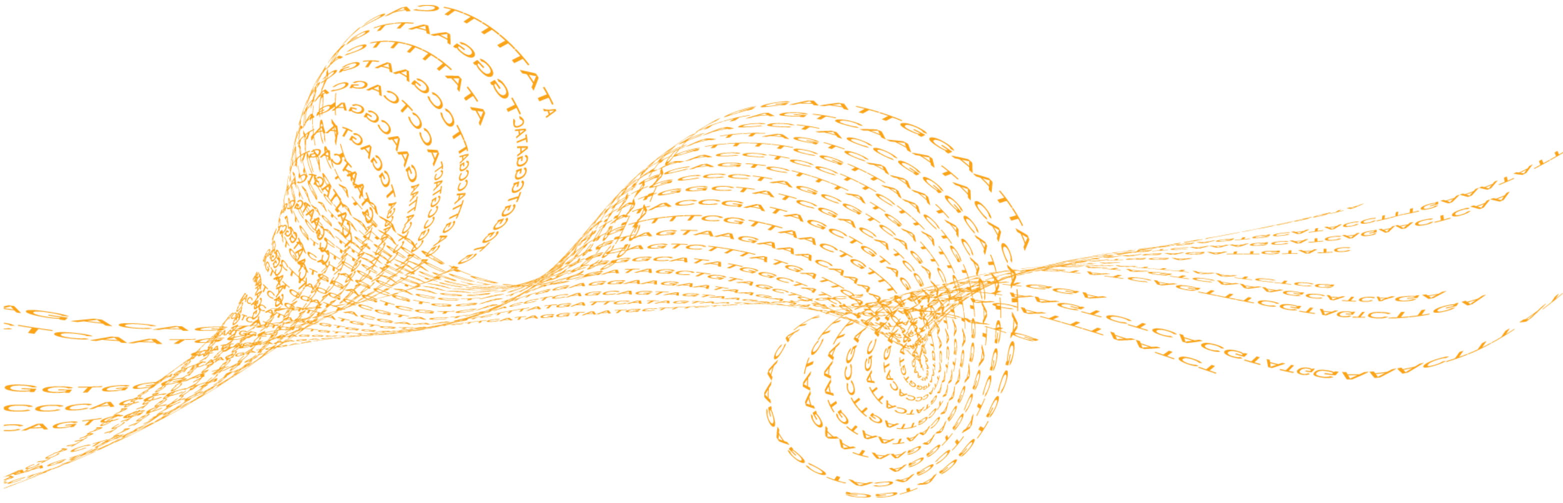
# Isaac Whole Genome Sequencing v2



ILLUMINA PROPRIETARY
15050954 Rev. B
August 2014

This document and its contents are proprietary to Illumina, Inc. and its affiliates ("Illumina"), and are intended solely for the contractual use of its customer in connection with the use of the product(s) described herein and for no other purpose. This document and its contents shall not be used or distributed for any other purpose and/or otherwise communicated, disclosed, or reproduced in any way whatsoever without the prior written consent of Illumina. Illumina does not convey any license under its patent, trademark, copyright, or common-law rights nor similar rights of any third parties by this document.

The instructions in this document must be strictly and explicitly followed by qualified and properly trained personnel in order to ensure the proper and safe use of the product(s) described herein. All of the contents of this document must be fully read and understood prior to using such product(s).

FAILURE TO COMPLETELY READ AND EXPLICITLY FOLLOW ALL OF THE INSTRUCTIONS CONTAINED HEREIN MAY RESULT IN DAMAGE TO THE PRODUCT(S), INJURY TO PERSONS, INCLUDING TO USERS OR OTHERS, AND DAMAGE TO OTHER PROPERTY.

ILLUMINA DOES NOT ASSUME ANY LIABILITY ARISING OUT OF THE IMPROPER USE OF THE PRODUCT(S) DESCRIBED HEREIN (INCLUDING PARTS THEREOF OR SOFTWARE) OR ANY USE OF SUCH PRODUCT(S) OUTSIDE THE SCOPE OF THE EXPRESS WRITTEN LICENSES OR PERMISSIONS GRANTED BY ILLUMINA IN CONNECTION WITH CUSTOMER'S ACQUISITION OF SUCH PRODUCT(S).

**FOR RESEARCH USE ONLY**

# Introduction

After BaseSpace® has generated FASTQ files containing the base calls and quality scores of a run, you can use the Isaac Whole Genome Sequencing app to analyze the sequencing data. The resulting data are analyzed in two steps: alignment to the reference genome followed by assembly and variant calling.

The Isaac Whole Genome Sequencing app uses the following modules for analysis:

- Alignment with the Isaac Alignment Software
- Variant calling with the Isaac Variant Caller
- Structural variant and large indels with the Large Indel and Structural Variant Caller.
- Copy number variation (CNV) analysis with the CNV Variant Caller.

These modules produce the following output:

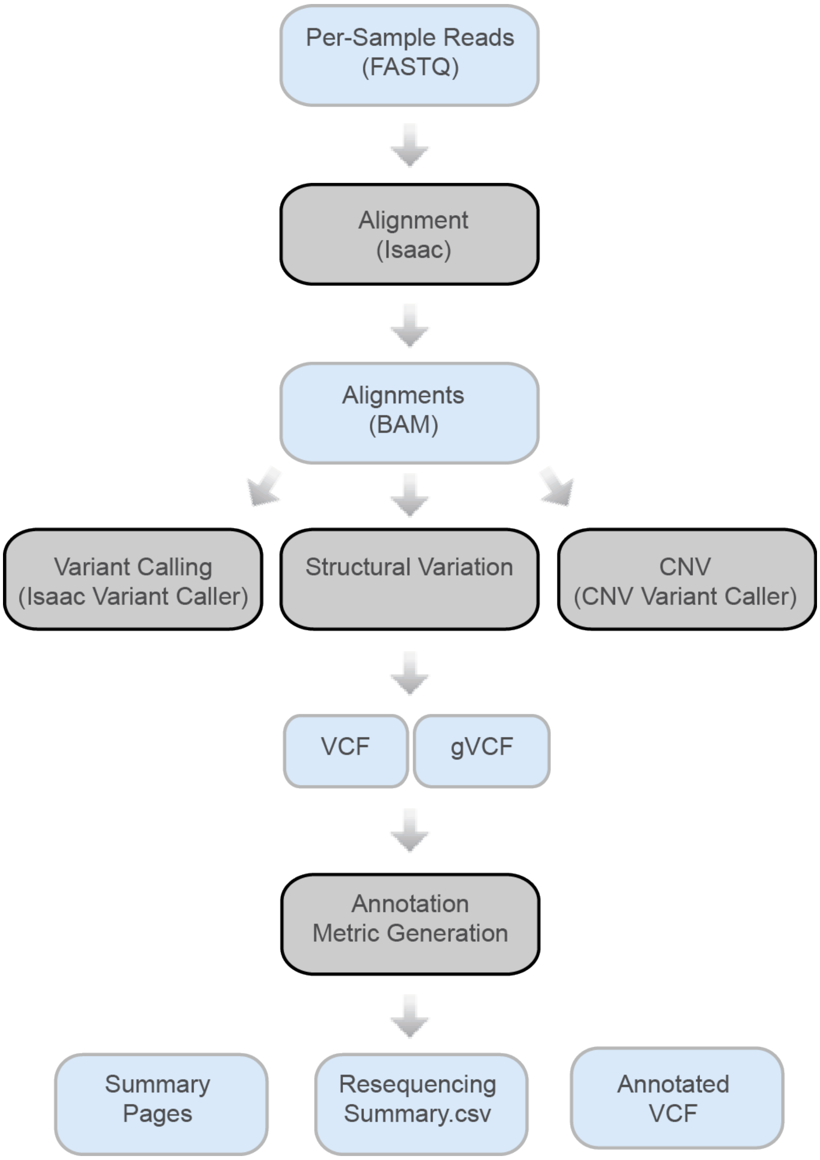
- Realigned and duplicate-marked reads in BAM file format.
- Variants in VCF file format.
- An additional Genome VCF (gVCF) file. This file features an entry for every base in the reference, which differentiates reference calls and no calls, and a summary of quality.

