Filetypes for Annotation & Alignment

J Fass | 20 June 2017

Filetypes

- Fasta
- Fastq
- GTF / GFF
- SAM / BAM / CRAM
- BED

See https://genome.ucsc.edu/FAQ/FAQformat.html

Fasta

>sequenceName | plus other junk | few maintain a standard here AGTGGAGCAAGCACAGAGAAGAAACTGCAGTCAGGACATAAAGTAAAGTA TTCCTGATATGTAATTAATCACTAAATGTCTTTAATGGGATCTCTTTCTA TTGAGATATTTGTAAACTTTCTTCATGTGATTGGTTTACAGATATTCAGG TTTCTGCAAATGGGTGCTGTCTATATTATAGAATTTTTAGTTGAAATTTT CAAAATACTCTTTGagtattctcttgtaattatattactttacaaggttt gtggggcatctctttcatttgtgattacatggttgcagtattctttttgt tcttagtcagactgtataattgtctgtgaagtccagtaaacttttgaaag

Assembly gap?

Soft-masked repetitive sequence? Low confidence assembly?

Fastq

```
@K00188:264:HG3WJBBXX:1:1101:6289:1595 1:N:0:TAGCTT
GGACTGCCTTTCAGCCCGTCGCAGAGGGAATGGGAGCCTCTGGAGCGGGTGCAGAGGCTCAGCAG
+
@header
<sequence>
+(sometimes header?)
<base qualities>
```

Fastq

0ct	Dec	Hex	Char	0ct	Dec	Hex	Char
000	0	00	NUL '\0' (null character)	100	64	40	@
001	1	01	SOH (start of heading)	101	65	41	Ā
002	2	02	STX (start of text)	102	66	42	В
003	3	03	ETX (end of text)	103	67	43	C
004	4	04	EOT (end of transmission)	104	68	44	D
005	5	05	ENQ (enquiry)	105	69	45	E
006	6	06	ACK (acknowledge)	106	70	46	F
007	7	07	BEL '\a' (bell)	107	71	47	G
010	8	08	BS '\b' (backspace)	110	72	48	Н
011	9	09	HT '\t' (horizontal tab)	111	73	49	1
012	10	0A	LF '\n' (new line)	112	74	4A	J
013	11	0B	VT '\v' (vertical tab)	113	75	4B	K

Fastq

0ct	Dec	Hex	Char
112	<mark>74</mark>	4A	J
<mark>74</mark> -	33 = 4	<mark>41</mark>	

Probability of error = $10 ^ (-41 / 10) \sim 0.0001$

41 is the "phred-scaled Q-value"

Standard FASTQ encodes qualities using "phred + 33" quality characters. See https://en.wikipedia.org/wiki/FASTQ_format for a good graphic about current and older encodings.

Common QC question: "how many reads have average of at least Q30?"

GTF / GFF

Gene Transfer Format / **Gene Feature Format** ... describes gene locations within the genome. If we align reads to the genome, reads that align to gene regions must have come from mRNA from that gene, yes?

GTF / GFF

- Human/mouse: GENCODE (uses Ensembl IDs) (http://www.gencodegenes.org/), but may need some manipulation to work with certain software
- Ensembl genomes (http://ensemblgenomes.org/) and Biomart (http://www.ensembl.org/biomart/martview/)
- Illumina igenomes (http://support.illumina.com/sequencing/ sequencing_software/igenome.html) provides indexes for some software, and files with extra info for Tophat/cufflinks.
- NCBI genomes (http://www.ncbi.nlm.nih.gov/genome/)
- Many specialized databases (Phytozome, Patric, VectorBase, FlyBase, WormBase)
- "Do it yourself" genome assembly and gene-finding (don't forget functional annotation)

GTF / GFF

chr12	unknown exon	4382902	4383401 .	+	
chr12	unknown CDS	4383207	4383401 .	+	
chr12	unknown start_codon	4383207	4383209 .	+	
chr12	unknown CDS	4385171	4385386 .	+	
chr12	unknown exon	4385171	4385386 .	+	
chr12	unknown CDS	4387926	4388085 .	+	
chr12	unknown exon	4387926	4388085 .	+	
chr12	unknown CDS	4398008	4398156 .	+	
chr12	unknown exon	4398008	4398156 .	+	:
chr12	unknown CDS	4409026	4409172 .	+	
chr12	unknown exon	4409026	4414522 .	+	
chr12	unknown stop_codon	4409173	4409175 .	+	

