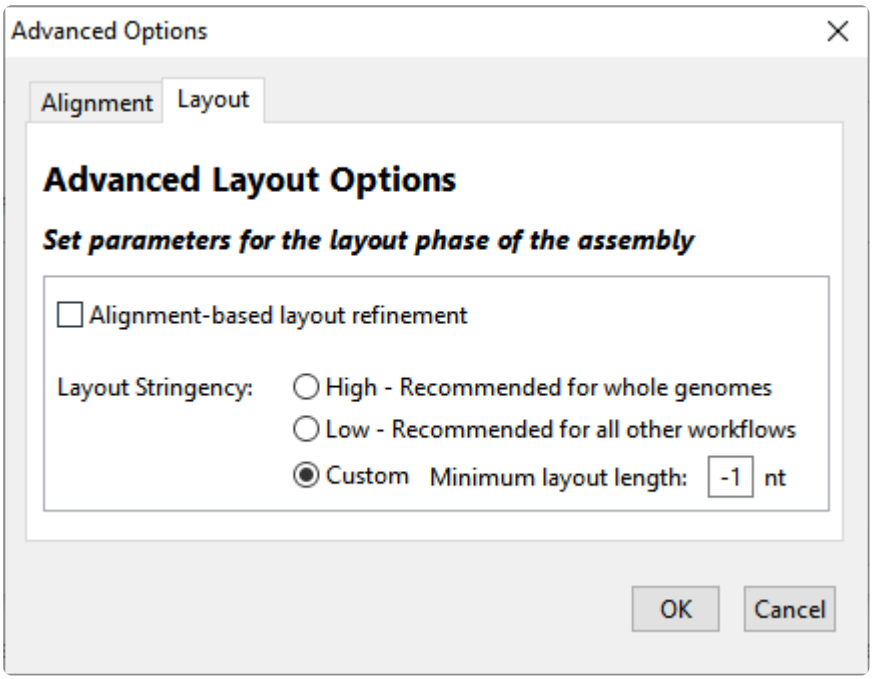
To specify a “soft” filter:

In a “soft” filter, data not matching the criterion are removed from the default display of the SeqMan SNP table. To set a soft filter:

- In the wizard – Use the **SNP filter stringency** radio buttons in the Analysis Options screen.
- In a manual script – Use the **computeSNP > snpFilter > pNotRefMinVal** parameter.

Layout tab

The Layout tab of the Analysis Options dialog is used to set parameters for the layout phase of the assembly. To access the tab from the Analysis Options screen, click the **Advanced Analysis Options** button then click on the **Layout** tab. The options available in this tab may vary depending on the workflow.



Parameter	Description
Alignment-based layout refinement	If you are following a whole-genome workflow, checking this button may improve the accuracy of the assembly.
Layout Stringency	Choose from the following: • **High** – Recommended for whole genomes. • **Low** – Recommended for all other workflows. • **Custom** – If you wish to enter the **Minimum layout length** manually. This is the minimum number of identical matching bases (from the mer analysis only) for a read to be included in the layout. It is specified by an integer, with a default of 50 nucleotides. For reads shorter than 100 bases, the setting is automatically adjusted to the mer size. Increasing this number decreases the false discovery rate (FDR) and true positive rate (TPR). Allowed values are 0-999. The default is -1 nt (nucleotides).

RNA-seq normalization methods

“Normalization” refers to the standardization of sequencing data on the basis of sequencing depth and gene length. Some versions of the Assembly Options”:#assembly-options screen allow you to specify a data normalization method, or to select **None**, in which case data will not be normalized. Some methods are described in the table below, while **DESeq2** and **edgeR** are discussed below the table.

Normalization method	Description
zRPKM	zRPKM (Krumm et al., 2012) is available for the CNV workflow and is calculated: zRPKM = ($RPKM_{exon,\ sample}$ – $Median_{exon}$) / $StdDev_{exon}$. This is the optimal normalization method for CNV projects with at least three groups/projects. Otherwise, we recommend using the RPK-CN normalization method.
RPK-CN	RPKM-CN (Krumm et al., 2012) is available for CNV experiments and is calculated as: RPKM-CN = RPKM / median of the exon’s RPKMs; where RPKM > 1. RPKM-CN calculates the copy number by taking the ratio of the RPKM of an exon versus the median RPKM of any exon in the experiment. The final number is a ratio (or log ratio) indicating a relative copy number with no units, since the units are cancelled out in the ratio. The variable M is a constant: the number of millions of mapped reads in the experiments. The ultimate meaning of the ratio comes from the different reads “R” and length “K” of each exon and the median. The constant, M, drops out of the equation and only affects scaling for initial filtering-out of low-coverage exons. We only recommend using RPKM-CN if you don’t have enough samples to provide a good standard deviation for each exon when using the zRPKM normalization method. Otherwise, zRPKM is the preferred method for the CNV workflow.
Quantile	Quantile is available for RNA-Seq workflows only. Quantile normalization adjusts all of the values in your project so that the distribution is the same across all of the experiments.
RMP	RPM (reads assigned per million mapped reads) is available for RNA-Seq, ChIP-Seq, and miRNA experiments, and is the only normalization method available for ChIP-Seq and miRNA experiments. When RPM is selected the signal values for each experiment will be divided by the total number of mapped reads divided by one million.
RPKM	RPKM (reads assigned per kilobase of target per million mapped reads) is available for RNA-Seq data. When RPKM is selected, the signal values for each experiment will be divided by the total bases of target sequence divided by one thousand; and the resulting number divided by the total number of mapped reads divided by one million.

DESeq2 and edgeR:

DESeq2 (Love et al. 2014) and edgeR (Robinson et al. 2010) are statistical packages in Bioconductor used to assess differential expression in RNA-Seq experiments.

DESeq2 or edgeR statistics for an assembly can be analyzed by opening the assembly in ArrayStar. For information about setting up an assembly suitable for analyzing DESeq2 or edgeR statistics in ArrayStar, see Create an assembly using DESeq2 or edgeR statistics.

Both methods require a control group to be specified, and both require replicate samples for each experimental condition and for the control. Note that when multiple experimental conditions are being considered, the same control group is used for multiple tests. The original P-values from the statistical tests are then adjusted using the Benjamini-Hochberg (1995) procedure.

Differences between DESeq2 and edgeR are shown in the table below:

Calculation	DESeq2	edgeR
Normalization method	Uses a median of ratios method to normalize read counts to account for sequencing depth and RNA composition. Provides two methods: regularized logarithm (*rlog*) and Variance Stabilizing Transformations (VST). DESeq2 does not attempt to account for transcript length since it is comparing counts between samples for the same gene and assumes the length does not change. This assumption holds true except in rare cases where the dominant transcript length changes between samples due to alternative splicing for example.	Uses "trimmed mean of M-values" (*TMM*) (Robinson & Oshlack, 2010|topic=Research References). The TMM normalized read count can be viewed in the ArrayStar tables, where counts are represented as log2(counts-per-million-reads). Normalized counts generated by a different method, *RLE*, are also available within ArrayStar but these values are not used for the actual statistical tests. RLE is similar to the RLOG normalization method used by DESEq2.
Statistical tests for differential expression	DESeq2 uses raw counts, rather than normalized count data, and models the normalization to fit the counts within a Generalized Linear Model (GLM) of the negative binomial family with a logarithmic link. Statistical tests are then performed to assess differential expression, if any.	Data are normalized to account for sample size differences and variance among samples. The normalized count data are used to estimate per-gene fold changes and to perform statistical tests of whether each gene is likely to be differentially expressed. EdgeR uses an exact test under a negative binomial distribution (Robinson and Smyth, 2008|topic=Research References). The statistical test is related to Fisher's exact test, though Fisher uses a different distribution.
Data reporting	In ArrayStar, the *rlog* values are used by default in the scatter plot and for clustering. VST values are	In ArrayStar, the $\log_2$(CPM) values calculated using TMM are used by

method	displayed as Gene Table data columns.	default in the scatter plot. In the Gene Table, values for fold change compared to the control are represented as log(fold change).

ChIP-seq peak detection methods

If you are following the ChIP-seq workflow, the Analysis Options screen allows you to specify a peak detection method. Available methods are described in the table below:

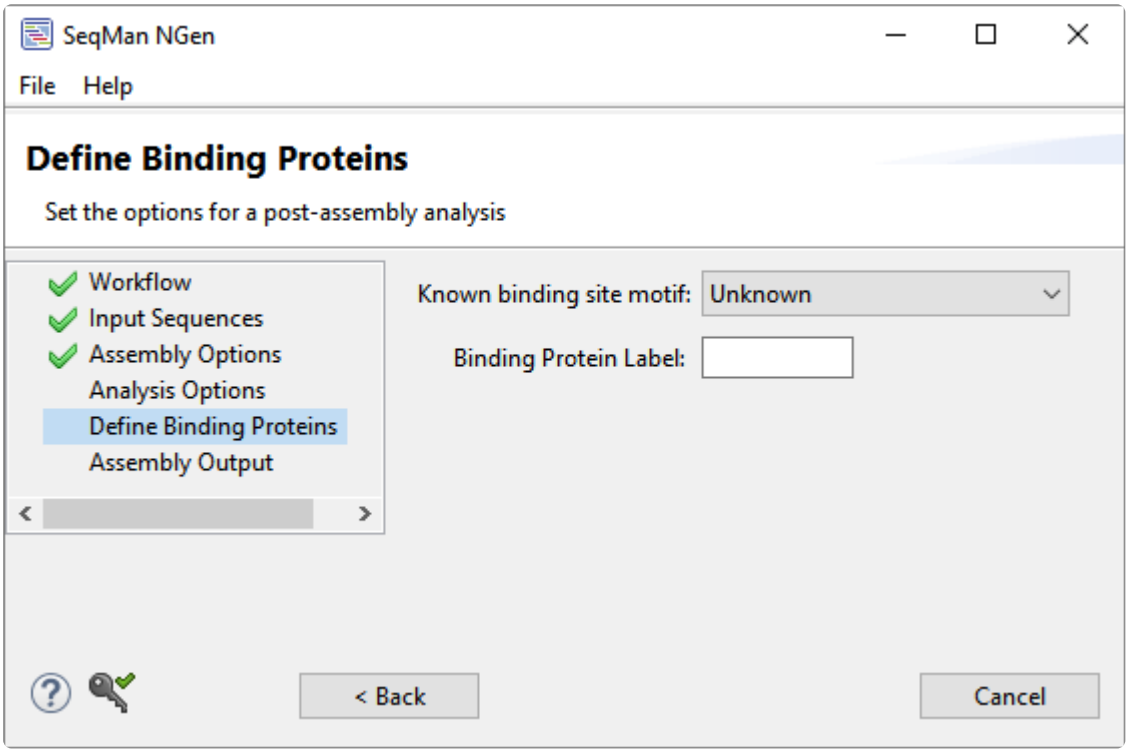
Name	Description
MACS	The **MACS** Peak Finder is based on the peak detection algorithm (Zhang, et al., 2008). This is a model-based algorithm that expects there to be paired peaks of reads on either side of a true binding site. Paired flanking peaks reflect the fact that only the 5' ends of immuno-precipitated fragments are usually sequenced. Because of this, the majority of the locations of sequences associated with peaks don't correspond to the location of the binding site. The MACS algorithm attempts to build a model of the distance between these peaks and takes this distance into account to shift reads forwards or backwards, resulting in a peak centered over the true binding site. The MACS Peak Detection algorithm reports the number of reads within a peak as the signal value for that peak. It also calculates a P-value based on the distribution of reads near a peak region to try to compensate for uneven background noise across the genome. When present, control data are used to filter the peaks that are called and to assign each peak an FDR score which is the false discovery rate likelihood that the peak is not valid.
ERANGE2	The **ERANGE2** Peak Finder (Johnson et al., 2007) is a simple "sliding window" peak detection algorithm that looks for a specified number of reads within a window of a specified length. If a peak is found, it is extended as long as there are reads within the window width. If control data are present, it can be used to disqualify any peaks that do not have a minimum fold enrichment over the same region in the control data.
ERANGE3	The **ERANGE3** Peak Finder is based on the ERANGE 3.1 Algorithm for ChIP-Seq and RNA-Seq Analysis (see Mortazavi et al., 2008). This peak detection algorithm calculates peaks in a normalized reads-per-million space. Features of this algorithm include simple read shifting and repeat read handling. This algorithm also considers the directionality of reads when calling peaks.

Define Binding Proteins

This wizard screen only appears for reference-guided ChIP-seq workflows. The Define Binding Protein screen allows you to define binding sites for your experiment.

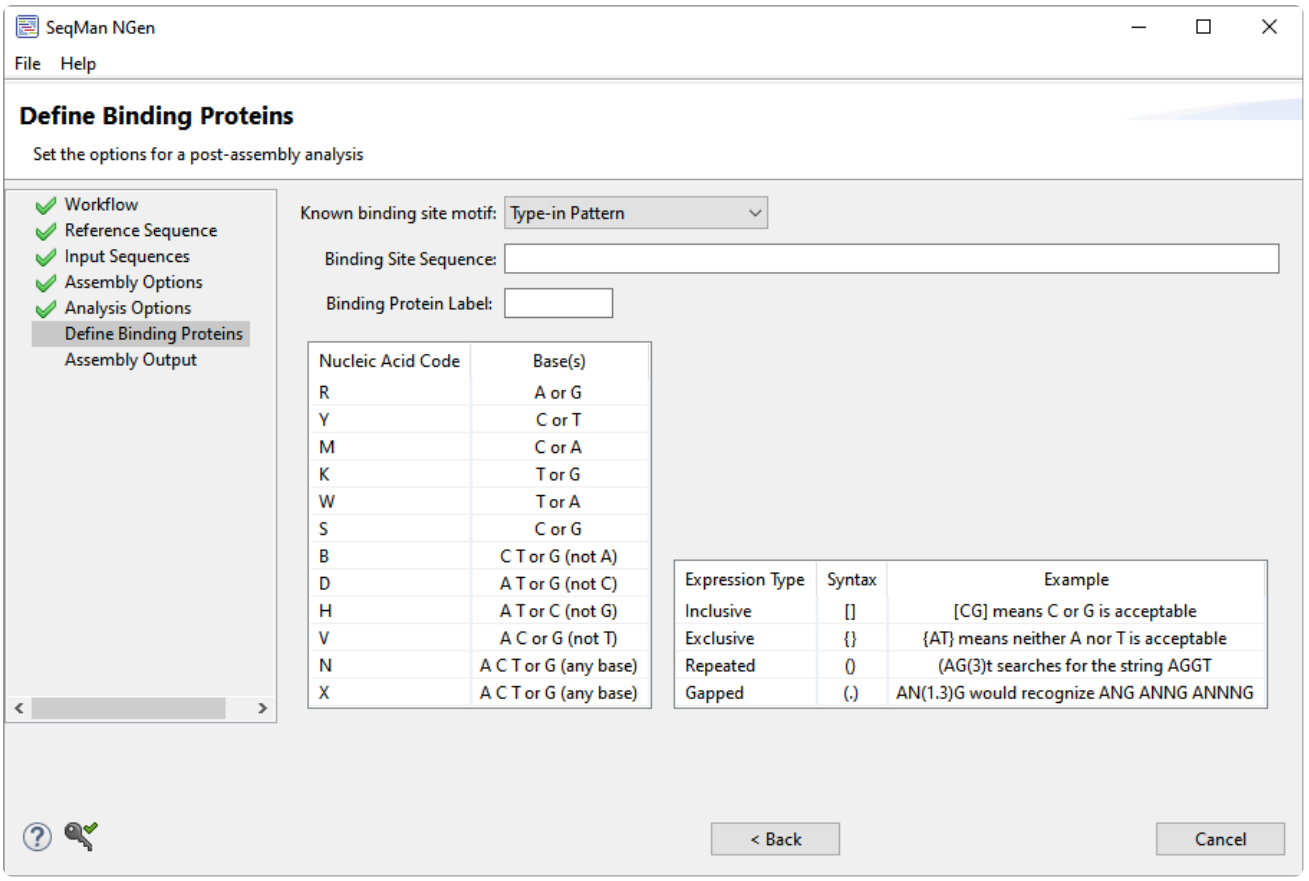
Start by making a selection from the **Known binding site motif** drop-down menu. The Define Binding Proteins screen will change and offer different options based on your selection.

Unknown

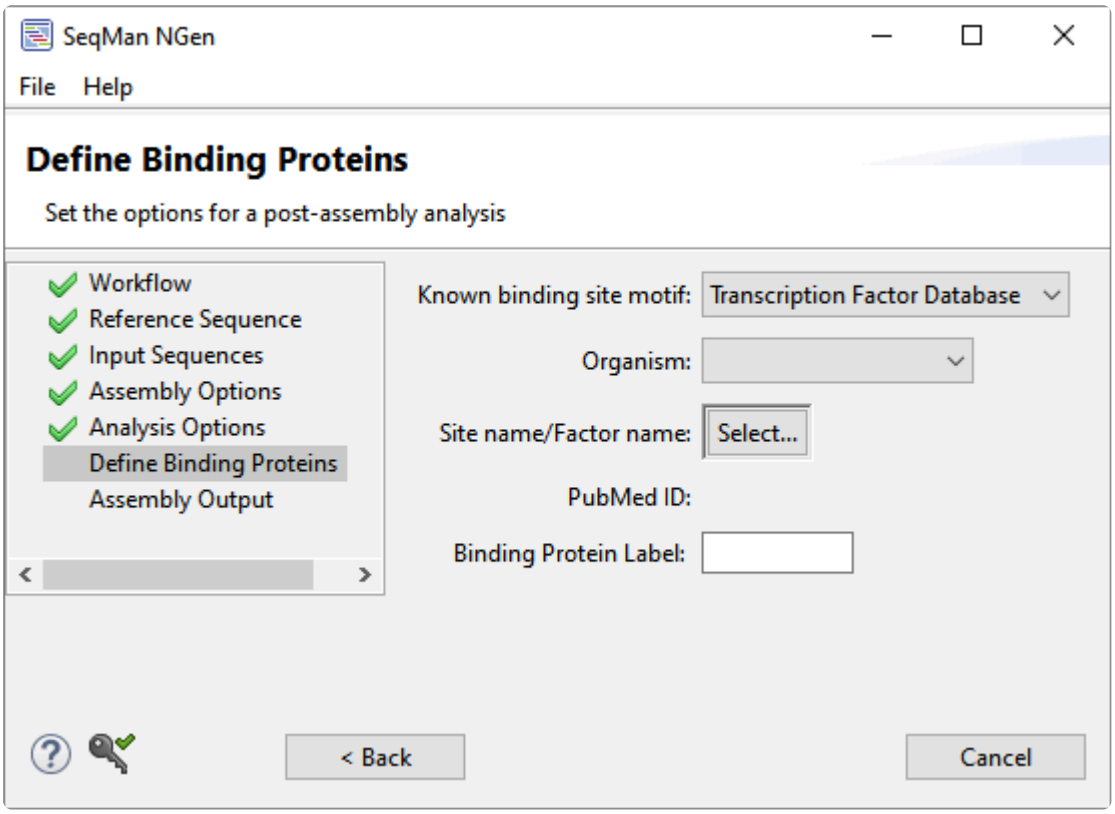


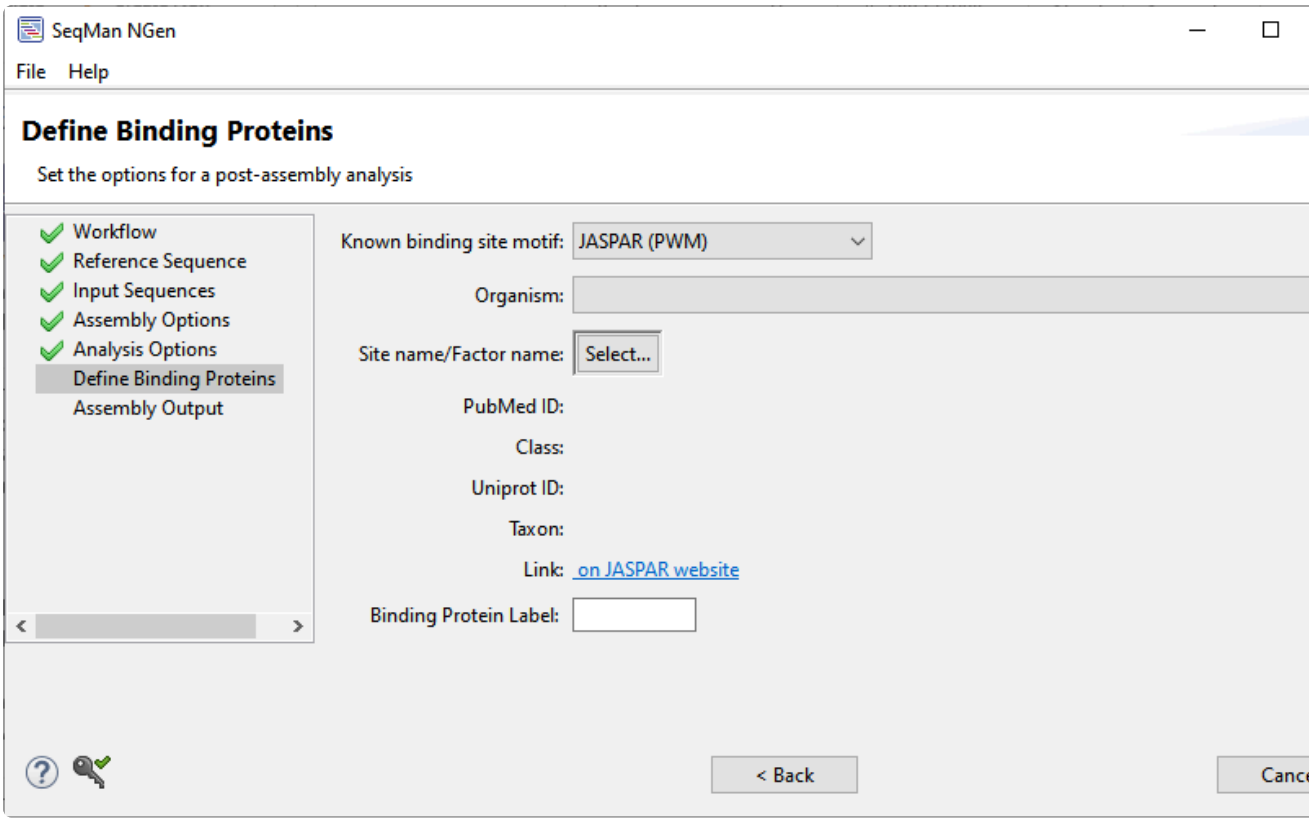
Choose this option to use the whole genome to find peaks. Type a name into the **Binding Protein Label** box.

Type-in pattern



Select this option to specify your own binding site pattern for your binding protein. Type a pattern into the **Binding Site Sequence** text box and a name into the **Binding Protein Label** text box. SeqMan NGen recognizes IUPAC nucleic acid and regular expression syntax for these patterns. A key to the syntax is provided within the wizard screen.

Transcription Factor database	 Choose this option to select a binding site pattern from DNASTAR's transcription factor database, which is prokaryotic organisms only. Select the **Organism** from the drop-down menu. Click the **Select** button to ch the site/factor name from a list. Make a selection and choose **OK**. The **PubMed ID** field will be filled in automatically. Finally, type a name into the **Binding Protein Label** text box.



JASPAR (PWM)

Select this option if you want to use the JASPAR position weight matrix to locate binding sites for eukaryo organisms. If you choose this option, SeqMan NGen will calculate the log-odds for each sequence given t selected matrix. The score for a single character at a particular position in the matrix is equal to the $\log_2$ o likelihood of seeing that character at that position in the data used to generate the matrix divided by the background likelihood of seeing that character at that position.

For example, if the matrix is derived from 80 sequences and in 70 of those sequences there is an “A” in po 1, the log odds score of seeing the character “A” in position 1 is $\log_2((70/80)/(20/80)) = 1.80$. If a “C” occur time in position 1 of the training sequences, the log odds score of seeing the character “C” in position 1 is $\log_2((1/80)/(20/80)) = -4.32$. To get the log odds score for the whole sequence, SeqMan NGen sums the lo odds scores of each character in the sequence.

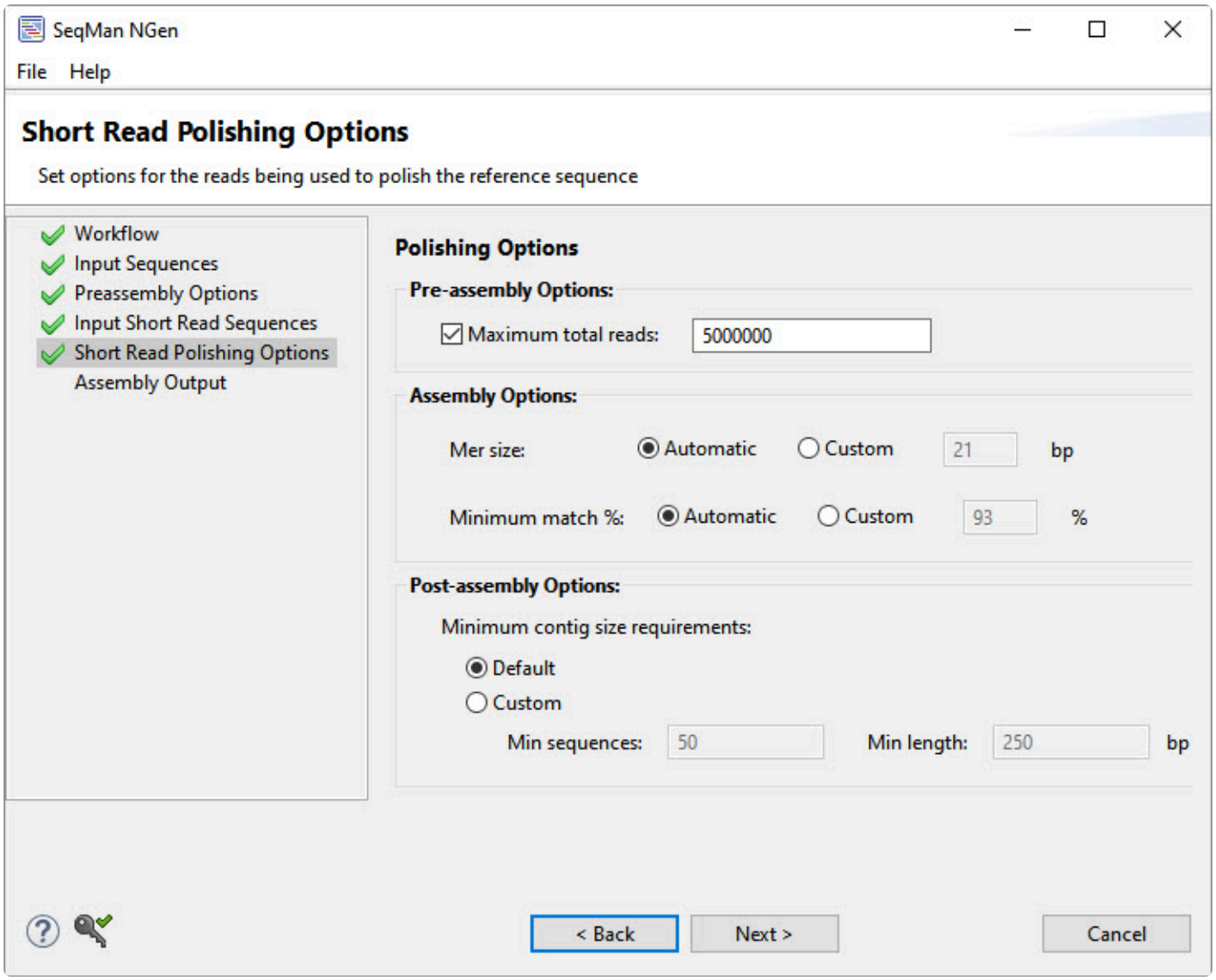
A sequence is considered to "match" the matrix if its score is greater than or equal to the specified Thresh By default the threshold value is half of the average of the log-odds scores of sequences that were used to the pattern. You can increase the threshold for more stringency or decrease it for more matches. Once yo initiate sequence assembly, all detected peaks will be scanned for the presence of sites that pass the JAS scoring threshold.

Once you have chosen this option, select the **Organism** from the drop-down menu. Click the **Select** butto choose the site/factor name from a list. Make a selection and choose **OK**. The remaining fields will be fille automatically. If you wish to view online entries for the selected site/factor, click on the corresponding links

Click **Next >** to proceed to the next wizard screen or **< Back** to return to the previous screen.

Refinement Options

If, in the Workflow screen, you selected the **De novo Genome assembly and editing** and choose the "NGS-based" workflow **Hybrid reference-guided/de novo genome assembly**, the Refinement Options screen will appear later in the wizard.



.

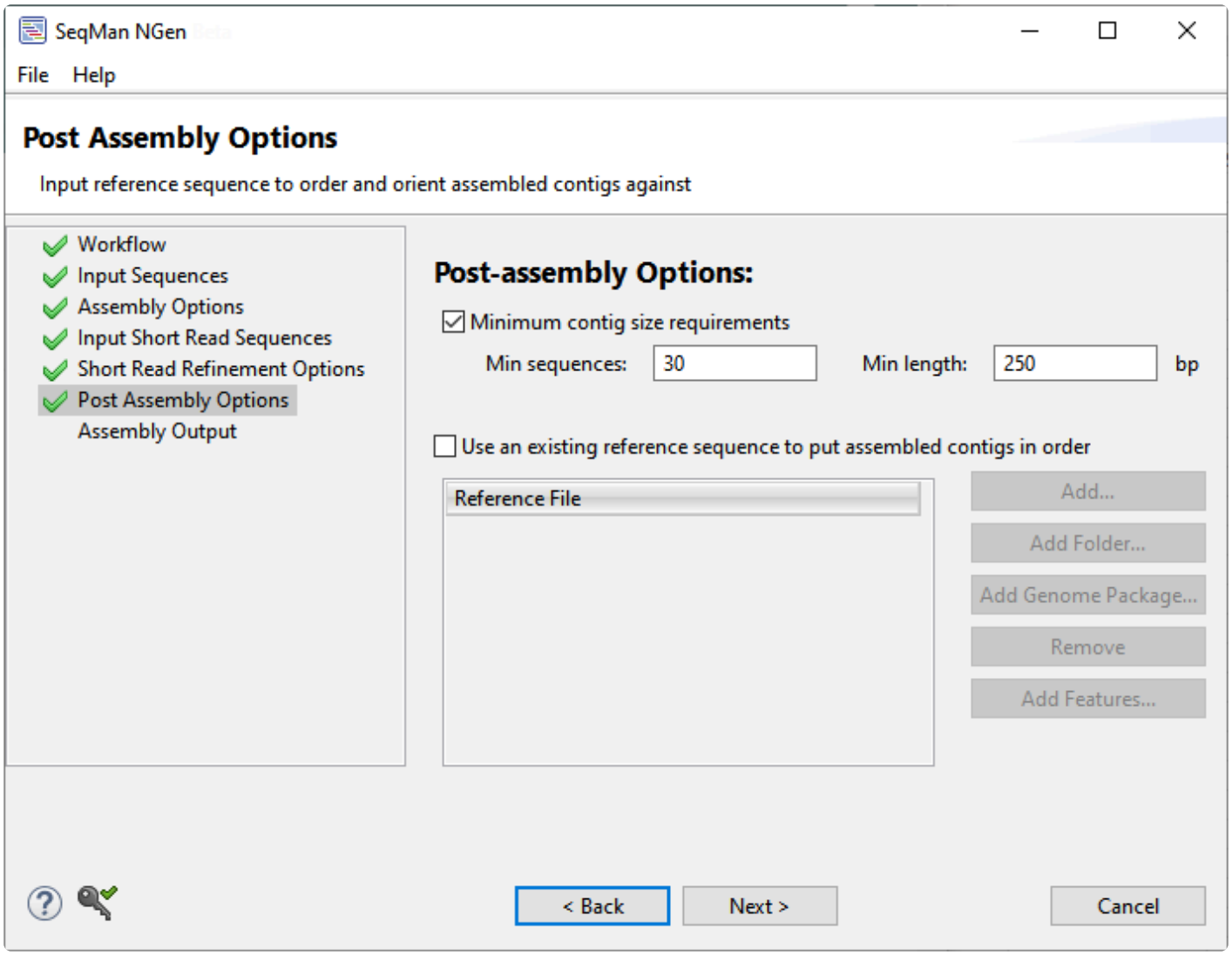
Category	Options and Descriptions
Pre-assembly Options	Check **Maximum total reads** and enter a value if you wish to limit the read depth. Utilizing this option can make the assembly proceed faster.
Assembly Options	**Mer size** is the minimum length of a mer (overlapping region of a fragment read), in bases, required to be considered a match when arranging reads into contigs. Mer size information is used to identify matches during the assembly layout phase. The default mer size is determined by the

	selected read technology and is shown in the window. For more information, see How mer tags are chosen. • **Automatic** – Select this button to automatically set the size based on assembly type and sequencing technology. • **Custom** – Select this button to choose the size yourself. You must enter the desired number of base pairs in the field at right. Lowering the mer size increases the sensitivity of finding matches, but also increases the likelihood of finding spurious matches in addition to the correct match. Lowering the mer size can also greatly increase the requirements for storing intermediate and temporary files with large projects.
	Minimum match % specifies the minimum percentage of matches in an overlap required to join two sequences in the same contig. SeqMan NGen determines the percentage to use based on the sequencing technology you specified in the Assembly Options dialog. For more information, see Calculation of match percentage. • **Automatic** – Select this button to automatically set the percentage based on assembly type and sequencing technology. • **Custom** – Select this button to designate the percentage yourself. You must enter a number in the field at right.
Post-assembly Options	Check **Minimum contig size requirements** and type values for one or both of the following to remove assembled, untemplated contigs that do not meet minimum thresholds. This can lead to a desirable decrease in project size. • **Min sequences** – Disassembles any untemplated contigs with fewer than the specified number of sequences. This option affects only untemplated contigs. No templated contigs are removed. • **Min length** – Using this option disassembles any untemplated contigs shorter than the specified length.This option affects only untemplated contigs. No templated contigs are removed.