In addition, there is an annotation and metric generation step with the following output:

- Summary pages.
- A Resequencing_summary.csv file.
- Annotated VCF file. This binary file can be loaded into VariantStudio for viewing; see www.illumina.com/clinical/clinical_informatics/illumina-variantstudio.ilmn.

See *Isaac Whole Genome Sequencing Methods* on page 23 and *Isaac Whole Genome Sequencing Output* on page 8 for more information.

Figure 1 Isaac Whole Genome Sequencing Workflow



## Versions

The following module versions are used in the Isaac Whole Genome Sequencing app:

- Isaac: 01.13.10.21
- Isaac Variant Caller: 2.0.17
- CNV Variant Caller (CNVseg): 2.2.4
- Structural Variant Caller (Grouper): 1.4.2
- Samtools: 0.1.18
- Tabix: 0.2.5 (r1005)

## Current Limitations

Before running the Isaac Whole Genome Sequencing app, be aware of the following limitations:

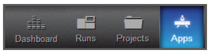
- Currently the app does not support mate-pair or other non-forward/reverse styles of paired-end sequencing.
- Currently the app does not support annotation of non-human genomes.
- Requires a minimum read length of 32 bp and a maximum read length of 150 bp.
- Minimum recommended data set size is enough data to yield 10 × coverage after alignment of the genome being sequenced. See Table 1 for recommended minimum number of reads for 10 × coverage.
- Maximum data set size must be fewer than 200 Gigabases, which equates to the following:
  - Approximately 1 billion reads assuming 2 × 100
  - Approximately 665 million reads assuming 2 × 150
- Completedjobinfo.xml may not print all statistics.
- Sample name length has a maximum of 32 characters.
- GQX can be entered as any value, although the maximum recommended value is 99.

For recommended minimum number of reads for 10 × coverage for different species, see Table 1. The number of reads listed yields 10 × coverage with an additional 5% to account for unaligned reads.

Table 1 Recommended Minimums for 10 × Coverage

| | Genome Size | Data Size | Reads for 2 × 100 (million) | Reads for 2 × 150 (million) |
|---|---|---|---|---|
| *Arabidopsis thaliana* | 63.4 Mb | 666 Mb | 3.33 | 2.22 |
| *Bos taurus* | 2.65 Gb | 28 Gb | 140.00 | 93.33 |
| *Escherichia coli* K-12 DH10B | 4.7 Mb | 50 Mb | 0.25 | 0.17 |
| *Escherichia coli* K-12 MG1655 | 4.6 Mb | 49 Mb | 0.25 | 0.16 |
| *Drosophila melanogaster* | 139.5 Mb | 1.5 Gb | 7.33 | 4.88 |
| Human | 3.3 Gb | 35 Gb | 175.00 | 116.67 |
| *Mus musculus* | 2.6 Gb | 28 Gb | 140.00 | 93.33 |
| PhiX (Illumina) | 5386 b | 57 Kb | 282.77 | 188.51 |
| *Rattus norvegicus* | 2.9 Gb | 31 Gb | 155.00 | 103.33 |
| *Rhodobacter sphaeroides* 2.4.1 | 4.6 Mb | 49 Mb | 0.25 | 0.16 |
| *Saccharomyces cerevisiae* | 12.2 Mb | 129 Mb | 0.65 | 0.43 |
| *Staphylococcus aureus* NCTC 8325 | 12.8 Mb | 135 Mb | 0.68 | 0.45 |

# Running Isaac Whole Genome Sequencing

1 Click the Apps button. 

2 Find **Isaac Whole Genome Sequencing v2** in the list and click the **Launch** button.

3 Read the End-User License Agreement and permissions, and click **Accept** if you agree.

4 Fill out the required fields in the Isaac Whole Genome Sequencing input form:

   a **Analysis Name**: Provide the analysis name. Default name is the app name with the date and time the app session was started.

   b **Save Results To**: Select the project that stores the app results.

   c **Sample**: Browse to the sample you want to analyze, and select the checkbox.

   d **Reference Genome**: Select the reference genome.

   e **Enable SV/CNV calling**: If selected, structural variants (SVs) and copy number variants (CNVs) are called using the tools Grouper and CNVSeg, respectively. For more information, see *Large Indel and Structural Variant Calls* on page 25 and *CNV Variant Caller* on page 26.

   f **Annotation**: Choose which gene and transcript annotation reference database to use.

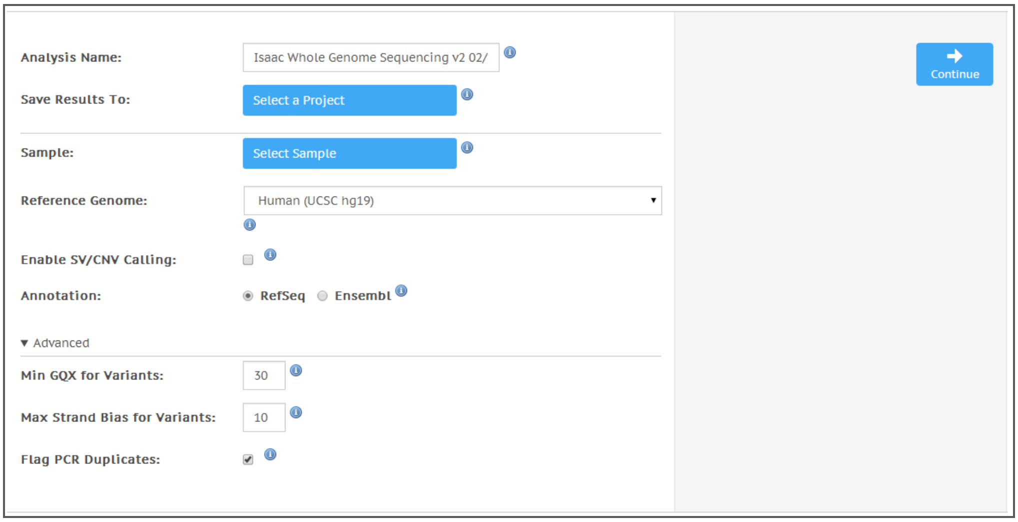
5 If desired, fill out the advanced fields in the Isaac WGS input form:

   a **Min GQX for Variants**: Enter GQX for variants. GQX is the minimum of the GQ (genotype quality) and QUAL (low quality filter), which makes it a conservative filter. Default value is 30, the maximum recommended is 99.

   b **Max Strand Bias for Variants**: Choose the maximum allowed strand bias for variant calling. This option filters for reads in which the differences in allele frequencies for forward- and reverse-strand reads is too high. Default is 10.

   c **FlagPCRDuplicates**: If selected, PCR duplicates are flagged in the BAM files and not used for variant calling. PCR duplicates are defined as paired-end reads generated from two clusters that have the exact same alignment positions for each read. Optical duplicates are already filtered out during RTA processing.