The left columns list source, feature type, and genomic coordinates

```
gene_id "CCND2"; gene_name "CCND2"; p_id "P6197"; transcript_id "NM_001759"; tss_id "TSS231"; gene_id "CCND2"; gene_name "CCND2"; p_id "P6197"; transcript_id "NM_001759"; tss_id "TSS231"; gene_id "CCND2"; gene_name "CCND2"; p_id "P6197"; transcript_id "NM_001759"; tss_id "TSS231"; gene_id "CCND2"; gene_name "CCND2"; p_id "P6197"; transcript_id "NM_001759"; tss_id "TSS231"; gene_id "CCND2"; gene_name "CCND2"; p_id "P6197"; transcript_id "NM_001759"; tss_id "TSS231"; gene_id "CCND2"; gene_name "CCND2"; p_id "P6197"; transcript_id "NM_001759"; tss_id "TSS231"; gene_id "CCND2"; gene_name "CCND2"; p_id "P6197"; transcript_id "NM_001759"; tss_id "TSS231"; gene_id "CCND2"; gene_name "CCND2"; p_id "P6197"; transcript_id "NM_001759"; tss_id "TSS231"; gene_id "CCND2"; gene_name "CCND2"; p_id "P6197"; transcript_id "NM_001759"; tss_id "TSS231"; gene_id "CCND2"; gene_name "CCND2"; p_id "P6197"; transcript_id "NM_001759"; tss_id "TSS231"; gene_id "CCND2"; gene_name "CCND2"; p_id "P6197"; transcript_id "NM_001759"; tss_id "TSS231"; gene_id "CCND2"; gene_name "CCND2"; p_id "P6197"; transcript_id "NM_001759"; tss_id "TSS231"; gene_id "CCND2"; gene_name "CCND2"; p_id "P6197"; transcript_id "NM_001759"; tss_id "TSS231"; gene_id "CCND2"; gene_name "CCND2"; p_id "P6197"; transcript_id "NM_001759"; tss_id "TSS231"; gene_id "CCND2"; gene_name "CCND2"; p_id "P6197"; transcript_id "NM_001759"; tss_id "TSS231"; gene_id "CCND2"; gene_name "CCND2"; p_id "P6197"; transcript_id "NM_001759"; tss_id "TSS231"; gene_id "CCND2"; gene_name "CCND2"; p_id "P6197"; transcript_id "NM_001759"; tss_id "TSS231"; gene_id "CCND2"; gene_name "CCND2"; p_id "P6197"; transcript_id "NM_001759"; tss_id "TSS231"; gene_id "CCND2"; gene_name "CCND2"; p_id "P6197"; transcript_id "NM_001759"; tss_id "TSS231"; gene_id "CCND2"; gene_name "CCND2"; p_id "P6197"; transcript_id "NM_001759"; tss_id "TSS231";
```

The right column includes attributes, including gene ID, etc.

chr12 unknown CDS 3677872 3678014 . + 2 gene_id "PRMT8"; gene_name "PRMT8"; p_id "P10933"; transcript_id "NM_019854"; tss_id "TSS4368";

GTF

```
Sequence Name (i.e., chromosome, scaffold, etc.)
                                                          chr12
Source (program that generated the gtf file or feature)
                                                          unknown
Feature (i.e., gene, exon, CDS, start codon, stop codon)
                                                          CDS
Start (starting location on sequence)
                                                          3677872
End (end position on sequence)
                                                          3678014
Score
Strand (+ or -)
Frame (0, 1, or 2: which is first base in codon, zero-based) 2
Attribute (";"-delimited list of tags with additional info)
           gene_id "PRMT8"; gene_name "PRMT8"; p_id "P10933";
           transcript_id "NM_019854"; tss_id "TSS4368";
```