Click **Next >** to proceed to the next wizard screen or **< Back** to return to the previous screen.

Post Assembly Options

The Post Assembly Options screen lets you enter a reference sequence against which to order and orient assembled contigs. You can also set requirements for minimum contig size, if you would like to do so.



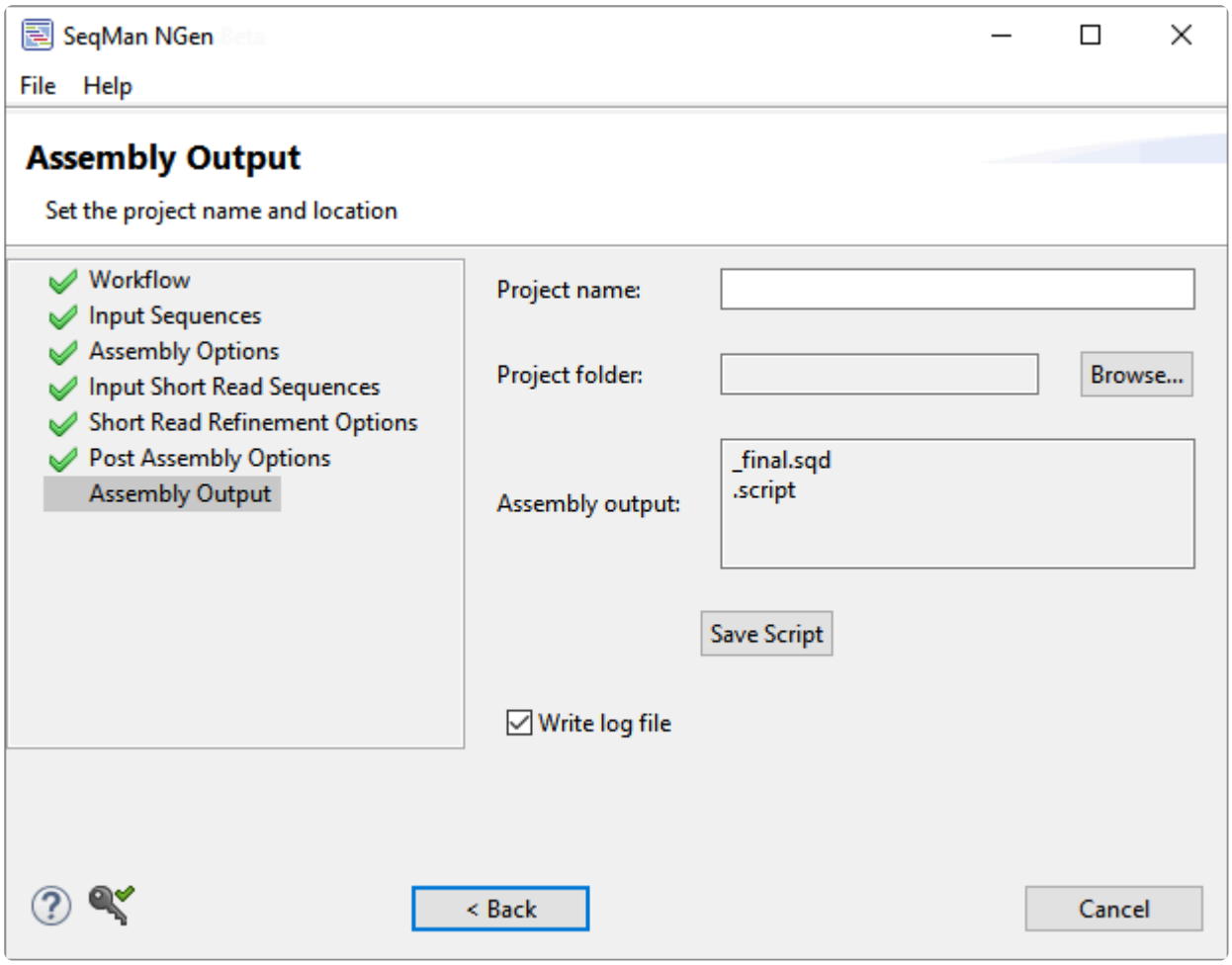
- Check the box if you wish to specify **Minimum contig size requirements**. If you check the box, you will need to enter the desired values for:
    - **Min(imum) sequences** – The minimum allowed number of sequences that make up a valid contig.
    - **Min(imum) length** -The size of the minimum allowed contig length for a contig to be included in the assembly.
- Check the box if you wish to **Use an existing reference sequence to put assembled contigs in order**. If you check this box, you will need to add at least one reference sequence using the various **Add** buttons in the lower half of the screen. To learn how to do this, see:

- Add a reference sequence from your computer or the Cloud
- Add a genome template from DNASTAR
- Remove a sequence from the list

Click **Next >** to proceed to the next wizard screen or **< Back** to return to the previous screen.

Assembly Output

When the Assembly Output screen appears in the wizard, you must select a name and location for your project before proceeding further.



- **Project name** – Enter a name for all output files, including the finished assembly. The finished assembly will be saved in BAM format.

- **Project folder** – Use the **Browse** button to select a location for your assembly output files.

    - For non-Cloud assemblies, **Browse** launches your file explorer. Navigate to the desired location and then click **Open** to exit. The required disk space may range from 1 GB to 5 TB, depending on a variety of factors. See our technical requirements page for more information.

      IMPORTANT: Never save the assembly output files directly to the desktop, as the many intermediate files and folders created during assembly may hamper or prevent further computer operations. However, files may be saved to a folder on the desktop.

      IMPORTANT: You must save to a writable location. For example, the **User** folder on Windows

is not writeable.

    - For Cloud assemblies, **Browse** opens the DNASTAR Cloud Data Drive and displays your files on the DNASTAR Cloud. Navigate to the desired location and highlight the target folder, then click the green check mark () to exit from the DNASTAR Cloud Data Drive.

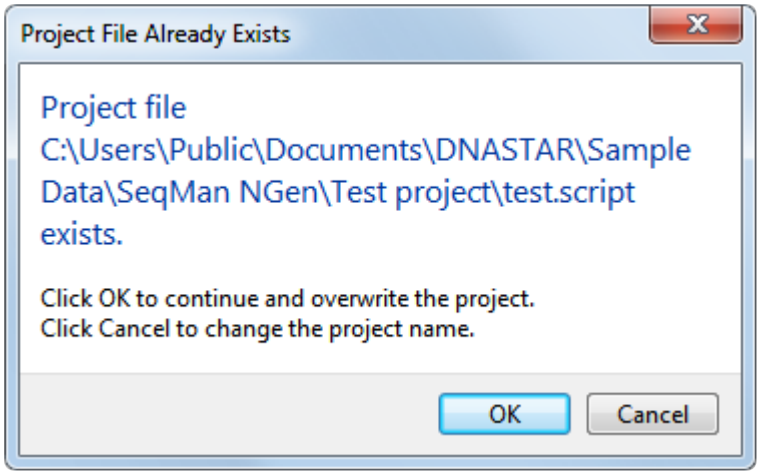
- **Assembly output** displays the output file names and extensions based on current workflow and other selections. This is for information only, and cannot be edited.

- **Save Script** – Press this button if you wish to save your project and convert your wizard choices into a SeqMan NGen assembly script (*.script*) prior to assembly. This button is not available for Cloud assemblies. The resulting assembly script is an editable text file that can be modified and re-run if desired. Note that if you use **Save Script** after having checked the **Run as separate assemblies** box in the Input Sequences screen, a set of three separate scripts will be saved for the project. If you save one or more of these scripts to a location other than the main project folder, any attempt to run the assemblies from the SeqMan NGen project script will fail. Moving the projects back to the main project folder will allow assembly to proceed.

- **Write log file** – To create and save a text-formatted log file that can be used for troubleshooting any issues with the assembly. The file is saved in the same project folder as the *.assembly* and is assigned the name *.log*. The log includes the SeqMan NGen script for the project, followed by a list of steps that were performed and their outcomes. This log is especially useful in troubleshooting an assembly that will not complete.

```
Preprocessing complete.

Mapping data

Mapping data to genes in project...

Loading data names and annotations

Loading signal data

Loading exon repeat_distrib_percent values for Ecoli templated project

Finishing
```

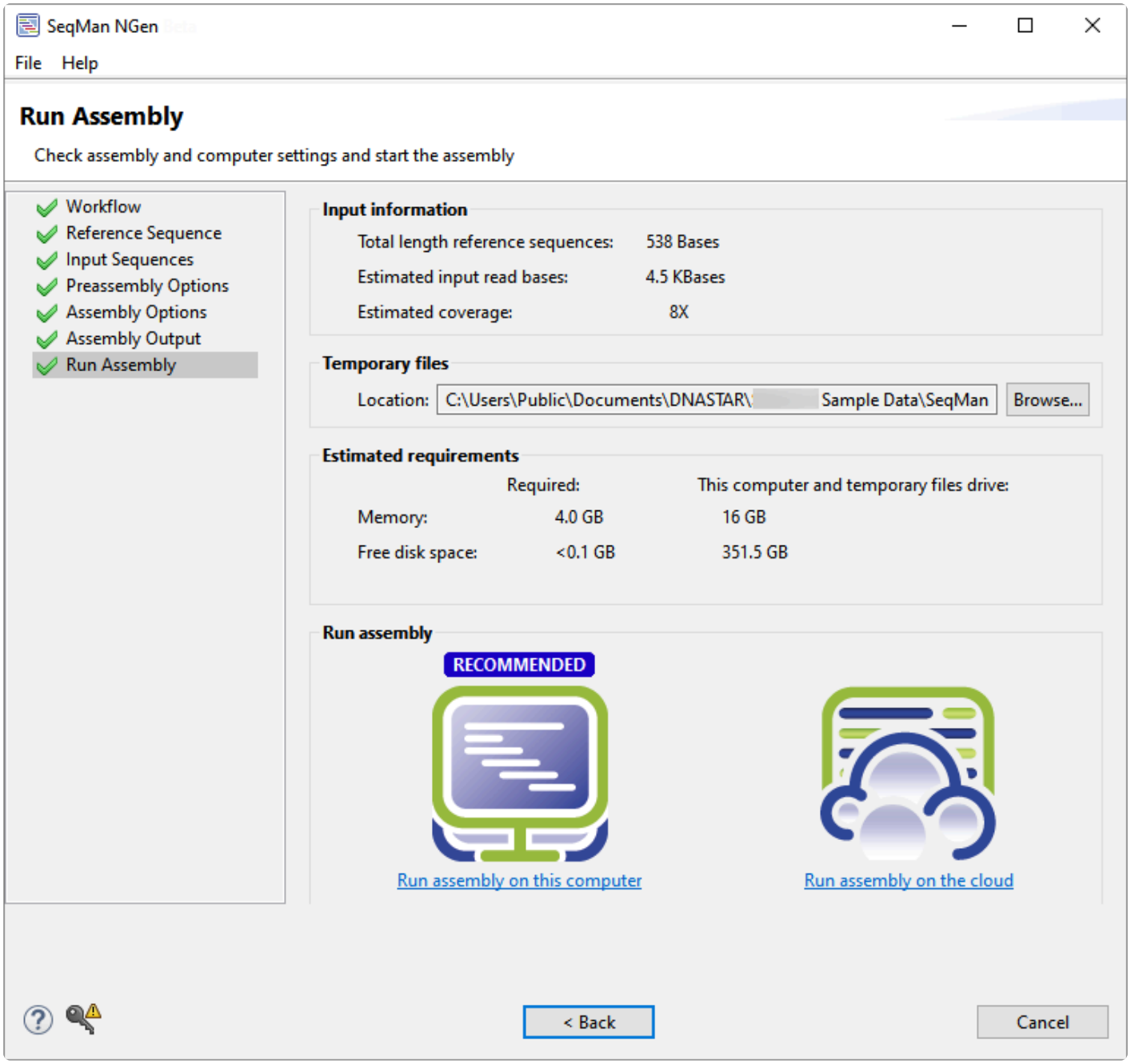
Click **Next >** to proceed to the next wizard screen or **< Back** to return to the previous screen. If you choose a name that already exists in the chosen location, clicking **Next >** will cause the following warning.



Click **OK** to continue and over-write the earlier project; or **Cancel** to return to the wizard screen, where you may change the project name and/or location.

Run Assembly Project

The Run Assembly Project screen follows the Assembly Output screen and prompts you to specify a name for the project and a location in which to save temporary files. You can also review system memory information that will help you make an educated decision whether to assemble on your local computer or on the cloud.



- The “Input information” section shows a quick overview regarding the size of the planned assembly, including the total length of the reference sequences, estimated size of the read files, and estimate coverage area.

- Under “Temporary files,” the **Location** box shows any location previously specified for temporary files created during the run. If you have not previously specified a location, the box will be blank. To specify or change the location, use the **Browse** button or open your computer’s file explorer and drag and drop the directory in the **Temporary file location** box. SeqMan NGen will remember and use the temporary file location for future assemblies. Here are some tips regarding temporary files:

    - If possible, use an external hard drive as the temporary file location.

    - Never save the assembly output files or temporary files directly to the desktop, as the many intermediate files and folders created during assembly may hamper or prevent further computer operations. However, files may be saved to a folder on the desktop.

    - By default, most temporary files are deleted when the assembly is complete. Other files (e.g., [template_name].FasInfo.sqlite and [template_name].mer) may remain in the temporary file location in order to facilitate efficient reassembly of data in the future.

    - You do not need to specify a temporary file location when following a *de novo* or special reference-guided workflow.

- The “Estimated requirements” section consists of two columns. The **Estimate** column shows the estimated **Memory** and **Free disk space** required for the current assembly to run without failure. The **This computer and temporary files** column shows the amount of memory and free disk space on your local computer. If either value on the left is larger than its equivalent on the right, we strongly recommend you run the assembly on the cloud. Unless you have an extremely powerful computer, most non-bacterial assemblies should be run on the cloud.

- In the “Run assembly* section, click the link to **Run assembly on this computer** or **Run assembly on the cloud**. Base your decision on the “Estimated requirements” section above. If you choose to run on the cloud and are not already logged in, you will be prompted to log in to Cloud Assemblies.

    - If you choose **Run assembly on this computer**, you must wait for one assembly to finish before beginning another. Note that *de novo* projects require ample amounts of RAM, while most reference-guided assemblies require large amounts of CPU and free hard drive space. Unless you have an unusually powerful computer, local assembly is best reserved for small, reference-guided assembly projects.

    - If you choose **Run assembly on the cloud**, you can perform any number of SeqMan NGen assemblies simultaneously, without using up your computer’s resources. The assembly takes place on a powerful Amazon cloud computer, and output data are stored securely on the cloud. To try Cloud Assembly for free*, contact your DNASTAR representative to request free trial access and a secure data storage folder.Matt 12/15: Upload happens, user sees only one line. Can open CDD and navigate to folder to see details. After assembly, finished stuff stays on Cloud and also goes to place you specified on local computer.

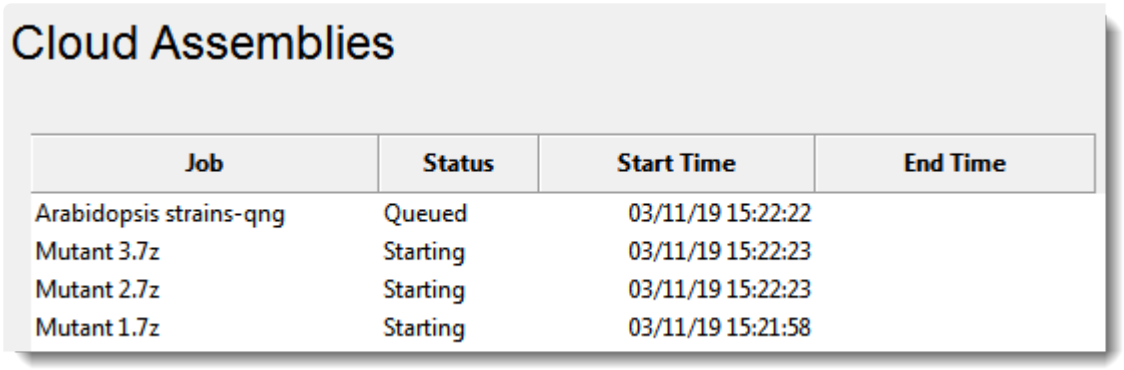
If you want to make changes to a screen before proceeding, click the screen name on the left of the wizard or choose **< Back** to return to the previous screen.

Monitor the progress of a Cloud Assembly

To monitor the progress of a Cloud Assembly:

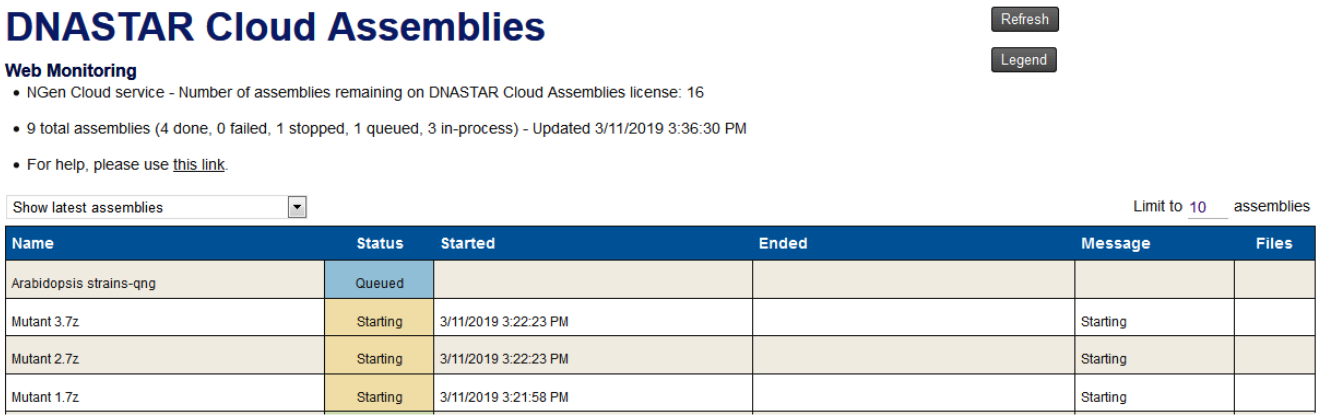
Once you press the **Start Assembly** button in the SeqMan NGen wizard, the assembly begins. The Cloud Assembly tutorials in this User Guide each take 30 minutes to several hours to complete. You can monitor the progress using either of two methods:

- From the wizard: – After pressing **Start Assembly**, wait for the **Next** button to become active. Press **Next** to go to the Cloud Assemblies page, where you can monitor the progress of the assembly.



  Close the wizard at any time by pressing **Finish**. After pressing **Finish**, you will need to use the other monitoring method (next bullet point) if you wish to continue monitoring the assembly.

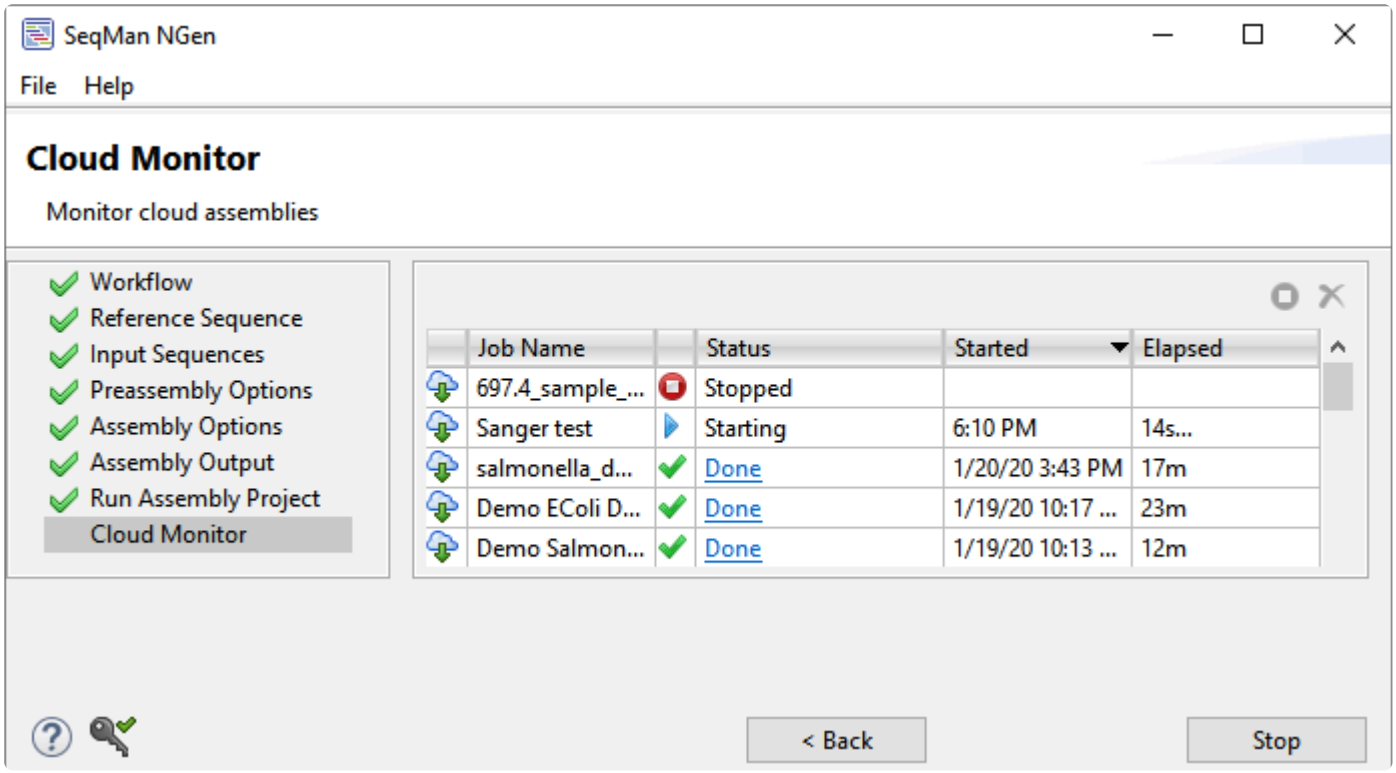
- From the DNASTAR website: – Open the Web Monitor, entering your DNASTAR login credentials if prompted to do so. See the “Web Monitor help“https://www.dnastar.com/Web_Monitor_Help/#!Documents/usethewebmonitor.htm to learn about features in this browser window.



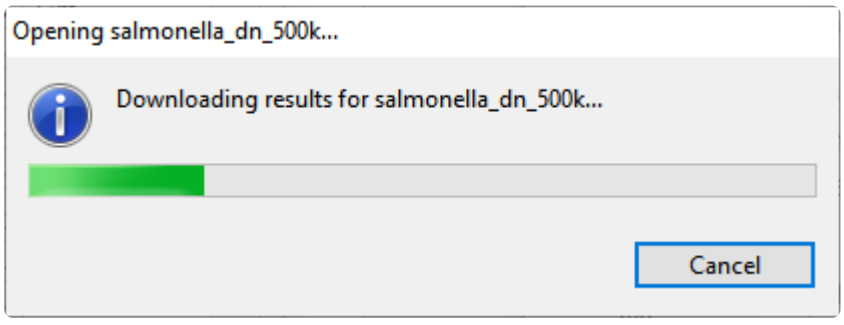
Once an assembly is **Done**, you can download the assembly results for analysis with SeqMan Pro, ArrayStar, GenVision Pro, etc.

Cloud Monitor

After pressing the **Run assembly on the Cloud** link in the Run Assembly screen, you will be taken to the Cloud Monitor screen. This is where you monitor in-progress and completed Cloud Assemblies. You can also get to this screen by launching SeqMan NGen and selecting **Monitor cloud assembly** from the Welcome screen.



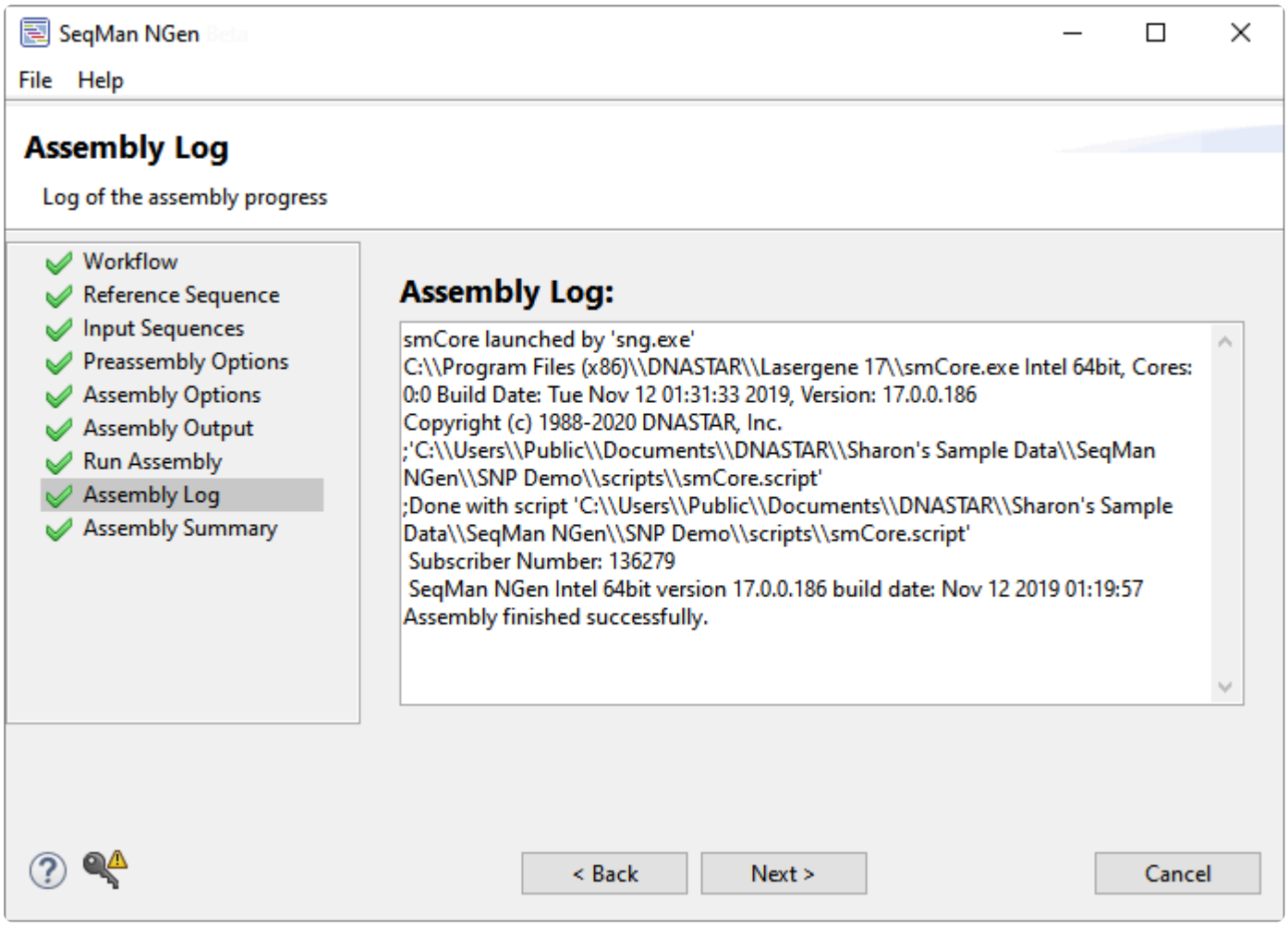
Once an assembly has finished successfully, the **Status** for that run is shown with the hyperlinked word "Done." Click on "Done" to download the result files. A progress bar may appear:



Once the download is complete, the results files will be displayed in your file explorer. You can then open these in a suitable application for downstream analysis, such as SeqMan Ultra, GenVision Pro or ArrayStar.

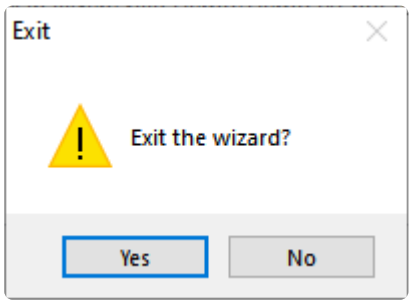
Assembly Log

After pressing the **Run assembly on this computer** link from the Run Assembly screen, the Assembly Log opens to show the status of the assembly.