Figure 2 Isaac Whole Genome Sequencing Input Form



Analysis Name: Isaac Whole Genome Sequencing v2 02/

Save Results To: Select a Project

Sample: Select Sample

Reference Genome: Human (UCSC hg19) ▾

Enable SV/CNV Calling: □

Annotation: ● RefSeq ○ Ensembl

▾ Advanced

Min GQX for Variants: 30

Max Strand Bias for Variants: 10

Flag PCR Duplicates: ☑

6 Click **Continue**.

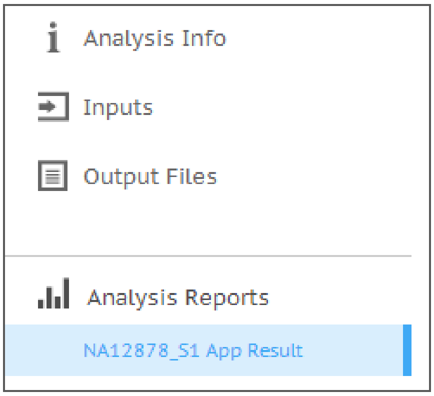
The Isaac Whole Genome Sequencing app now starts analyzing your sample. When completed, the status of the app session is automatically updated, and you receive an email.

NOTE
If needed, you can merge sampleFASTQ files in BaseSpace. See the BaseSpace User Guide for more information.

# Isaac Whole Genome Sequencing Output

This chapter describes the output that the Isaac Whole Genome Sequencing app produces. To go to the results, click the **Projects** button, then the project, then the analysis.

Figure 3 Isaac Whole Genome Sequencing Output Navigation Bar



When the App Session is completed, you can access your output through the left navigation bar, which provides the following:

- **Analysis Info**: an overview of the app session settings. See *Analysis Info* on page 11 for a description.
- **Inputs**: overview of input settings, see *Inputs* on page 12
- **Output Files**: access to the output files, organized by sample and app session. See *Isaac Whole Genome Sequencing Files* on page 12 for descriptions.
- **Analysis Reports**: access to analysis metrics for each sample. See *Analysis Reports* on page 8 for a description.

## Analysis Reports

The Isaac Whole Genome Sequencing app provides an overview of statistics per sample on the sample pages. A brief description of the metrics is below.

### Alignment Summary

| Statistic | Definition |
|---|---|
| Number of reads | Total number of reads passing filter for this sample. |
| Coverage | Total number of aligned bases divided by the genome size. |
| Percent Duplicate Paired Reads | Percentage of paired reads that have duplicates. |
| Fragment Length Median | Median length of the sequenced fragment. The fragment length is calculated based on the locations at which a read pair aligns to the reference. The read mapping information is parsed from the BAM files. |

| Statistic | Definition |
|---|---|
| Fragment Length Standard Deviation | Standard deviation of the sequenced fragment length. |

- Read Statistics

| Statistic | Definition |
|---|---|
| Percent Aligned | Percentage of reads passing filter that aligned. |
| Percent Q30 | The percentage of bases with a quality score of 30 or higher. |
| Mismatch Rate | The average percentage of mismatches across both reads 1 and 2 over all cycles. |

## Small Variants Summary

This table provides metrics about the number of SNVs, insertions, and deletions.

| Statistic | Definition |
|---|---|
| Total Passing | Total number of variants present in the data set that pass the variant quality filters. |
| Percent found in dbSNP | 100*(Number of variants in dbSNP/Number of variants). |
| Het/Hom Ratio | Number of heterozygous variants/Number of homozygous variants. |
| Ts/Tv Ratio | Transition rate of SNVs that pass the quality filters divided by transversion rate of SNVs that pass the quality filters. Transitions are interchanges of purines (A, G) or of pyrimidines (C, T). Transversions are interchanges between purine and pyrimidine bases (for example, A to T). |

## Variants by Sequence Context

| Statistic | Definition |
|---|---|
| Number in Genes | The number of variants that fall into a gene. |
| Number in Exons | The number of variants that fall into an exon. |
| Number in Coding Regions | The number of variants that fall into a coding region. |
| Number in UTR Regions | The number of variants that fall into an untranslated region (UTR). |
| Number in Splice Site Regions | The number of variants that fall into a splice site region. |
| Number in Mature microRNA | The number of variants that fall into a mature microRNA. |

## Variants by Consequence

| Statistic | Definition |
|---|---|
| Frameshifts | The number of variants that cause a frameshift. |
| Non-synonymous | The number of variants that cause an amino acid change in a coding region. |
| Synonymous | The number of variants that are within a coding region, but do not cause an amino acid change. |
| Stop Gained | The number of variants that cause an additional stop codon. |
| Stop Lost | The number of variants that cause the loss of a stop codon. |

## Structural Variants Summary

This table breaks structural variant output into the classes of variants called, and reports the total number and their overlap with annotated genes. All counts are based on PASS filter variants.

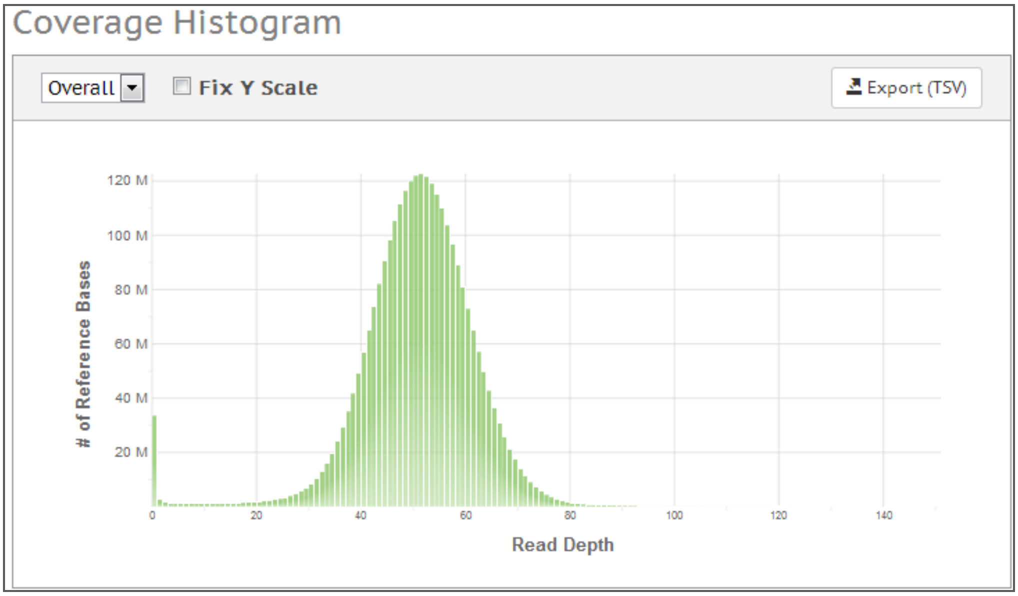
| Variant Class | Definition |
|---|---|
| CNV | Number of copy number variations. |
| Insertions | Number of insertions |
| Tandem duplications | Number of tandem duplications |
| Deletions | Number of deletions |
| Inversion | Number of inversions |

## Coverage Histogram

The coverage histogram shows the number of reference bases plotted against the depth of coverage (read depth). It has the following features:

- The dropdown menu allows you to look at the overall picture, or highlight a particular chromosome.
- The **Fix Y Scale** checkbox allows you to keep the Y Scale the same when comparing multiple chromosomes.
- The Export TSV link allows you to export the coverage data in a tab-separated TXT file.