GTF file with questionable attributes ...

```
gene id "AAEL005599";
gene id "AAEL005599"; transcript id "AAEL005599-RA";
gene id "AAEL005599"; transcript id "AAEL005599-RA"; exon number "1 of 4";
gene id "AAEL005599"; transcript id "AAEL005599-RA"; exon number "2 of 4";
gene id "AAEL005599"; transcript id
                                    "AAEL005599-RA"; exon number "3 of 4";
gene id "AAEL005599"; transcript id
                                    "AAEL005599-RA"; exon number "4 of 4";
transcript id "AAEL005599-RA";
gene_id "AAEL005599"; transcript id
                                    "AAEL005599-RA";
gene id "AAEL005599"; transcript id
                                    "AAEL005599-RA"; exon number "1 of 4";
gene id "AAEL005599"; transcript id
                                    "AAEL005599-RA"; exon number "2 of 4";
                                    "AAEL005599-RA"; exon number "3 of 4";
gene id "AAEL005599"; transcript id
                                    "AAEL005599-RA"; exon number "4 of 4";
gene id "AAEL005599"; transcript id
gene id "AAEL005599"; transcript id
                                    "AAEL005599-RA";
gene id "AAEL016379"; transcript id
                                    "AAEL005599-RA"; exon number "5 of 4";
                                    "AAEL005599-RA"; exon number "6 of 4";
gene id "AAEL016380"; transcript id
```

SAM / BAM / CRAM!

http://www.htslib.org/

See also samtools man page: http://samtools.sourceforge.net/

SAM spec grew out of 1000 Genomes Project (see Li et al. 2009 *Bioinformatics* 25:2078)

SAM is plain text; BAM is binary, compressed version of SAM; CRAM is further compressed but not widely used / recognizable by many tools.

```
[...]
    SN:ctg103993
@SQ
             LN:217
@SQ
    SN:ctg103994
             LN:222
                                 Header lines (start with "@")
@SQ
    SN:ctg103995
             LN:205
@SQ
   SN:ctg103996
            LN:210
@PG
   ID:bwa PN:bwa VN:0.7.13-r1126 CL:bwa mem -t 4 -M ../../01 Reference/Transcriptome-Contigs-Build2.fna
../../02-Cleaned/3E/3E SE.fastq
   ID:bwa-7BC92A6F PN:bwa VN:0.7.13-r1126 CL:bwa mem -t 4 -M ../../01_Referen Alignment in @ (one line for alignment)
@PG
../../02-Cleaned/3E/3E_R1.fastq ../../02-Cleaned/3E/3E_R2.fastq
K00188:264:HG3WJBBXX:1:1116:14692:35180#0
                              cta2
                                  128
                                      58
                                         101M =
                                               128
AAGTCTCGACCAAGTGGTTCAGATGGTGACACAGATGTTAGCCCCATCCACCATTCAGTTGCCGTTTTGATAGCTGGAAATCCTGTAAACACAA
K00188:264:HG3WJBBXX:1:1116:14692:35180#0
                           181 cta2
                                 128 0
                                                128
TTTAGTTTTAATTTTTGACTTTGAATAGCGGGAGTCCAGATCGTGTGAACACAGCAGACTGAGCACTCCATTGACAGCCTTCTTCTGTACTTTAGC
K00188:264:HG3WJBBXX:1:1202:11028:9596#0
                              cta5
                                  45
                                        101M =
TTCTTTTTCTACAGTTCATTGTCTGTATAAAGTATGCATCAGGAACAATCTGACTAGGAAGGTAAATAATGTAAAACAGATGATTATTGTATGAAA
K00188:264:HG3WJBBXX:1:1202:11028:9596#0
                           181
                              ctg5
                                 45
```

```
[...]
    SN:ctg103993
@SQ
             LN:217
@SQ
    SN:ctg103994
             LN:222
@SQ
    SN:ctg103995
             LN:205
@SQ
   SN:ctg103996
            LN:210
@PG
   ID:bwa PN:bwa VN:0.7.13-r1126 CL:bwa mem -t 4 -M ../../01 Reference/Transcriptome-Contigs-Build2.fna
../../02-Cleaned/3E/3E SE.fastq
@PG
   ID:bwa-7BC92A6F PN:bwa VN:0.7.13-r1126 CL:bwa mem -t 4 -M ../../01_Reference/TranscriptReach RairB(identical headers)
../../02-Cleaned/3E/3E R1.fastg ../../02-Cleaned/3E/3E R2.fastg
K00188:264:HG3WJBBXX:1:1116:14692:35180#0
                              cta2
                                  128
                                     58
                                         101M =
                                               128
AAGTCTCGACCAAGTGGTTCAGATGGTGACACAGATGTTAGCCCCATCCACCATTCAGTTGCCGTTTTGATAGCTGGAAATCCTGTAAACACAA
K00188:264:HG3WJBBXX:1:1116:14692:35180#0
                                                128
                           181 cta2
                                  128 0
TTTAGTTTTAATTTTTGACTTTGAATAGCGGGAGTCCAGATCGTGTGAACACAGCAGACTGAGCACTCCATTGACAGCCTTCTTCTGTACTTTAG¢
K00188:264:HG3WJBBXX:1:1202:11028:9596#0
                              cta5
                                        101M =
TTCTTTTTCTACAGTTCATTGTCTGTATAAAGTATGCATCAGGAACAATCTGACTAGGAAGGTAAATAATGTAAAACAGATGATTATTGTATGAAA
K00188:264:HG3WJBBXX:1:1202:11028:9596#0
                          181
                              ctg5
                                 45
```

