Sequence Alignment/Map Optional Fields Specification

The SAM/BAM Format Specification Working Group

17 Feb 2025

The master version of this document can be found at `https://github.com/samtools/hts-specs`.
This printing is version f3bb8bc from that repository, last modified on the date shown above.

This document is a companion to the *Sequence Alignment/Map Format Specification* that defines the SAM and BAM formats, and to the *CRAM Format Specification* that defines the CRAM format.[1] Alignment records in each of these formats may contain a number of optional fields, each labelled with a *tag* identifying that field's data. This document describes each of the predefined standard tags, and discusses conventions around creating new tags.

1 Standard tags

Predefined standard tags are listed in the following table and described in greater detail in later subsections. Optional fields are usually displayed as `TAG:TYPE:VALUE`; the *type* may be one of `A` (character), `B` (general array), `f` (real number), `H` (hexadecimal array), `i` (integer), or `Z` (string).

Tag	Type	Description
AM	i	The smallest template-independent mapping quality in the template
AS	i	Alignment score generated by aligner
BC	Z	Barcode sequence identifying the sample
BQ	Z	Offset to base alignment quality (BAQ)
BZ	Z	Phred quality of the unique molecular barcode bases in the OX tag
CB	Z	Cell identifier
CC	Z	Reference name of the next hit
CG	B,I	BAM only: CIGAR in BAM's binary encoding if (and only if) it consists of >65535 operators
CM	i	Edit distance between the color sequence and the color reference (see also NM)
CO	Z	Free-text comments
CP	i	Leftmost coordinate of the next hit
CQ	Z	Color read base qualities
CR	Z	Cellular barcode sequence bases (uncorrected)
CS	Z	Color read sequence
CT	Z	Complete read annotation tag, used for consensus annotation dummy features
CY	Z	Phred quality of the cellular barcode sequence in the CR tag
DI	Z	Duplicate Identity, for identifying the queryname that this read is a duplicate of
DS	i	Duplicate-set Size containing the size of the duplicate set
DT	Z	Duplicate type, used to identifying duplicate reads as coming from the library-construction (LB) or sequencing (SQ)
E2	Z	The 2nd most likely base calls
FI	i	The index of segment in the template

[1] See `SAMv1.pdf` and `CRAMv3.pdf` at `https://github.com/samtools/hts-specs`.

BQ:Z:`qualities` Offset to base alignment quality (BAQ), of the same length as the read sequence. At the i-th read base, $\mathrm{BAQ}_i = Q_i - (\mathrm{BQ}_i - 64)$ where Q_i is the i-th base quality.

CC:Z:`rname` Reference name of the next hit; '`=`' for the same chromosome.

CG:B:I,`encodedCigar` Real CIGAR in its binary form if (and only if) it contains >65535 operations. This is a BAM file only tag as a workaround of BAM's incapability to store long CIGARs in the standard way. SAM and CRAM files created with updated tools aware of the workaround are not expected to contain this tag. See also the footnote in Section 4.2 of the SAM spec for details.

CP:i:`pos` Leftmost coordinate of the next hit.

DI:Z:`rname` (For duplicate templates) The queryname of the template that is not marked as duplicate and that this template is a duplicate of.

DS:i:`count` Size of the duplicate set that the template is part of.

DT:Z:`str` (For duplicate templates) Either LB or SQ indicating the source of the duplication. Use LB if the duplication occured during library-construction (e.g., via PCR). Use SQ if the duplication occured during sequencing (e.g., due to imaging error, aka "optical duplicates", or due to over-aggressive bridge-amp).

GL:f:`score`

E2:Z:`bases` The 2nd most likely base calls. Same encoding and same length as SEQ. See also `U2` for associated quality values.

FI:i:`int` The index of segment in the template.

FS:Z:`str` Segment suffix.

H0:i:`count` Number of perfect hits.

H1:i:`count` Number of 1-difference hits (see also `NM`).

H2:i:`count` Number of 2-difference hits.

HI:i:*i* Query hit index, indicating the alignment record is the i-th one stored in SAM.

IH:i:`count` Number of alignments stored in the file that contain the query in the current record.

MC:Z:`cigar` CIGAR string for mate/next segment.

MD:Z:`[0-9]+(([A-Z]|\^[A-Z]+)[0-9]+)*`

String encoding mismatched and deleted reference bases, used in conjunction with the CIGAR and SEQ fields to reconstruct the bases of the reference sequence interval to which the alignment has been mapped. This can enable variant calling without requiring access to the entire original reference.

The `MD` string consists of the following items, concatenated without additional delimiter characters:

- `[0-9]+`, indicating a run of reference bases that are identical to the corresponding SEQ bases;
- `[A-Z]`, identifying a single reference base that differs from the SEQ base aligned at that position;
- `\^[A-Z]+`, identifying a run of reference bases that have been deleted in the alignment.

As shown in the complete regular expression above, numbers alternate with the other items. Thus if two mismatches or deletions are adjacent without a run of identical bases between them, a '`0`' (indicating a 0-length run) must be used to separate them in the `MD` string.

Clipping, padding, reference skips, and insertions ('`H`', '`S`', '`P`', '`N`', and '`I`' CIGAR operations) are not represented in the `MD` string. When reconstructing the reference sequence, inserted and soft-clipped SEQ bases are omitted as determined by tracking '`I`' and '`S`' operations in the CIGAR string. (If the