Figure 4 Isaac Whole Genome Sequencing Coverage Histogram



## Analysis Info

This app provides an overview of the analysis on the Analysis Info page.

A brief description of the metrics is below.

| Row | Definition |
|---|---|
| Name | Name of the app session. |
| Application | App that generated this analysis. |
| Date started | Date and time the app session started. |
| Date completed | Date and time the app session completed. |
| Duration | Duration of analysis. |
| Session Type | The number of nodes used. |
| Size | Total size of all output files. |
| Status | Status of the app session. |

### Log Files

Clicking the **Log Files** link at the bottom of the Analysis Info page provides access to Isaac Whole Genome Sequencing app log files. Log files are located in a folder in the Output Files section.

The key log files to help follow data processing and debugging are the following:

- **CompletedJobInfo.xml**: Contains information about the completed job.
- **Logging.zip**: Contains all detailed workflow log files for each step of the workflow.
- **SampleSheetUsed.csv**: A copy of the sample sheet, generated at the end of a run.

- **WorkflowError.txt**: Workflow standard error output (contains errors messages created while running the workflow).
- **WorkflowLog.txt**: Workflow standard output (contains details about workflow steps, command line calls with parameters, timing, and progress).

The following files contain additional information in case components (such as mono) do not work as expected:

- **monoErr.txt**: Wrapper mono call error output (contains anything that WorkflowError.txt does not catch; in most cases empty, except one line).
- **monoOut.txt**: Wrapper mono call standard output (contains command calling the workflow and anything that WorkflowLog.txt does not catch).

NOTE
For explanation about mono, see www.mono-project.com.

### Isaac Whole Genome Sequencing Status

The status of the Isaac Whole Genome Sequencing app session can have the following values:

1. Preparing Run Data
2. Finished Preparing Run Data
3. Analysis Started
4. Alignment for Sample {SampleName}
5. *If SV/CNV is selected:* Detect CNV for Sample {SampleName}
6. *If SV/CNV is selected:* Detect SV for Sample {SampleName}
7. Variant analysis for Sample {SampleName}
8. Statistics evaluation for Sample {SampleName}
9. Report generation for Sample {SampleName}
10. Analysis Completed for Sample {SampleName}
11. Finalizing Analysis Results for Sample {SampleName}
12. Finished Finalizing Analysis Results

## Inputs

The Isaac Whole Genome Sequencing app provides an overview of the input samples and settings that were specified when setting up the Isaac Whole Genome Sequencing run.

## Isaac Whole Genome Sequencing Files

The Files page provides access to the output files. See the following pages for descriptions:

- *BAM Files* on page 13
- *VCF Files* on page 13
- *gVCF Files* on page 17
- *Resequencing_summary.csv* on page 20
- *Sample Summary Report* on page 22

## BAM Files

The Sequence Alignment/Map (SAM) format is a generic alignment format for storing read alignments against reference sequences, supporting short and long reads (up to 128 Mb) produced by different sequencing platforms. SAM is a text format file that is human-readable. The Binary Alignment/Map (BAM) keeps the same information as SAM, but in a compressed, binary format that is only machine readable.

### Detailed Description

The file naming convention for aligned reads in BAM format is as follows: SampleName_S#.bam (where # is the sample number determined by ordering in the sample sheet).

Go to samtools.sourceforge.net/SAM1.pdf to see the exact SAM specification.

BWA adds some custom fields to the BAM output. See bio-bwa.sourceforge.net/bwa.shtml#4 for a description.

## VCF Files

VCF is a text file format that contains information about variants found at specific positions in a reference genome. The file format consists of meta-information lines, a header line, and then data lines. Each data line contains information about a single variant.

More information is available here: www.1000genomes.org/wiki/Analysis/Variant%20Call%20Format/vcf-variant-call-format-version-41.

### VCF File Format

The file naming convention for VCF files is as follows: SampleName_S#.vcf (where # is the sample number determined by ordering in the sample sheet).

The header of the VCF file describes the tags used in the remainder of the file. A description of the tags is also provided here and on www.broadinstitute.org/gatk/guide/article?id=1268.

| Setting | Description |
|---|---|
| CHROM | The chromosome of the reference genome. Chromosomes appear in the same order as the reference FASTA file (generally karyotype order) |
| POS | The 1-based position of this variant in the reference chromosome. The convention for *.vcf files is that, for SNPs, this base is the reference base with the variant. For indels or deletions, this base is the reference base **immediately before** the variant. Variants are ordered by position. |
| ID | The rs number for the SNP obtained from dbSNP. If there are multiple rs numbers at this location, the list is semi-colon delimited. If no dbSNP entry exists at this position, the missing value ('.') is used. |

| Setting | Description |
|---|---|
| **REF** | The reference genotype. For example, a deletion of a single T can be represented as reference TT and alternate T. |
| **ALT** | The alleles that differ from the reference read. For example, an insertion of a single T can be represented as reference A and alternate AT. |
| **QUAL** | A Phred-scaled quality score assigned by the variant caller. Higher scores indicate higher confidence in the variant (and lower probability of errors). For a quality score of Q, the estimated probability of an error is 10-(Q/10). For example, the set of Q30 calls has a 0.1% error rate. Many variant callers assign quality scores (based on their statistical models) which are high relative to the error rate observed in practice. |
| **FILTER** | See *VCF FILTER Entries* on page 16 for possible entries. |
| **FORMAT** | See *VCF FORMAT Entries* on page 16 for possible entries. |
| **INFO** | See *VCF INFO Entries* on page 16 for possible entries. |