UC Davis Genome Center | Bioinformatics Core | J Fass

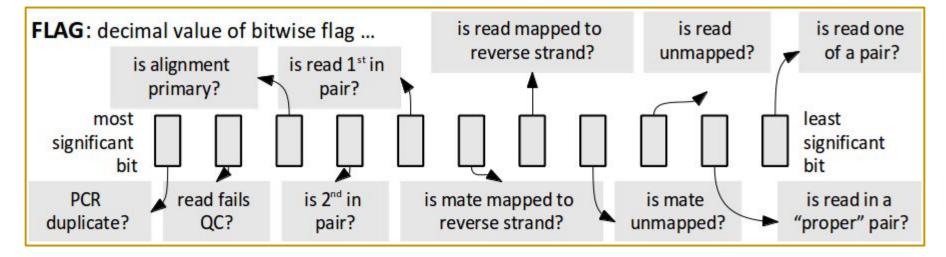
Col	Field	Type	Regexp/Range	Brief description
1	QNAME	String	[!-?A-~]{1,254}	Query template NAME
2	FLAG	Int	[0,2 ¹⁶ -1]	bitwise FLAG
3	RNAME	String	* [!-()+-<>-~][!-~]*	Reference sequence NAME
4	POS	Int	$[0,2^{31}-1]$	1-based leftmost mapping POSition
5	MAPQ	Int	$[0,2^8-1]$	MAPping Quality
6	CIGAR	String	* ([0-9]+[MIDNSHPX=])+	CIGAR string
7	RNEXT	String	* = [!-()+-<>-~][!-~]*	Ref. name of the mate/next read
8	PNEXT	Int	$[0,2^{31}-1]$	Position of the mate/next read
9	TLEN	Int	$[-2^{31}+1,2^{31}-1]$	observed Template LENgth
10	SEQ	String	* [A-Za-z=.]+	segment SEQuence
11	QUAL	String	[!-~]+	ASCII of Phred-scaled base QUALity+33

QNAME: Query name

Read IDs are truncated at first whitespace (spaces / tabs), which can make them *non-unique*. Illumina reads with older IDs have trailing "/1" and "/2" stripped (this information is recorded in the next field). Illumina reads with newer IDs have second block stripped (read number is recorded in the next field).

```
@FCC6889ACXX:5:1101:8446:45501#CGATGTATC/1 ⇒ @FCC6889ACXX:5:1101:8446:45501
```

@HISEQ:153:H8ED7ADXX:1:1101:1368:2069 1:N:0:ATCACG ⇒ @HISEQ:153:H8ED7ADXX:1:1101:1368:2069



```
HISEQ:153:H8ED7ADXX:1:1104:8193:69947
```

```
99 (decimal) = 00001100011 (binary)

4773690

50

101M

=
4773721

132
```

```
... so, (from right to left): read is in a pair; the pair is proper; read is mapped (double neg); mate is mapped (double neg); read is mapped to forward strand (double neg); mate is mapped to reverse strand; read is 1st in pair ... remaining bits not used
```