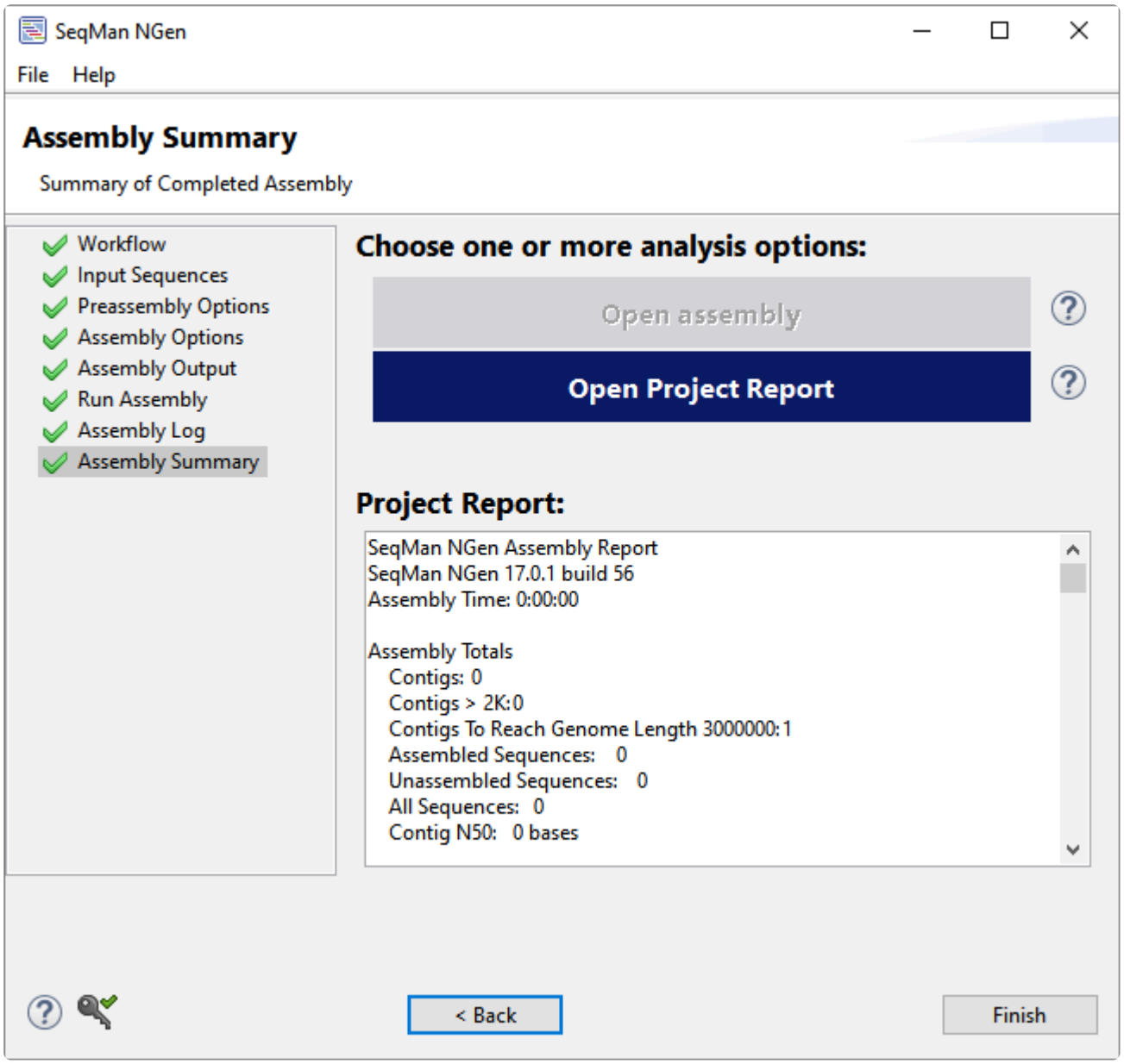
Once the assembly runs to completion without failure, the following text will be displayed: “Assembly finished successfully.”

Click **Next >** to proceed to the Assembly Summary screen or **< Finish** to exit from SeqMan NGen. In the latter case, the following confirmation popup will appear:



Assembly Summary

After a local assembly has finished in the Assembly Log, clicking **Next >** takes you to the Assembly Summary screen.



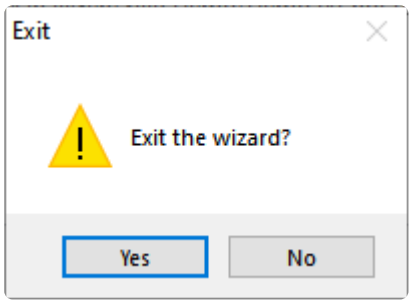
If assembly failed, the dialog displays the message “Assembly failed. No report available.” Otherwise, you will see the Project Report information in the body of the screen.

Under “Choose one or more analysis options,” select the button of interest. The availability of a button depends on the workflow and/or operating system and will include a subset of the following:

Button name	Result

Open assembly	Launches the results in SeqMan Ultra. If multiple *.assembly* projects were created, you will be prompted to choose the names of those you wish to open. If you are following a variant workflow, we recommend downstream analysis using SeqMan Ultra's Variant view or searching for **Variants** in the search bar of the Analysis view.
Compare variants / differential gene expression between samples	Launches the results in ArrayStar. If you are using Macintosh, a warning message will appear asking you to open the assembly on Windows. Move the completed assembly to a Windows computer (or Macintosh running Parallels) in order to view the assembly in ArrayStar.
View peaks	This option is available for ChIP-seq only and opens the results in GenVision Pro.
Open Project Report	Opens the Project Report in your default text editor. For other ways to view the report and to learn about the report contents, see View the Project Report.
View and Compare Sashimi Plots	This option is available for RNA-seq workflows only and opens the completed assembly in GenVision Pro.

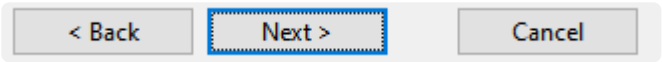
Click **< Back** to return to the previous screen or **Finish** to exit from SeqMan NGen. In most cases, you will choose **Finish**. The following confirmation popup will appear:



Navigate between wizard screens

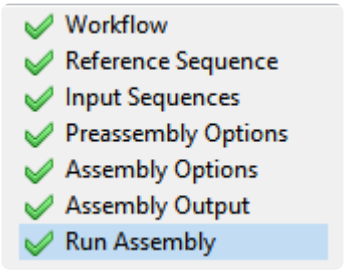
There are two different ways to navigate between screens in the SeqMan NGen wizard.

- Whether or not you have yet been to a screen, you can navigate using the buttons at the bottom of each dialog.



  - Click the **Help** button (Win) or the question mark icon (Mac) to launch the user's guide topic for the current panel.
  - Click **< Back** and **Next >** to navigate to the previous or next panel.
  - Click **Quit** to exit SeqMan NGen.

- If you have already visited a particular screen, you can return to it by clicking its name from the menu at the left of the SeqMan NGen wizard. The menu contains a list of wizard screens that you have visited so far. If you return to a previous screen and make a selection that changes the workflow type, subsequent screens that are not part of the new workflow will be removed from the list



Add and remove files in the wizard

Many wizard screens allow or require you to specify files that will be used in the assembly. There are several ways to add these files, each discussed in a separate topic:

- Add sequences from your computer or the cloud
- Add a genome template from DNASTAR
- Add a genome template from NCBI

You can also remove a sequence from the list if you change your mind.

Add sequences from your computer or the cloud

Some wizard screens require you to add at least one sequence or genome template before proceeding. In these cases, the “Next” button only appears after the item has been added.

- To add one or more sequences from your computer (i.e. “local” sequences), press **Add** or **Add Folder**. Navigate to and select the desired file(s) or folder of files, then click **Open**.

  > **Note:** If your reference sequence is a GFF feature file, with or without a separate FASTA file, we recommend adding the GFF file first. If the FASTA is built into the GFF file, you don’t need to do anything further. If the FASTA file is separate, SeqMan NGen will automatically prompt you to add the accompanying FASTA file.

- To add one or more sequences from the DNASTAR Cloud Data Drive, press **Add from Cloud** or **Add Folder from Cloud**. Navigate to and select the desired file(s) or folder of files, then click the green check mark ()

  The first time you press a button associated with Cloud Assemblies (or the Cloud Data Drive), you may be prompted to enter your **Email** and **Password**. Enter the information that you use when logging in to the DNASTAR website. Check the **Save password** box if you would like SeqMan NGen to save this information so you do not need to retype it in the future. If you are a licensed Cloud user, a message will display the number of assemblies remaining on your license. Click **OK** to close the message and continue to the next screen. If you are *not* licensed, a different message will appear, asking you to purchase a Cloud license through DNASTAR Support (support@dnastar.com).

  The following brief video shows how easy it is to upload your data and then perform an assembly on the cloud:

Add a genome template from DNASTAR

If your workflow includes the Reference Sequence or Set Contaminant screen, you must add at least one reference sequence, biome genome, or genome template before proceeding.

Curated and up-to-date DNASTAR genome template packages are available for common model organisms. Each template package contains template sequence, annotations, and database linking information. If you wish to use DNASTAR's database association features (e.g., dbSNP, GERP, and COSMIC), you must input one of these genome packages in the appropriate screen for your workflow.

To add a DNASTAR genome template from your local computer:

If you have not previously downloaded the genome package to your local computer, start with step 1. Otherwise, start at step 5.

1. Press **Download Genome Package**. If the button is disabled, see the first and last notes below.
2. Select a package from the list.
3. Click **Select**. and choose a save location. The package is saved with the extension *.genometemplate*.
4. When the download finishes, click **OK**.
5. Press **Add Genome Package**.
6. Navigate to the location where you saved the automatically downloaded (or manually downloaded & extracted) package, and click **Open**.

To add a DNASTAR genome template from the Cloud:

1. Press **Add Genome Package**.
2. Select a package from the list and click **Select**.

After creating an assembly with a genome template, you can access dbSNP information as described in the following brief video:

Notes:

- The **Add Genome Package** and **Download Genome Package** buttons are disabled if you have already added files using the **Add** or **Add Folder** buttons.

- SeqMan NGen can read and produce output using a variety of common chromosome naming conventions, including “chr1” and “ch1,” as well as Arabic and Roman numerals. Chromosome names are captured from genome template packages and used to assign contig IDs to entries from BED, VCF and manifest files.

- Chromosome names are captured from genome template packages and used to assign contig IDs to entries from BED and Manifest files.

- In the human genome template packages provided by DNASTAR, the “unlocated contig” is actually a concatenated, multi-sequence contig containing the alternate loci sequences. These loci are used for large regions where the human population contains variation so divergent that it cannot be adequately described by simple substitutions and small indels. Examples of these regions include the LRC/KIR complex on chromosome 19 and the MHC on chromosome 6.

- If you are performing a local assembly, there are some circumstances in which it is necessary to download and extract the package manually prior to using the wizard. To do this, go to DNASTAR’s Genome Template Packages web page and download a template package with the genome of interest. Downloaded genome packages are saved on your computer as ZIP files, and must be extracted prior to use. On Macintosh, double-click on the ZIP file. The files will be automatically extracted via the Archive Utility. On Windows, use any archive utility to extract the files. One method is to double-click on the ZIP file. In the ensuing Explorer window, click **Extract all files** from the top left. Choose a location for the files and select **Extract**.

Add a genome template from NCBI

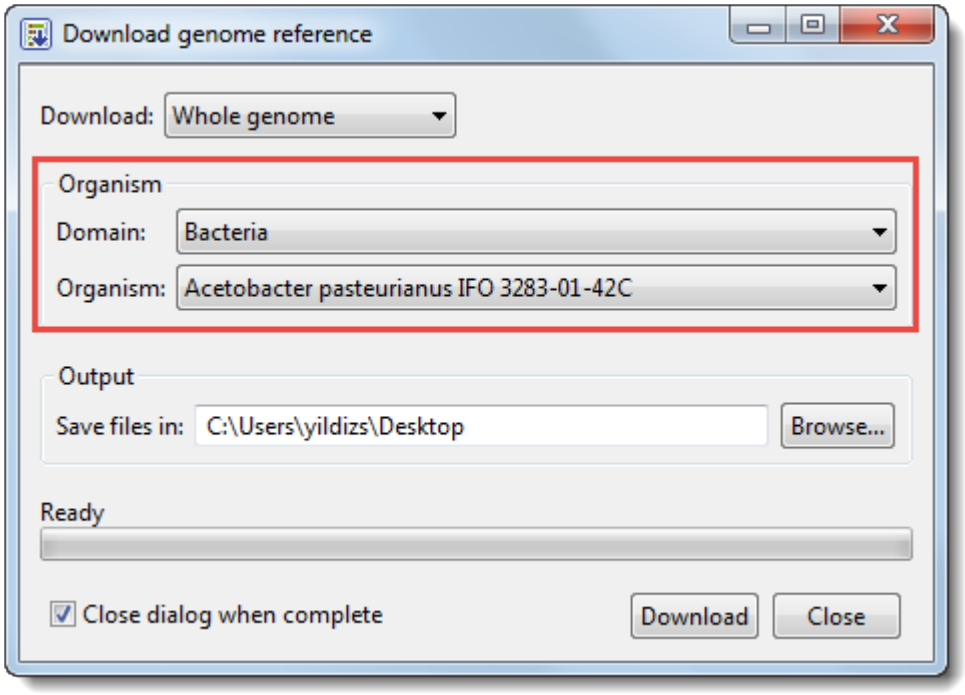
If your workflow includes the Reference Sequence or Set Contaminant screen, you must add at least one reference sequence, biome genome, or genome template before proceeding.

If you are doing a local (i.e. non-Cloud) assembly, you may download and/or add genomes directly from the NCBI database in either GenBank or FASTA formats.

1. Press the **Download NCBI Genomes** button. This launches the Download genome reference dialog.

2. Use the **Download** dropdown menu to choose between downloading a **Whole genome** or **By accession numbers**.

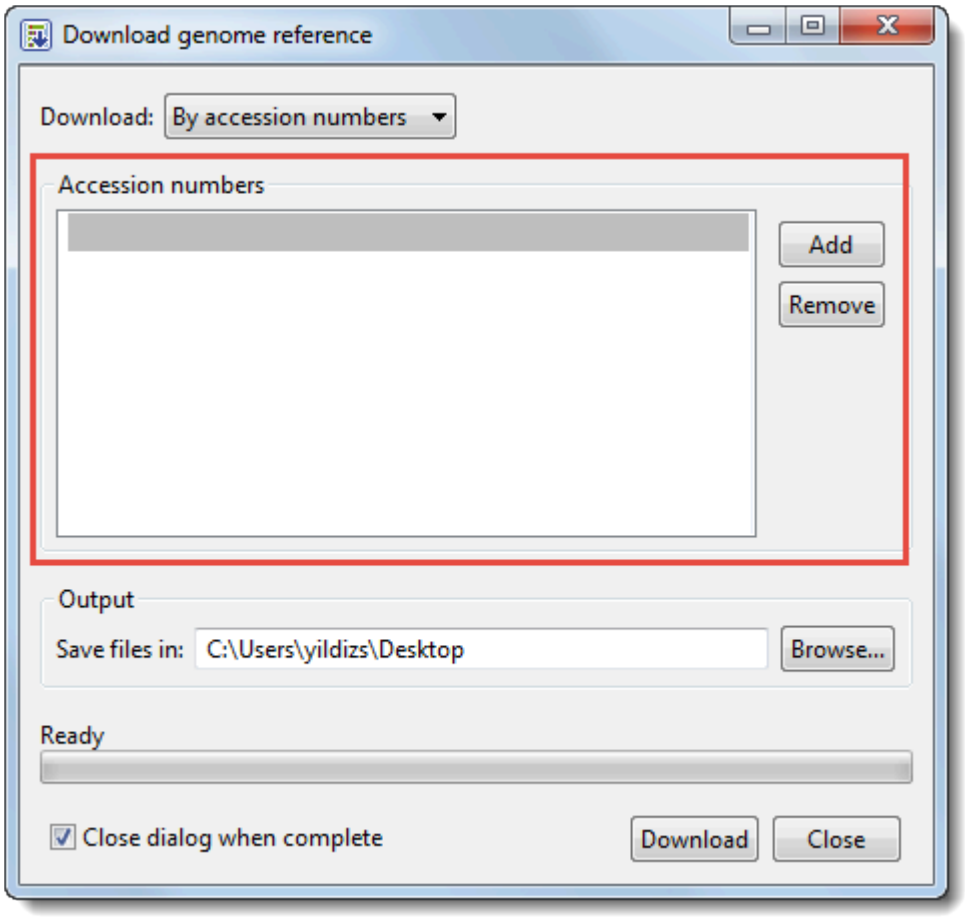
    - If you select **Whole genome**, SeqMan NGen will retrieve the most recent build of the selected genome. Use the next two drop-down menus to select the **Organism type** and **Organism**.



      SeqMan NGen will download all the reference sequences from the NCBI Entrez Genome Project database for the selected genome. These downloads may include auxiliary genomes such as mitochondria and chloroplasts. They may also include some contigs which have not yet been placed by the genome finishing process.

    - If you select **By accession numbers**, the Organism section disappears and is replaced by an Accession numbers entry area. Type an accession number or paste it from your clipboard, then press **Add** to add a number to the list. You can type accession numbers with or without explicit version numbers (e.g., NC_000913.3 or NC_000913). If the version number is omitted, then NCBI's latest version is returned.

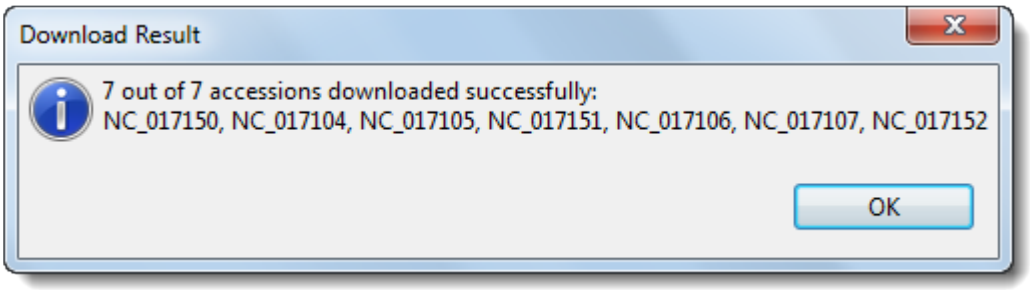
Continue adding numbers, as desired. Multiple accession numbers should be separated using a space, comma, semi-colon or line break. To remove an accession number from the list, select it and click **Remove**.



3. Click **Browse** to select a name and location in which to save the downloaded genome files.

4. If you do not need to download additional genomes, you can check **Close dialog when complete**. Otherwise, leave the box unchecked to keep the dialog open after initiating the current download.

5. Press **Download**. Once the download is complete, a message like the one below will appear.



6. Click **OK** to close the dialog and add the accessions to the Reference Sequence screen. If you

checked **Close dialog when complete**, the Download genome reference dialog will also close. Otherwise, it will remain open so that you can download additional NCBI genomes.

Remove a sequence from the list

To remove a sequence that has been added to the list on the left of the Input Sequences screen, select the file from the list and press **Remove**.

Use editing commands in the wizard

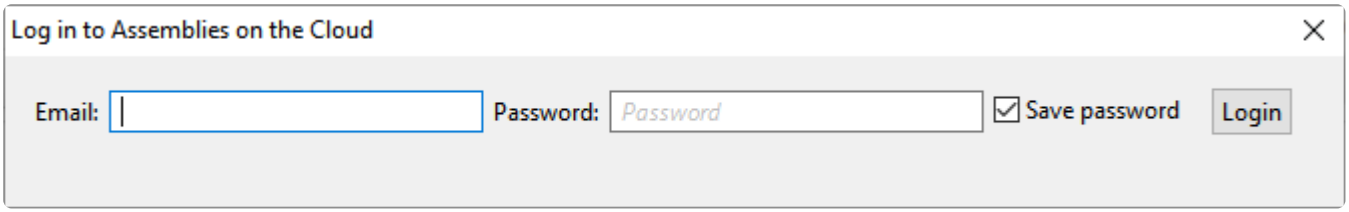
SeqMan NGen's basic editing commands are similar to those found in Microsoft Windows and other text editing programs. These commands may be available through context (right-click) menu options in text boxes throughout the SeqMan NGen wizard.

Task	Keyboard shortcut
To remove the selected portion of text and place it on the clipboard	**Ctrl/Cmd+X**
To copy the selected portion of text to the clipboard	**Ctrl/Cmd+C**
To paste an item in the clipboard at the cursor insert point	**Ctrl/Cmd+V**
To delete a portion of text without placing it on the clipboard	**Delete** key
To select all text	**Ctrl+A**

Log in to Cloud Assemblies

To log in to Cloud Assemblies or to access the Cloud Data Drive:

If you are not logged into your DNASTAR account and press a button associated with Cloud Assemblies or the Cloud Data Drive (e.g., the **Add from Cloud** button in the Reference Sequence screen), a popup dialog will prompt you to enter your **Email** and **Password**.



You can also access this popup dialog by pressing the “key” icon in the lower left corner of the SeqMan NGen wizard. The key has different appearances depending whether you are currently logged in ( ) or not ( ).

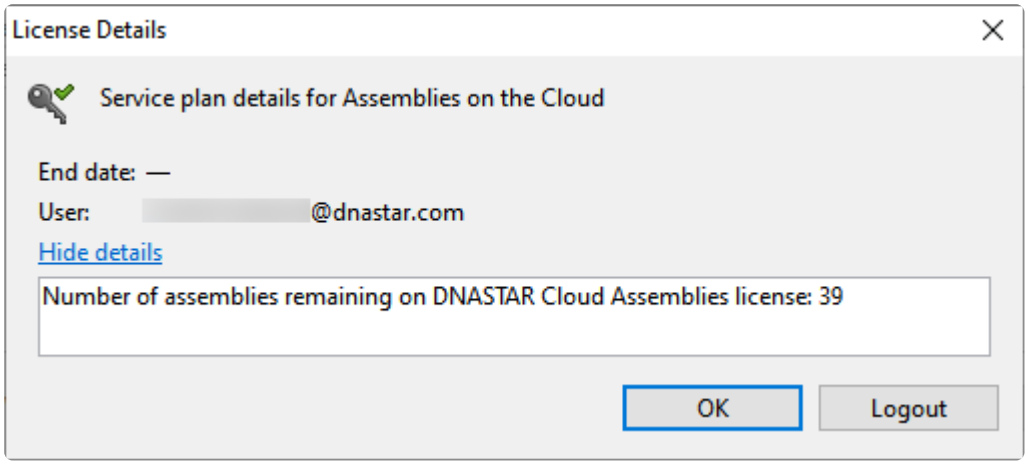
To log in, enter the information that you use when logging in to the DNASTAR website. Check the **Save password** box if you would like SeqMan NGen to save this information so you do not need to retype it in the future. Press **Log in**.

If you forgot your login credentials, click the **Forgot password** link.

To view your Cloud Assembly service plan details:

Once you are logged in as above, click the “key” icon () to display the License Details popup. You service plan **End date** and your **User** name are shown. To see additional details, click the **Show details** link.

- If you have assemblies remaining on a current service plan, the popup expands to show the number of assemblies remaining on your license. Click **OK** to close the message or **Logout** to exit from Cloud Assemblies.



- If no assemblies remain or your service plan has expired, the message will instead ask you to purchase a Cloud license through DNASTAR Support (support@dnastar.com).

Use the DNASTAR Cloud Data Drive

The DNASTAR Cloud Data Drive works as both a Cloud file browser and a mechanism for transferring data between your desktop/laptop computer and the DNASTAR Cloud. Your data and results are stored in a private, encrypted Amazon Web Services (AWS) account visible only to you.

As a file browser:

The DNASTAR Cloud Data Drive lets you view files and folders, and create/delete folders and subfolders.

As a data transfer tool:

The DNASTAR Cloud Data Drive lets you upload data files and folders to the DNASTAR Cloud, or download them from the Cloud to your own computer. The status and progress of uploads and downloads are continuously displayed in the lower pane of the dialog.

The following brief video shows how easy it is to upload your data and then perform a SeqMan NGen Cloud Assembly.

License and Credential Requirements

Requirements for utilizing the DNASTAR Cloud Data Drive:

You must have Internet access and an active license for Assemblies on the Cloud, NovaFold, or the DNASTAR Cloud Desktop. To purchase these items, please request an online quote.

Licensing options for the DNASTAR Cloud:

There are two licensing options:

- DNASTAR provides you with credentials for Assemblies on the Cloud, DNASTAR Cloud Desktop, and/or NovaFold. These are accessed through your DNASTAR account.
- You set up an account with Amazon Web Services (AWS) and create an Access Key for the DNASTAR Cloud Desktop or NovaFold. To sign up for AWS, follow the instructions on this AWS page.

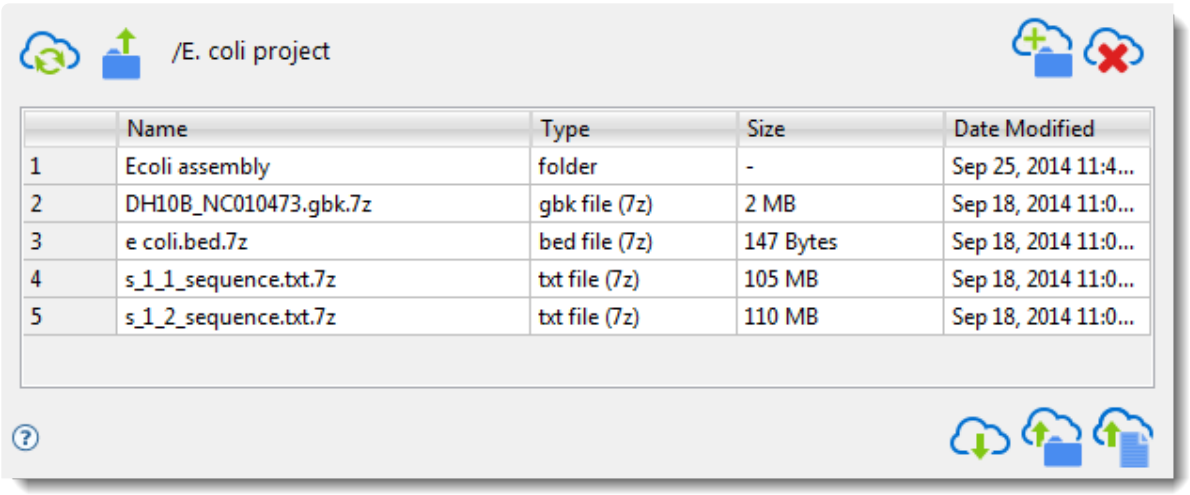
You may need to enter these credentials the first time you attempt to access the DNASTAR Cloud Data Drive.

Note: During the time that your DNASTAR Cloud license is in effect, we recommend that you not change the access key. For more information about managing access keys, please refer to this AWS page.

The DNASTAR Cloud Data Drive User Interface

The DNASTAR Cloud Data Drive consists of an upper section with colorful tool icons and a lower section with buttons.

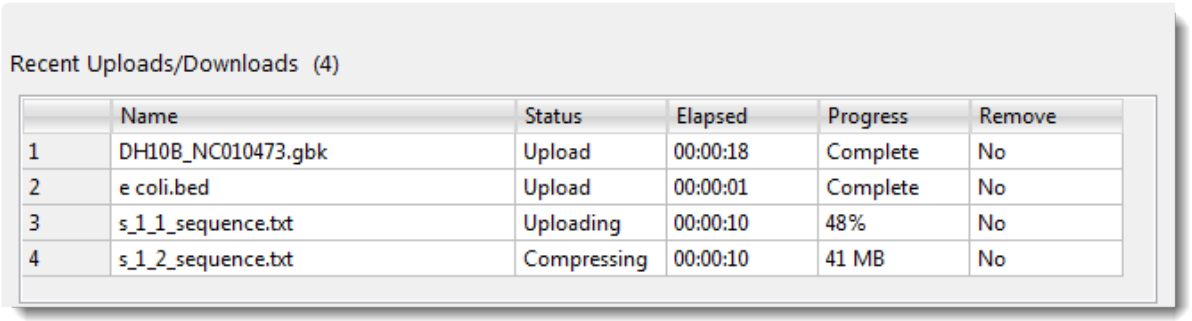
The table in the top half of the dialog shows files and folders already being stored in the Cloud.



Task	How to…
To refresh the current Cloud folder	Click this icon: . Use this option if a file/folder that you recently uploaded is not yet visible in the top table of the Cloud Data Drive, or if a file/folder you recently deleted continues to be visible there.
To create a new folder on the Cloud	Click this icon: . See Create a New Cloud Folder for detailed instructions.
To open a folder	Double-click on the folder. Note that before uploading data (see second table in this help topic), you must open the folder in which the data will be contained.
To move up one folder level	Click this icon: .
To delete the selected items from the Cloud	Click this icon: . See Permanently Remove Files and Folders from the Cloud for detailed instructions.
To upload a folder and its contents from your physical computer to the Cloud	Click this icon: . Once uploaded, folders cannot be moved into or out of another folder. This means that before you upload a folder, you need to open the parent folder (if any) that will contain it. For details on uploading, see Transfer a Folder from a Physical Computer to the Cloud.

To upload files from your physical computer to the Cloud	Click this icon:  . Once uploaded, files cannot be moved into or out of another folder. This means that before you upload a file, you need to open the parent folder (if any) that will contain it. For details on uploading, see Transfer Files from a Physical Computer to the Cloud.
To download files from the Cloud to your physical computer	Click this icon:  . For details, see Transfer Files or Folders from the Cloud to a Physical Computer.
To view the online help	Click the help icon:  .

The table in the bottom half of the dialog displays files or folders that are in the process of being transferred between the Cloud and a desktop/laptop computer.



The following table contains descriptions for each of the columns:

Column	Description
Name	The file name and extension.
Status	The stage of the upload or download process that is currently in progress. • For an upload, the steps are: **Compressing**, **Compression**, **Uploading**, **Upload** (shown both when **Progress** = **Pending** and **Complete**). • For a download, the steps are: **Decompressing**, **Decompress**, **Download** (shown both when **Progress** = **Pending** and **Complete**).
Elapsed	Time elapsed since initiating the upload/download in **Hours:Minutes:Seconds**.
Progress	During compression/decompression of files, this column displays the size of the file. During upload/download, it instead displays information about the percent of the job completed so far: **Pending** (i.e., 0%), **n%**, **Complete** (i.e., 100%).
Remove	Press the **Remove** button if you would like to remove the file from the queue and thereby cancel the

	upload/download.

The buttons at the bottom right of the Cloud Data Drive are used to exit from the application.

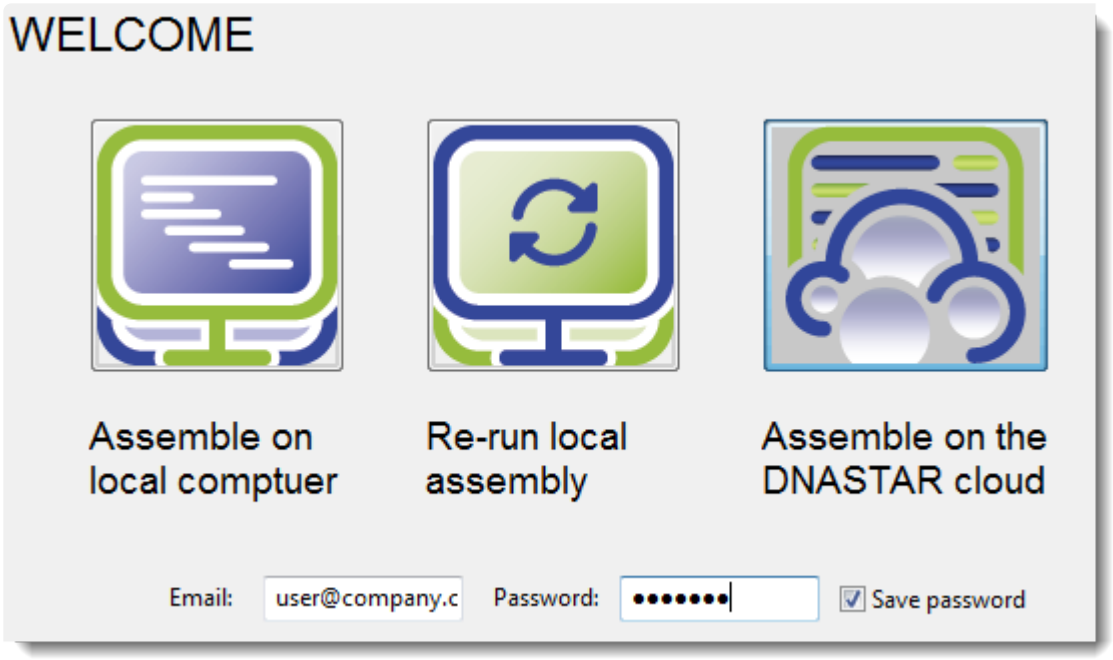
Access the DNASTAR Cloud Data Drive

Once you have met the license and credential requirements, the Data Drive can be accessed in several ways:

Through the SeqMan NGen wizard:

1. In the Welcome screen, select **Assemble on the DNASTAR Cloud** (currently available on Windows and Macintosh only) if you are licensed for this option and wish to perform one or more assemblies on the Cloud, rather than using your own physical machine.