<table>
<tr><th>Setting</th><th>Description</th></tr>
<tr><td>INFO</td><td>Illumina Annotation Service (IAS) provided annotations are:<br>• CSQT – Transcript consequence as predicted by Variant Effect Predictor (www.ensembl.org/info/docs/tools/vep/index.html) version 72. Only canonical transcripts are included in the VCF file to maintain readability. The ANT file contains consequences for all affected transcripts. This binary file can be loaded into VariantStudio for viewing; see www.illumina.com/clinical/clinical_informatics/illumina-variantstudio.ilmn.<br>A comma-separated list for each affected gene is provided. Each entry in the list includes the HGNC gene symbol (when available), transcript ID, and functional consequences in a delimited format: HGNC|TranscriptID|Consequence. If the annotation source selected was RefSeq, then many of the TranscriptIDs begin with `NM_`. If the selected annotation source was Ensembl, then the TranscriptIDs begin with `ENST`. The consequences are indicated using valid Sequence Ontology (SO) terms (www.ensembl.org/info/genome/variation/predicted_data.html#consequences).<br>• CSQR – Regulatory consequence as predicted by Variant Effect Predictor (www.ensembl.org/info/docs/tools/vep/index.html) version 72. A comma-separated list for each affected regulatory region (including transcription factor binding sites) is provided using the following delimited format: RegulatoryID|Consequence. The annotations provided in this field come from the Ensembl database of regulatory features even if RefSeq was selected as the annotation source. Many of the RegulatoryIDs begin with `ENSR`. The consequences are indicated using valid Sequence Ontology (SO) terms (www.ensembl.org/info/genome/variation/predicted_data.html#consequences) and typically are either `regulatory_region_variant` or `TF_binding_site_variant`.<br>• AF – The allele frequency from all populations of 1000 genomes data<br>• AA – The inferred allele ancestral to the chimpanzee/human lineage<br>• GMAF – Global minor allele frequency (GMAF); technically, the frequency of the second most frequent allele. Format: GlobalMinorAllele|AlleleFreqGlobalMinor<br>• EVS – Allele frequency, sample count, and coverage taken from the Exome Variant Server (EVS). Format: AlleleFreqEVS|EVSCoverage|EVSSamples<br>• cosmic – The numeric identifier for the variant in the Catalogue of Somatic Mutations in Cancer (COSMIC) database (cancer.sanger.ac.uk/cancergenome/projects/cosmic/).<br>• clinvar – Clinical significance from the ClinVar database (www.ncbi.nlm.nih.gov/clinvar/).<br>• phastCons – Denotes if the variant is an identical or similar sequence that occurs between species and maintained between species throughout evolution</td></tr>
<tr><td>SAMPLE</td><td>The sample column gives the values specified in the FORMAT column. One MAXGT sample column is provided for the normal genotyping (assuming the reference). For reference, a second column is provided for genotyping assuming the site is polymorphic.</td></tr>
</table>

NOTE
The full set of provided annotations from the Illumina Annotation Service (IAS) can be accessed through a binary annotation (ANT) file that accompanies the VCF file.

## Isaac Whole Genome Sequencing VCF Entries

The VCF files for Isaac Whole Genome Sequencing can have the following entries in the FILTER, FORMAT, and INFO fields:

Table 2 VCF FILTER Entries

| Entry | Description |
|---|---|
| IndelConflict | Locus is in region with conflicting indel calls |
| SiteConflict | Site genotype conflicts with proximal indel call, typically a heterozygous SNV call made inside of a heterozygous deletion |
| LowGQX | Locus GQX is less than 30 or not present |
| HighDPFRatio | The fraction of base calls filtered out at a site is greater than 0.4 |
| HighSNVSB | SNV strand bias value (SNVSB) exceeds 10 |
| HighDepth | Locus depth is greater than 3x the mean chromosome depth |

Table 3 VCF FORMAT Entries

| Entry | Description |
|---|---|
| GQX | Minimum of {Genotype quality assuming variant position,Genotype quality assuming non-variant position} |
| GT | Genotype |
| GQ | Genotype Quality |
| DP | Filtered base call depth used for site genotyping |
| DPF | Base calls filtered from input before site genotyping |
| AD | Allelic depths for the ref and alt alleles in the order listed. For indels, this value only includes reads that confidently support each allele (posterior probability 0.999 or higher that read contains indicated allele vs all other intersecting indel alleles) |
| DPI | Read depth associated with indel, taken from the position preceding the indel. |

Table 4 VCF INFO Entries

| Entry | Description |
|---|---|
| SNVSB | SNV site strand bias |
| SNVHPOL | SNV contextual homopolymer length |
| CIGAR | CIGAR alignment for each alternate indel allele |
| RU | Smallest repeating sequence unit extended or contracted in the indel allele relative to the reference. RUs longer than 20 bases are not reported. |
| REFREP | Number of times RU is repeated in reference. |

| Entry | Description |
|---|---|
| IDREP | Number of times RU is repeated in indel allele. |
| END | End position of the region described in this record |
| BLOCKAVG_ min30p3a | Non-variant site block. All sites in a block are constrained to be non-variant, have the same filter value, and have all sample values in range [x,y], y <= max(x+3,(x*1.3)). All printed site block sample values are the minimum observed in the region spanned by the block |

## gVCF Files

This application also produces the Genome Variant Call Format file (gVCF). gVCF was developed to store sequencing information for both variant and non-variant positions, which is required for human clinical applications. gVCF is a set of conventions applied to the standard variant call format (VCF) 4.1 as documented by the 1000 Genomes Project. These conventions allow representation of genotype, annotation, and other information across all sites in the genome in a compact format. Typical human whole-genome sequencing results expressed in gVCF with annotation are less than 1 Gbyte, or about 1/100 the size of the BAM file used for variant calling. If you are performing targeted sequencing, gVCF is also an appropriate choice to represent and compress the results.

gVCF is a text file format, stored as a gzip compressed file (*.genome.vcf.gz). Compression is further achieved by joining contiguous non-variant regions with similar properties into single 'block' VCF records. To maximize the utility of gVCF, especially for high stringency applications, the properties of the compressed blocks are conservative. Block properties like depth and genotype quality reflect the minimum of any site in the block. The gVCF file can be indexed (creating a *.tbi file) and used with existing VCF tools such as tabix and IGV, making it convenient both for direct interpretation and as a starting point for tertiary analysis.

For more information, see sites.google.com/site/gvcftools/home/about-gvcf.

The following conventions are used in the variant caller gVCF files.

### Samples per File

There is only one sample per gVCF file.

### Non-Variant Blocks Using END Key

Contiguous non-variant segments of the genome can be represented as single records in gVCF. These records use the standard 'END' INFO key to indicate the extent of the record. Even though the record can span multiple bases, only the first base is provided in the REF field to reduce file size.

The following is a simplified segment of a gVCF file, describing a segment of non-variant calls (starting with an A) on chromosome 1 from position 51845 to 51862.

```
##INFO=<ID=END,Number=1,Type=Integer,Description="End position
   of the variant described in this record">#CHROM POS ID REF
   ALT QUAL FILTER INFO FORMAT NA19238chr1 51845 . A . . PASS
   END=51862
```

Any field provided for a block of sites, such as read depth (using the DP key), shows the minimum value that is observed among all sites encompassed by the block. Each sample value shown for the block, such as the depth (DP), is restricted to a range where the maximum value is within 30% or 3 of the minimum. For example, for sample value range [x,y], y <= x+max(3,x*0.3). This range restriction applies to each of the sample values printed in the final block record.

### Indel Regions

Sites that are "filled in" inside of deletions have additional changes:

All deletions:

- Sites inside of any deletion are marked with the deletion filters, in addition to any filters that have already been applied to the site.
- Sites inside of deletions cannot have a genotype or alternate allele quality score higher than the corresponding value from the enclosing indel.

