

# Importing and Demultiplexing Illumina BCL Files

Primary sequencing output of an Illumina sequencer are per-cycle base call (bcl) files, which first need to be converted to fastq format, so that the data can be pushed to downstream applications. Partek Flow software comes with a conversion tool that can be used to import data in the bcl file format. In addition to the file conversion, this tool also demultiplexes the bcl files in the same step and outputs demultiplexed fastq files as the result.

We recommend you start by transferring the entire Illumina run folder to the Partek Flow server. To start a new project with bcl files, first select **bcl** under the **Other** import tab (Figure 1)

Single cell Bulk **Other**

Select the format

**Data repository**  
Use a previously published data node to data repository

**Create a blank sample**  
Add blank samples one at a time to then associated files with the samples

**Generic Count matrix**  
Import quantified data (e.g. gene counts per sample). Acceptable file types are txt, csv, tsv, txt.gz, csv.gz, tsv.gz

**fasta**  
Import unaligned reads. Acceptable file types are fasta, fasta.gz, fasta.bz2, fa, fa.gz, fa.bz2

**SRA**  
Import unaligned reads from the Sequence Read Archive from NCBI. Acceptable file type is sra

**bcl**  
Import raw base call files for demultiplexing and fastq file generation. You need an entire Illumina run folder, the RunInfo.xml file and a sample sheet csv file

Figure 1. Bcl file import setup dialog. Required input includes: RunInfo.xml file, SampleSheet.csv file, and a directory hosting .bcl files

The resulting window shows the configuration dialog (Figure 2).

**Data directory**  
The Data directory is located at the top level of the run folder

**Run info file**  
The RunInfo.xml is located at the top level of the run folder (-R)

[Transfer files to the server](#)

**Sample sheet file**  
The sample sheet specifies for every index in every lane which sample it belongs to (--sample-sheet)

☒ Partek Flow Server ☐ URL

[Transfer files to the server](#)

**Advanced options**

**Option set**

Figure 2. Bcl file import setup dialog. Required input includes: RunInfo.xml file, SampleSheet.csv file, and a directory hosting .bcl files

The bcl files hold the base calls and are in the *Data directory* within the whole Illumina run folder. Note that the *Data directory* file path needs to point to the directory, not to an individual bcl file.

The RunInfo.xml file is generated by the primary analysis software and contains information on the run, flow cell, instrument, time stamp, and the read structure (number of reads, number of cycles per read, whether a read is an index read). This file is typically stored at the top level in the Illumina run folder.

The SampleSheet.csv file provides the information on the relationship between the samples and indices specified during library creation. Although it has four sections, two sections (Settings and Data) are important for the data import and conversion. For more information on the files, consult Illumina documentation.

Selecting the **Configure** option under the *Advanced options* section enables a granular control of the import (Figure 3).

Figure 3. Advanced options of bcl importer

The *Select tiles* option (--tiles) enables the user to process only a subset of tiles available in the flow cell. The input for this option is a comma-separated list of regular expressions.

*Min trimmed read length* (--minimum-trimmed-read-length) specifies the minimum read length after adapter removal.

*Mask short adapter reads* (--mask-short-adapter-reads) applies when a read is trimmed below the length specified by *Min trimmed read length*. If the number of bases after adapter removal is less than *Min trimmed read length*, it forces the read length to be equal to *Min trimmed read length* by replacing the adapter bases that fall below the specified length by Ns. If the number of remaining bases falls below *Mask short adapter sequences*, then it replaces all the bases in a read with Ns.

*Adapter stringency* (--adapter-stringency) specifies the minimum match rate that triggers the masking or trimming of adapters. The rate is calculated as  $\text{MatchCount} / (\text{MatchCount} + \text{MismatchCount})$ . Only the reads exceeding the specified rate of sequence identity with adapters are trimmed.

*Barcode mismatches* (--barcode-mismatches) controls the number of allowed mismatches per index sequence.

*Use bases mask* (--use-bases-mask) defines a custom read structure that may be different to the structure specified in the RunInfo.xml file. The input for this option is a comma-separated list where Y and I are followed by a number indicating how many sequencing cycles to include in the fastq file. For example, if the option is set to Y26,I8,Y98, 26 cycles (26bp) will be used to generate the R1 sequence, 8 cycles (8bp) will be used for the sample index, and 98 cycles (98bp) will be used to generate the R2 sequence.

*Do not split files by lane* (--no-lane-splitting) prevents splitting of fastq files by lane, i.e. the converter will merge multiple lanes and generate one fastq file per sample.

*Create fastq for index reads* (--create-fastq-for-index-reads) creates an extra fastq file for each sample containing the sample index sequence for each read. This will be imported as an extra sample into the project.

*Ignore missing bcls* (--ignore-missing-bcl) will interpret missing base call files as N.

*Ignore missing filter* (--ignore-missing-filter) will ignore missing filter files and assume all clusters pass the filter.

*Ignore missing positions* (--ignore-missing-positions) will write new, unique coordinates into the header line if the cluster location files are missing.

*Ignore missing controls* (--ignore-missing-control) will interpret missing control files as missing not-set control bits.

*Save undetermined fastq* will take the reads that could not be assigned to a sample index and collect them into an Undetermined\_S0.fastq file, which will be imported as a new sample.

The result of the import is an *Unaligned reads* data node, containing demultiplexed fastq files.

For more information about the BCL to FASTQ conversion tool, including information on the proper folder structure and instructions for formatting the SampleSheet.csv file, please consult the [bcl2fastq2 Conversion Software Guide](#).

## Additional Assistance

If you need additional assistance, please visit [our support page](#) to submit a help ticket or find phone numbers for regional support.



Your Rating: Results: 32 rates