The Variant Call Format and VCFtools

Petr Danecek[1], Adam Auton[2], Goncalo Abecasis[3], Cornelis A. Albers[1], Eric Banks[4], Mark A. DePristo[4], Bob Handsaker[4], Gerton Lunter[5], Garbor Marth[6], Steve Sherry[7], Gilean McVean[8], Richard Durbin[1,*] and 1000 Genomes Project Analysis Group[9]

[1]Wellcome Trust Sanger Institute, Cambridge, CB10 1SA, UK; [2]University of Oxford, Wellcome Trust Centre for Human Genetics, Oxford, OX3 7BN, UK; [3]Center for Statistical Genetics, Department of Biostatistics, University of Michigan, Ann Arbor, M48109, USA; [4]Broad Institute of MIT and Harvard, Cambridge, MA 02141, USA; [5]University of Oxford, Department of Physiology, Anatomy and Genetics, Oxford, OX1 3QX, UK; [6]Boston College, Department of Biology, MA 02467, USA; [7]National Institutes of Health National Center for Biotechnology Information, MD 20894, USA; [8]University of Oxford Department of Statistics, Oxford, OX1 3TG, UK; [9]http://www.1000genomes.org

Abstract

One of the main uses of next-generation sequencing is to discover variation amongst large populations of related samples. Recently the format for storing next-generation read alignments has been standardised by the SAM/BAM file format specification. This has significantly improved the interoperability of next-generation tools for alignment, visualisation, and variant calling. We propose the Variant Call Format (VCF) as a standarised format for storing the most prevalent types of sequence variation, including SNPs, indels and larger structural variants, together with rich annotations. VCF is usually stored in a compressed manner and can be indexed for fast data retrieval of variants from a range of positions on the reference genome. The format was developed for the 1000 Genomes Project, and has also been adopted by other projects such as UK10K, dbSNP, or the NHLBI Exome Project. VCFtools is a software suite that implements various utilities for processing VCF files, including validation, merging and comparing, and also provides a general Perl and Python API. The VCF specification and VCFtools are available from http://vcftools.sourceforge.net.

Example

VCF header:

```
##fileformat=VCFv4.0
##fileDate=20100707
##source=VCFtools
##reference=NCBI36
##INFO=<ID=AA,Number=1,Type=String,Description="Ancestral Allele">
##INFO=<ID=H2,Number=0,Type=Flag,Description="HapMap2 membership">
##FORMAT=<ID=GT,Number=1,Type=String,Description="Genotype">
##FORMAT=<ID=GQ,Number=1,Type=Integer,Description="Genotype Quality (phred score)">
##FORMAT=<ID=GL,Number=3,Type=Float,Description="Likelihoods for RR,RA,AA genotypes (R=ref,A=alt)">
##FORMAT=<ID=DP,Number=1,Type=Integer,Description="Read Depth">
##ALT=<ID=DEL,Description="Deletion">
##INFO=<ID=SVTYPE,Number=1,Type=String,Description="Type of structural variant">
##INFO=<ID=END,Number=1,Type=Integer,Description="End position of the variant">
```

Body:

```
#CHROM POS ID     REF  ALT    QUAL FILTER INFO                FORMAT     SAMPLE1   SAMPLE2
1      1   .      ACG  A,AT    .   PASS   .                   GT:DP      1/2:13    0/0:29
1      2   rs1    C    T,CT    .   PASS   H2;AA=T             GT:GQ      0|1:100   2/2:70
1      5   .      A    G       .   PASS   .                   GT:GQ      1|0:77    1/1:95
1      100 .      T    <DEL>   .   PASS   SVTYPE=DEL;END=300 GT:GQ:DP   1/1:12:3  0/0:20
```

Mandatory header lines

Optional header lines (meta-data about the annotations in the VCF body)

Reference alleles (GT=0)

Alternate alleles (GT>0 is an index to the ALT column)

Deletion

SNP

Large SV

Insertion

Other event

Phased data (G and C above are on the same chromosome)

Types of variants

SNPs

Alignment	*VCF representation*		
ACGT	POS	REF	ALT
ATGT	2	C	T

Insertions

Alignment	*VCF representation*		
AC-GT	POS	REF	ALT
ACTGT	2	C	CT

Deletions

Alignment	*VCF representation*		
ACGT	POS	REF	ALT
A--T	1	ACG	A

Complex events

Alignment	*VCF representation*		
ACGT	POS	REF	ALT
A-TT	1	ACG	AT

Large structural variants

VCF representation

POS	REF	ALT	INFO
100	T	<DEL>	SVTYPE=DEL;END=300

Extensible meta-data

Annotations may apply to the variant as a whole (the **INFO** column) or to each genotype (the **FORMAT** column). In addition to genotype, other commonly used annotations include genotype likelihoods, dbSNP membership, ancestral allele, read depth, mapping quality, and others.

VCFtools

- Format validation
- Annotating
- Comparing, calculating basic statistics
- Merging
- Creating intersections and complements

Examples

```
# Validate VCF files
vcf-validator file.vcf.gz

# Compare VCF files
compare-vcf A.vcf.gz B.vcf.gz C.vcf.gz

# List positions present in at least two of the files
vcf-isec -n +2 A.vcf.gz B.vcf.gz C.vcf.gz > out.vcf
```

VCF highlights

- Meta-data - fexible and extensible
- Text format - easy to generate and parse
- Stored compressed - compact size
- Indexed by `tabix` - fast random access by genomic position
- Open source implementation - VCFtools, GATK, ... (C++, Java, general Perl and Python API)

http://vcftools.sourceforge.net/specs.html