2. Enter your **Email** and **Password**. Enter the information that you use when logging in to the DNASTAR website. Check the **Save password** box if you would like SeqMan NGen to save this information so you do not need to retype it in the future.



3. Click **Next**. If you are licensed, a message will display the number of assemblies remaining on your license. Click **OK** to close the message and continue to the next screen.

> **Note:** If you are *not* licensed, a message will appear, asking you to purchase a Cloud license through DNASTAR Support (support@dnastar.com).

4. In the Cloud Assembly page:

    - If you intend to start a new assembly project, click the **Upload Data** button to access the DNASTAR Cloud Data Drive. From within the Cloud Data Drive, upload any files or folders that

will be needed for the assembly. As you progress through the SeqMan NGen wizard, you can access the files on the Cloud Data Drive by using the **Add**, **Add Folder** or **Browse** buttons.

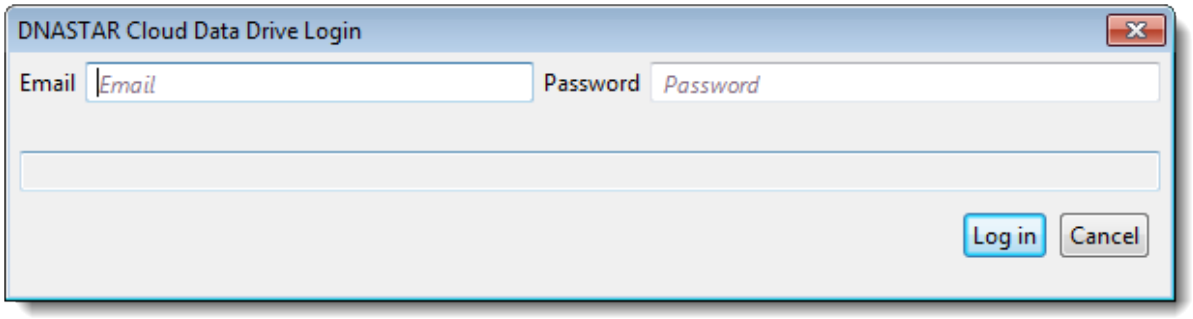
or…

- If you want to monitor an existing assembly, click **Manage/monitor existing cloud projects** and click **Next** to open the Cloud Assemblies screen. From this screen, make a selection in the upper table and then choose either **Download Assembly** or **View Data** to open the DNASTAR Cloud Data Drive.

As a standalone application:

From your computer's hard drive or from the DNASTAR Cloud Desktop, launch the DNASTAR Navigator and then click on **DNASTAR Cloud Data Drive**.

If you are not currently logged in to the DNASTAR Cloud Data Drive, you will first be prompted to log in.

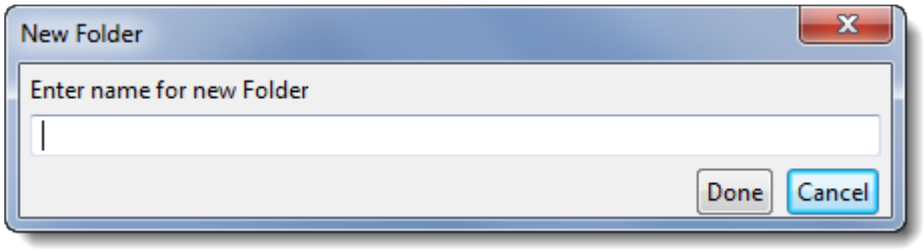


Enter the User ID and Password from your DNASTAR account, then click **Log in**.

Create a New Cloud Folder

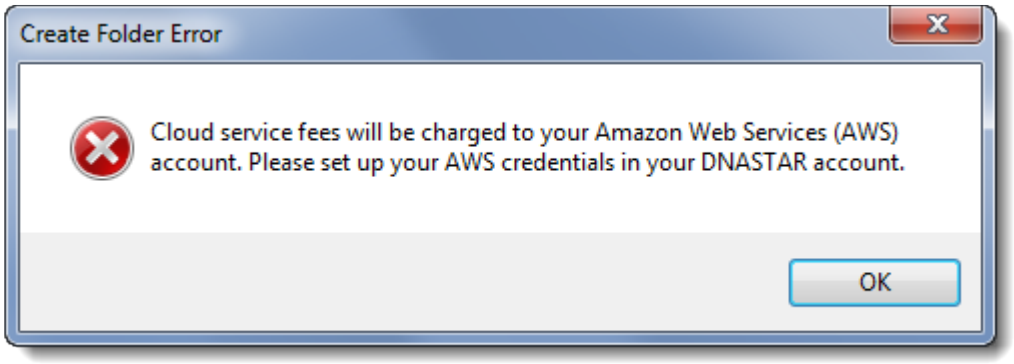
To create a new folder in the DNASTAR Cloud:

Click the **Create a new folder on the Cloud** tool ( ) and type in a name for the new folder.



Press **Done**; or choose **Cancel** to exit without creating a folder.

If your AWS credentials have not been set up, you will receive the following error message:



Click **OK** to exit from the message. To address the credential issue, refer to License and Credential Requirements or contact DNASTAR Support at support@dnastar.com.

Transfer a Folder from a Physical Computer to the Cloud

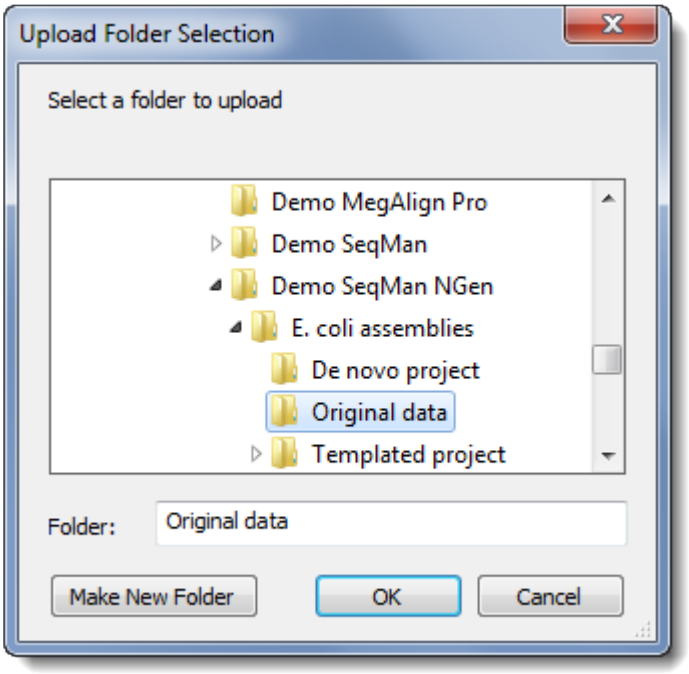
To transfer a folder and its contents from your desktop or laptop computer to the DNASTAR Cloud:

1. Within the Cloud Data Drive, open the folder (if any) that will contain the folder you are transferring.

> **Note:** Once a folder has been transferred to the DNASTAR Cloud Data Drive, you cannot move it into a different "containing folder."

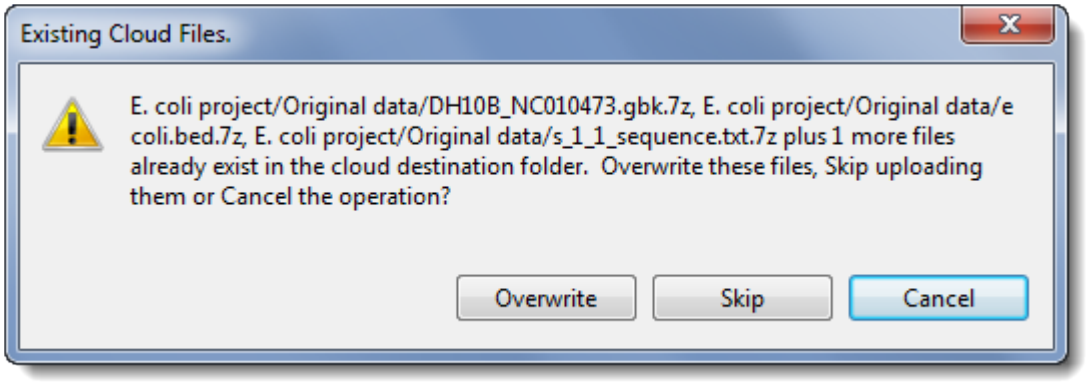
2. Click the **Upload a folder and all its contents to the Cloud** tool).

3. Navigate to the file or folder location and select it.



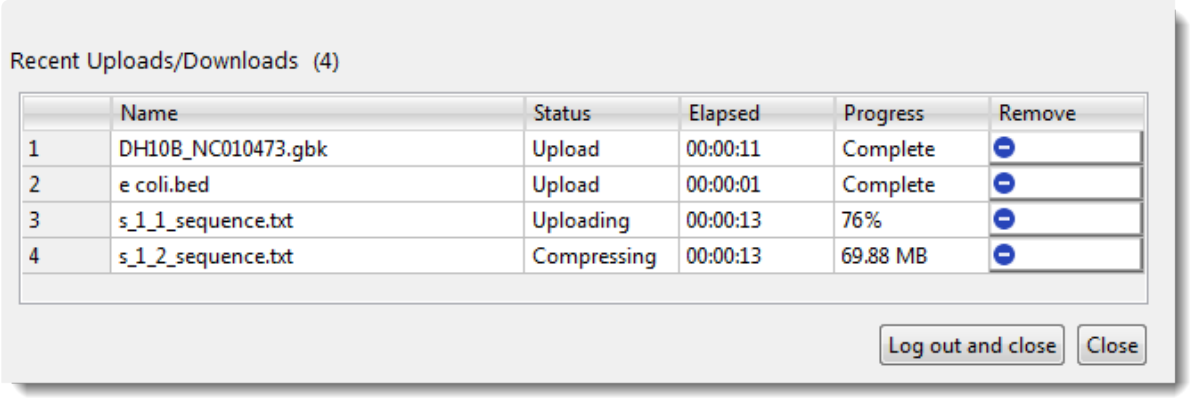
4. Press **OK**.

    - If the same folder has been uploaded previously to the same location, the following message will appear.



Choose whether to **Overwrite** the existing folder, **Skip** uploading the folder or **Cancel** the entire upload operation. In the case of a folder, the final two options are identical.

- If you have not previously uploaded the folder and its files, or if you have chosen to **Overwrite** existing files, the files within the selected folder will begin uploading in the lower half of the Cloud Data Drive dialog. If the folder was not previously uploaded, the folder will be created in the upper half of the Cloud Data Drive dialog and the files placed within.



At the completion of the upload, the folder will be displayed in the upper table of the dialog. Double-clicking on the folder name will reveal the folder contents.

Transfer Files from a Physical Computer to the Cloud

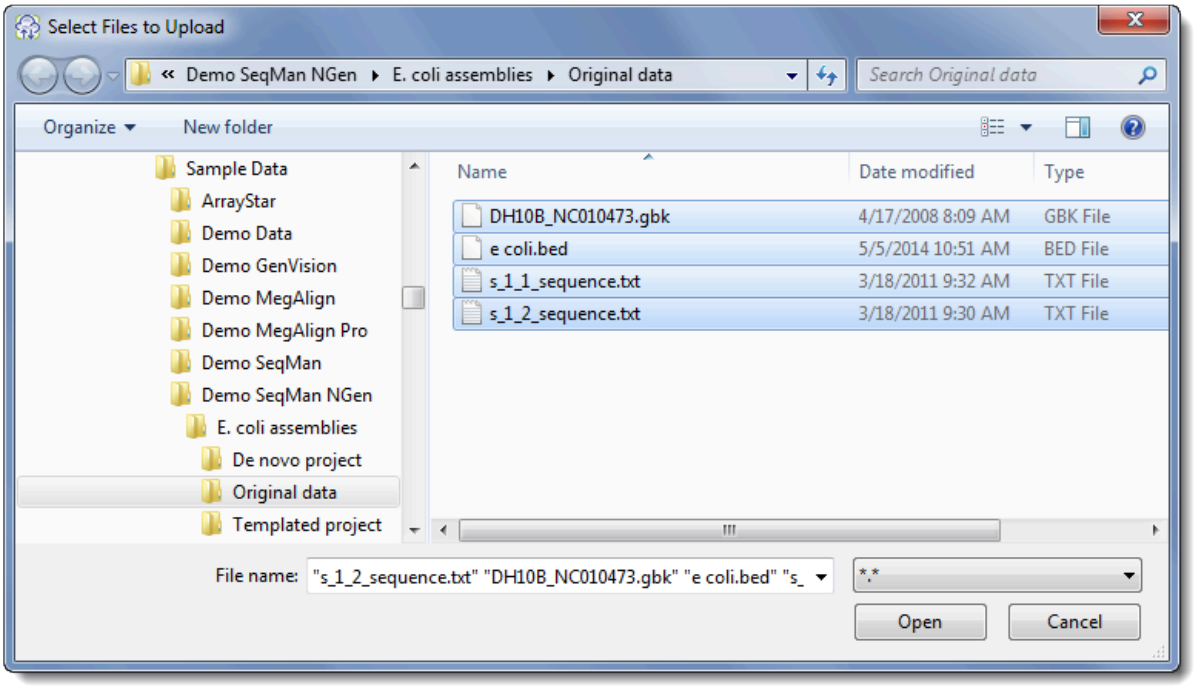
To transfer one or more files from your desktop or laptop computer to the DNASTAR Cloud:

1. Within the Cloud Data Drive, open the folder (if any) that will contain the files you are transferring.

> **Note:** Once a file has been transferred to the DNASTAR Cloud Data Drive, you cannot move it into a different "containing folder."

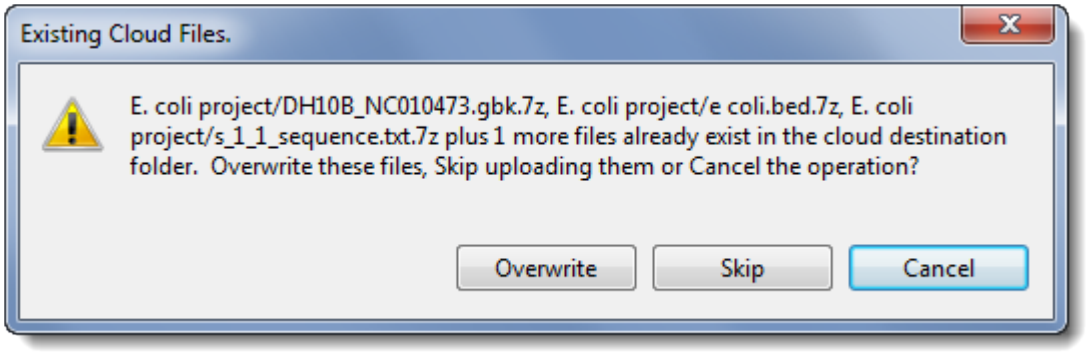
2. Click the **Upload files to the Cloud** tool ( ).

3. Navigate to the file location and select the desired file(s).



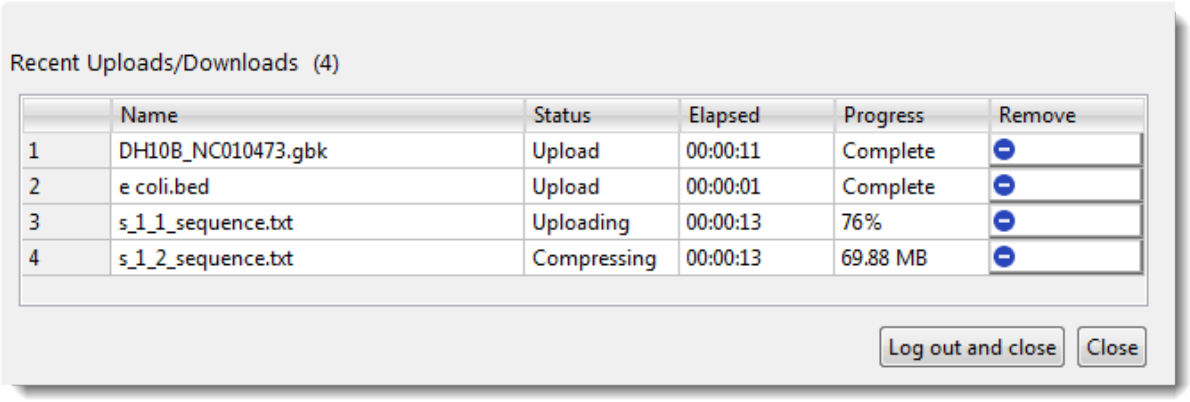
4. Press **Open**.

   - If one or more files have been uploaded to the same location previously, a warning message will appear.



Choose whether to **Overwrite** the file(s), **Skip** uploading the duplicate file(s) or **Cancel** the entire upload operation.

- If you have not previously uploaded the files, or if you have chosen to **Overwrite** existing files, the files will begin uploading in the lower half of the Cloud Data Drive dialog. If the files were not previously uploaded, they will be added to the upper half of the Cloud Data Drive dialog.



At the completion of the upload, the files will be displayed in the upper table of the dialog.

Transfer Files or Folders from the Cloud to a Physical Computer

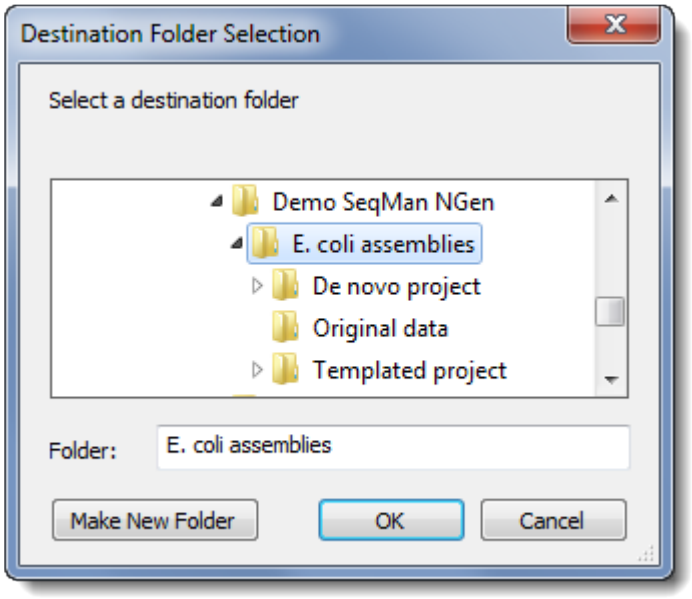
To download files or folders from the Cloud to a physical computer:

1. Select the files and/or folders from the upper table by clicking, **Shift+clicking** or **Ctrl/Cmd+clicking**.

   > **Note:** If you do not make a selection, you will receive a warning message after Step 2.

2. Select the **Download** tool ( ).

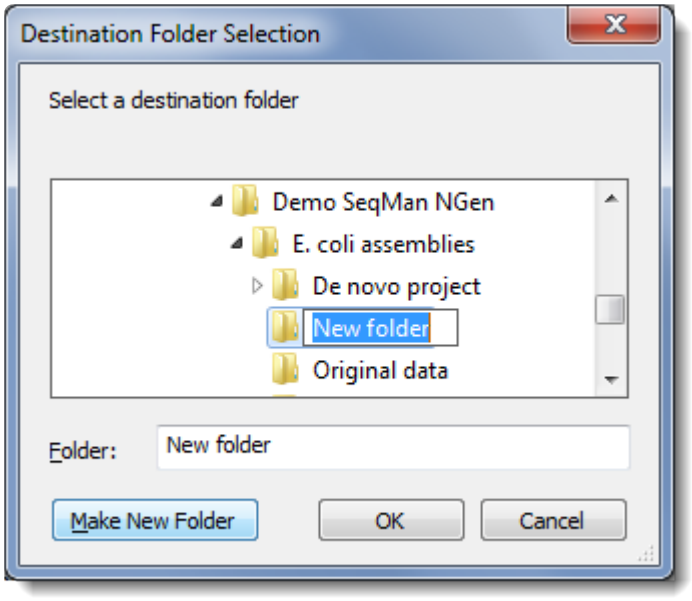
3. In the ensuing dialog, navigate to the destination folder.



4. (optional) If desired, create a new sub-folder in the selected directory as follows:

   a. Select the parent folder.

   b. Click **Make New Folder**. A new folder is created, ready to be renamed.



c. Type a name for the folder.

5. Press **OK**.

   The progress of the download(s) will be shown in the bottom half of the Cloud Data Drive. After downloading is complete, the files/folders can be found in the destination directory chosen in Steps 3 or 4.

Permanently Remove Files and Folders from the Cloud

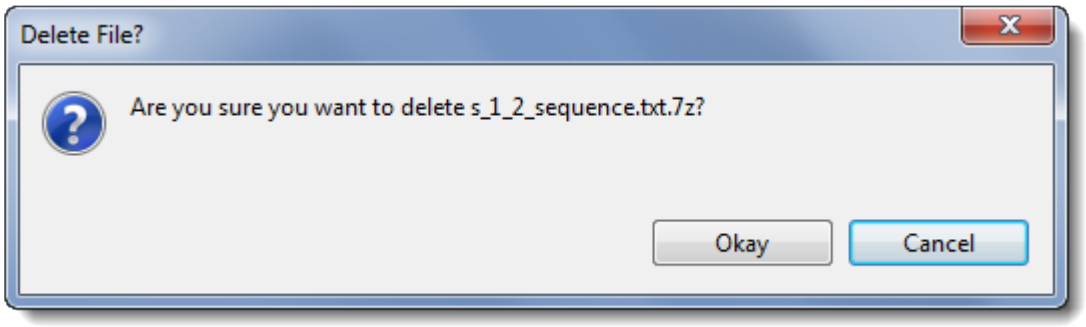
To permanently remove files and/or folders from the DNASTAR Cloud:

1. Select their rows from the upper table by clicking, **Shift+clicking** or **Ctrl/Cmd+clicking**.

   > **Note:** If nothing is selected, you will receive a warning message after Step 2.

2. Press the **Delete** tool ( ).

3. In the ensuing confirmation dialog(s), confirm that you wish to delete the file(s)/folder(s) by pressing **Okay**. Alternatively, you may abort the deletion process for a given file by clicking **Cancel**.



Close the DNASTAR Cloud Data Drive

To close the DNASTAR Cloud Data Drive application:

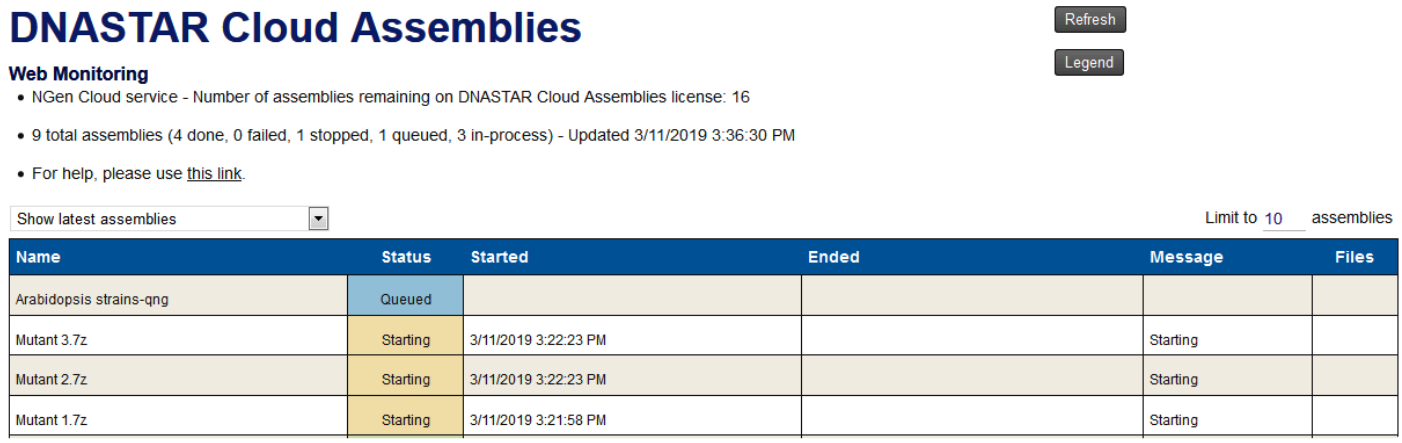
- Click **Close** to exit.

  or…

- Click **Log out and close** to both close the Cloud Data Drive and log off from your DNASTAR account. After using this button, the SeqMan NGen wizard will “forget” your saved credentials, and will prompt you for your **Email** and **Password** the next time you request access to Assemblies on the Cloud.

Monitor the progress of a cloud assembly

If you choose to perform an assembly on the cloud, you can monitor assembly progress using the online Web Monitor. Click this link, and enter your DNASTAR login credentials if prompted to do so. See the Web Monitor help to learn about features in this browser window.



Access and understand output files

The output file structure for a SeqMan NGen assembly varies depending upon the workflow. For a description of output files, see Reference-guided workflow output or *De novo* workflow output.

Note that FASTQ files “created with the SeqMan NGen wizard will have a *.fastq* extension, while those created via a command line script will have a *.fas* extension.

View the Project Report

The Project Report summarizes the assembly statistics, including the parameters used, the number of assembled/unassembled sequences and contigs in your project, and the average quality scores.

Opening the Project Report:

Use any of the methods below:

- Click the **Project Report** button in the Assembly Summary screen.
- Open the output file *Report.txt* in any suitable text editor.
- Open the assembly in SeqMan Pro and choose **Project > Report**.
- Open the assembly in SeqMan Ultra and choose **Project > Project Details**.

Project Report contents:

- Click these links for a list of the Project Report contents for reference-guided and *de novo* workflows.
- The contigs in the project are named as follows:
    - If you performed a reference-guided workflow, the resulting contig will take the name of the reference sequence name.
    - If you scanned your assembly for known repeats, then the contigs in your project containing sequences flagged as possible repeats will be named: Repeat-00001, Repeat-00002, Repeat-00003, etc.
    - If none of the above applies, the contigs in your project will be named Contig 00001, Contig 00002, Contig 00003, etc.
- If you've opted to create both an *.sqd* and an *.assembly* output, you may notice that the files do not exactly match. That's because *.sqd* files, unlike *.assembly* files, allow sequences to extend beyond either end of the reference sequence.

Project Report contents for reference-guided workflows

The Project Report for reference-guided assemblies will contain a subset of the following results:

Run Statistics	
Reference Seq Cnt	The total number of sequences in the reference (template).
Sequence Cnt	The total number of reads in the sample.
Total Reads Assembled	
Pair Seqs Cnt	The number of paired sequences included in the assembly.
Single Pair Seq Cnt	The number of paired sequences of which only one pair was included in the assembly.
Split Seq Cnt	The number of sequences that were split in the assembly.
Bad Split Seq Cnt	The number of sequences that were split, and of which only one portion was included in the assembly.
Single Seq Cnt	The number of single (unpaired) sequences in the assembly.
Consistent Pair Cnt	The number of paired sequences that met pair constraints. One “pair” in this statistics represents two sequences.
Inconsistent Pair Cnt	The number of putative paired sequences that did not meet pair constraints.
Seqs score < 80%	Percentage of reads that exactly matched the template (i.e. “alignment score”).
Seqs score < 90%	
Seqs score < 100%	
Seqs score 100%	
Unassembled Sequences	
Unaligned Cnt	Total number of reads not included in the finished assembly.
LayoutMiss Cnt	The number of reads that didn’t match the template at all. In other words, the number of sequences that contained no mer which matched a mer on a template sequence. This number is affected by the assembly parameters merSize and merSkip. **Example:** A sequence that has no 21-mer in common with the template but does have a matching 17-mer would be included in **LayoutMiss Cnt** at a mer size of 21, but not at a mer size of 17.
LayoutPoor Cnt	The number of reads with an insufficient number of mer matches to be included in the

	assembly. This number is affected by the assembly parameter merLayoutMin.
Bad Seq Cnt	The number of reads with ≥25% ambiguous Ns in the sequence. Filtered Illumina data is sometimes included in this count, as well.
Excluded Seq Cnt	The number of contaminated reads.
ExcessiveCov. Seq Cnt	The number of reads unused due to excessive coverage.
SNP Info	
Found SNP Cnt (incl. indel lengths)	The number of SNP positions plus the total number of coalesced bases, minus the number of multi-base indel entries.
Found User SNP	The number of SNPs found that match those in the user-supplied VCF SNP file.
Missing User SNP coverage	The number of SNPs from the user-supplied VCF SNP file that were *not* found, even though the area had coverage.
Missing User SNP zero coverage	The number of SNPs from the user-supplied VCF SNP file that were not found because the area had no coverage.
Assembly Parameters	
merSize	The values specified in the SeqMan NGen wizard prior to assembly.
merSkip	
merSkipQuery	
merLayoutMin	
templateHitCntThresh	

Project Report contents for de novo workflows

The Project Report for *de novo* assemblies will contain a subset of the following results:

Assembly Totals	
Contigs	Total number of contigs assembled.
Contigs > 2K	Total number of assembled contigs that are more than 2000 base pairs in length.
Contigs to Reach Genome Length 'x'	The number of contigs needed to cover the genome length specified in the Workflow pane.
Contigs removed due to small size	The number of contigs removed due to being smaller than the threshold value.
Assembled Sequences	The number of sequences utilized in the assembly.
Unassembled Sequences	The number of sequences excluded from the assembly. These may be further categorized as: 1) Sequences not assembled due to complete trimming, and 2) Sequences removed due to small contig size.
All Sequences	Total number of sequences in the project.
Contig N50	Contig size at which 50% of the sequence data are represented. **Note:** In a typical microbial genome assembly, Contig N50 values exceed 80K base pairs and genome coverage is attained in less than 100 contigs. In many assemblies, contig N50 exceeds 100K with genome coverage attained in 25 contigs. If paired-end Roche 454 Life Sciences data are used, contigs can be ordered into a handful of large scaffolds to attain genome coverage that greatly facilitates gap closure and completion of the genome assembly.
Average Coverage	Average depth of coverage in the assembly.
Average Totals	
Sequences Per Contig	Average number of sequences used for each contig.
Average Lengths	

Contigs	Average contig length.
Assembled Sequences	Average length of sequences used in the assembly.
Unassembled Sequences	Average length of sequences excluded from the assembly.
All Sequences	Average length of all sequences in the project.
Average Quality	
Assembled Sequences	Average quality score of sequences used in the assembly.
Unassembled Sequences	Average quality score of sequences excluded from the assembly.
All Sequences	Average quality score of all sequences in the project.
Assembled Pair Statistics	
Read Pairs	Total number of paired reads in the project.
Assembled Pairs	The number of paired reads included in the assembly.
Pairs Consistent Within a Contig	The number of paired sequences within a single contig that met pair constraints. One “pair” in this statistics represents two sequences.
Pairs Inconsistent Within a Contig	The number of putative paired sequences within a contig that did not meet pair constraints.
Split Pair Statistics (Ion Torrent paired reads and 454 data only)	
Reads Split into Pairs	The number of reads that were split into pairs at the linker.
Unsplit Reads with Pair Linker(s)	The number of reads that were not split into reads because the linker was too far to one side.
Unsplit Reads without Pair	The number of reads that were not associated with a linker.