Heterozygous deletions:

- Sites inside of heterozygous deletions are altered to have haploid genotype entries (e.g. "0" instead of "0/0", "1" instead of "1/1").
- Heterozygous SNV calls inside of heterozygous deletions are marked with the "SiteConflict" filter and their genotype is unchanged.

Homozygous deletions:

- Homozygous reference and no-call sites inside of homozygous deletions have genotype "."
- Sites inside of homozygous deletions that have a non-reference genotype are marked with a “SiteConflict” filter, and their genotype is unchanged.
- Site and genotype quality are set to "."

The described modifications reflect the notion that the site confidence is bound within the enclosing indel confidence.

On occasion, the variant caller produces multiple overlapping indel calls that cannot be resolved into two haplotypes. If this case, all indels and sites in the region of the overlap are marked with the *IndelConflict* filter.

### Genotype Quality for Variant and Non-variant Sites

The gVCF file uses an adapted version of genotype quality for variant and non-variant site filtration. This value is associated with the key GQX. The GQX value is intended to represent the minimum of {Phred genotype quality assuming the site is variant, Phred genotype quality assuming the site is non-variant}. The reason for using this value is to allow a single value to be used as the primary quality filter for both variant and non-variant sites. Filtering on this value corresponds to a conservative assumption appropriate for applications where reference genotype calls must be determined at the same stringency as variant genotypes, i.e.:

- An assertion that a site is homozygous reference at GQX >= 30 is made assuming the site is variant.
- An assertion that a site is a non-reference genotype at GQX >= 30 is made assuming the site is non-variant.

### Section Descriptions

The gVCF file contains the following sections:

- Meta-information lines start with ## and contain metadata, config information, and define the values that the INFO, FILTER, and FORMAT fields can have.

- The header line starts with # and names the fields that the data lines use. These fields are #CHROM, POS, ID, REF, ALT, QUAL, FILTER, INFO, FORMAT, followed by one or more sample columns.
- Data lines that contain information about one or more positions in the genome.

If you extract the variant lines from a gVCF file, you produce a conventional variant VCF file.

## Field Descriptions

The fixed fields #CHROM, POS, ID, REF, ALT, QUAL are defined in the VCF 4.1 standard provided by the 1000 Genomes Project. The fields ID, INFO, FORMAT, and sample are described in the meta-information.

- **CHROM**: Chromosome: an identifier from the reference genome or an angle-bracketed ID String ("<ID>") pointing to a contig.
- **POS**: Position: The reference position, with the first base having position 1. Positions are sorted numerically, in increasing order, within each reference sequence CHROM. There can be multiple records with the same POS. Telomeres are indicated by using positions 0 or N+1, where N is the length of the corresponding chromosome or contig.
- **ID**: Semi-colon separated list of unique identifiers where available. If this ID is a dbSNP variant, it is encouraged to use the rs number. No identifier is present in more than one data record. If there is no identifier available, then the missing value is used.
- **REF**: Reference bases: A,C,G,T,N; there can be multiple bases. The value in the POS field refers to the position of the first base in the string. For simple insertions and deletions in which either the REF or one of the ALT alleles would otherwise be null/empty, the REF and ALT strings include the base before the event. This modification is reflected in the POS field. The exception is when the event occurs at position 1 on the contig, in which case they include the base after the event. If any of the ALT alleles is a symbolic allele (an angle-bracketed ID String "<ID>"), the padding base is required. In that case, POS denotes the coordinate of the base preceding the polymorphism.
- **ALT**: Comma-separated list of alternate non-reference alleles called on at least one of the samples. Options are:
  - Base strings made up of the bases A,C,G,T,N
  - Angle-bracketed ID String (”<ID>”)
  - Break-end replacement string as described in the section on break-ends.

  If there are no alternative alleles, then the missing value is used.
- **QUAL**: Phred-scaled quality score for the assertion made in ALT. i.e. -10log_10 probability (call in ALT is wrong). If ALT is ”.” (no variant), this score is -10log_10 p (variant). If ALT is not ”.”, this score is -10log_10 p(no variant). High QUAL scores indicate high confidence calls. Although traditionally people use integer Phred scores, this field is permitted to be a floating point to enable higher resolution for low confidence calls if desired. If unknown, the missing value is specified. (Numeric)
- **FILTER**: PASS if this position has passed all filters, i.e. a call is made at this position. Otherwise, if the site has not passed all filters, a semicolon-separated list of codes for filters that fail. gVCF files use the following values:
  - *PASS*: position has passed all filters.
  - *IndelConflict*: Locus is in region with conflicting indel calls.
  - *SiteConflict*: Site genotype conflicts with proximal indel call, which is typically a heterozygous SNV call made inside of a heterozygous deletion.

  - *LowGQX*: Locus GQX (minimum of {Genotype quality assuming variant position,Genotype quality assuming non-variant position}) is less than 30 or not present.
  - *HighDPFRatio*: The fraction of base calls filtered out at a site is greater than 0.3.
  - *HighSNVSB*: SNV strand bias value (SNVSB) exceeds 10. High strand bias indicates a potential high false-positive rate for SNVs.
  - *HighSNVHPOL*: SNV contextual homopolymer length (SNVHPOL) exceeds 6.
  - *HighREFREP*: Indel contains an allele that occurs in a homopolymer or dinucleotide track with a reference repeat greater than 8.
  - *HighDepth*: Locus depth is greater than 3x the mean chromosome depth.
- **INFO**: Additional information. INFO fields are encoded as a semicolon-separated series of short keys with optional values in the format: <key>=<data>[,data]. gVCF files use the following values:
  - *END*: End position of the region described in this record.
  - *BLOCKAVG_min30p3a*: Non-variant site block. All sites in a block are constrained to be non-variant, have the same filter value, and have all sample values in range [x,y], y <= max(x+3,(x*1.3)). All printed site block sample values are the minimum observed in the region spanned by the block.
  - *SNVSB*: SNV site strand bias.
  - *SNVHPOL*: SNV contextual homopolymer length.
  - *CIGAR*: CIGAR alignment for each alternate indel allele.
  - *RU*: Smallest repeating sequence unit extended or contracted in the indel allele relative to the reference. If longer than 20 bases, RUs are not reported.
  - *REFREP*: Number of times RU is repeated in reference.
  - *IDREP*: Number of times RU is repeated in indel allele.
- **FORMAT**: Format of the sample field. FORMAT specifies the data types and order of the subfields. gVCF files use the following values:
  - *GT*: Genotype.
  - *GQ*: Genotype Quality.
  - *GQX*: Minimum of {Genotype quality assuming variant position, Genotype quality assuming non-variant position}.
  - *DP*: Filtered base call depth used for site genotyping.
  - *DPF*: Base calls filtered from input before site genotyping.
  - *AD*: Allelic depths for the ref and alt alleles in the order listed. For indels, this value only includes reads that confidently support each allele (posterior probability 0.999 or higher that read contains indicated allele vs all other intersecting indel alleles).
  - *DPI*: Read depth associated with indel, taken from the site preceding the indel.
- **SAMPLE**: Sample fields as defined by the header.

## Resequencing_summary.csv

The Isaac Whole Genome Sequencing app produces an overview of statistics for each sample in a comma-separated values (CSV) format: the *.resequencing_summary.csv. The Resequencing_summary.csv presents the same data as the Sample Summary Report, but in an easier to parse format. These files are located in the results folder.