(0 / NO .. 1 / YES)

```
AS:i:0 XN:i:0 XM:i:0 XO:i:0 XG:i:0 NM:i:0 MD:Z:101 YT:Z:UU XS:A:- NH:i:1
```

FLAG: still confused?

```
https://broadinstitute.github.io/picard/explain-flags.html
Common flags for SR (single reads): 0, 4, 16, sometimes 20 (hmm..)
Common flags for PE (paired ends): 99/147, 83/163, 77/141, 65/129, 81/161 ...
HISEQ:153:H8ED7ADXX:1:1104:8193:69947
99
chr1
4773690
50
101M
4773721
132
GTGCCATCTGTGGGCTGGTGATC[...]AGCAGCATGCTCCATGGTCTCTACATG
AS:i:0 XN:i:0 XM:i:0 XO:i:0 XG:i:0 NM:i:0 MD:Z:101 YT:Z:UU XS:A:- NH:i:1
```

```
RNAME: reference sequence name

Reference sequence ID (from fasta header), possibly truncated at first whitespace (still unique??)

>chromosome 1
... becomes ...
chromosome
... (!)
```

```
POS: 1-based leftmost position of (post-clipping) aligned read
       ... 4,773,680 4,773,690 4,773,700 4,773,710 ...
REF: ... TACCCAATGGGGATGACATAAGGTGCCATCTGTGGGCTGGTGATTCCATAGTAGAC...
READ:
                               GGTGCCATCTGTGGGCTGGTGATCCCATAGTAGAC...
HISEQ:153:H8ED7ADXX:1:1104:8193:69947
99
chr1
4773690
50
101M
4773721
132
GTGCCATCTGTGGGCTGGTGATC[...]AGCAGCATGCTCCATGGTCTCTACATG
AS:i:0 XN:i:0 XM:i:0 XO:i:0 XG:i:0 NM:i:0 MD:Z:101 YT:Z:UU XS:A:- NH:i:1
```

```
POS: 1-based leftmost position of (post-clipping) aligned read
       ... 4,773,680 4,773,690 4,773,700 4,773,710 ...
REF:
        ...CCAATGGGGATGACATAAGTGCCATCTGTGGGCTGGTGATCAGTAGAC...
                               GTGCCATCTGTGGGCTGGTGATCAGTAGAC...
READ:
HISEQ:153:H8ED7ADXX:1:1104:8193:69947
99
chr1
4773690
50
49H101M
4773721
132
GTGCCATCTGTGGGCTGGTGATC[...]AGCAGCATGCTCCATGGTCTCTACATG
AS:i:0 XN:i:0 XM:i:0 XO:i:0 XG:i:0 NM:i:0 MD:Z:101 YT:Z:UU XS:A:- NH:i:1
```

```
POS: 1-based leftmost position of (post-clipping) aligned read
       ... 4,773,680 4,773,690 4,773,700 4,773,710 ...
REF:
        ...CCAATGGGGATGACATAAGTGCCATCTGTGGGCTGGTGATCAGTAGAC...
READ:
                            GCCGTGCCATCTGTGGGCTGGTGATCAGTAGAC...
HISEQ:153:H8ED7ADXX:1:1104:8193:69947
99
chr1
4773690
50
3S101M
4773721
132
GCCGTGCCATCTGTGGGCTGGTGATC[...]AGCAGCATGCTCCATGGTCTCTACATG
AS:i:0 XN:i:0 XM:i:0 XO:i:0 XG:i:0 NM:i:0
                                   MD:Z:GCC101(?) YT:Z:UU XS:A:- NH:i:1
```

MAPQ: mapping quality (phred scaled)

Mapping quality is used by some aligners, in different ways. It's generally a function of the edit distance (mismatches, indels), and the uniqueness of the alignment. Multiple equivalent best alignments yield a mapping quality of zero; alignments with an edit distance close to the best alignment lower the mapping quality.

CIGAR: extended CIGAR string (Compact Idiosyncratic Gapped Alignment Report)

Format: [0-9][MIDNSHP][0-9][MIDNSHP]...

M = match / mismatch (!), I/D = insertion / deletion, N = skipped bases on reference, S/H = soft / hard clip (hard clipped bases no longer appear in the sequence field), P = padding ... e.g. "101M" means that all bases in the read align to bases in the reference, starting with position (4,773,690), always in the order of the reference.