Linker(s)	
Assembly Parameters	
Match Size	The values specified in the SeqMan NGen wizard prior to assembly.
Match Spacing	
Minimum Match Percentage	
Match Score	
Mismatch Penalty	
Gap Penalty	
Max Gap	
Genome Length	
Expected Coverage	

Reference-guided workflow output

Reference-guided workflows vary in the number and contents of output files and folders. Only a subset of items in the table below may appear for a particular workflow.

In the file names below, the project name should be understood to precede any hyphen (-) or period (.) used at the beginning of file and folder names.

Single assemblies:

The project folder has the name specified in the Assembly Output screen and contains:

- **.script** file
- **.assembly** package (called **.transcriptome** for the transcript annotation workflow)
- **-noSplit.assembly** package (Reference-guided assembly with gap closure workflows only)
- -**Reports** folder
    - -**zinternal** folder
        - **info** folder

All Exome and Gene Panel projects and multiple sample assemblies run as separate assemblies:

The project folder contains the name specified in the Assembly Output screen followed by the suffix **_assemblies** (called **_RNA-Seq** in the reference-guided RNA-seq workflows). This folder contains:

- **.script** file (if saved in the Run Assembly screen).
- **Results.txt** file - Overview information and statistics for each assembly.
- **.table.txt** file
- **.template.script** file
- **_arstar.script** file – A script to load all assemblies as a SNP project in ArrayStar.
- **_arstarValidation.script** file – (only if validation control was present) A script to load the validation control assembly and associated VCF file as a SNP project in ArrayStar and to automatically calculate the accuracy statistics.
- **.assembly** packages (one per sample)
- **-Reports** folders (one per sample)
    - **-zinternal** folder
        - **info** folder

Contents of the .assembly package

The **.assembly** package is part of the output for XNG workflows. (The contents of the **-noSplit.assembly** package are similar to those of the **.assembly** package.)

In the file names below, the project name should be understood to precede any hyphen (-) or period (.) used at the beginning of file and folder names.

Note for Windows users: To open text reports with the correct formatting displayed, we recommend using Wordpad, Notepad++, or Microsoft Excel®, and not the default Windows text editor, Notepad.

File Suffix	Description
It is intended that the entire .assembly package be opened in SeqMan Pro or SeqMan Ultra for viewing and analysis of the assembly. However, the following individual files also contain useful information.	
.vcf	A VCF file (**.vcf**) is automatically created for all assemblies with variants. The file is modified in three ways adhering to the Variant Call File (VCF) v. 4.2 specification: * In the FILTER field, each row is marked with one of three qualifiers to show whether or not a position was covered: ** “PASS” for positions where a call could be determined based on the sequence read data. ** “NC” for positions with no sequence read coverage (this will be denoted at the top of the file under ##FILTER.) ** “.” for positions when data for a call is missing or a call could not be made. These changes to the FILTER field apply to both single-sample and multi-sample VCFs, but not to VCFs lacking any sample information. * In the QUAL field, a Phred-scaled quality score is provided for the assertion made in the ALT column. The score is calculated as -10 $\log_{10}$ prob (call in ALT is wrong). ** In rows where the ALT column contains ‘.’ (i.e. no variant was called), the column contains -10$\log_{10}$ prob(variant). ** In rows where the ALT does not contain ‘.’ (i.e. a variant call), the column contains -10$\log_{10}$ prob(no variant). ** A missing value is specified as “.” * The PA field contains the Pnotref value. Note that the QUAL scale is reversed relative to Pnotref when ALT is "."; that is, when a position is in the

	reference. However, in one direction or the other, it will scale logarithmically with Pnotref. This does mean that it will be closer to Qcall (or "GQ") in cases where there isn't "homozygous vs. heterozygous" call ambiguity. However, when the ambiguity is present, it will diverge.
.bed, .txt, etc.	The target region file (.bed or manifest) for the assembly, if one was specified.
.templateInfo	Contains general information for each contig in the assembly.
.enrichment_Summary.txt	Contains the textual information for the **Project > Show coverage of target regions** option in SeqMan Pro and SeqMan Ultra.
.sqd	This file is only created when the *.assembly* is first opened in SeqMan Pro or SeqMan Ultra. It contains saved display specific information such as SNP filtering criteria. Double-clicking on this file will open the *.assembly* package in SeqMan Pro or SeqMan Ultra.
-Transcriptome table folder containing the file .table.txt	This folder and its file, showing the putative gene identity for each transcript, are created for the de novo transcriptome RNA-seq workflow only.
There is normally no reason to open the following files.	
-0.assemblyInfo	Contains information about assembly parameters which can be used for combining multiple assemblies. This file is not present for SeqMan NGen assemblies made prior to version 14.0. In 14.0 and later, it is present in templated miRNA, ChIP-Seq, and RNA-Seq workflows.
[project name]Transcriptome.table.txt	This file is present in RNA-seq workflows that used a *.Transcriptome* package as a template. It is equivalent to the *.table.txt* file in the Transcriptome table folder of the *.Transcriptome* package.
.auxPair	(internal use only)
.bam	The BAM formatted alignment file.
.bam.bai	The BAM index file.
.capture.userSNP.vcf	(internal use only)
.combined.snpExt	(internal use only)
.coverage	Contains information at each position along the contig where the coverage changes.
.coverage2	Contains information for the maximum coverage of 100 base pair intervals across the contig.
.coverage4	Contains information for the maximum coverage of 10,000 base pair intervals across the contig.
.coverage.missingSNP	Contains information about positions in dbSNP that had coverage and were called the reference base in the assembly.

.exomeCapture-features	(internal use only)
.info	Contains files used by SeqMan Pro and SeqMan Ultra.
.midinfo	(internal use only)
missing.fas	A fasta file of reads with no mers matching the reference.
missing.fas.qual	A base quality file of reads with no mers matching the reference.
.nocoverage.missingSNP	Contains information about positions in dbSNP that had no coverage in the assembly.
outofOrder.txt	A text file of sequence reads not included in the final assembly due to excessive trimming during the alignment phase.
.pair	(internal use only)
.pairDist	Contains information about the position and distance between paired end reads.
pairSpecifiers.txt	(internal use only)
poor.fas	A fasta file of reads rejected at the layout phase due to match scores below the threshold.
poor.fas.qual	A base quality file of reads rejected at the layout phase due to match scores below the threshold.
.quant	Reprises information in the .coverage4 .coverage2 and /or .coverage files.
.region_capture.bed	(internal use only)
report.txt	Contains the textual information for the **Project > Report** option in SeqMan Pro. See View the Project Report for information about the report contents for XNG and SNG workflows.
.snp	Contains all the information for SNPs called using the “Simple” method.
.snpExt	Contains all the information for SNPs called using either the “Diploid” or “Haploid” method.
SNPs.log	An optional text form of the .snpExt table that contains information on how each was calculated. If you encounter a problem, this file is useful for DNASTAR Support to help you with trouble-shooting.
.splitExt	(internal use only)
.template-comment	Contains the comment information for that contig.
.template-features	Contains the feature information for that contig.
.template-features2	(internal use only)

.template.fof	A file-of-files containing the path and file names of the reference sequences.
.template-gapped-seq	A .seq file of the template containing gaps.
.template-gaps	A binary file of the template gap information.
.template-seq	A .seq file of the template without gaps.
unaligned.fas.qual	A base quality file of reads rejected at the alignment phase.

Contents of the -reports folder

The *-reports* folder is part of the XNG *.assembly* package.

In the table below (and in the sentence above), it should be understood that the project name precedes any hyphen (-) or period (.) used at the beginning of file and folder names.

File Suffix or Extension	Description
-zinternal	See Contents of the -zinternal folder for details.
-enrichment_Summary.txt	(internal use only)
-perAssemblyResults.txt	The table consists of one row of overview information and assembly statistics per assembly. Information from subsequent assemblies is appended to the existing table. Here are some statistics of interest from **perAssemblyResults.txt**: * **NumSeqs** – The total number of sequences. * **Export_Split_Cnt** – The number of exported split reads. * **Export_Aligned** – The number of exported reads. **Note:** Since split reads are counted in each location where they align, it is possible for **Export_Aligned** to exceed **NumSeqs**.
-perTemplateResults.txt	Overview information and assembly statistics per contig.
-projectReport.txt	Overview information of the assembly. The same report can be viewed within SeqMan Pro using the **Project > Report** menu command.
-unassembled.fastq	The unassembled reads from the assembly in Fastq format. If production of this file is not specified in the script, three files are created instead: * **missing.fastq** – Uunassembled reads with no hits to any template. * **poor.fastq** – Unassembled reads with scores too low to include in the layout. * **unaligned.fastq** – Unassembled reads included in the layout, but rejected by the aligner.

Contents of the -zinternal folder

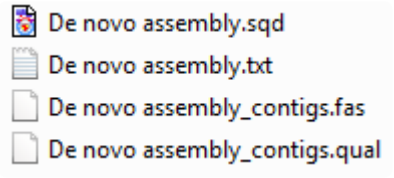
The *-zinternal* folder is located in the *-reports* folder, which in turn is part of the XNG *.assembly* package.

In the table below (and the nomenclature used in the sentence above), it should be understood that the project name precedes any hyphen (-) or period (.) used at the beginning of file and folder names.

File Suffix or Extension	Description
info	The files **-[templateID].insertion2** and **-[templateID].sV_Edges.txt** both contain structural variation information used in SeqMan Pro reports.
bamToSQD.script	For converting the assembly to .sqd format.
pairScheme.info	(internal use only)
results.txt	Here are some statistics of interest from the **results.txt** file: * **templateCoverage%** – The percentage of the length of the template that is covered by one or more reads. * **medianCoverage** – The average depth for regions that with nonzero coverage.
The following files instruct SeqMan NGen to convert unassembled reads into a separate SQD project:	
batchUnassembled	A UNIX executable file for *de novo* assembly of unassembled reads with v. and vi.
batchUnassembled.table.txt	A table of values for running an SNG assembly of the missing .fas reads against the template sequence.
batchUnassembled.template.script	The SNG script containing variables that are specified by the batchMissing.table.txt.

De novo workflow output

De novo workflows output a results folder containing the following files:



File Suffix	Description
.sqd	The main assembly output. To view and analyze the assembly, open this file with SeqMan Pro or SeqMan Ultra.
.txt	(internal use only)
-contigs.fas	Created when contigs are saved in FASTA format.
-contigs.qual	Created when contigs are saved in FASTA format. The values in the file are the sum of the base qualities at each position in the contig, up to a maximum of 90.

Note for Windows users: To open text reports with the correct formatting displayed, we recommend using Wordpad, Notepad++, or Microsoft Excel®, and not the default Windows text editor, Notepad.

RNA-Seq reference-guided workflow output

If you are following a reference-guided RNA-Seq workflow, output results are saved in an .assembly package folder labeled with the user-specified project name and the suffix **_RNA-Seq**. This folder contains the following files:

Subfolder Name	File Name	Description
	[project name].astar	An ArrayStar project file containing data from all of the samples.
	[project name].script	The main script for a reference-guided assembly project.
	[project name].table.txt	The table containing project descriptions and corresponding files; used by the main script.
	[project name].template.script	The reference-guided workflow script, which is applied to all projects listed in **.table.txt**.
	[project name].astar.script	The script file executed by ArrayStar when the **Launch in ArrayStar** button is pressed in the Project Report.
	query'n'.script	The script containing query sequences. This script is generated only if a multiple project assembly includes groups that contain non-equal numbers of files.
	_tableScript.template.script	The script template, executed by the XNG assembler in order to run the QNG assembler. This script is not present before the run and is generated by XNG.
	[project name]-qng.script	After SeqMan NGen finishes running the XNG part of the assembly, QNG assembly is performed using this script. In multi-sample projects, this file references **.table.txt**.
	[project name].Results.txt	A tab-separated table showing the assembly results.
Sample'n'.assembly	**sample'n'-0.isoforms-features**	Each .assembly project contains a set of files relating to a single contig.
	sample'n'-0.genes-features	
	sample'n'-0.bam	
	sample'n'-0.coverage	
Sample'n'-Reports	**'n'-0.report.txt**	A summary of the run; also shown in the Project Report.

RNA-Seq de novo transcriptome workflow output

If you are following the de novo transcriptome RNA-Seq workflow, output results are saved in a folder called *[project name] De Novo Transcriptome Assembly*. This folder contains the following subfolders and files:

Subfolder	File/Folder Name	Description
	[project name]_rnaAssemble.script	Input script used to create the assembly results. This file can be opened in SeqMan Pro in order to examine isoforms using the Feature Table.
Assemblies	**[project name]_novel_transcripts.sqd**	SQD assembly of all contigs that did not have a database match.
	[project name]_unassembled.fastq	Multi-sequence FASTQ file with all unclustered and unassembled sequences.
	sub_0 (folder)k	Folder containing sub-folders (sub_0, sub_1, etc.) with a separate .sqd document for each final assembly. If available, gene and organism names are used to create the file names.
Intermediate Assembly Results	**cluster (folder)**	Intermediate results are deleted by default at the end of the assembly, but can be retained by designing the input script such that the **assembleTemplate** command's **deleteIntermediates** parameter is set to **false**.
	combine (folder)	
	intermediateFiles (folder)	
Reports	**[project name].AllTranscripts.SearchResults.txt**	Excel file containing summary information for each of the final assembled contigs. The table automatically opens for viewing when you open a .Transcriptome package in SeqMan Pro. The table, known in SeqMan Pro as the "All Transcripts" table, contains the following columns: * **Assembly ID** – Name assigned to the assembled sequence, using the criteria specified in the wizard. * **Gene name**, **Custom column #1*** – Best matching gene meeting criteria defined in the wizard. * **Organism name**, **Custom column #2*** – Organism from which the best matching gene came.

* **Accession number**, **Custom column #3*** – Accession number of the best match.

* **Description**, **Custom column #4*** – Description of the best match.

***Custom columns:** The four “custom columns,” above, use default names (e.g., **Gene name**, **Organism name**) if one of the default RefSeq databases was used in the SeqMan NGen assembly. However, if you used a custom GREP expression or a custom database that did not include these fields, these columns may have different names or be absent from the table.

* **Database** – Database (e.g. RefSeq, Custom, etc.) from which the best matching gene came.

* **Transcript length** – Length of the assembled sequence, in bases.

* **Transcript start** – Position in the assembled sequence where the match begins.

* **Transcript end * – Position in the assembled sequence where the match ends.**

* ***% Transcript match** – Length of the matching segment in the transcript x 100, divided by the total length of the transcript.

* **Gene length** – Length of the database entry, in bases.

* **% of Full length** – Length of the assembled sequence x 100, divided by the length of the corresponding database entry. Values greater than 100% indicate that the assembled sequence is longer than the database entry.

* **Gene start** – Position in the database entry where the match begins.

		* **Gene end** – Position in the database entry where the match ends. * **% Gene match** – Length of the matching segment in the database entry x 100, divided by the total length of the database entry. * **% Identity** – Total number of identical bases in the matching region x 100, divided by the total number of bases in the matching region. * **Bit score** – Normalized value calculated from the raw score and expressed in units of “bits,” a common measure in information theory. * **eValue** – “Expectation value,” an estimate of the probability of obtaining the observed alignment score with two random sequences. Expectation values are less sensitive to length than **Bit scores** and are therefore are generally a better measure of alignment quality. * **Assembled reads** – Total number of assembled reads for that sequence.
	[project name].AllTranscripts.Table.txt	Excel file containing summary information for each of the final assembled contigs. The table contains the following columns: * **Assembly ID** – Name assigned to the assembled sequence, using the criteria specified in the wizard. * **Type** – Type of matching gene (e.g., mRNA, tRNA, rRNA, etc.) * **Gene length** – Length of the database entry, in bases. * **% of Full length** – Length of the assembled sequence x 100, divided by the length of the corresponding database entry. Values greater than 100% indicate that the assembled sequence is longer than the database entry.

		* **Assembled reads** – Total number of assembled reads for that sequence. * **Depth** – Average depth of coverage.
Transcripts	**[project name]_identified_transcripts.fas**	Multi-sequence *.fasta* file containing the consensus sequences from all the assembled contigs that had a database match. Header lines for each entry contain the name and sequence length.
	[project name]_novel_transcripts.fas	Multi-sequence *.fasta* file containing the consensus sequences from all the assembled contigs that did not have a database match. Header lines for each entry contain the name and sequence length.

Appendix

The Appendix contains the following topics:

SeqMan NGen calculations

This following topics describe how SeqMan NGen handles various situations or makes calculations:

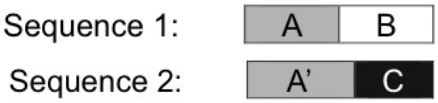
Calculation of “match percentage”

By default, SeqMan NGen uses a local match percentage which requires that the match percentage threshold be met in each overlapping window of 50 bases. The size of this window can be adjusted by specifying a different value for the **match window** parameter.

An example containing a repeated region follows.

A genome fragment has repeated regions labeled A and A’, and two unique regions labeled B and C.

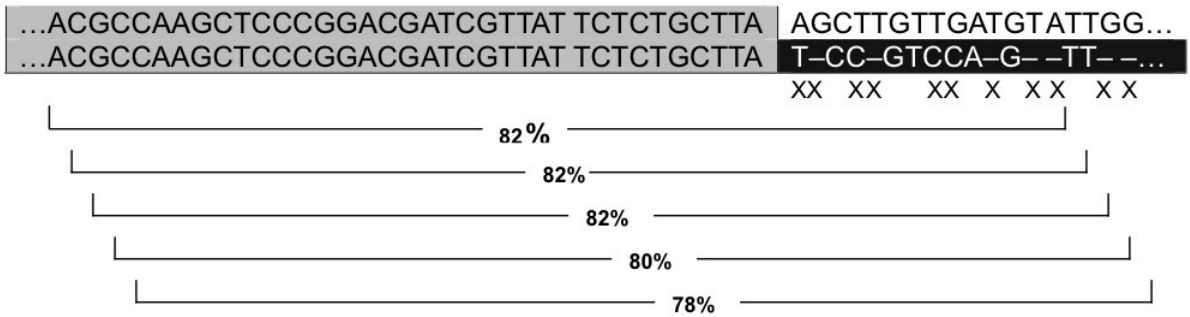
When the fragment is sequenced, one of the sequences contains parts of regions A and B, and another contains parts of regions A’ and C:



In this example, a **minimum match percentage** of 80% is used. When the two sequences are aligned, the 400 bases in the overlapping A and A’ regions match 100%. The 200 bases in the overlapping B and C regions match 42%. Over the entire alignment, 484 out of 600 bases match, yielding a global match percentage of 81%.

However, SeqMan NGen checks the match percentage for every alignment of 50 bases. The alignment below shows the last 36 overlapping bases of A and A’ and the first 18 overlapping bases of B and C. Each mismatch in the overlap is marked by an X below the alignment. In the first 50 bases shown, there are 41 matches, and the match percentage is 82%. This is above the threshold of 80%, so the match percentage of the next 50 bases is checked and is also found to be 82%.

Each fifty bases are checked along the overlap as long as the match percentage is at or above the threshold. In this case, the alignment fails once it gets far enough into the overlap of the unique regions, B and C, that the match percentage drops to 78%. The sequences will not be assembled together into a contig, which is correct for this data set.



Detection of structural variations

In addition to SNPs and small insertions and deletions, genetic variation can also involve large scale rearrangements. These rearrangements may include large insertions and deletions, inversions, and translocations — collectively known as *structural variations* (*SV's*). To view a tabular report with structural variation findings for an assembly, open the assembly in SeqMan Pro and select **Contig > Structural Variation Report**.

The rest of this topic describes how these structural variations were detected during assembly with SeqMan NGen.

During a templated data assembly, SeqMan NGen automatically detects insertions or deletions (*indels*) greater than 10 bp based on a combination of two data types:

Coverage – read depth can be suggestive of larger indels, including duplications or the collapse of a duplication.

Split reads – reads spanning a deletion in the new genome relative to the reference genome can be "split" into two segments based on matches to discontinuous regions on the reference. For example, the following split read alignment indicates there is a 35bp deletion in the new genome:

Ref: AGGCTGACCTCCTGGGCCTAAGACACTGAGTGCCCCAATATGACTCGACTAGCA

Read: AGGCTGACCTCGACTAGCA

Ref: AGGCTGACCTCCTGGGCCTAAGACACTGAGTGCCCCAATATGACTCGACTAGCA

Split: AGGCTGACCTC GACTAGCA

The SeqMan NGen algorithm requires that four criteria be met for splitting a read:

1. At least 20 bases on each "half" of the split must match the reference. This means that reads must be at least 40bp long, though in practice they should be > 60 bp.

2. The first mer match must be within 10 bases of the start of the read, and the final mer match must be within 10 bases of the end of the read. This increases the likelihood that the entire read will align after splitting.

3. The distance between the two closest mers on either side of the split must be within 20% of the total read length. For example, in a 100 base read where bases 5-30 make up the mer match on the 5' "half" of the read, then the first mer match on the 3' half must start between bases 31 and 50 (30+(100*0.2)=50) of the read. This relatively simple requirement allows for SNPs or sequencing errors near the actual split to be tolerated and resolved during alignment.

4. The two “halves” must be aligned in the same orientation.

In practice, two copies of the read are given to the aligner: one seeded with the 5’ mer match and the other with the 3’ mer match. The aligner then extends the alignment on both sides of each copy, and then trims each copy to maximize the final alignment score. It is the final trimmed internal position for each copy that is reported in SeqMan Pro’s Structural Variation Report.

Note: Because of the trimming and the flexibility in the location where the nearest internal mer match must begin (criteria #3 above), it is possible that base substitutions are present in the split region that will not be displayed in SeqMan Pro’s Alignment View. Thus, while split reads have far greater resolution power than coverage, the breakpoints identified must be considered as provisional.

Handling of repeats

Repeat handling parameters compute a threshold for deciding the number of identical subsequences of bases (mers) used to indicate a putative repeat. Mers that are common to two or more fragment reads are aligned to determine the overall layout of reads. For additional information, see How mer tags are chosen.

Repeat handling is controlled via the **Place repeat reads** drop-down menu in the Alignment tab of the Advanced Assembly Options dialog.

The repeat threshold can also be computed by multiplying the **Match repeat percent** parameter value in the Advanced Assembly Options dialog by the **Expected coverage**. Any mer that occurs more frequently than the computed threshold is not considered for use as a mer tag in determining overlaps. Coverage can be determined in two ways:

- By using the length of the genome/fragment being sequenced, as specified by the **Expected genome length** parameter in the Assembly Options dialog. If this option is used, SeqMan NGen calculates expected coverage by dividing the **Expected genome length** value by the total length of all sequences in the project.

- By using a fixed number for expected coverage, as specified by the **Expected coverage** parameter in the Assembly Options dialog.

Handling of sex chromosomes

When using the SeqMan NGen wizard, certain workflows allow you to specify the subject's **Gender** in the Assembly Options screen.

> **Note for command-line users:** When using the command-line version of XNG, the sex of the subject is instead specified using the **assembleTemplate** command by setting the **query** parameter value to **male**, **female**, or **unknown**.

SeqMan NGen treats all non-template package chromosomes as diploid, unless otherwise specified. However, SeqMan NGen's XNG assembler does recognize sex chromosomes in DNASTAR genome template packages. Since some regions of the X and Y chromosomes are homologous, read placement may be more accurate in females when reads are not falsely assigned to the Y.

Special handling of sex chromosomes occurs in two circumstances:

During the placement/layout of reads:

All samples in an assembly must be of the same sex; different sexes cannot be specified for individual MID samples. Chromosomes are recognized by GTP shortname (X Y W Z). One of the names must match exactly. Humans and many other animals have X/Y, while the chicken *(Gallus gallus)* has a W/Z. For some GTPs such as the cow *(Bos taurus)*, Y exists biologically, but has not been sequenced and is not provided in the genome template package.

Sex influences read placement as follows:

- If sex is left **Unknown**, all chromosomes will be available.
- If sex is set to **Female**, no reads are placed against the Y chromosome.
- If sex is set to **Male**, the haploid variant caller is used for both X and Y chromosomes; no reads are placed against the W chromosome.

+When calling SNPs and using the Bayesian SNP caller: +

Bayesian SNP calling is also modified by taking the sex chromosome into account. The XNG script normally controls whether calling is done as diploid or haploid. Whenever haploid is specified as a SNP method all templates are haploid. However even when diploid is chosen:

- chromosomes Y or Z are always haploid

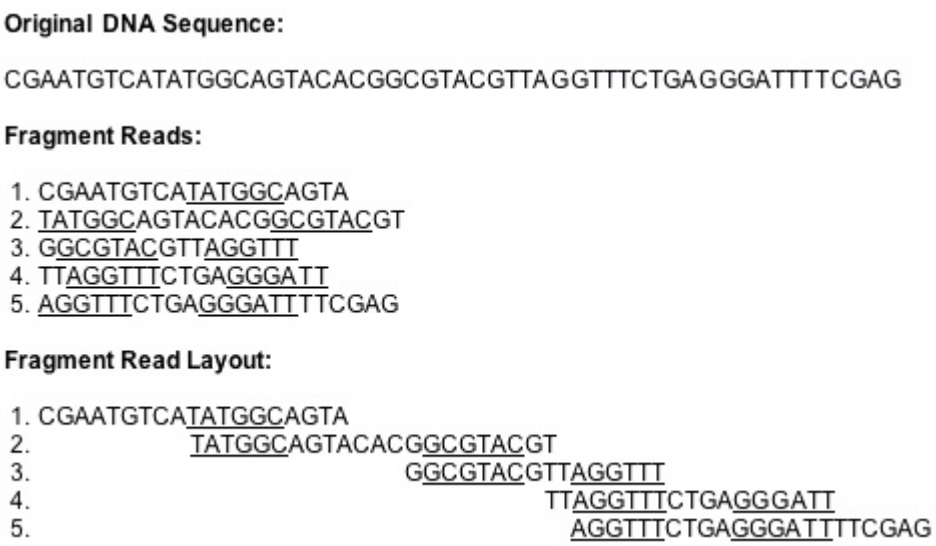
- chromosome X is haploid for males, diploid for female or unknown
- chromosome W is haploid for females, diploid for male or unknown

The mitochondrion is considered diploid, though it would usually be polyploid. If mitochondria are of interest, a separate assembly should be done using the simple SNP method, since the Bayesian caller depends on knowing that there are one or two chromosomes.

Special handling of sex chromosomes can be disabled in an XNG script by setting noSexChromosomes:true. Changing the shortname of a chromosome in a GTP will cause it to revert to normal autosomal behavior in all scripts using the GTP.

How “mer tags” are chosen

The SeqMan NGen layout algorithm relies on unique subsequences of bases, or *mers*, which occur in overlapping regions of fragment reads. Mers that are common to two or more fragment reads are aligned to determine the overall layout of reads. Overlapping reads have many mers in common, but only a few mers per overlapping region are needed to identify the overlap. These mers are called *mer tags*. The use of mers to tag fragments and identify overlaps is illustrated in the following figure:



As shown in the above figure, a 54bp original DNA sequence is covered by five overlapping fragment reads. The 6-mer tags for each fragment read are underlined. Matching mer tags are aligned to determine the layout of the reads.

The power of using mer tags relies on the ability of SeqMan NGen to choose mers that are most likely to occur only once in the original DNA sequence. It is important to avoid choosing mers that occur in repeated regions since the result may be fragment reads that are incorrectly aligned together.

Three parameters are involved in choosing mer tags: **Match Size**, **Repeat Handling**, and **Match Spacing**. All of these parameters can be adjusted in the Advanced Assembly Options dialog.

The **Match Size** and **Repeat Handling** parameters help to choose tags that are most likely to be unique in the original DNA sequence. **Match Size** sets the length of the mers. The longer the mer, the higher the probability that it is unique. **Repeat Handling** parameters help to identify which mers are not likely to be unique. If a mer occurs more often than expected in the dataset, the mer may be part of a repeated region.

Match Spacing specifies the preferred distance between mer tags. The smaller the **Match Spacing** parameter value, the more memory and more time the assembly will take. If a fragment read is shorter than the **Match Spacing** value, multiple mer tags are still chosen for the read.

Note: During assembly, any given read will only be assigned to one contig, even if it matches the hit criteria for more than one contig. If there is no information linking the read to a specific contig (e.g. a unique SNP or a paired-end constraint), SeqMan NGen will assign the sequence randomly to one of the contigs for which it meets the criteria.

Run SeqMan NGen through the command line

To create a script:

Either:

- Use the SeqMan NGen wizard to create the script. Rather than pressing the **Assemble** button at the end, simply save the script.
- Create a text file from scratch using the commands and parameters for XNG or SNG. Save the script with the file extension *.script*.

To run the script using the command line:

1. Open the command shell (Win) or Terminal (Mac).
2. Type one of the following commands:
    - sng *[path to script]*
    - xng *[path to script]*

Note: If SeqMan NGen does not run immediately, it is possible that the application has not been added to your environment variable PATH. After installing SeqMan NGen on a computer for the first time, command-line users may want to restart the computer before using the application. Restarting will add the application to your environment variable PATH. If you are unable to restart your computer, the following two options are available:

* Use the SeqMan NGen wizard to run the script.

* At the command-line, specify the path to the application, followed by the path to the script.

XNG, SNG, and QNG assemblers

SeqMan NGen uses three powerful assemblers: XNG, SNG and QNG.

The XNG assembler:

The XNG assembler (patent pending) is used for all reference-guided assemblies. This assembler features an algorithm for fast, accurate assembly of extremely large genomes and is capable of assembling data sets of any size, given sufficient disk resources and modest RAM requirements. The XNG assembler uses multiple cores, but the exact number varies over the course of the assembly.

The primary output is an assembly folder (Win) or package (Mac) containing a binary *.bam* file for each reference sequence to which reads could be mapped. The assembly folder also contains accessory files. Note that BAM files cannot be edited. Reference-guided workflows use the XNG assembler. The reference-guided assembly with gap closure uses both the XNG and SNG assemblers, but the output files are most similar to XNG outputs.

SNG/SMNG assembler:

The SNG assembler generates finished assemblies in SeqMan Pro / SeqMan Ultra (.sqd) or BAM (.assembly) formats. The *.sqd* files are editable within SeqMan Pro, but the number of data reads is limited to 10 million or fewer. BAM files of any size can be created, but may not be edited. The SNG/SMNG assembler uses one core during assembly. SNG/SMNG workflows include:

- All *de novo* assemblies. SNG/SMNG generates finished assemblies in editable SeqMan Pro / SeqMan Ultra format.
- Reference-guided assemblies for small genome (less than 30MB) reconstruction projects where editing is required. In order to perform editing, an output format of SeqMan Pro / SeqMan Ultra must be selected.

QNG assembler:

In this type of assembly, XNG is used to make a first-pass assembly. Then, a version of ArrayStar's QSeq algorithm is applied to the results to create the finished assembly.

For a list of the output files for a given workflow, see Access and Understand Output Files and its subtopics. In addition, View the Project Report describes how to access the most commonly-viewed output file.

XNG commands

The following commands can be used in the script for an XNG assembly. Click on the name of a command in the table below (shown in alphabetical order) to see a description and example, and to see the parameters associated with that command. All XNG commands and parameters are assumed to be optional unless the description states that it is required.

assembleTemplate	exportVCF	pause
computeSNP	extractPairs	quit
createGenomeTemplate	include	removeDuplicateSeqs
diskPath	loadAssembly	runScript
dumpConsensus	loadBAM	set
dumpSNP	mergeIonTorrentShortReads	setDefaultDirectory
execute	message	setMachineMemory
exportSplits	pairFilePattern	setParam

assembleTemplate

Note: All parameters are assumed to be optional unless the description is prefaced by "required."

assembleTemplate is a required command, and Initiates the assembly of the loaded sequences using the specified template as a reference.

