NOTE
There can be only one Unaligned directory by default. If you want multiple Unaligned directories, you will have to use the option `--output-dir` to generate a different output directory.

FASTQ Files

As of 1.8, CASAVA converts *.bcl files into FASTQ files, and uses these FASTQ files as sequence input for configureAlignment. The files are located in the Unaligned/Project_<ProjectName>/Sample_<SampleName> directories.

NOTE
Reads that were identified as sample prep controls in the control files are not saved in the FASTQ files.

Naming

Illumina FASTQ files use the following naming scheme:

```
<sample name>_<barcode sequence>_L<lane (0-padded to 3
   digits)>_R<read number>_<set number (0-padded to 3
   digits>.fastq.gz
```

For example, the following is a valid FASTQ file name:

```
NA10831_ATCACG_L002_R1_001.fastq.gz
```

In the case of non-multiplexed runs, <sample name> will be replaced with the lane numbers (lane1, lane2, ..., lane8) and <barcode sequence> will be replaced with "NoIndex".

Set Size

The FASTQ files are divided in files with the file size set by the --fastq-cluster-count command line option of configureBclToFastq.pl.The different files are distinguished by the 0-padded 3-digit set number.

TIP
If you need to combine the divided fastq gzipped files into one unique fastq gzipped file for use in a third-party tool, you can use the `cat` command, for example, like this:

```
cd Unaligned/Project_ID/sample_ID
cat NA10831_ATCACG_L002_R1_001.fastq.gz NA10831_ATCACG_
L002_R1_002.fastq.gz > NA10831_ATCACG_L002_R1_
combined.fastq.gz
```

Compression

FASTQ files are saved compressed in the GNU zip format, an open source file compression program. This is indicated by the .gz file extension. CASAVA automatically unzips the files before using them.

Format

Each entry in a FASTQ file consists of four lines:

- Sequence identifier
- Sequence
- Quality score identifier line (consisting of a +)
- Quality score

Each sequence identifier, the line that precedes the sequence and describes it, needs to be in the following format:

```
@<instrument>:<run number>:<flowcell ID>:<lane>:<tile>:<x-pos>:<y-pos> <read>:<is filtered>:<control number>:<index sequence>
```

The elements are described below.

Element	Requirements	Description
@	@	Each sequence identifier line starts with @
<instrument>	Characters allowed: a-z, A-Z, 0-9 and underscore	Instrument ID
<run number>	Numerical	Run number on instrument
<flowcell ID>	Characters allowed: a-z, A-Z, 0-9	
<lane>	Numerical	Lane number
<tile>	Numerical	Tile number
<x_pos>	Numerical	X coordinate of cluster
<y_pos>	Numerical	Y coordinate of cluster
<read>	Numerical	Read number. 1 can be single read or read 2 of paired-end
<is filtered>	Y or N	Y if the read is filtered, N otherwise
<control number>	Numerical	0 when none of the control bits are on, otherwise it is an even number
<index sequence>	ACTG	Index sequence

An example of a valid entry is as follows; note the space preceding the read number element:

```
@EAS139:136:FC706VJ:2:5:1000:12850 1:Y:18:ATCACG
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
+
BBBBCCCC?<A?BC?7@@???????DBBA@@@@A@@
```

NOTE

Be aware that the CASAVA 1.8 FASTQ files contain all reads, both the reads that passed filtering, as well as the reads that did not pass filtering. If you use third-party software that cannot handle this, we recommend that clean up the FASTQ files first using the <is filtered> field described above. you can use the

```
cd /path/to/project/sample
mkdir filtered
for fastq in *.fastq.gz ; do zcat $fastq | grep -A 4 '^@.* [^:]*:N:[^:]*:' > filtered/$fastq ; done
```

Quality Scores

A quality score (or Q-score) expresses an error probability. In particular, it serves as a convenient and compact way to communicate very small error probabilities.

Given an assertion, A, the probability that A is not true, P(~A), is expressed by a quality score, Q(A), according to the relationship:

$$Q(A) = -10 \log_{10}(P(\sim A))$$