A brief description of the metrics is below.

| Statistic | Definition |
|---|---|
| Sample ID | IDs of samples reported on in the file. |
| Run Folder | Run folders for samples reported on in the file. |
| Fragment length median | Median length of the sequenced fragment. The fragment length is calculated based on the locations at which a read pair aligns to the reference. The read mapping information is parsed from the BAM files. |
| Fragment length min | Minimum length of the sequenced fragment. |
| Fragment length max | Maximum length of the sequenced fragment. |
| Fragment length SD | Standard deviation of the sequenced fragment length. |
| Number of Reads | Total number of reads passing filter for this sample. |
| Percent Aligned (per read) | Percentage of reads passing filter that aligned. |
| Percent Q30 (per read) | The percentage of bases with a quality score of 30 or higher. |
| MismatchRate (per read) | The average percentage of mismatches across both reads 1 and 2 over all cycles. |
| SNVs All | Total number of Single Nucleotide Variants present in the data set passing the quality filters. |
| SNVs Passing Filters | SNVs passing variants filter. |
| SNVs (Percent found in dbSNP) | 100*(Number of SNVs in dbSNP/Number of SNVs). |
| SNV Ts/Tv ratio | Transition rate of SNVs that pass the quality filters divided by transversion rate of SNVs that pass the quality filters. Transitions are interchanges of purines (A, G) or of pyrimidines (C, T). Transversions are interchanges of purine and pyrimidine bases (for example, A to T). |
| SNV Het/Hom ratio | Number of heterozygous SNVs/Number of homozygous SNVs. |
| Indels | Total number of indels present in the data set passing the quality filters. |
| Insertions Passing Variants | Insertions passing variant filters. |
| Deletions Passing Variants | Deletions passing variant filters. |
| Indels (Percent found in dbSNP) | 100*(Number of Indels in dbSNP/Number of Indels). |
| Insertions (Percent found in dbSNP) | 100*(Number of insertions in dbSNP/ Number of insertions) |
| Deletions(Percent found in dbSNP) | 100*(Number of deletions in dbSNP/ Number of deletions) |
| Indel Het/Hom ratio | Number of heterozygous indels/Number of homozygous indels. |

| Statistic | Definition |
|---|---|
| Insertion Het/Hom ratio | Ratio of the number of heterozygous to homozygous insertions. |
| Deletion Het/Hom ratio | Ratio of the number of heterozygous to homozygous deletions. |
| SmallVariantStatisticsFlag | Flags whether SmallVariantStatistics was run (1 means that it was run) |
| SVStatisticsFlag | Flags whether SVStatistics was run (1 means that it was run) |
| CNVStatisticsFlag | Flags whether CNVStatistics was run (1 means that it was run) |

## Sample Summary Report

The Sample Summary Report presents the same data as the Resequencing_summary.csv, but in an easier to read format for humans. These files are located in the results folder.

For a description of the presented metrics, see *Resequencing_summary.csv* on page 20.

# Isaac Whole Genome Sequencing Methods

This chapter describes the methods that are used in the Isaac Whole Genome Sequencing app.

## Isaac Aligner

The Isaac aligner aligns DNA sequencing data, single or paired-end, with read lengths and low error rates using the following steps:

- **Candidate mapping positions**—Identifies the complete set of relevant candidate mapping positions using a 32-mer seed-based search.
- **Mapping selection**—Selects the best mapping among all candidates.
- **Alignment score**—Determines alignment scores for the selected candidates based on a Bayesian model.
- **Alignment output**—Generates final output in a sorted duplicate-marked BAM file and summary file.

[1] Come Raczy, Roman Petrovski, Christopher T. Saunders, Ilya Chorny, Semyon Kruglyak, Elliott H. Margulies, Han-Yu Chuang, Morten Källberg, Swathi A. Kumar, Arnold Liao, Kristina M. Little, Michael P. Strömberg and Stephen W. Tanner (2013) Isaac: Ultra-fast whole genome secondary analysis on Illumina sequencing platforms. Bioinformatics 29(16):2041-3
bioinformatics.oxfordjournals.org/content/29/16/2041

### Candidate Mapping

To align reads, the Isaac aligner first identifies a small but complete set of relevant candidate mapping positions. The Isaac aligner begins with a seed-based search using 32-mers as seeds. After the initial single-seed search, Isaac performs a multi-seed search for only those reads that were not mapped unambiguously with a single seed.

### Mapping Selection

Following a seed-based search, the Isaac aligner selects the best mapping among all the candidates. For paired-end data sets, all mappings where only one end is aligned (called orphan mappings) trigger a local search to find additional mapping candidates. These candidates (called shadow mappings) are defined through the expected minimum and maximum insert size. After optional trimming of low quality 3' ends and adapter sequences, the possible mapping positions of each fragment are compared. This step takes into account pair-end information (when available), possible gaps using a banded Smith-Waterman gap aligner, and possible shadows. The selection is based on the Smith-Waterman score and on the log-probability of each mapping.

### Alignment Scores

The alignment scores of each read pair are based on a Bayesian model, where the probability of each mapping is inferred from the base qualities and the positions of the mismatches. The final mapping quality is the alignment score, truncated to 60 for scores above 60, and possibly corrected to known ambiguities in the reference as flagged in the seeds. Following alignment, reads are sorted. Further analysis is performed to identify duplicates and optionally to realign indels.

### Alignment Output

After sorting the reads, the Isaac aligner generates compressed binary alignment output files, called BAM (*.bam) files, using the following process:

- **Marking duplicates**—Detection of duplicates is based on the location and observed length of each fragment. The Isaac aligner identifies and marks duplicates even when they appear on oversized fragments or chimeric fragments. Optical duplicates are already filtered out during RTA processing.
- **Realigning indels**—The Isaac aligner tracks previously detected indels, over a window large enough for the current read length, and applies the known indels to all reads with mismatches.
- **Generating BAM files**—The first step in BAM file generation is creation of the BAM record, which contains all required information except the name of the read. The Isaac aligner reads data from base call (BCL) files that were written during primary analysis on the sequencer to generate the read names. Data are then compressed into blocks of 64 kb or less to create the BAM file.

## Isaac Variant Caller

The Isaac Variant Caller identifies single nucleotide polymorphisms (SNPs) and small indels using the following steps:

- **Read filtering**—Filters out reads failing quality checks.
- **Indel calling**—Identifies a set of possible indel candidates and realigns all reads overlapping the candidates using a multiple sequence aligner.
- **SNP calling**—Computes the probability of each possible genotype given the aligned read data and a prior distribution of variation in the genome.
- **Indel genotypes**—Calls indel genotypes and assigns probabilities.
- **Variant call output**—Generates output in a VCF file and a compressed genome variant call (gVCF) file. See *VCF Files* on page 13 and *gVCF Files* on page 17 for details.

### Indel Candidates

Input reads are filtered by removing any of the following:

- Reads that failed primary analysis quality checks.
- Reads marked as PCR duplicates.
- Paired-end reads not marked as a proper pair.
- Reads with a mapping quality less than 20.