Example:

XNG script used in the "clustering" step of the de novo transcriptome RNA-seq workflow:

merSize: 25
minNewClusterSize: 5
minSingleMergeClusterSize: 7
minMultiMergeClusterSize: 7
minMultiMergeIgnoreFactor: (currently not used by default)
minClusterSizeToOutput: 100

Parameter	Description	Allowed va (default underlin
alignmentCutoff	Used in the "clustering" step of the de novo transcriptome RNA-seq workflow.	[number] Default = 20
assemble	Specifies whether to use the part of the query that matches the contaminant sequence(s), the part that doesn't match, or both.	[matchCont noMatchCor all]
assemblyInfo	Contains information about the assembly.	[text string]
assemblyInfoAlt	Contains pairs of keys and values which will be written to the -0.assemblyInfo file.	
autoTrim	Specifies whether mismatching ends of reads should automatically be trimmed.	[true / false
autoTrim	Specifies whether mismatching ends of reads should automatically be trimmed.	[true / false
boneyardAssembly	Specifies whether sequences not used in the original or incremental XNG assemblies should be added to the assembly project by the SNG assembler. This command pertains only to reference-guided assemblies with gap closure. By default, during this type of assembly, the XNG	[true / false

	assembler first finds structural variations (SVs) then splits the contig after each SV. Elements of this process can be modified using this command. (**Note:** “Boneyard” is a term for sequences that were not assigned to any contig).	
combineDuplicateSeqs	Specifies whether the duplicate reads will be clustered.	[true / false]
contaminant	Use of this parameter partitions the query data by running an additional mer-match (layout) against the specified contaminant sequence(s). A full assembly is then run using the part of the query that either matches or does not match the contaminant sequence(s). This parameter can be used for removing reads originating from an organism(s) that may have also been present in the query data set (e.g., reads from human DNA present in a metagenomic sample from the human gut). file: [directory/filename enclosed in quotes] the file with contaminant sequences. assembleContam: [matchContam/noMatchContam/all] merLayoutMin: [number] unassembled: [directory/filename enclosed in quotes] the file containing no contaminant reads.	[directory/ filename enc in quotes]
dbSNPTable	(Intended for internal use only).	[directory/ filename enc in quotes]
delayAlignInserts	Use of this flag turns the delay reads that cause inserts on or off. ‘True’ means that gap causing reads will be delayed. Reads will be added such that reads causing the lowest number of inserts (length of inserts is not considered) will be added before those causing more inserts.	[true / false] Defaults: tru named read technologies false for ‘Oth read techno
deleteIntermediates	Specifies whether intermediate files are saved or deleted. These files can be large with large-scale projects.	[true / false / / all / notTemplate
directoryMer	Specifies the path and directory where both the template and query data mer files will be stored. Alternatively, separate directories for the template and query mer files can be specified using the parameters below. If no directory is specified, the mer file will be created in the directory containing the sequence data.	[directory/ filename enc in quotes]
directoryQueryMer	(required) Specifies the path and directory where the query mer file will be stored.	[directory/ filename enc in quotes]

directoryTemplateMer	(required) Specifies the path and directory where the template mer file will be stored.	[directory/ filename enc in quotes]
filterDeepLayout	(optional) Specifies that XNG remove superfluous sequences in areas of deep coverage. Wizard equivalent: Using 'true' is equivalent to selecting the **Advanced Assembly Options > Alignment tab > Limit all deep coverage regions** radio button	[true / false Set to 'false, default, exce projects invo miRNA or microbial genomes, w it is set to 'tr
filterDeepLayoutOrganelle	(optional) Specifies that XNG remove superfluous sequences in areas of deep coverage. Wizard equivalent: Using 'true' is equivalent to selecting **Advanced Assembly Options > Alignment tab > Only limit deep coverage regions for Mitochondria and Chloroplasts** radio button	[true / false Set to 'false, default, exce projects invo a mitochond chloroplast template (i.e those with a name of 'MT or 'CHL' or 'chloro'), wh is set to 'true
forceFullForwardAlign	Start the alignment at the 5' end of the sequence.	[true / false
forceMake	Specifies whether new intermediate mer files will be created. A value of 'false' means that existing valid intermediate files will be used.	[true / false query / hit / layout]
format	Specifies the format of the alignment output file. If 'none' is entered, the assembly is run to include the alignment phase, but no alignment output is generated. This parameter can be used to remove reads from a contaminant source.	[BAM / SQD NONE / NONE_align Aux_align]
gap5Prime	Put the gap on the 5' side of the sequence.	[true / false
gapPenalty	The penalty for opening or extending a gap during an alignment. This penalty is deducted from the pairwise score used to calculate match percentage. A high gap penalty suppresses gapping, while a low value promotes gapping.	[number] Default = 30 most workflo 50 for the de transcriptom

		RNA-seq workflow.
gapExtensionPenalty	Used in the “clustering” step of the de novo transcriptome RNA-seq workflow.	[number] Default = 5
geneticCode	This parameter specifies the genetic code to use with a reference sequence.	[filepath/star Lasergene genetic code name]
hits	(required) Specifies the path and name of the hit file. Incomplete paths will be appended to the default directory.	[directory/ filename enc in quotes]
increaseRunGapPen	This parameter is a flag to increase the gap open penalty in HP runs.	[true / false
layout	(required) Specifies the path and name of the layout file. Incomplete paths will be appended to the default directory.	[directory/ filename enc in quotes]
layoutAlign	Specifies that a pairwise alignment should be performed at the payout phase in order to pick the best position for a given read.	[true / false
layoutMaxTemplateGap	The maximal number of gaps introduced into the alignment used during layout.	[number]
layoutRSRange	The maximal Register Shift difference used while building the layout.	[number]
layoutType	Specifies how reads are to be laid out.	[unique / on multiple / multipleAll]
matchScore	The score for a base match during an alignment. This score contributes to the pairwise score used to calculate match percentage. Increasing the matchScore value allows for longer or more frequent gaps, thus forcing bases that match to be assembled together.	[number] Default = 10
MaxGap	The theoretical maximum length of a gap that could be inserted. In practice, the maximum gap size will usually be about half of this value.	[number fro 0-99] Default = 6 f most workflo 30 for the de transcriptom RNA-seq workflow
maxMergeSize	When linking clusters into a scaffold, only link them together if the overall	

	number of reads in the scaffold would not exceed this threshold. Used in the “clustering” step of the de novo transcriptome RNA-seq workflow.	
maxNCnt	(optional) This parameter removes sequential reads of the IUPAC ambiguity code ‘N’ that are greater than or equal to the number specified. Use of this parameter may help in assemblies whose reads contain large clusters of spurious N’s.	[integer]
maxSecondaryTrimLength	During alignment, a read can be trimmed from both ends. This parameter defines the longest allowable length for the smaller of the two trimmed ends.	[number]
maxSeqs	Specifies the maximum number of query sequences to add to an assembly. Use of this command can speed up assembly.	[number]
merCntThresh	Minimum number of mers needed in order to be recorded in the mer file.	[number]
merLayoutMin	Specifies the minimum length (in bases) of at least one stretch of matching mers used to identify matches between the reference and query data. The minimum value is equal to the mer. The maximum value is the read length, which would require the entire read be an exact match. For example, with a merSize of 19 and a merLayoutMin of 21, at least one stretch of three consecutive mers in a read would have to match for the read in order to be included in the layout.	[number fro 11-1000] Default = 25
merMinimizer	(Intended for internal use only)	[number]
merSize, merLength or matchSize	(required) Specifies the length (in bases) of mers used to identify matches between the reference and query data.	[number]
merSkip	(Intended for internal use only) Specifies the number of positions to ignore or “skip” when creating the template mer file. Normally, mers are only skipped in the query (see **merSkipQuery**, below). The first and last mer of every read are always included. Increasing the value reduces the size of the intermediate files as well as the overall assembly time. However, larger values can also reduce the number of reads included in the assembly, especially with short read data. 0 = do not skip 2 = skip every second base 3 = skip every third base etc.	[number] Default = 0
merSkipQuery	Specifies the number of positions to ignore or “skip” when creating the query mer file. The first and last mer of every read are always included. Increasing the value reduces the size of the intermediate files as well as the overall assembly time. However, larger values can also reduce the number of reads included in the assembly, especially with short read	[number] Default = 0

	data. 0 = do not skip 2 = skip every second base 3 = skip every third base etc.	
method	Defines how to handle splits in the assembly: * **normal** – normal assembly method * **splitOnly** – only reads which have been split will be included in the assembly * **noSplit** – no reads will be split	[normal/split noSplit]
minAlignedLength	Specifies the minimum number of bases that must align after trimming for a read to be included in the assembly.	[number fro 11-999] Default = 25 most workflo 50 for the de transcriptom RNA-seq workflow.
minClusterSizeToOutput	Threshold for the number of reads that a cluster must contain in order for the cluster to be passed along to SNG for assembly in the next step of the program. Used in the “clustering” step of the de novo transcriptome RNA-seq workflow. Note that this command is present only for the **clusterParam** block of the **rnaAssemble** command.	[number]
minMatchPercent	The minimum percentage of matches in an overlap required to join two sequences in the same contig.	[number] Default = 93 most workflo 60 for the de transcriptom RNA-seq workflow.
minMultiMergeClusterSize	When two or more clusters overlap the same k-mer, the minimum number of reads (depth) required at that k-mer for a cluster to consider that cluster significant. If three or more clusters exceed this threshold, the k-mer is considered	[number]

	"noisy" and a potential false join, and will not be merged. This is reported as a "multi-cluster link that was not merged". If two significant clusters overlap and have similar enough depth, the clusters are considered linked and are scaffolded together. Otherwise, if only one cluster is significant, all reads at that k-mer which have no assigned cluster are merged directly into it as described for the **minSingleMergeClusterSize** option. This parameter is used in the "clustering" step of the de novo transcriptome RNA-seq workflow. Note that this command is present only for the **clusterParam** block of the **rnaAssemble** command.	
minMultiMergeIgnoreFactor	When two or more clusters overlap the same k-mer and may be linked, they must be within this ratio of one other. Used in the "clustering" step of the de novo transcriptome RNA-seq workflow. Note that this command is present only for the **clusterParam** block of the **rnaAssemble** command.	[number]
minSeqsPerTemplate	Minimum number of sequences sufficient to build the layout or alignment.	[number]
minSingleMergeClusterSize	The minimum number of reads (depth) matching an existing cluster at a single k-mer required to extend that cluster by immediately adding all new reads for that k-mer to the cluster. Used in the "clustering" step of the de novo transcriptome RNA-seq workflow. Note that this command is present only for the **clusterParam** block of the **rnaAssemble** command.	[number]
minNewClusterSize	Minimum number of matching reads at a single k-mer (i.e., "depth") required to create a new cluster. Used in the "clustering" step of the de novo transcriptome RNA-seq workflow. Note that this command is present only for the **clusterParam** block of the **rnaAssemble** command.	[number]
mismatchPenalty	The penalty for a base mismatch during an alignment. This penalty is deducted from the pairwise score used to calculate match percentage.	[number] Default = 20
noSexChromosomes	Disables special handling of sex chromosomes.	[true / false
noSVPairSort	Specifies whether to turn off the calculation of pairs for structural variations. This may potentially reduce XNG assembly time.	[true / false
onePackage	Specifies whether an assembly containing multiple reference sequences should be bundled into a single .assembly package. If 'false' is entered,	[true / false

	one .assembly package is created per contig.	
openInSeqman	(optional) Specifies whether the completed assembly should immediately be launched in SeqMan.	[true / false]
output	(required) Specifies the path and directory of the output files. Incomplete paths are appended to the default directory.	[directory/ filename enc in quotes]
pairDist	(Intended for internal use only)	[true/ false]
pickTemplate	Defines the number of templates from which to choose, and finds the template that is the best match for the input sequence.	[number]
placeHit	(Intended for internal use only)	[true / false
probe	(Intended for internal use only)	[number]
query	(required) Specifies the directory and file name(s) of the query data to be assembled. A folder with one or data files can also be used in place of individual file names. Properties for query: file: [directory/filename enclosed in quotes] Specifies the directory and file/folder. isPair: [true/false] Specifies whether the query files contain paired end data. minDist: [number] (required if isPair is 'true') Specifies the minimum expected distance in bases between paired end reads. Default is 0. maxDist: [number] (required if isPair is 'true') Specifies the maximum expected distance in bases between paired end reads. Defaults are 750 for Illumina; 4500 for 454 and Sanger, 7500 for Other, and user-defined for Ion Torrent seqTech: [unknown|IonTorrent||IlluminaLongReads|454|PacBio|normalScore|Other] Specifies the offset to be used when converting compressed quality scores into numerical values. These are the offsets used for the technology specified:	[directory/ filename enc in quotes]

Data Type	Value	Offset
IonTorrent	IonTorrent	33
Illumina	IlluminaLongReads	33
Roche 454	454	33
Other types	normalScore	33

Note 1: For 454,quality scores for homopolymeric runs of ≥ 2 are oriented from 5' to 3' on the top strand.

Note 2: If possible, the data type of unknown data is determined automatically based on the first data file.

pairTech : [unknown|LucigenRsaI|LucigenBfaI|Rsa1|Bfa1|Custom]

pairLinker: [string]

groupName: [string] The name of a group this file belongs to. Used for running multiple samples in one file.

sex: [unknown|female|male]

trim: [true / false] Specifies whether vector trimming needs to be applied to the reads.

sngTrim: contains parameters for fast vector trimming (See the SNG command **trimVector**)

scan: [true / false] Specifies whether reads needs to be scanned for contaminants

contaminantScan: Contains the assembleTemplate command with contaminant file used as a template and parameters: directoryTemplateMer, hits, layout, output, unassembled, results, format, mersize, ignorePolyMers and deleteIntermediates. The format parameter has valuenone_ALIGN.

Example:

query: {{file: "/data/home/proj/Illumina_s_5_1.txt"}
{file: "/data/home/proj/Illumina_s_5_2.txt "}
isPair: true
minDist: 400
maxDist: 700

	seqTech: Illumina}	
recordSplitsOnly	Functional only when used in the same program as splitTemplateContigs or recordStructVariations (both described below). Specifies whether or not to turn off contig splitting while still recording SVs for later inclusion in the Structural Variation Report.	[true / false
recordStructVariations	Specifies under which circumstances structural variations (SVs) should be calculated and recorded. 0|false = Don't calculate SVs 1|true = Calculate SVs at zero coverage 2 = Calculate SVs at insertions and deletions 3 = Calculate SVs at zero coverage and at insertions	[integer bet 0-3 / true / fa Default = 2
removeDuplicateSeqs	Completely removes clonal reads after the alignment phase of assembly. Clonal reads, where the endpoints of both reads in a pair match those in another pair, are usually the result of PCR artifacts. If 'true,' the reads will not be scored, and will not be included in SNP calculations. Marking this parameter to 'true' may substantially increase the time needed for assembly.	[true / false
removeUniqueInserts	Removes reads that cause an insert which no other read would create. This parameter is only enabled when **delayAlignInserts** (described under the assembleTemplate command) is true.	[true / false Defaults: tru Illumina and Torrent read technologies false for all types.
repeatPenaltyScale	Indicates the quality penalty (using the Phred scale) to use for a read which places in two locations identically. Higher repeat counts are further penalized relative to this on a $\log_2$ scale such that repeats placing in four locations have a double penalty, in eight locations have a triple penalty, and so on. This penalty is applied to a ceiling of Phred score 30 if the other methods are disabled or have a higher score.	[number] Default = 8
repeatThreshMax	Specifies the maximum number of occurrences of a mer in the reference sequence(s) for it to be considered repeated. Mers exceeding this number will not be used for identifying matches.	[number fro 1-10000] Default = 10
repeatThreshMin	Specifies the minimum number of occurrences of a mer in the reference sequence(s) for it to be considered repeated. Mers less than this number will not be used for identifying matches.	[number]
reportFiles	Defines the kind of report file to be generated.	

	perProject: [true / false] Generate a per project report. perTemplate: [true / false] Generate a per template report. removeInteral: [true / false] Remove intermediate reports.	
repeatmermax	Threshold number of occurrences in a data set for a mer to be considered “repeated.” Used in the “clustering” step of the de novo transcriptome RNA-seq workflow.	
results	Specifies the path and name of the result summary file. This file contains a compilation of assembly statistics and uses the extension fileSize.txt. Incomplete paths will be appended to the default directory.	[directory/ filename enc in quotes]
saveUnSplitAssembly	Specifies whether XNG should save both the normal assembly output, [filename].assembly, and the unsplit intermediate assembly, [filename]-noSplit.assembly. The latter file contains SVs but no SNPs, and can be used to validate splits in the final assembly.	[true / false]
sex	Specifies the sex of the subject, used for read placement and SNP calling. See How sex chromosomes are handled for details.	[male / fema unknown]
showCDSVariant	Specifies whether or not XNG should show all variants of a CDS feature contacted by a SNP. The version number for the CDS variant will then appear in brackets when viewed in the SNP report in SeqMan Pro.	[true / false
sngConvertOptions	(Intended for internal use only)	[text string]
snp	Specifies whether or not a SNP detection pass of the gapped alignment should be made during the assembly.	[true / false
snp_checkStrandedness	Specifies whether or not the strand that each read comes from is considered in the SNP calculation. This is ignored by the simple SNP calling method (used when genome ploidy is “Heterogeneous”).	[true / false
snp_combineSubs	This parameter is used to coalesce adjacent substitutions.	[true / false
snp_excludeBases3p	(internal use only) This parameter causes the specified number of bases from the 3’ end of each read to not be considered during variant calling.	[integer]
snp_excludeBases5p	(internal use only) This parameter causes the specified number of bases from the 5’ end of each read to not be considered during variant calling.	[integer]
snp_excludeBasesEdge	This parameter causes the specified number of bases from both the 5’ and 3’ ends of each read to not be considered during variant calling.	[integer] For the simp SNP calling method (use when genon

		ploidy is “Heterogene the default is For the Baye SNP calling methods (us when genom ploidy is **Dip** or **Haploid**), default is 0.
snp_limitEndPos	Specifies the 3’ most coordinate of the specified template from which to stop calculating SNPs.	[number bet 1 and the le of the templa
snp_limitStartPos	Specifies the 5’ most coordinate of the specified template from which to begin calculating SNPs. A value between 1 and the length of the template must be entered.	[number] Default = 1
snp_limitTemplateID	Specifies a single template ID for which to calculate SNPs.	[number] Default = 0
snp_logEndPos	Specifies the 3’ most coordinate of the specified template from which to stop storing a detailed log of SNP information. A value between 1 and the length of the template must be entered.	[number] Default = 1
snp_logLevel	Specifies the level of detailed logging to store in the “shared” project directory as “SNP.log.” Level 0 specifies that no log will be stored. Level 1 stores detailed info on the SNPs which were called, level 2 also logs columns where the preliminary filtered passed but the final filtering failed, and level 3 logs all columns. This is ignored by the simple SNP calling method (used when genome ploidy is “Heterogeneous”).	[whole numb from 0-3] Default = 0
snp_logStartPos	Specifies the 5’ most coordinate of the specified template from which to begin storing a detailed log of SNP information. A value between 1 and the length of the template must be entered.	[number] Default = 1
snp_logTemplateID	Specifies a single template from which to store a detailed log of SNP information.	[number] Default = 0
snp_maxRun	Specifies the maximum length of a homopolymeric run for an indel to be considered during variant calling. For example, a snp_maxRun of ‘5’ will allow a portion of sequence up to 5 bases in length to be called as a SNP.	[integer] Defaults are 454 and Ion Torrent read technologies

		for all others
snp_maxStrandBias	Strand Bias (SB) for a SNP is the bias for the SNP appearing on one strand versus the other. It is measured relative to the strand bias in the assembly at the location of the SNP. For example, in a column with 60 forward reads and 40 backward reads, 6 SNP bases on the forward strands, and 4 on the reverse strands would be unbiased. SB is given by the formula: SB = |SNP% $_f$ – SNP% $_r$ | / Total SNP% …where SNP% $_f$ and SNP% $_r$ are the percentage of reads containing the variant on the forward (top) and reverse (bottom) strands, respectively; and SNP% is the total percentage of reads containing the variant. SB is calculated based on an “absolute value,” and will therefore be a positive number. The effect of different SB thresholds is shown below: -1 – A negative number cannot normally be generated by the equation above. However, you may use ‘-1’ in the script to turn off the **snp_maxStrandBias** parameter. In the wizard, SeqMan NGen indicates the parameter is turned off by making **Maximum strand bias** (see Variants tab) either blank or absent. 0 – Perfectly balanced (unbiased) strands. Reads with variants are present on both strands, and variants appear equally on both strands. Between 0-1, not inclusive – As the number ‘1’ is approached, more variants are called with unbalanced variants containing reads at that position 1 – All variant-containing reads are on a single strand. **Note:** In cases where all the reads covering a base are on one strand only, the SNP% of the other strand cannot be calculated (due to a “division by zero” error). These positions will not be removed by the snp_maxStrandBias filter. To remove these variants, instead set snp_minStrandCov to ≥ 1. Example: In a homozygous case (SNP% = 100) with a depth of 100, where 75 variant containing reads are on the top strand (75%) and 25 variant	[integer] Defaults for Bayesian SN calling meth (used when genome ploi “Diploid” or “Haploid”) a for 454 and Torrent read technologies shown (blan all others. Defaults for simple SNP calling meth (used when genome ploi “Heterogene are 0.25 for read technologies

	containing reads are on the bottom strand (25%), the strand bias would equal: (75 – 25)/100 = 0.5.	
snp_minHomopolDelDepth	Specifies the minimum read depth required to call a deletion in a homopolymeric run.	[integer] Default = 0
snp_minHomopolDelFrac	Specifies the minimum fraction of reads required to call a deletion in a homopolymeric run.	[integer] Default = 0
snp_minHomopolInsDepth	Specifies the minimum read depth required to call an insertion in a homopolymeric run.	[integer] Default = 0
snp_minHomopolInsFrac	Specifies the minimum fraction of reads required to call an insertion in a homopolymeric run.	[integer] Default = 0
snp_minPctToScore	Specifies minimum percentage of reads in a column which must differ from the reference in order to score the column. For the simple SNP calling method (used when genome ploidy is “Heterogeneous”), this is the only criteria used to call a SNP. For the Bayesian SNP calling methods (used when genome ploidy is “Diploid” or “Haploid”), this is a filter applied before the other parameters.	[number fro Default = 0.0
snp_minProbNonrefToCall	Specifies the minimum probability of a SNP column which is required to call a SNP, expressed as a number from 0 and 1. The probabilities of all genotypes other than Homozygous Reference are totaled and checked against this number. This is the final filter applied during the Bayesian SNP calling methods (used when genome ploidy is “Diploid” or “Haploid”) and is ignored by the simple SNP calling method (used when genome ploidy is “Heterogeneous”).	[number fro Default = 0.1 requiring a minimum 10 change.
snp_minStrandCov	Specifies the minimum number of reads from each strand required to call a variant at a given position.	[integer] In the Bayes SNP calling methods (us when genom ploidy is “Di or “Haploid”) default is 0. simple SNP calling meth (used when genome ploi

		“Heterogene the default i
snp_minVariantDepthToScore	(required if “snp” is true) Specifies the minimum depth required for a specific base (or deletion) in a column before it is considered usable for SNP calling. This is the second filter applied during the Bayesian SNP calling methods (used when genome ploidy is “Diploid” or “Haploid”) and is ignored by the simple SNP calling method (used when genome ploidy is “Heterogeneous”).	[number fro 0-100] Default = 2
snp_minWeight	Called “Minimum base quality score” in the SeqMan NGen wizard, this parameter specifies the minimum quality score for a base to be considered in the SNP calculation.	[number] In the simple calling meth (used when genome ploi “Heterogene the default i In the Bayes SNP calling methods (us when genom ploidy is “Di or “Haploid” default is 5.
snp_reportUserMissing	Specifies what kind of positions to put in the missingUser file, including one or more of the following: dbSNP = dbSNP Pos user = in user VCF SNP file zeroCoverage = include zero coverage regions cosmic = in COSMIC database allcaptured = include all positions in capture regions captured = include only positions in capture regions Example: snp_reportUserMissing: [user allcaptured captured] [kParamTypeStrFixedVocab]	
snp_runVar	Uses a Bayesian probabilistic model to exclude heterozygous insertions and deletions in homopolymeric runs. Intended for use with Ion Torrent data.	[true / false Defaults: tru 454 and Ion Torrent read

		technologies false for all others.
snp_showAllFeatures	Specifies whether XNG should count SNPs multiple times if the SNP contacts different versions (variants) of a CDS feature.	[true / false
snp_writeExtended	Specifies whether the additional values produced by the Haploid or Diploid SNP calculation methods are included in the SNP table. Wizard equivalent: **Advanced Options > Alignment tab > Trim to targeted regions**	[true / false
snpMethod	Specifies the SNP detection method to use. Simple produces a count of each type of base in the column and calculates the percent of non-reference bases. Haploid uses a Bayesian statistical model to calculate a probability score that the position contains a polymorphism and give a quality score for the base called at that position. Diploid uses a Bayesian statistical model to calculate a probability score that the position contains a polymorphism and give a quality score for the base(s) called at that position. Based on the scores, it also calls the genotype at each position.	[simple / ha diploid]
splitTemplateContigs	Specifies under which circumstances contigs should be cut after a templated assembly. Any split contigs will be grouped into scaffolds with a defined position to allow for easy sorting when the project is viewed in SeqMan Pro. This command pertains only to reference-guided assemblies with gap closure. By default, during this type of assembly, the XNG assembler first finds structural variations (SVs) then splits the contig after each SV. Elements of this process can be modified using this command. 0|false = Don't split 1|true = Split at locations with zero coverage 2 = Split at insertions and deletions 3 = Split at zero coverage and at insertions	[integer bet 0-3 / true / fa Default = 2
template	(required) Specifies the directory and file name of the reference sequence file. A folder with one or more reference sequence files can also be used in place of individual file names. Each entry must also be enclosed by brackets. If more than template entry is used, the list must also be enclosed by an additional set of brackets. Properties for template: file: [directory/filename enclosed in quotes]	[directory/ filename enc in quotes]

	Specifies the directory and file/folder. feature: [directory/filename enclosed in quotes] (optional) Specifies the directory and file name for annotated features when the reference sequence and feature annotations are in separate files. transcriptKind: [both|identified|novel] if the .*Transcriptome* package is used as a template, defines which transcripts will be used as a template. userSNP: [directory/filename enclosed in quotes] exomeCapture: file: [directory/filename enclosed in quotes] The BED file name. track: [string] the region of interest (Optional) merMask: [true / false] Specifies if mers from outside of the capture region should be excluded from assembly. Examples for template: Sequence and annotation in one file: AssembleTemplate template: {{file: “/data/home/proj/MG1655.gbk”} {file: “/data/home/proj/W3110.gbk”}} Sequence and annotation in separate files: AssembleTemplate template: {file: “/Library/ABC_proj/references/MG1655.fas” feature: “/Library/ABC_proj/references/MG1655.gff”}	
templateHitCntThresh	(Intended for internal use only)	[number]
trimToTargetRegions	Controls whether reads are trimmed, by default, to the boundaries of the targeted regions, as defined by the .bed or manifest file. The default of true indicates that the reads are trimmed to the stated boundaries. If conditions are not met, the SeqMan NGen wizard does not change this parameter to ‘false,’ but instead omits it from the script. The parameter status is only shown in the script for control workflows.	[true / false

	Wizard equivalent: **Trim to targeted regions** in the Alignment tab of the Advanced Assembly Options dialog.	
unassembled		[directory/ filename enc in quotes]
verify		[true / false

computeSNP

Sets parameters for the SNP computation phase of the assembly. The command is designed for use with existing BAM files that have not been analyzed for SNPs, or to re-analyze an existing file with different parameters.

Most of the parameters for **computeSNP** are identical to parameters for **assembleTemplate** and are discussed in that topic:

showCDSVariant
snp_logLevel
snp_minProbNonrefToCall
snp_checkStrandedness
snp_logStartPos
snp_minStrandCov
snp_combineSubs
snp_logTemplateID
snp_minVariantDepthToScore
snp_excludeBases3p
snp_maxRun
snp_minWeight

All other parameters are described in the table below:

Parameter	Description	Allowed values (defaults are underlined)
calcJunctionSeqs	In the structural variation workflow, specifying 'false' prevents junction sequences from being calculated.	[true / false]
concurrentAligns	(Intended for internal use only)	[number]
file	(required) Specifies the path and name of one or more .assembly projects from which to compute SNPs.	[directory/filename enclosed in quotes]
snp_writeMissingDBSnps	In a SNP assembly, specifying 'false' causes missing SNPs not to be recorded, saving time and file space.	[true / false]
snpFilter	Specifies whether SNP filtering is turned on or off. Properties for snpFilter:	

capture: [true / false]
Specifies whether there is an exome capture file. If an exon capture file is added in the SeqMan NGen wizard or through a script, this value is set to ‘true.’ In the absence of an exome capture file, the SeqMan NGen wizard automatically sets this property to ‘false.’

pNotRefMinVal: [number]
In the unusual case that the hard filter is missing, this property is used to set the minimum value that can be displayed in the SeqMan SNP table. Otherwise, this property is ignored. Default is 10.

userOnly: [true / false / All]
Specifies whether there is a VCF SNP file. The SeqMan NGen wizard always calls this as ‘true’ (or ‘yes’) but ignores the property if no VCF SNP file has been loaded.

pNotRef: [number]
This parameter is equivalent to Wizard option **Assembly Options > SNP Filter Stringency > pNotRef**. This parameters is a “soft” filter used to specify a PnotRef threshold. Data not matching the criterion are removed from the default display of the SeqMan Pro SNP table. This option is only available for the Bayesian SNP calling methods (used when genome ploidy is “Diploid” or “Haploid”). Wizard values include Low (90%), Medium (99%) and High (99.9%).

minSnpFilter: [number]
This parameter does not relate to any setting in the SeqMan NGen wizard, but corresponds to “SNP%” in SeqMan Pro and “minSNPFilter” in ArrayStar. In the simple SNP calling method (used when genome ploidy is “Heterogeneous”), the default is 5% for 454 and Ion Torrent read technologies; 1% for all others. In Bayesian SNP calling methods (used when genome ploidy is “Diploid” or “Haploid”), the default depends on stringency and ploidy rather than the read technology. The default for Diploid is 15% for all stringency levels. The default for Haploid is 25% for low stringency, 50% for medium and 75% for high.

minDepth: [number]
(optional) Specifies a minimum sequence depth threshold. This parameter does not relate to any setting in the SeqMan NGen wizard, but corresponds to **Depth** in SeqMan Pro and **minDepth** in ArrayStar. In the simple SNP calling method (used when genome ploidy is “Heterogeneous”), the default is 50. In Bayesian SNP

	calling methods (used when genome ploidy is **Diploid** or **Haploid**), the default is 20. The following set of SNP filters are used by ArrayStar and SeqMan Pro: **codonOnly**: [Coding / CodingChange / Nonsense / All] **maxDepth**: [number] **maxCodingFeatureDistance**: [number] **minSnpFilter**: [number] **qCall**: [number] **synonymousCodingChange**: [true/false] **substitionCodingChange**: [true/false] **noStartCodingChange**: [true/false] **noStopCodingChange**: [true/false] **nonsenseCodingChange**: [true/false] **frameshiftCodingChange**: [true/false] **notCodingCodingChange**: [true/false] **inFrameIndelCodingChange**: [true/false] **refOnly**: [Reference/Unique/All] **cosmicOnly** : [Yes/No/All] **minIndelSize**: [number] **gerpScore**: [number] **substitution**: [true/false] **showIndels**: [true/false]	
userSNP	Specifies a location for storing the VCF SNP table.	[directory/ filename enclosed in quotes]

createGenomeTemplate

The command **createGenomeTemplate** is intended for internal use only.