MRNM: reference sequence to which the *mate* of this read is aligned

"=" ... mate is aligned to the same reference sequence is this read

"*" ... this is a single read; no mate exists

MPOS: 1-based, left-most position of 1st (post-clipping) nucleotide of mate read

"0" ... no mate exists

```
TLEN: inferred insert size / template length ... "0" if no mate ... "-#" if second read(?)

reference

template length
```

SEQ and **QUAL:** read's nucleotides and base qualities, *always in the order of the reference (forward, top) strand!* ... includes any insertions, deletions, etc. present in the read.

Reads aligned to reverse strand appear in reverse, with reversed base qualities.

AS:i:0

OPT: various pre-defined and user-defined tags in the format TAG:VTYPE:VALUE ... VTYPE is one of [A (printable character); i (signed integer); f (floating point); z (printable string); H (hex string)].

NM:i:0

YT:Z:UU XS:A:-

e.g.: XS:A:- was set by TopHat, RNA that was read was coded by the reverse strand

XN:i:0 XM:i:0 XO:i:0 XG:i:0

e.g.: NM:i:0 means zero mismatches in this alignment

SAM - quick summary

```
12345678901234
                                      5678901234567890123456789012345
               coor
               ref
                      AGCATGTTAGATAA**GATAGCTGTGCTAGTAGGCAGTCAGCGCCAT
Paired-end-
               r001+
                            TTAGATAAAGGATA*CTG
               r002+
                           aaaAGATAA*GGATA
               r003+
                         gccta AGCTAA
               r004+
                                        ATAGCT.....
                                                            TCAGC
 Multipart-
               r003-
                                               ttagct TAGGC
               r001-
                                                             CAGCGCCAT
              @SQ SN:ref LN:45
Ins & padding
                             7 30 8M2I4M1D3M = 37
                                                    39 TTAGATAAAGGATACTA
 Soft clipping
               r002
                             9 30 3S6M1P1I4M *
                                                     Ø AAAAGATAAGGATA
               r003
                      0 ref
                             9 30 5H6M
                                                     0 AGCTAA
                                                                     NM:i:1
     Splicing
               r004
                      0 ref 16 30 6M14N5M
                                                     0 ATAGCTTCAGC
Hard clipping
               r003
                     16 ref 29 30 6H5M
                                                     0 TAGGC
                                                                     NM:i:0
               r001
                     83 ref 37 30 9M
                                                 7 -39 CAGCGCCAT
               ref 7 T 1 .
                            Iref 12 T 3 ...
                                                      Iref 17 T 3 ...
               ref 8 T 1 .
                               ref 13 A 3 ...
                                                       ref 18 A 3 .-1G..
               ref 9 A 3 ...
                               ref 14 A 2 .+2AG.+1G.
                                                       ref 19 G 2 *.
               ref 10 G 3 ...
                               ref 15 G 2 ..
                                                       ref 20 C 2 ..
               ref 11 A 3 ...C
                                ref 16 A 3 ...
                                                       . . .
```

google "Heng Li slides" - Challenges and Solutions in the Analysis of Next Generation Sequencing Data (2010)

BAM

BAMs are compressed SAMs (so, binary, not human-readable text ... don't look directly at them!). They can be indexed to allow rapid extraction of information, so alignment viewers do not need to uncompress the whole BAM file in order to look at information for a particular read or coordinate range, somewhere in the file.

Indexing your BAM file, myCoolBamFile.bam, will create an index file, myCoolBamFile.bam.bai, which is needed (in addition to the BAM file) by viewers and other downstream tools. An occasional downstream tool will require an index called myCoolBamFile.bai (notice that the ".bai" replaces the ".bam", instead of being appended after it).

CRAM

Available as of SAMtools 1.0, and is a binary format like BAM. Uses data-specific compression tools (i.e. compressing letters is different than compressing numbers), *specifically* reference-based compression (e.g. for aligned reads, only *mis-matching* bases need to be stored). Also can employ *lossy* compression of base qualities, which appears to have a negligible effect on, say, variant calling (see Illumina *white paper*).

Indexing your CRAM file, myCoolBamFile.cram, will create an index file, myCoolBamFile.cram.crai, which is needed (in addition to the CRAM file) by viewers and other downstream tools.

This is still a *relatively recent development*, so it may be a while before many tools are CRAM-capable.

?'s ...?