### Indel Calling

The variant caller proceeds with candidate indel discovery and generates alternate read alignments based on the candidate indels. As part of the realignment process, the variant caller selects a representative alignment to be used for site genotype calling and depth summarization by the SNP caller.

### SNP Calling

The variant caller runs a series of filters on the set of filtered and realigned reads for SNP calling without affecting indel calls. First, any contiguous trailing sequence of N base calls is trimmed from the ends of reads. Using a mismatch density filter, reads having an unexpectedly high number of disagreements with the reference are masked, as follows:

- The variant caller treats each insertion or deletion as a single mismatch.
- Base calls with more than two mismatches to the reference sequence within 20 bases of the call are ignored.
- If the call occurs within the first or last 20 bases of a read, the mismatch limit is applied to a 41-base window at the corresponding end of the read.
- The mismatch limit is applied to the entire read when the read length is 41 or shorter.

### Indel Genotypes

The variant caller filters out all bases marked by the mismatch density filter and any N base calls that remain after the end-trimming step. These filtered base calls are not used for site-genotyping but appear in the filtered base call counts in the variant caller output for each site.

All remaining base calls are used for site-genotyping. The genotyping method heuristically adjusts the joint error probability that is calculated from multiple observations of the same allele on each strand of the genome. This correction accounts for the possibility of error dependencies.

This method treats the highest-quality base call from each allele and strand as an independent observation and leaves the associated base call quality scores unmodified. Quality scores for subsequent base calls for each allele and strand are then adjusted. This adjustment is done to increase the joint error probability of the given allele above the error expected from independent base call observations.

### Variant Call Output

After the site and indel genotyping methods are complete, the variant caller applies a final set of heuristic filters to produce the final set of non-filtered calls in the output.

The output in the genome variant call (gVCF) file captures the genotype at each position and the probability that the consensus call differs from reference. This score is expressed as a Phred-scaled quality score.

## Large Indel and Structural Variant Calls

The large indel and structural variant caller uses the series of modules described here, and then generates output files in VCF 4.1 format.

#### Before ReadBroker

- **StatsGenerator**—Computes summary statistics on insert sizes, read orientation, and alignment scores for each input BAM file.
- **AnomalousReadFinder**—Grouper processes chromosomes in chunks. This method enables parallel execution and, therefore, faster performance. AnomalousReadFinder examines all alignments in a block and classifies reads and read pairs as follows:
  - Classifies reads as either shadow (unaligned) or semi-aligned partial or clipped alignment).
  - Classifies read pairs as either InsertionPair, DeletionPair, InversionPair, TandemDuplicationPair, or ChimericPair, according to which type of structural variant an anomalously mapped read pair is associated.

- **ClusterFinder**—Clusters reads based on their type and the position of their alignment. Only reads of the same type are clustered together at this stage, except shadow and semi-aligned reads, which can be clustered together.
- **ClusterMerger**—Associates clusters of various anomalous read types with shadow/semi-aligned read clusters, which breakpoints can cause. A breakpoint is a pair of bases that are adjacent in the sample genome but not in the reference. Two clusters are merged if they share the read or if they agree on the position and length of the structural variant. This information is inferred from read alignment orientation and distance.

### ReadBroker

- Interchromosomal translocations yield chimeric read pairs where one read aligns to one chromosome and its partner aligns to another. Because Grouper examines each chromosome individually, the ReadBroker step is performed to join the information from chimeric read pairs across chromosomes.

### After ReadBroker

- **SmallAssembler**—Assembles reads in clusters into contigs using a *de Bruijn* method and iteratively assembles reads into contigs until all reads in the cluster are assembled. It also produces a file containing the reads that were used to assemble the contig, with a realignment to the contig sequence.
- **SpanContigs**—Uses the presence of nearby anomalous read pairs to determine whether to extend the search range used by the subsequent AlignContig step from its default.
- **AlignContig**—Computes a dynamic programming alignment of a contig to a region of the reference genome; merges full or partial duplicate calls of the same event into a single call.
- **VariantFilter**—Removes all structural variants that overlap with gaps identified in UCSC gaps. The UCSC gaps file defines regions of the genome that have not been sequenced.
- **DeletionGenotyper**—Assigns a genotype to all deletions.

## CNV Variant Caller

CNV variant caller is designed to identify copy number variants (CNVs) in diploid genomes using Hidden Markov Models (HMM) or unbalanced Haar wavelets. The method adopts a count-based approach for CNV calling and comprises two main steps:

1. Pre-processing step, during which read depth is computed at each position and then filtered based on CpG islands, assembly gaps, telomeric/centromeric regions. Either alignability tracks or coverage tracks obtained from a pool of reference sample are used to normalize the data. Counts or count ratios are produced as an output.
2. Segmentation of read counts/ratios using fixed or variable bin size and a copy number assignment.

### Normalization

A single sample or a pool of reference samples is used for normalization, by deriving a ratio between a test and the reference. Window size is fixed (by default to 100 bp). The HMM model with Gaussian emission distribution is used for segmentation. A bin exclusion criterion (less than 10% of build coverage in both samples) is applied.

The reference for CNV normalization is an alignability measure that is meant to gauge the probability of a position aligning to a single unique region of the genome. In detail, the notion of alignability for reads of length k is as follows: given a map M that, for a fixed read length k and any position P in a genome G, stores at M(P) the number of occurrences in G of the k-mer that starts at P for a given position P in G, define the overlap set of P as the k-mers that overlap P. The alignability of P is the proportion of this overlap set that is unique.

## Variant Scoring

After copy number assignment, each CNV call is assigned a quality score based on a two-sample t-test. Each counts/ratio in a 1 kb window on each size of a breakpoint (or half the length of a variant call, whichever is smaller) is compared using t-test. This test is based on the null hypothesis that there is no difference in coverage on each size of the breakpoint. Obtained p-values are then reported as Q-scores on a Phred scale as -10 log10.

# Notes

# Technical Assistance

For technical assistance, contact Illumina Technical Support.

Table 5 Illumina General Contact Information

| | |
|---|---|
| **Illumina Website** | www.illumina.com |
| **Email** | techsupport@illumina.com |

Table 6 Illumina Customer Support Telephone Numbers

| Region | Contact Number | Region | Contact Number |
|---|---|---|---|
| North America | 1.800.809.4566 | Italy | 800.874909 |
| Austria | 0800.296575 | Netherlands | 0800.0223859 |
| Belgium | 0800.81102 | Norway | 800.16836 |
| Denmark | 80882346 | Spain | 900.812168 |
| Finland | 0800.918363 | Sweden | 020790181 |
| France | 0800.911850 | Switzerland | 0800.563118 |
| Germany | 0800.180.8994 | United Kingdom | 0800.917.0041 |
| Ireland | 1.800.812949 | Other countries | +44.1799.534000 |

## Safety Data Sheets

Safety data sheets (SDSs) are available on the Illumina website at support.illumina.com/sds.html.

## Product Documentation

Product documentation in PDF is available for download from the Illumina website. Go to support.illumina.com, select a product, then click **Documentation & Literature**.

Illumina
San Diego, California 92122 U.S.A.
+1.800.809.ILMN (4566)
+1.858.202.4566 (outside North America)
techsupport@illumina.com
www.illumina.com