Parameter	Description	Allowed values (defaults in bold/ underline)
file	Specifies the directory and file/folder of the input file.	[directory/filename enclosed in quotes]
output	The path and name of the output file.	[directory/filename enclosed in quotes]

diskPath

The **diskPath** command is required, and defines the default directory where temporary intermediate files from the assembly will be stored. The files can be large with large scale projects. Visit our website to view space requirements for a range of representative projects.

Parameter	Description	Allowed values (defaults underlined)
clean	Specifies whether or not to clean the merge disk. When automated scripts are being run simultaneously or sequentially, this command can be useful for emptying the merge disk between assemblies.	[true|false]
pathMac	Specifies the default path and file name for Macintosh.	[directory/ filename enclosed in quotes]
pathWin	Specifies the default path and file name for Windows.	[directory/ filename enclosed in quotes]
path	(required) Specifies the default path and file name. Example: diskPath path: “/data/proj/”	[directory/ filename enclosed in quotes]

dumpConsensus

The command **dumpConsensus** is intended for internal use only, and is used to convert the binary consensus file created during assembly into a text file.

Parameter	Description	Allowed values (defaults underlined)
file	Specifies the directory and file/folder.	[directory/filename enclosed in quotes]

dumpSNP

The command **dumpSNP** is intended for internal use only, and creates a tab delimited text file from one or more SNP containing binary files generated during assembly. SNP binary files include those with the .snpExt suffix contained in an .assembly package as well as those with either the .coverage.missingSNP or .nocoverage.missingSNP suffix contained in the _shared folder. To convert all the .snpExt files in a package simply use the .assembly name.

Parameter	Description	Allowed values (defaults underlined)
file	(required) Specifies the path and name of .assembly package (all SNP files will be included), one or more individual .snpExt files or either/both of the missingSNP files.	[directory/filename enclosed in quotes]
output	(required) Specifies the path and name of the output file.	[directory/filename enclosed in quotes]
refPos_end	To export SNPs with positions lower than this value.	[number]
refPos_start	To export SNPs with positions higher than this value.	[number]
snp_maxProbNonrefToCall	Lower limit for probability scores for exported SNPs.	[number]
snp_minProbNonrefToCall	Lower limit for probability scores for exported SNPs.	[number]
snp_type	Specifies which SNP file from the .assembly to use as an input.	[simple|SNP|missing|user|stats|userIDOnly]
templateID	Defines the template for which the SNP will be exported.	[number]
onefile	Defines whether all SNPs should be placed into one file.	[true|false]

execute

The command **execute** executes any shell script command.

Parameter	Description	Allowed values
command	Text for any shell script command.	[text string]

exportSplits

The command **exportSplits** is intended for internal use only, and is used to convert the binary splits file created during assembly into a text file.

Parameter	Description	Allowed values
file	Specifies the directory and file/folder.	[directory/filename enclosed in quotes]
output	The path and name of the output file.	[directory/filename enclosed in quotes]

exportVCF

The command **exportVCF** is used to accept the exome capture file and VCF file and builds another VCF file containing SNPs only in the capture regions.

Parameter	Description	Allowed values
userSNP	User SNP file.	[directory/filename enclosed in quotes]
exomeCapture	**file**: [directory/filename enclosed in quotes] Exome capture file. **track**: [text string] The name of the region of interest.	
output	The output VCF file.	[directory/filename enclosed in quotes]

extractPairs

The command **extractPairs** is used to create a tab delimited table of pair end information.

Parameter	Description	Allowed values
file	The path and name of any pair distance file (*.pairdist* file) from within a project's shared folder.	[directory/filename enclosed in quotes]
output	The path and name of the output file.	[directory/filename enclosed in quotes]

include

The command **include** is used to call up additional lines of script previously stored in a text file. In this way, a group of commands can be shared between two or more scripts.

Parameter	Description	Allowed values
file	Specifies the directory and file/folder.	[directory/filename enclosed in quotes]

loadAssembly

The command **loadAssembly** is intended for internal use only.

Parameter	Description	Allowed values
file	Specifies the directory and file/folder.	[directory/filename enclosed in quotes]

loadBAM

The command **loadBAM** is used to set parameters for analyzing existing BAM files. It allows ungapped BAM files to be converted into a fully gapped assembly file or to re-gap an existing file with different parameters. The command also permits SNPs to be calculated or re-calculated with different parameters starting with an existing BAM file. The associated parameters are also available for full assemblies and are described under the **assembleTemplate** command.

Parameter	Allowed values
align	
delayAlignInserts	[true|false]
format	
gapPenalty	
increaseRunGapPen	[true|false]
layout	
matchScore	
minAlignedLength	[0-999]
minMatchPercent	
mismatchPenalty	
output	
removeUniqueInserts	[true|false]
snp	
snp_checkStrandedness	
snp_clusteredPosFilterMinDev	[number]
snp_clusteredPosFilterMinFromEdge	[number]
snp_hetKnownThresh	[number]
snp_hetThresh	[number]
snp_limitEndPos	
snp_limitStartPos	
snp_limitTemplateID	
snp_logEndPos	

snp_logLevel	
snp_logStartPos	
snp_logTemplateID	
snp_maxRun	[number]
snp_maxStrandBias	[number]
snp_minHomopolDelDepth	[number]
snp_minHomopolDelFrac	[number]
snp_minHomopolInsDepth	[number]
snp_minHomopolInsFrac	[number]
snp_minPctToScore	
snp_minProbNonrefToCall	
snp_minStrandCov	[number]
snp_minVariantDepthToScore	
snp_minWeight	
snp_nlMutationRate	The chance that any single base is different from the reference. The default value of 0.0013 is equivalent to ~4 million variations in a Human sample against the reference or several thousand in a bacterial genome. [number]
snp_observedInControlFilterMaxCount	[number]
snp_observedInControlFilterMaxFrac	[number]
snp_proximalGapFilterMaxDel	[number]
snp_proximalGapFilterMaxIns	[number]
snp_proximalGapFilterWindowSize	[number]
snp_reportUserMissing	[dbSNP|user|zeroCoverage|cosmic|allcaptured|captured]
snp_runVar	[true|false]
snp_showAllFeatures	[true|false]
snp_writeExtended	
snp_writeMissingDBSnps	[true|false]
snpMethod	

snpRefAsm	[quoted file name]
template	

mergeIonTorrentShortReads

When using Ion Torrent data, the **mergeIonTorrentShortReads** command causes overlapping short reads to merge into mini-contigs.

Parameter	Description	Allowed values
output	(required) Specifies the path and directory of the output files.	[directory/ filename enclosed in quotes]
query	(required) Specifies the directory and file name(s) of the query data to be assembled. A folder with one or data files can also be used in place of individual file names.	[directory/ filename enclosed in quotes]

message

The **message** command is used to write out the string to the standard output.

Parameter	Description	Allowed values
str	Specifies the string to be written to the standard output.	[text string]

pairFilePattern

The **pairFilePattern** command allows you to specify the pattern for pair files using the GREP language.

Example:

pairFilePattern
  forward: “(?‘name’.*)_R1_(?‘ext’.*)\fastq
  reverse: “(?‘name’.*)_R2_(?‘ext’.*)\fastq

Parameter	Description	Allowed values
forward	A naming pattern to match forward clones.	[text string enclosed in quotes]
reverse	A naming pattern to match reverse clones.	[text string enclosed in quotes]

pause

The command **pause** is used to create a pause. It can be used when running table scripts to stop at any point.

Example:

pause
  prompt: “Table script paused. Press Enter to continue.”

Parameter	Description	Allowed values
prompt	Text that should appear in the console. The pause is terminated by hitting the **Enter** key.	[text string enclosed in quotes]

quit

The **quit** command is used to terminate a script. This command does not have any parameters.

removeDuplicateSeqs

The **removeDuplicateSeqs** command is used to coalesce multiple identical reads at the same position into a single read, provided the reads match the reference exactly. If this feature is active, at the end of assembly, XNG will print the message: “Coalesced $lld identical reads that matched the template exactly.” This command does not have any associated parameters.

Allowable values are [true|false] The default is false.

runScript

The **runScript** command allows you to batch multiple projects of the same type (e.g. assembly, computeSNPs). There are required three file: 1) a runScript file with variables, 2) a file with a table of values for the variables, and 3) a script file specifying the action to be carried out.

Example (runScript file):

```
setDefaultDirectory directory: “.”
  set $force: false
  set $DataDisk: “/Volumes/Raid/DataDisk”
  set $ResultDisk: “/Volumes/ResultDisk”
  set $MergeDisk: “/Volumes/MergeDisk0”
  set $snp:true
  set $snpMethod:”Diploid”
  set $repCnt:100
  set $merLayoutMin:19
  diskPath path: {“${MergeDisk}/mergeSort Data”}}
  runScript table: “testAssembly.txt” script: “testAssembly.template.script”
```

Example (table file):

```
defaultDir template query isPair seqTech project merSize snp snpMethod
“${ResultDisk}/rice” ${DataDisk}/rice.genome ${DataDisk}/rice FALSE Illumina rice 21 TRUE Diploid
“${ResultDisk}/ecoli” ${DataDisk}/Ecoli.gbk ${DataDisk}/ecoli TRUE Illumina Ecoli 21 TRUE Diploid
“${ResultDisk}/Exome” ${DataDisk}/GRCh37.gbk ${DataDisk}/Sample1 FALSE 454 HuEx 19 TRUE Diploid
```

Example (script file):

```
; “assembly.template.script”
setMachineMemory memory:32
setDefaultDirectory directory: $defaultDir
compareSeqs template: $template
query: {file: $query
isPair: $isPair
seqTech: $seqTech}
directoryMer: “intermediateFiles”
; directoryQueryMer: “intermediateFiles”
hits: “intermediateFiles/${project}.hits”
layout: “intermediateFiles/${project}.layout”
output: “results_${mersize}_${merSkipQuery}/${project}”
; results per project results: “${project}.results.txt”
```

```
; aggregate all results
results: “${ResultDisk}/assembly.results.txt”
merSize: $mersize
merSkipQuery: $merSkipQuery
repeatCnt: $repCnt
merLayoutMin: $merLayoutMin
layoutType: once
maxGap: 6
format: BAM
onePackage: true
snp: $snp
snpMethod: $snpMethod
; snp_writeExtended: true
forceMake: $force
```

Parameters for this command are described below:

Parameter	Description	Allowed values
script	The filename and location of the script.	[directory/filename enclosed in quotes]
table	The filename and location of the file containing text strings and numbers values for each variable.	[directory/filename enclosed in quotes]
inline	Executes the list of commands and parameters.	

set

The command **set** is used to set variables. It does not have any associated parameters. See the example below and those under the **runScript** command.

Example:

set $snp:true
set $snpMethod:”Diploid”

setDefaultDirectory

The **setDefaultDirectory** command (required) defines the default directory for the project. When a default directory is specified (see table below), files located in that directory only need to be identified by their subfolder and/or file name in subsequent commands.

Parameter	Description	Allowed values
directory or defaultDirectory	(required) Specifies the default directory. Previously called defaultDirectory.	[directory/filename enclosed in quotes]
directoryMac or defaultMacDirectory	Specifies the default directory for Macintosh. Previously called defaultMacDirectory.	[directory/filename enclosed in quotes]
directoryWin or detaultWinDirectory	Specifies the default directory for Windows. Previously called defaultWinDirectory.	[directory/filename enclosed in quotes]

Example:

```
setDefaultDirectory
   directory: “/data/home/proj/”
```

setMachineMemory

The **setMachineMemory** command defines the amount of random access memory (RAM) that the program will use. Limiting the amount of RAM available to the assembler allows you to use the computer for other purposes while an assembly is running. However, this will likely slow down the assemblies and is not recommended for large projects.

Parameter	Description	Allowed values
memory	(required) Amount of RAM (in GB) to be used, entered in multiples of four. Entering a value greater than the available RAM causes all RAM to be used.	[number that is a multiple of 4]

Example:

```
setMachineMemory
  memory: 32
```

setParam

The **setParam** command adjusts the stringency of one or more of the assembling parameters for the project. SeqMan NGen will use the default values for any parameter that is not specified within the script.

All of the parameters for setParam are identical to the same parameters described in the **assembleTemplate** command topic:

delayAlignInserts
gapPenalty
increaseRunGapPen
matchScore
minAlignedLength
minMatchPercent
mismatchPenalty
removeUniqueInserts

SNG commands

The following commands can be used in the script for an SNG assembly. Click on the name of a command in the table below (shown in alphabetical order) to see a description and example, and to see the parameters associated with that command. All SNG commands and parameters are assumed to be optional unless the description states that it is required.

Project management	File loading	Parameter settings	Preprocessing and assembling
closeProject	load454PairedEnd	setContaminantParam	appendToAssembly
runScript	loadConstraint	setContaminantParam	assemble
saveProject	loadContaminant	setParam	convertReads
saveReport	loadLayout	setQualityParam	extendContigs
writeUnassembledSeqs	loadRepeat	setRepeatParam	fixedTrim
	loadSeq	setVectorParam	include
	loadTemplate		makeSeqNamesUnique
	loadVector		realignContigs
	openProject		removeSmallContigs
	setDefaultDirectory		set
			setAssemblyReport
			setPairSpecifier
			splitLinkerReads
			splitMIDSeqs
			splitPairs
			splitTemplates
			trimVector

Project management commands

SNG “project management” commands include:

- closeProject
- runScript
- saveProject
- saveReport
- writeUnassembledSeqs

closeProject

The command **closeProject** closes the current project and frees the memory in use so that the system is ready for additional assemblies. This can be useful if you want to run multiple assemblies in one script.

runScript

The **runScript** command allows you to run a table script within the current script. A table script references variable values for specified parameters and other elements in a script. This enables you to run multiple projects from the same script, substituting new parameter values and other variables each time. SeqMan NGen will run the table script repeatedly, using the variable values from one row of the table for each iteration of the script until all of the rows have been used.

Parameter	Description	Allowed values
file	Specifies the directory and file/folder.	[directory/filename enclosed in quotes]
script	(required) Specifies the directory and file name of the table script you wish to run.	[directory/filename enclosed in quotes]
table	(required) Specifies the delimited text file containing the variable values.	[directory/filename enclosed in quotes]

Example:

runScript
  script: “/Library/abc_Project/abc_script.script”
  table: “/Library/abc_Project/table.txt”

saveProject

The **saveProject** command saves the assembly to a project file. By default, the SeqMan Pro / SeqMan Ultra project file format (*.sqd*) is used. Phrap (*.ace*) and FASTA (*.fas*) formats may also be specified by using the format parameter, and specifying the desired file extension using the file parameter.

Note: As a command-line tool, SeqMan NGen will not prompt you if you try to save a new project file with the same name as an existing file in the same location. When you run a script multiple times, be sure to change the file name of the project to be saved each time to prevent existing project files from being overwritten.

Parameter	Description	Allowed values
file	(required) Specifies the directory and file name of the project file to be saved.	[directory/filename enclosed in quotes]
format	Specifies the output file format. • SeqMan - Saves a 64-bit SeqMan Pro project file (*.sqd*) that is compatible with SeqMan Ultra or SeqMan Pro version 8.1 and higher (default). • SeqMan8 - Saves a 32-bit SeqMan Pro project file (*.sqd*) that is compatible with SeqMan Ultra or SeqMan Pro version 8.0 and higher. • SeqMan7 - Saves a 32-bit SeqMan Pro project file (*.sqd*) that is compatible with SeqMan Ultra or SeqMan Pro version 7.2 and higher. Note that this project file will be much bigger than the same project created in either of the SeqMan formats listed above.	[SeqMan|SeqMan8|SeqMan7|Phrap|Fasta|BAM|SAM]

	• Phrap - Saves an *.ace* file. • Fasta - Saves *.fas* and *.qual* files of the consensus sequence for each contig. • BAM - Saves a BAM file (SNG/SMNG reference-guided assemblies only). • SAM - Saves a SAM file (SNG/SMNG reference-guided assemblies only).	
onePackage	Specifies whether an assembly containing multiple reference sequences should be bundled into a single .assembly package. If 'false' is entered, one .assembly package is created per contig.	[true|false]
openInSeqMan	Specifies whether to automatically launch SeqMan Pro / SeqMan Ultra and open the completed assembly once the script has completed.	[true|false]

Example:

SaveProject
  file: "/Library/My projects/ABC_project.sqd"
  format:seqman
  openInSeqMan:true

saveReport

The **saveReport** command exports a report as a text file that summarizes assembly statistics, including the parameters used, the number of assembled/unassembled sequences and contigs, average quality scores, and the number of sequences excluded from the assembly due to exceeding the **maxAssemblyCoverage** parameter of the *setParam* command.

The same information contained within this report is also saved within the SeqMan Pro project file (*.sqd*) regardless of whether you choose to export the report by setting this parameter. The report can be viewed in SeqMan Pro / SeqMan Ultra using the **Project > Report** command.

Parameter	Description	Allowed values
file	(required) Specifies the directory and file name of the report to be saved.	[directory/filename enclosed in quotes]

Example:

saveReport
  file: “/Library/abc_Project/abc_report.txt”

writeUnassembledSeqs

The **writeUnassembledSeqs** command saves all sequences that were not assembled in the project as *.fas* and *.qual* files.

Parameter	Description	Allowed values
file	(required) Specifies the directory and file name of the unassembled sequences to be saved.	[directory/filename enclosed in quotes]
saveTrimmed	Specifies whether to save only the trimmed portion of the unassembled sequences.	[true / false]

File loading commands

SNG “file loading” commands include:

- load454PairedEnd
- loadConstraint
- loadContaminant
- loadLayout
- loadRepeat
- loadSeq
- loadTemplate
- loadVector
- openProject
- setDefaultDirectory

load454PairedEnd

The **load454PairedEnd** command loads a file of Roche 454 sequences and checks for the presence of a linker defining the paired end sequences. If the linker is found, the linker is removed and the remaining portion is split into two sequences linked with a paired end constraint.

Parameter	Description	Allowed values
DiscardLinkerless	Specifies whether to discard any read where no portion of the mate pair linker was found. In this way, reads that do not have a linker sequence will be discarded from the assembly.	[true|**false**]
file	The directory and file name of the .fas, .fna, or .sff file containing the 454 sequences.	[directory/filename enclosed in quotes]
linker	The directory and file name of the .fas, fna, or .sff file containing the 454 linker sequences. If not specified, SeqMan NGen will use its default 454 linker sequence: GTTGGAACCGAAAGGGTTTGAATTCAAACCCTTTCGGTTCCAAC.	[directory/filename enclosed in quotes]
max, maxDistance	The maximum distance for the paired end constraint.	[number] Default = 10000
min, minDistance	The minimum distance for the paired end constraint.	[number] Default = 0

Example:

load454PairedEnd
  file: “/Library/454 data/123_Pairedend.fas”
  linker: “/Library/454 data/123_linkerseqs.fas”
  min: 0
  max: 10000
  DiscardLinkerless: false

loadConstraint

The **loadConstraint** command loads a constraint file. The file can be in the NCBI ancillary file format, or in the CAP3 constraint file format. SeqMan NGen uses constraint files to identify paired end reads, similar to using the **setPairSpecifier** command. Constraint files in the NCBI ancillary file format also contain trimming information, which SeqMan NGen will load and use. SeqMan NGen will create a CAP3 file when saving a Phrap project (*.ace*) that used paired end constraints.

Parameter	Description	Allowed values
file	The directory and file name of the constraint sequence file.	[directory/filename enclosed in quotes]

Example:

loadConstraint
  file: “/Library/constraints/123_xyz.con”

loadContaminant

The **loadContaminant** command loads a contaminant sequence file to be used to identify known contaminants, such as primers, in the assembly. Sequences that contain at least 12 matching 17-mers are flagged as contaminant sequences and will be removed from the assembly.

Parameter	Description	Allowed values
file	The directory and file name of the contaminant sequence file. A folder may also be specified, in which case all of the sequence files within that folder will be loaded and used for contaminant screening.	[directory/ filename enclosed in quotes]

Example:

```
loadContaminant
  file: “/Library/contaminants/123_abc.seq”
```

loadLayout

The **loadLayout** command loads a layout file to be used for an assembly. The format may be either a SOLiD General Feature Format file (.gff) or a File of Filenames file (.fof). When this command is used, SeqMan NGen still aligns each read from the file to the reference, but uses the information contained within the specified file to determine the overall layout of reads.

Parameter	Description	Allowed values
layoutFile	(required) Specifies the directory and file name of the layout file. Both .gff and .fof formats are accepted	[directory/filename enclosed in quotes]
templateFile	(required) Specifies the directory and file name of the reference sequence file.	[directory/filename enclosed in quotes]

Example:

loadLayout
  templateFile: “/Library/123_project/template.seq”
  layoutFile: “/Library/123_project/layoutfile.gff”

loadRepeat

The **loadRepeat** command loads a sequence file to be used to identify repeat sequences in the assembly. All sequences identified as repeats will be added to the assembly last, after all non-repeats have been assembled. See our website for a list of supported file types.

Parameter	Description	Allowed values
file	(required) Specifies the directory and file name of the repeat sequence file. A folder may also be specified, in which case all of the sequence files within that folder will be loaded and used as repetitive sequences.	[directory/ filename enclosed in quotes]

Example:

```
loadRepeat
  file: “/Library/repetitive_seqs/123_repeat.seq”
```

loadSeq

The **loadSeq** command loads a sequence file or files for assembly. See our website for a list of supported file types.

Parameter	Description	Allowed values
blockContig	Used in the reference-guided workflow.	[text string]
blockContigID	Used in the reference-guided workflow.	[number]
blockName	Used in the reference-guided workflow.	[text string]
blockPos	Used in the reference-guided workflow.	[number]
DiscardLinkerless	Specifies whether reads that do not have a linker sequence should be discarded from the assembly.	[true / false]
file	(required) Specifies the directory and file name of the sequence file(s) to be loaded. A folder may also be specified, in which case all of the sequence files within that folder will be loaded.	[directory/filename enclosed i
groupName	Used to identify the multi-sample group name for a read file.	[text string]
isPair	Specifies whether the query files contain paired end data.	[true / false]
linker	The directory and file name of the .fas, fna, or .sff file containing the 454 linker sequences. If not specified, SeqMan NGen will use its default 454 linker sequence: GTTGGAACCGAAAGGGTTTGAATTCAAACCCTTTCGGTTCCAAC.	[directory/filename enclosed i
max	The maximum distance for the paired end constraint.	[number] Default = 10000
maxSeqs	Specifies the maximum number of reads to load from a file.	[number]
mergePairs	Specifies whether the reads are paired end data that overlap and should therefore be merged.	[true / false]
min	The minimum distance for the paired end constraint.	[number] Default = 0
minSeqLen	Minimum length of a sequence required to include it in the assembly.	[number]
multiplex	Specifies whether reads are from a multi-sample run.	[true / false]
seqTech	Specifies the offset to be used when converting compressed quality scores into numerical values. These are the offsets used for the technology specified:	[IonTorrent|SOLiD|Illumina|45

	Data Type / **Value** / **Offset** IonTorrent / IonTorrent / 33 Applied Biosystems SOLiD / SOLiD / 33 Illumina / Illumina / 64 Roche 454 / 454 / 33 Other types / normalScore / 33 **Note 1**: For 454,quality scores for homopolymeric runs of ≥ 2 are oriented from 5’ to 3’ on the top strand. **Note 2**: If possible, the data type of unknown data is determined automatically based on the first data file.	
templateFragment	Used in reference-guided assemblies with gap closure.	[number]

Example:

loadSeq
  file: “/Library/ABC_project/ABC_sequences.fas”

loadTemplate

The **loadTemplate** command loads a sequence file to be used as a reference for all other sequences to be assembled to. The sequence will be displayed as a “reference” sequence in SeqMan Pro for SNP analysis.

Parameter	Descriptions	Allowed values
file	(required) Specifies the directory and file name of the reference sequence file to be loaded. A folder may also be specified, in which case all of the sequence files within that folder will be loaded and treated as reference sequences.	[directory/ filename enclosed in quotes]

Example:

loadTemplate
  file: “/Library/abc_Project/abc_template.seq”

loadVector

The **loadVector** command loads a vector sequence file to be used for vector trimming.

Parameter	Description	Allowed values
cloneSite	This parameter specifies the position of the cloning site on the vector where insertion occurs.	[number]
file	(required) Specifies the directory and file name of the vector sequence file to be used for vector trimming.	[directory/filename enclosed in quotes]

Example:

```
loadVector
  file: “/Library/vectors/123_vector.seq”
  cloneSite:826
```

openProject

The **openProject** command loads an existing assembly project into memory.

Parameter	Description	Allowed values
file	(required) Specifies the directory and file name of the project file to be loaded.	[directory/filename enclosed in quotes]

setDefaultDirectory

The **setDefaultDirectory** command is required, and defines the default directory for the project. When a default directory is specified, files located in that directory only need to be identified by their subfolder and/or file name in subsequent commands.

Parameter	Description	Allowed values
directory	(required) Specifies the default directory.	[directory/filename enclosed in quotes]
defaultMacDirectory	Specifies the default directory for Macintosh.	[directory/filename enclosed in quotes]
defaultWinDirectory	Specifies the default directory for Windows.	[directory/filename enclosed in quotes]

Example:

setDefaultDirectory: “/Library/ABC_proj/”

Once you have set a default directory, you may use two periods before a file name to specify that the file you wish to use is located in the parent folder of the default directory you specified. For example, the following line specifies that the vector file, 123Vector.fas, is located in the ABC Data folder, the parent folder of the default directory.

loadVector file: “../123Vector.fas”

Parameter settings commands

SNG “parameter setting” commands include:

- setContaminantParam
- setParam
- setQualityParam
- setRepeatParam
- setVectorParam

setContaminantParam

The **setContaminantParam** command allows you to adjust the parameters used for scanning for contaminant sequences. In order to be applied, this command must appear in the script before the **loadContaminant** command, and the **contamScan** parameter for the **assemble** command must be set to **true**.

Parameter	Description	Allowed values
MerLength	The minimum length of a mer required to be considered an exact match when scanning for contaminants. Wizard equivalent: **Advanced Trim/Scan Options > Mer length**.	[number from 5-50] Default = 17
MinMerMatch	The minimum number of matching mers required to mark the sequence as a contaminant. Wizard equivalent: **Advanced Trim/Scan Options > Minimum matches**.	[number from 1-50] Default = 12

Example:

setContaminantParam MerLength:17
setContaminantParam MinMerMatch:12

setParam

The **setParam** command allows you to adjust the stringency of one or more of the assembling parameters for the project. SeqMan NGen will use the default values for any parameter that is not specified within the script.

Parameter	Description	Allowed values
AllowConstraintBased	Specifies whether the assembler should use constraints during assembly.	[true / false]
AssembleBoneyard	Specifies whether, after a reference-guided assembly has been completed, the unassembled sequences remaining should be assembled into contigs. If the reference has been split, SeqMan NGen will attempt to join the split contigs together in new arrangements. (Note: “Boneyard” is a term for sequences that were not assigned to any contig). Wizard equivalent (*de novo* or special reference-guided workflows only): **Assembly Options** > ***De novo* assemble unassembled reads**.	[true / false]
CoverageType	Specifies the type of coverage to be used for repeat handling. ‘Genome’ uses the length of the genome being assembled to calculate the expected coverage. ‘Fixed’ uses a fixed value as the expected coverage. If you know the length of the genome/fragment being assembled, we recommend using ‘genome’ for this parameter and then specifying the length using the **genomeLength** parameter. If you do not know the genome/fragment length, use ‘fixed’ and provide the most accurate estimate of expected coverage for the **FixedCoverage** value.	[genome / fixed]
DefaultQuality	The value used for the base quality of sequences without quality scores. Wizard equivalent (*de novo* or special reference-guided workflows only):	[number from 5-100] Default = 15

	Assembly Options > **Default quality**.	
FixedCoverage	The estimated depth of the sequencing, which can be used instead of the genome length for repeat handling. Use caution when estimating the value for fixedCoverage. If the value you use is significantly lower than the actual depth, the assembly may take a much longer time to complete and may have too many mers flagged as repeats.	[number from 1-65535] Default = 20
GapPenalty	The penalty for opening or extending a gap during an alignment. This penalty is deducted from the pairwise score used to calculate match percentage. A high gap penalty suppresses gapping, while a low value promotes gapping. Wizard equivalent (*de novo* or special reference-guided workflows only): **Assembly Options** > **Gap penalty**.	[number from 0-1000] Default = 30 for most workflows; 50 for the de novo transcriptome RNA-seq workflow.
GenomeLength	Specifies the length of the genome or fragment being assembled. This is used to calculate expected coverage in determining repeat handling. (Note: this parameter was called “setGenomeParam” prior to SeqMan NGen 2.0.).	[number from 0-1015 ULL] Default = 0
HaploidSNP	Specifies whether to use the second most common base at a position when performing SNP passes. (See the **snpPasses** parameter). Using this parameter will increase the SNP percentage for SNPs occurring on one allele of a diploid genome in a reference-guided assembly. When haploidSNP is set to ‘true,’ the lowCoverageThreshold parameter value should be greater than zero.	[true / false]
HaploidThreshold	The minimum number of times that the second most common base must occur at a position in order for it to be used to find SNPs during haploid SNP passes.	[number from 0-100] Default = 0

	(See the **haploidSNP** parameter above).	
LowCoverageThreshold	The minimum coverage required in an assembly to be excluded from SNP passes. SeqMan NGen will include regions in an assembly that have coverage less than the value specified as well as regions with zero coverage when it performs SNP passes. (See the **snpPasses** parameter). Wizard equivalent (*de novo* or special reference-guided workflows only): **Assembly Options** > **SNP low cover cutoff**.	[number from 0-10000] Default = 0
MatchRepeatPercent	The percent frequency a mer occurs compared to its expected frequency. Mers exceeding this value are flagged as repeated and not used as mer tags in determining overlaps. (Note: this parameter was called "**maxCoverageRatio**" prior to SeqMan NGen 2.0.). Wizard equivalent (*de novo* or special reference-guided workflows only): **Assembly Options** > **Match repeat percent**.	[number from 100-1000] Default = 150
MatchScore	The score for a base match during an alignment. This score contributes to the pairwise score used to calculate match percentage. Increasing the matchScore value will allow for longer or more frequent gaps, thus forcing bases that match to be assembled together. Wizard equivalent (*de novo* or special reference-guided workflows only): **Assembly Options** > **Match score**.	[number from 1-1000] Default = 10
MatchSize	The minimum number of matching consecutive bases required to determine the overlap of sequence reads. If an even number is entered, SeqMan NGen will automatically increase the value to the next odd number. (Note: this parameter was	[odd whole number] Default = 21

	called **setParamMerLength** prior to SeqMan NGen 2.0.). Wizard equivalent (*de novo* or special reference-guided workflows only): **Assembly Options > Mer size**.	
MatchSpacing	The length of the window of a sequence read where at least one mer tag will be chosen. (Note: this parameter was called “merTagWindow” prior to SeqMan NGen 2.0.). Wizard equivalent (*de novo* or special reference-guided workflows only): **Assembly Options > Match spacing**.	[number from 1- 1000000] Default = 50
MatchWindowLength	The size of the window used to calculate the match percentage. Wizard equivalent (*de novo* or special reference-guided workflows only): **Assembly Options > Match window**.	[number from 10-1000] Default = 50
MaxAssemblyCoverage	The maximum depth of coverage allowed in the reference-guided assembly. SeqMan NGen will not exceed the coverage specified by this threshold. This parameter is only available for reference-guided assemblies, and should be used with caution as it will limit the number of sequences included in the assembly. A value of 0 indicates unlimited coverage. Wizard equivalent (*de novo* or special reference-guided workflows only): **Assembly Options > Maximum coverage**.	[number from 0-65535] Default = 0
MaxContigs	The maximum number of contigs to write to an .assembly project. This command is not generally needed due to SeqMan’s capacity to handle a very large number of contigs.	[number]
MaxGap	The theoretical maximum length of a gap that could be inserted. In practice, the maximum gap size will usually be about half of this value. Wizard	[number from 0-99] Default = 6

	equivalent (*de novo* or special reference-guided workflows only): **Assembly Options > Max gap**.	
MaxUsableCount	Any mers occurring more frequently than FixedCoverage multiplied by MaxUsableCount are disregarded as mer tags from the assembly. Wizard equivalent (*de novo* or special reference-guided workflows only): **Assembly Options > Max usable**.	[number from 1-65535] Default = 25
MinContigSeqs	The minimum number of sequences in a contig. After an assembly has been completed, any contigs without a reference sequence will be disassembled if they contain fewer sequences than the number specified. The use of this parameter is recommended when performing *de novo* assemblies using data from Next Generation sequencing technologies, such as Illumina, as these types of assemblies can produce tens of thousands of very small contigs.	[number from 0-10000] Default = 0
Minimizer	(Intended for internal use only). An experimental way of choosing mer tags that may save time and memory. The accuracy of this parameter has not been verified by DNASTAR.	[number]
MinMatchPercent	The minimum percentage of matches in an overlap required to join two sequences in the same contig. (Note: this parameter was called “minMatchPercentage” prior to SeqMan NGen 2.0.). Wizard equivalent (*de novo* or special reference-guided workflows only): **Assembly Options > Minimum match percentage**.	[number from 0-100] Default = *+93+
MismatchPenalty	The penalty for a base mismatch during an alignment. This penalty is deducted from the pairwise score used to calculate match percentage. Wizard	[number from 0-1000] Default = 20

	equivalent (*de novo* or special reference-guided workflows only): **Assembly Options > Mismatch penalty**.	
SkipRealign	This parameter only affects *de novo* assemblies, and specifies whether to skip the realignment step of the assembly. The realignment step will then analyze each sequence at the nucleotide level to determine the exact position of each sequence in the alignment. Wizard equivalent (*de novo* or special reference-guided workflows only): **Assembly Options > Realign reads after assembly**.	[true / false]
SNP	Specifies whether a SNP detection pass of the gapped alignment is made during the assembly.	[true / false]
snp_checkStrandedness	Specifies whether the strand that each read comes from is considered in the SNP calculation. This is ignored by the simple SNP calling method (used when genome ploidy is “Heterogeneous”).	[true / false]
snp_minPctToScore	Specifies minimum percentage of reads in a column which must differ from the reference in order to score the column. For the simple SNP calling method (used when genome ploidy is “Heterogeneous”), this is the only criteria used to call a SNP. For the Bayesian SNP calling methods (used when genome ploidy is “Diploid” or “Haploid”), this is a filter applied before the other parameters.	[number from 0-1] Default = 0.05
snp_minProbNonrefToCall	Specifies the minimum probability of a SNP column which is required to call a SNP, expressed as a number from 0 and 1. The probabilities of all genotypes other than Homozygous Reference are totaled and checked against this number. This is the final filter applied	[number from 0-1] Default = 0.1, requiring a minimum 10% change

	during the Bayesian SNP calling methods (used when genome ploidy is “Diploid” or “Haploid”). This is ignored by the simple SNP calling method (used when genome ploidy is “Heterogeneous”).	
snp_minVariantDepthToScore	(required if “snp” is true) Specifies the minimum depth required for a specific base (or deletion) in a column before it is considered usable for SNP calling. This is the second filter applied during the Bayesian SNP calling methods (used when genome ploidy is “Diploid” or “Haploid”). This is ignored by the simple SNP calling method (used when genome ploidy is “Heterogeneous”).	[number from 0-100] Default = 2
snp_minWeight	Called “Minimum base quality score” in the SeqMan NGen wizard, this parameter specifies the minimum quality score for a base to be considered in the SNP calculation.	[number]
SNPMatchPercentage	The minimum match percentage required during passes to fill in SNP regions. See the **snpPasses** parameter. Wizard equivalent (*de novo* or special reference-guided workflows only): **Assembly Options > SNP match percent**.	[number from 0-100] Default = 90
snpMethod	Specifies the SNP detection method to use. Simple produces a count of each type of base in the column and calculates the percent of non-reference bases. Haploid uses a Bayesian statistical model to calculate a probability score that the position contains a polymorphism and give a quality score for the base called at that position. Diploid uses a Bayesian statistical model to calculate a probability score that the position contains a polymorphism and give a quality score for the base(s) called at	[simple|haploid|*diploid*|population]

	that position. Based on the scores, it also calls the genotype at each position.	
SNPPasses	The number of times SeqMan NGen will cycle through a reference-guided assembly, attempting to fill in regions with low coverage or no coverage due to SNPs. Wizard equivalent (*de novo* or special reference-guided workflows only): **Assembly Options > SNP passes**.	[number from 0-10] Default = 2
SplitFalseJoins	Specifies whether the assembler should identify and splits false joins based on the set of false join parameters indicated.	[true / false]
SplitTemplateContigs	Specifies whether, after a reference-guided assembly has been completed, the template should be split into contigs at areas where there is zero coverage. Split contigs will be grouped into scaffolds with a defined position to allow for easy sorting when the project is viewed in SeqMan Pro. Annotations on the reference sequence will also be split, and any /codon_start qualifiers will be adjusted to stay in frame.	[true / false]
TemplateDefaultQuality	The value used for the base quality of template sequences without quality scores. Wizard equivalent (*de novo* or special reference-guided workflows only): **Assembly Options > Default template quality**.	[number from 5-50000] Default = 500
TrimToMer	Specifies whether to trim the reads to the matching mer tags within the read. For each read, SeqMan NGen looks for mers that exist in the template (for templated assemblies) or in any other read in the assembly (for *de novo* assemblies). It then sets the trimming for the read to the start of the first mer found and the end of the last mer	[true / false]

	found. Trimming to mer may be useful when assembling data without accurate quality scores, data with very short linkers, or when assembling SOLiD data.	
UseRepeatHandling	Specifies whether to use the repeat probabilities to determine if a mer occurs too frequently to use. This parameter should only be used for *de novo* assemblies, unless the **assembleBoneyard** parameter is set to 'true' for the templated assembly. Wizard equivalent (*de novo* or special reference-guided workflows only): **Assembly Options > Repeat handling**.	[true / false]

Example A:

setParam SNP: true
setParam snp_minVariantDepthToScore: 2
setParam snp_minWeight: 5
setParam snp_combineSubs: true
setParam snp_excludeBasesEdge: 0
setParam snp_maxRun: -1
setParam snp_maxStrandBias: -1
setParam snp_minHomopolDelDepth: 0
setParam snp_minHomopolDelFrac: 0
setParam snp_minHomopolInsDepth: 0
setParam snp_minHomopolInsFrac: 0
setParam snp_minSoftDepth: -1
setParam snp_minSoftPnotRefPct: -1
setParam snp_minSoftSnpPct: -1
setParam snp_minStrandCov: 0
setParam snp_runVar: false
setParam snp_checkStrandedness: false
setParam snp_minProbNonrefToCall: 0.1
setParam SNPmethod: diploid
setParam snp_minPctToScore: 0.05

Example B:

In the de novo transcriptome RNA-seq workflow, reads clustered with XNG are reassembled using SNG. In order to minimize mis-joins, the initial phase of the assembly is done at high stringency using the following parameters:

setParam
merLength: 21
minMatchPercent: 97
useRepeatHandling: false
minContigSeqs: 101

Two assembly passes are performed for each read cluster. During the first pass, contigs are assembled from the reads after which those with less than 101 reads are dis-assembled and added to the unassembled sequences pool for that cluster. During a second pass SNG attempts to merge the assembled contigs and add any of the unassembled sequence reads from the first pass. To facilitate merging, **minMatchPercent** is lowered to 85 for this pass.

setParam
minMatchPercent: 85

setQualityParam

The **setQualityParam** command allows you to adjust the parameters used for quality trimming. In order to be applied, the **trimEnds** parameter for the **assemble** command must be set to 'true.'

Parameter	Description	Allowed values
EndRegion	The number of bases at the end of a sequence considered to be the "end region" which is used by other quality parameters.	[number from 1-100] Default = 5
MaxN	The maximum number of "N" bases permitted in the window used for N-based quality trimming.	[number from 1-100] Default = 2
MaxNHiQual	The maximum number of "N" bases permitted in the window used for N-based quality trimming to meet the high-quality threshold.	[number from 0-100] Default = 1
MinAveHiQual	The minimum averaged quality score of the evaluated window required to be considered high-quality.	[number from 10-40] Default = 22
MinAveLowQual	The minimum averaged quality score of the evaluated window required to be considered low-quality. Wizard equivalent: **Advanced Trim/Scan Options > Minimum quality**.	[number from 5-40] Default = 20
MinEndBaseQual	The minimum quality base score required in the specified end region.	[number from 5-40]

		Default = 15
NTrimWinLength	The length of the window used for “N-based” quality trimming. N-based quality trimming trims bases that are called “N” and is used only when quality scores are not available.	[number from 5-100] Default = 7
WinLength	The length of the window used for averaging quality scores. Wizard equivalent: **Advanced Trim/Scan Options > Window**.	[number from 2-100] Default = 5

Example:

setQualityParam winLength:30
setQualityParam minAveLowQaul:14
setQualityParam minAveHiQaul:18
setQualityParam minEndBaseQaul:15
setQualityParam endRegion:15
setQualityParam nTrimWinLength:50
setQualityParam maxN:2
setQualityParam maxNHiQual:1

setRepeatParam

The **setRepeatParam** command allows you to adjust the parameters used for scanning for repetitive sequences. In order to be applied, this command must appear in the script before the **loadRepeat** command, and the **repeatScan** parameter for the **assemble** command must be set to 'true.'

Parameter	Description	Allowed values
AlignCutoff	The minimum acceptable alignment score. When the alignment score drops below the specified value, this indicates that the end of the alignment between the read and the repeat has been reached, and the alignment will stop.	[number from 10-1000000] Default = 100
MaxMerGap	The maximum distance between two mers required to be considered a matching pair.	[number from 0-50] Default = 10
MerLength	The minimum length of a mer required to be considered an exact match when scanning for repeats. Wizard equivalent: **Advanced Trim/Scan Options > Mer length**.	[number from 5-50] Default = 17
MinEndFlagLen	The minimum length required for a mer to be flagged as a repeat if the segment is bound by the end of the read.	[number from 5-1000000] Default = 25
MinFlagLength	The minimum length required for a mer to be flagged as a repeat. Wizard equivalent: **Advanced Trim/Scan Options > Flag length**.	[number from 5-1000000] Default = 50
MinMerMatch	The minimum number of matching mers required to start an alignment. Wizard equivalent: **Advanced Trim/Scan Options > Minimum matches**.	[number from 2-25] Default = 2

Example:

setRepeatParam merLength:17

setRepeatParam minMerMatch:2
setRepeatParam maxMerGap:10
setRepeatParam minFlagLength:50
setRepeatParam alignCutoff:100
setRepeatParam minEndFlagLength:25

setVectorParam

The **setVectorParam** command allows you to adjust the parameters used for vector trimming. In order to be applied, this command must appear in the script before the **loadVector** or **TrimVector** command, and the **vectScan** parameter for the **assemble** command must be set to 'true.'

Parameter	Description	Allowed values (defaults in bold)
AlignCutoff	The minimum acceptable alignment score. When the alignment score drops below the specified value, this indicates that the end of the alignment between the read and the vector has been reached, and the alignment will stop.	[number from 10-1000000] Default = 100
EndCutOff	The distance to the endpoint where trimming will go all the way to the end of the sequence. Wizard equivalent: **Advanced Trim/Scan Options > Trim to end**.	[number from 0-1000000] Default = 25
EndMerMatch	The minimum number of mer matches required to start an alignment in the specified end region.	[number from 1-25] Default = 1
EndRegion	The number of bases at the end of a sequence where a lower stringency for matching and trimming is used.	[number from 0-1000000] Default = 15
MaxMerGap	The maximum distance between two mers required to be considered a matching pair.	[number from 0-50] Default = 5
MergeTrimGap	Maximum distance between two trim segments that will cause the segments to be merged. **MergeTrimGap** limits trimming to the ends of sequence reads, while **EndCutOff** doesn't. Controls how sensitive trimming should be in areas where some portions of the sequence match a vector and other portions don't. The higher the number the more likely the vector trimmer will find all the vector sequence in a region of poor	[number from 0-1000000] Default = 7, which is

	quality. The smaller the number, the more confidence there is that the bases trimmed are actually vector and not a spurious match.	suitable for trimming linkers from the ends of sequences.
MerLength	The minimum length of a mer required to be considered an exact match when searching for vector. Wizard equivalent: **Advanced Trim/Scan Options > Mer length**.	[number from 5-25] Default = 9
MinEndTrimLength	The minimum length to be trimmed when a vector matches the end of a read. This parameter can be useful in preventing small spurious matches from being trimmed, which may be significant with short read technologies.	[number from 5-1000000] Default = 5
MinMerMatch	The minimum number of matching mers required to start an alignment. Wizard equivalent: **Advanced Trim/Scan Options > Minimum matches**.	[number from 1-25] Default = 3
MinTrimLength	The minimum length required for a mer to be considered as a match for vector trimming. Wizard equivalent: **Advanced Trim/Scan Options > Trim length**.	[number from 5-1000000] Default = 30

Example:

```
setVectorParam merLength:9
setVectorParam minMerMatch:3
setVectorParam MerGap:5
setVectorParam minTrimLength:30
setVectorParam minEndTrimLength:5
setVectorParam alignCutoff:100
setVectorParam endRegion:15
setVectorParam endCutoff:25
setVectorParam endMerMatch:1
```

Preprocessing and assembling commands

SNG “preprocessing and assembling commands” include:

- appendToAssembly
- assemble
- convertReads
- extendContigs
- fixedTrim
- include
- makeSeqNamesUnique
- realignContigs
- removeSmallContigs
- set
- setAssemblyReport
- setPairSpecifier
- splitLinkerReads
- splitMIDSeqs
- splitPairs
- splitTemplates
- trimVector

appendToAssembly

The **appendToAssembly** command is for the reference-guided workflow and is intended for internal use only.

assemble

The **assemble** command is required and reprocesses and assembles the sequences that have been loaded. Preprocessing may include quality trimming, and scanning for vector, repetitive, and contaminant sequences.

Parameter	Description	Allowed values
assembleBlocks	Specifies whether the assembly is a reference guided assembly.	[true / false]
contamScan	If true, sequences will be scanned for the specified contaminant sequences before assembling. Also see **loadContaminant**	[true / false]
doAssemble	If false, only the preprocessing will be done, and the sequences will not be assembled.	[true / false]
repeatScan	If true, sequences will be scanned for the specified known repetitive sequences before assembling. Also see **loadRepeat**	[true / false]
trimEnds	If true, the sequences will be trimmed based on quality scores before assembling. Wizard equivalent: **Read options > Quality end trim**.	[true / false]
vectScan	If true, the sequences will be scanned and trimmed for vector before assembling. Also see **loadVector**.	[true / false]

Example:

```
assemble
  trimEnds:false
  vectScan:false
  repeatScan:false
  contamScan:false
  doAssemble:true
```

convertReads

The **convertReads** command converts a sequence from one file format to another. This command is particularly useful for converting SOLiD .csfasta files into .fastq files that can be used by the XNG assembler.

Parameter	Description	Allowed values
destination	The location and filename for the output.	[directory/filename enclosed in quotes]
file, reads	The input file containing the reads.	[directory/filename enclosed in quotes]
format	Specifies the format of the output file. If ‘genbank’ is entered, the output will be in *.gbk* format. If **fastq** is entered, the output will be in *.fastq* format.	[genbank / **fastq**]

extendContigs

The **extendContigs** command is intended for internal use only.

Parameter	Description	Allowed values
extendPasses		[number]
mergeContigsInScaffold		[true/false]

fixedTrim

The **fixedTrim** command trims reads prior to assembly using fixed values. Based on the parameter settings for this command, SeqMan NGen will trim reads either by a specified number of bases from each end, or to a specified range.

Parameter	Description	Allowed values
end3	If **trimRelative** (see below) is set to 'true,' then this value indicates the number of bases for SeqMan NGen to trim from the 3' end of each read. If trimRelative is set to 'false,' then this value indicates the specific 3' coordinate to which reads should be trimmed. Wizard equivalent: **Advanced Trim/Scan Options > 3' trim**.	[number from 0-1000000] Default = 0
end5	If **trimRelative** (see below) is set to 'true,' then this value indicates the number of bases for SeqMan NGen to trim from the 5' end of each read. If trimRelative is set to 'false,' then this value indicates the specific 5' coordinate to which reads should be trimmed. Wizard equivalent: **Advanced Trim/Scan Options > 5' trim**.	[number from 0-1000000] Default = 0
trimRelative	Specifies whether the value for the end3 and end5 parameters should indicate the number of bases for SeqMan NGen to trim from the 3' or 5' end of each read. When 'false,' the value specified for the **end3** or **end5** parameter indicates the specific coordinate to which reads should be trimmed.	[true / false]

Example:

```
fixedTrim
  end5:10
  end3:20
  trimRelative:true
```

include

When building a script, the **include** command can be used to call up additional lines of script previously stored in a text file. In this way, a group of commands can be shared between two or more scripts.

Parameter	Description	Allowed values
file	Specifies a directory and name for the file.	[directory/filename enclosed in quotes]

makeSeqNamesUnique

The **makeSeqNamesUnique** command is intended for internal use only.

realignContigs

The **realignContigs** command causes SeqMan NGen to perform another pass through a reference-guided assembly once the initial assembly is complete, and realigns contigs as needed. (This step occurs automatically for *de novo* assemblies.) Using this command may improve the accuracy of the final assembly by correcting occasional misalignments that can occur in gapped regions, however note that this step may significantly increase the time to assemble. This command must appear in the script after the assemble command.

removeSmallContigs

The **removeSmallContigs** command disassembles any contigs without reference sequences that have fewer than the specified number of sequences.

Parameter	Description	Allowed values
minLength	Specifies the minimum length of a contig to prevent it from being disassembled. Wizard equivalent (*de novo*, special reference-guided workflows only): **Assembly Options > Minimum length**.	[number] Default = 0
minSeqs	(required) Specifies the minimum number of sequences necessary in a contig to prevent it from being disassembled. Wizard equivalent (*de novo*, special reference-guided workflows only): **Assembly Options > Minimum sequences**.	[number] Default = 100

set

The **set** command is used to set variables. See the example below and those under the **runScript** command.

Example:

set $snp:true
set $snpMethod:”Diploid”

setAssemblyReport

The **setAssemblyReport** command is intended for internal use only. It is used to designate a file for a tab delineated report, similar to a report that XNG generates. This is useful during development to test how code changes impact results.

Parameter	Description	Allowed values
file name	Specifies the folder and file name.	[directory/filename enclosed in quotes]

setPairSpecifier

The **setPairSpecifier** command defines the paired end pair specifier for the paired Sanger and Illumina sequences in the assembly. This command must appear in the script before the assemble command, but after sequences have been loaded using the **loadSeq** command. For more information on assembling 454 paired end data, see the **load454PairedEnd** command. Pair specifiers define the naming convention for sequence pairs, as well as requirements for a minimum and maximum distance between the opposite ends of the inserts. Expressions for forward and reverse naming conventions should be created using the paired end specification language. Forward and reverse sequences must have identical names except for the unique portion that determines the direction of the clone.

Parameter	Description	Allowed values
pairs	This parameter lists the paired end constraints, specified by the following four values. Each value should be separated by a space and the list of values enclosed in double brackets {}. An additional set of brackets is required around all of the paired end constraints, regardless of whether one or multiple pair constraints are specified.	[forward|reverse|min|max]
forward	A naming pattern to match forward clones.	[text string enclosed in quotes]
max	The maximum distance for the paired end sequences to be separated.	[number]
min	The minimum distance for the paired end sequences to be separated.	[number]
reverse	A naming pattern to match reverse clones.	[text string enclosed in quotes]

Example:

(defines 2 pair specifiers each with different size ranges)

```
setPairSpecifier
  pairs:{{forward:”(.*)(2kb)(.*)-FP.*$”reverse:”(.*)(2kb)(.*)-RP.*$” min: 1500 max: 2500}
    {forward:”(.*)(8kb)(.*)-FP.*$” reverse:”(.*)(8kb)(.*)-RP.*$” min: 7000 max: 9000}}
```

splitLinkerReads

The **splitLinkerReads** command splits specified reads based on their match to given linker sequences. Reads that align to the linker and include the linker site (as specified by the linkerSite parameter or by the cloneSite option in an .fof file) will be split into two reads. The two newly split reads will be designated by _A and _B appended to the name.

Parameter	Description	Allowed values
linkerFile	The directory and file name of the linker file.	[directory/filename enclosed in quotes]
linkerSite	The position indicating where reads should be split.	[number]
seqFile	The directory and file name of the sequence reads.	[directory/filename enclosed in quotes]

Example:

splitLinkerReads
  seqFile: “/Library/123_project/reads.fas”
  linkerFile: “/Library/123_project/linker.fas”
  linkerSite:30

splitMIDSeqs

The **splitMIDSeqs** command is used to split 454 MID reads into individual files with one file per MID tag.

Parameter	Description	Allowed values
destination	The location and filename for the output.	[directory/filename enclosed in quotes]
file, reads	The input file containing the reads.	[directory/filename enclosed in quotes]

splitPairs

The **splitPairs** command is used to split 454 or ion torrent mate pair files into forward and reverse (and singleton) files.

Parameter	Description	Allowed value
destination	The location and filename for the output.	[directory/filename enclosed in quotes]
DiscardLinkerless	Specifies that reads without a linker sequence should be discarded from the assembly.	[true / false]
file, reads	The location and filename for the input.	[directory/filename enclosed in quotes]
seqTech	Specifies the offset to be used when converting compressed quality scores into numerical values. These are the offsets used for the technology specified: **Data Type** / **Value** / **Offset** IonTorrent / IonTorrent / 33 Applied Biosystems SOLiD / SOLiD / 33 Illumina / Illumina / 64 Roche 454 / 454 / 33 Other types / normalScore / 33 Wizard equivalent: **Input Sequence Files > Read technology**. **Note 1**: For 454,quality scores for homopolymeric runs of ≥ 2 are oriented from 5’ to 3’ on the top strand. **Note 2**: If possible, the data type of unknown data is determined automatically based on the first data file.	[IonTorrent|SOLiD|Illumina|454|normalScore|Other]

Example:

```
SplitPairs
destination:”c:data\splitReads\”
reads: {
{ file:”C:data\reads\file1.fas” format: IonTorrent }
{ file: “C:data\reads\file2.fas” format:454 discardLinkerless: true}
}
```

splitTemplates

The **splitTemplates** command splits reference contigs into multiple contigs in areas where there is zero coverage. Split contigs will be grouped into scaffolds with a defined position to allow for easy sorting when the project is viewed in SeqMan Pro / SeqMan Ultra. Annotations on the reference sequence will also be split, and any /codon_start qualifiers will be adjusted to stay in frame.

trimVector

The **trimVector** command is used for fast trimming vector sequence. Each read file is processed and the trimmed file is saved to the destination folder. If the file with the same name exists, the number will appended to the file name. The file is saved in .fastq format, including trimming statistics.

Parameter	Description	Allowed values
file, reads	The location and filename for the input.	[directory/filename]
LinkerFile	The location of file or folder with vector sequence.	[directory/filename]
destination	The location of output folder.	[directory]

Example:

```
setVectorParam
  EndCutOff: 130
  MatchSize: 11
  MinTrimLength: 15

TrimVector
  reads: {
  file: “C:\data\input.fastq”
  }
  LinkerFile: “c:\data\adapter.fas”
  destination: “c:\data\Out\”
```

Specifying XNG or SNG/SMNG when running a script

SeqMan NGen utilizes several assemblers with different capabilities and scripting languages. Therefore, it is essential to match the correct assembler with the type of assembly project to be done.

To specify which assembler to use to run your script, type **xng** or **sng** followed by the path and script file name after the command prompt. Alternatively, add either the **#!/usr/bin/xng** or **#!/usr/bin/sng** command as first line of the script and execute through the command line.

Turn off usage logging

By default, usage logging is enabled in Lasergene version 11 and later. To opt out of usage logging, launch the DNASTAR Navigator and go to **View > Preferences > Lasergene** (Win) or **DNASTAR Navigator > Preferences > Lasergene** (Mac). Uncheck the box and click **Apply** and then **OK**.

Non-English keyboards

SeqMan NGen recognizes only standard English-keyboard characters as input. If you are using a non-English keyboard, we recommend that you switch to a “virtual” English keyboard. Click a link for instructions: Windows 7 & 8, Macintosh 10.11.

Installed Lasergene file locations

The following file names use ‘x’ to represent the version number.

File Category	Application	Path
Application[2]	ArrayStar	Windows: C:\Program Files (x86)\DNASTAR\Lasergene x\ ArrayStar
	SeqNinja (command line)	Windows: C:\Program Files (x86)\DNASTAR\Lasergene x\ SeqNinjaCL
	All others	Windows: C:\Program Files (x86)\DNASTAR\Lasergene x Macintosh: /Applications/ DNASTAR/Lasergene x
Data Manager[2] (*DMx, DMx.exe)*	SeqBuilder Pro, Protean 3D, GeneQuest, MegAlign Pro	Windows: C:\Program Files (x86) \DNASTAR\Lasergene x Macintosh: /Applications/ DNASTAR/Lasergene x
Data Manager State File	SeqBuilder Pro, Protean 3D, GeneQuest, MegAlign Pro	Windows: C:\Program Data\ DNASTAR\DataManager Macintosh: ~/Library/ Application Support/ DNASTAR/DataManager, ~/Library/Preferences/ DNASTAR/DataManager
License Manager[2]	All	Windows: C:\Program Files (x86)\DNASTAR\License Manager Macintosh: /Applications/ DNASTAR/License Manager
Commuter License Manager	ArrayStar	Windows: C:\Program Files (x86)\DNASTAR\ArrayStar ‘x’

		Macintosh: /Applications/ DNASTAR/ArrayStar ‘x’
	SeqMan NGen	Windows: C:\Program Files (x86)\DNASTAR\SeqMan NGen ‘x’ Macintosh: /Applications/ DNASTAR/SeqMan NGen ‘x’
	All others	Windows: C:\Program Files (x86)\DNASTAR\Lasergene ‘x’ Macintosh: /Applications/ DNASTAR/Lasergene ‘x’
Server License File (*lservrc*), Server License Manager [2], *Server Executables (_lserv, lservnt.exe*)	All	Windows: C:\Program Files (x86)\DNASTAR-LicenseServer\Server Macintosh: ~/Library/ DNASTAR-LicenseServer
Standalone & Trial Licenses (**.license)*, License Server Client License (**.lshost)*, Key Server Client License (**.keyhost*)	All	Windows: C:\Program Data\ DNASTAR\Licenses Macintosh: ~/Library/ Application Support/ DNASTAR/Licenses
Preferences	Protean 3D, Navigator, SeqNinja (DNA*), GenVision Pro	Windows: C:\Users\<user>\DNASTAR Macintosh: ~/Library/ DNASTAR
	ArrayStar	Windows: C:\Users\<User>\AppData\ Roaming\DNASTAR\ArrayStar
	All others	Windows: C:\Users\<user>\AppData\ Local\DNASTAR\ Macintosh: ~/Library/ Preferences

AppData is a hidden folder in Windows. To unhide the folder, go to **Organize > Folder** and **Search Options > View > Show Hidden files and folders**.

Troubleshoot failure to launch

If you attempt to launch SeqMan NGen, without an updated .NET Framework Service Pack 1 (Windows) or Mono package (Mac) installed, you may receive an error message, and SeqMan NGen may fail to open.

The best way to ensure the correct software is installed is to perform all recommended updates when prompted by your operating system.

To instead resolve this issue manually:

- Windows – Download and install .NET Framework 4.7.2 from www.microsoft.com/net/download/windows/run.
- Macintosh – Download and install the Mono 5.10.1 stable package from www.mono-project.com/download/stable/#download-mac.

Research references

Benjamini Y and Hochberg Y (1995). “Controlling the false discovery rate: a practical and powerful approach to multiple testing.” *Journal of the Royal Statistical Society. Series B (Methodological)*, Vol. 57, No. 1 (1995), pp. 289-300. Read online

Fitzgerald DM, Bonocora RP, Wade JT (2014). “Comprehensive Mapping of the Escherichia coli Flagellar Regulatory Network.” PLoS Genet 10(10): e1004649. doi:10.1371/journal.pgen.1004649.

Johnson et al. (2007). "Genome-Wide Mapping of in Vivo Protein-DNA Interactions." Science June 8, 2007; 316(5830):1497-502.

Krumm N, Sudmant PH, Ko A, et al. (2012). "Copy number variation detection and genotyping from exome sequence data." Genome Res. published online May 14, 2012. (This reference pertains to the zRPKM and RPKM-CN normalization methods.)

Li H, Ruan J, and Durbin R (2008). “Mapping short DNA sequencing reads and calling variants using mapping quality scores.” Genome Res. 2008 Nov;18(11):1851-8. doi.org/10.1101/gr.078212.108. Epub 2008 Aug 19.

Love MI, Huber W and Anders S (2014) “Moderated estimation of fold change and dispersion for RNA-seq data with DESeq2.” Genome Biology 15:550. doi.org/10.1186/s13059-014-0550-8.

Mortazavi A, Williams BA, McCue K, Schaeffer L and Wold B (2008). “Mapping and quantifying mammalian transcriptomes by RNA-Seq.” Nature Methods, 5, 621-628. (Pertains to the ChIP-Seq Peak Finder algorithm.)

Mukherjee, Supratim & Huntemann, Marcel & Ivanova, Natalia & Kyrpides, Nikos & Pati, Amrita. (2015). Large-scale contamination of microbial isolate genomes by Illumina PhiX control. Standards in genomic sciences. 10. 18. 10.1186/1944-3277-10-18. Read online

Nookaew I et al. (2012). “A comprehensive comparison of RNA-Seq-based transcriptome analysis from reads to differential gene expression and cross-comparison with microarrays: a case study in *Saccharomyces cerevisiae*.” Nucleic Acids Res. 2012 Nov 1;40(20):10084-97. doi: 10.1093/nar/gks804 view on PubMed.

Robinson MD, McCarthy DJ, Smyth GK (2010). “edgeR: a Bioconductor package for differential expression analysis of digital gene expression data.” Bioinformatics, 26(1), 139-140.

Robinson, MD, and Oshlack, A (2010). “A scaling normalization method for differential expression analysis of RNA-seq data.” Genome Biology 11, R25

Robinson, MD, and Smyth, GK (2008). “Small sample estimation of negative binomial dispersion, with applications to SAGE data.” Biostatistics 9, 321 – 332.

Zhang Y, Liu T, Meyer CA et al. (2008) Model-based Analysis of ChIP-Seq (MACS). Genome Biol 9, R137. Read online